Risk Aware Warning and Control for Interactive Traffic Safety

John M. Dolan (PI) (https://orcid.org/0000-0003-2062-100X)
Yiwei Lyu (https://orcid.org/0009-0007-0136-5369)
Paul Schulte (https://orcid.org/0009-0002-0755-9352)

FINAL REPORT

July 31, 2024

DISCLAIMER

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, under [grant number 69A3552344811] from the U.S. Department of Transportation's University Transportation Centers Program. The U.S. Government assumes no liability for the contents or use thereof.

Contents

Chapter 1

Introduction

Interactions become significantly more complex when humans are involved due to their inherently unpredictable nature [4]. Surprisingly, interactions solely among humans tend to be safer than mixed interactions involving both humans and robots. This safety is attributed to the fact that human behavior is heavily reliant on agreed-upon norms governing how people should behave or respond. Thus, achieving harmonious human-robot interaction necessitates the incorporation of human factors into the system design. Approaches like inverse reinforcement learning have been extensively explored for human modeling and behavior replication [21, 7]. However, its effectiveness is highly contingent on the quality of the training data, making it susceptible to challenges in adapting to changes in scenarios and lacking formal safety guarantees.

Facing the new challenges brought by human involvement into interactions, our belief centers on empowering robots with the ability to emulate humans' impact-aware decision-making processes. This involves considering the potential consequences and impact of robot actions on humans, taking into account factors such as safety, ethics, and social norms. Through the utilization of model-based control approaches and risk quantification techniques, we have introduced a comprehensive risk assessment [9, 10] encompassing sufficiently inclusive factors. This approach equips robots with contextual awareness for making decisions mindful of potential risks. We further leverage the proposed notion of risk to formulate the safe interaction problem for mixed multi-agent systems as a responsibility reasoning challenge for robots. This involves determining the percentage of responsibility robots should take or the individual effort they should contribute toward a collective goal. My work in [14] proposed the first responsibility-oriented safety-critical controller, that enables robots to make responsible decisions by explicitly reasoning over the propagation of behavior among multiple entities and its impact on the rest of the group. The dynamical evaluation of the inter-agent influence allows robots to deter-

mine the appropriate portion of responsibility to contribute towards the collective goal without excessive risk or dominance, leading to improved task performance with formal safety guarantees. This marks a meaningful step forward, encouraging robots to contemplate their roles as collaborative partners when interacting with humans. Moreover, this framework does not require any task-related specifications and therefore can be widely applied in different robotics applications across various domains.

Chapter 2

Related Work

To achieve human-centered autonomy, it is essential to incorporate advanced risk assessment methods into the design of robot controllers, ensuring that robots are equipped with situational awareness to make informed decisions when interacting safely within dynamic and uncertain environments. In multi-agent systems where agents interact with each other in close proximity, the risk each individual agent faces, e.g. reflecting how likely one will collide with other agents, can differ based on various factors, and this information should also be reflected in controller design. Here we focus on both the direct and aggregated risks caused by the existence of surrounding agents and agents' motion under uncertainty. Conventional risk evaluation methods for collision avoidance only consider agent positions and estimate the risk based solely on the inter-agent proximity [23]. However, it is important to also consider agent motion when evaluating risk [18, 24], as two agents close to the same position moving away from or towards each other will have different collision likelihoods. Recently, Control Barrier Function (CBF) [2] as a model-based approach has been widely studied to render a set forward invariant. CBF is often used to characterize safety in terms of collision avoidance between pairwise agents based on factors such as agent positions, motion, safety radius, and agent behavior conservativeness [25, 8]. It provides a means to constrain the robot motion, so that if the robot is initially inside the defined safe set, CBF can always ensure the robot stays within the safe set with formally provable guarantees. Existing works mostly use CBF as a constraint for optimization-based controllers, and nominal control is only revised when the system is approaching the boundary of the safe set [27, 25]. Different from simply using CBF as a binary verification of whether the system is safe given the nominal control, in this work we propose a CBF-inspired risk evaluation framework to characterize to what extent the system is safe or unsafe, in order to make the best use of the information CBF provides.

To keep a multi-agent system safe, the collision-free configuration is required

for every pairwise agent during interactions. There has been some work addressing risk-aware control in dynamic environments with moving obstacles [5, 28], by posing multiple pairwise safety constraints between the individual agent under control and its neighboring agents. However, we argue that to better characterize an agent's collision risk, considering only the effect of its neighboring agents is not enough. We also need to consider the neighbors of the neighbors. The rationale behind this is that even if no collision occurs, the amount of risk the agent faces can vary depending on the existence and behavior of other agents in the shared environment. Therefore, another goal of this work is to propose a risk measurement framework that can accumulate the risk each agent faces to take neighbors of neighbors into account, providing situational awareness of the interactions among surrounding agents. The work in [18] constructs a risk level set using a hand-crafted cost function to quantify the influence of other agents positions and movement. However, it focuses on planning a safe trajectory for a single robot within the risk level set corresponding to a fixed risk threshold, which may be difficult to calibrate beforehand in order to keep a sizable admissible safe space in the interaction-intensive environment without exceeding the individual robot's risk tolerance.

On the other hand, realistic factors such as agents' motion uncertainty could also contribute to the risk imposed on individual agents. Chance constraint in the form of $\Pr(\cdot) \geq \alpha$ has been a popular tool to account for uncertainty by translating probabilistic constraints into deterministic ones [6, 11, 8]. Given a user-defined confidence level $\alpha \in (0, 1)$, the chance constraint is able to justify whether or not the deterministic condition is met with the satisfying probability. However, it is unable to quantify accumulated risk on an individual agent from different surrounding agents it may face. Compared to that, Conditional Value at Risk (CVaR) is considered as a more suitable tool for measuring risk from uncertainty [31]. With the user-defined confidence level α, $\text{CVaR}_\alpha(X) := \min_{z \in \mathbb{R}} \mathbb{E}\left[z + \frac{(X-z)^+}{1-\alpha}\right]$ [20] quantifies how bad the expected loss could be if the condition is violated. It maps the risk of uncertainty to a real number, allowing for better embedding of uncertainty information in risk measurement. Works in [1, 22] introduce the CVaR barrier function and Risk Control Barrier Function to enforce CVaR-safety and guarantee finite-time reachability to a desired set under uncertainty respectively. Different from these works, we focus on how to design risk-aware decentralized controllers by quantifying the cumulative risk the agents face in a crowded dynamic environment under uncertainty.

With that, the motivation for developing decentralized safe controllers is twofold: centralized control of a large-scale multi-agent system is computationally expensive, and inter-agent communication is not always available. Different approaches have been explored to translate centralized safe control into a decentralized setting, by splitting the safety constraints and separately solving individual optimization

problems with split constraints. In this way, agents only need to make decisions based on local information, without the need to predict what others are going to do. For example, Voronoi Cells-based methods are a common approach to achieve decentralized safe control by partitioning the joint state space of robots [32, 19]. On the other hand, CBF-based decentralized controllers are able to translate constraints from the state space into control space, making the problem more directly solvable. Various criteria for the control space partition between agents have been employed, such as agents' relative actuation limits [27] and social personalities, e.g. aggressive vs. conservative. In this chapter, we demonstrate how to embed the proposed notion of risk into both Voronoi-based Tessellations and Control Barrier Function-based decentralized safe control approaches. We show that with our proposed approaches, the individual agents can adapt to the dynamic changes in the surrounding environments with other agents to achieve implicit coordination and improved collective safety. Rigorous proof of safety guarantees are provided for both approaches.

Chapter 3

Risk Assessment

3.1 Background on Control Barrier Functions

A Control Barrier Functions (CBF) [2] is used to define an admissible control space for safety assurance of dynamical systems. One of its important properties is its forward-invariance guarantee of a desired safety set. Consider the following nonlinear system in control affine form:

$$\dot{x} = f(x) + g(x)u \tag{3.1}$$

where $x \in \mathcal{X} \subset \mathbb{R}^n$ and $u \in \mathcal{U} \subset \mathbb{R}^m$ are the system state and control input with f and g assumed to be locally Lipschitz continuous. A desired safety set $x \in \mathcal{H}$ can be denoted by the following safety function:

$$\mathcal{H} = \{x \in \mathbb{R}^n : h(x) \geq 0\} \tag{3.2}$$

Thus the control barrier function for the system to remain in the safety set can be defined as follows [2]:

Definition 1. *(Control Barrier Function) Given a dynamical system (3.1) and the set $\mathcal{H}$ defined in (3.2) with a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$, then h is a control barrier function (CBF) if there exists an extended class $\mathcal{K}_\infty$ function for all $x \in \mathcal{X}$ such that*

$$\sup_{u \in \mathcal{U}} \{L_f h(x) + L_g h(x) u\} \geq -\kappa\big(h(x)\big) \tag{3.3}$$

where $\dot{h}(x, u) = L_f h(x) + L_g h(x) u$ with $L_f h, L_g h$ as the Lie derivatives of h along the vector fields f and g. Similar to [26], in this paper we use the particular choice of extended class $\mathcal{K}_\infty$ function with the form as $\kappa(h(x)) = \alpha h(x)$ where $\alpha \geq 0$ is a CBF design parameter controlling system behaviors near the boundary of $h(x) = 0$. Hence, the admissible control space in Eq. 3.3 can be redefined as

$$\mathcal{B}(x) = \{u \in \mathcal{U} : \dot{h}(x, u) + \alpha h(x) \geq 0 \ \} \tag{3.4}$$

It is proved in [2] that any controller $u \in \mathcal{B}(x)$ will render the safe state set $\mathcal{H}$ forward-invariant, i.e., if the system (3.1) starts inside the set $\mathcal{H}$ with $x(t=0) \in \mathcal{H}$, then it implies $x(t) \in \mathcal{H}$ for all $t > 0$ under controller $u \in \mathcal{B}(x)$.

Here we consider the particular choice of pairwise vehicle safety function $h^s_{em}(x)$, safety set $\mathcal{H}^s$, and admissible safe control space $\mathcal{B}^s(x)$ as follows.

$$\begin{aligned} \mathcal{H}^s &= \{x \in \mathcal{X} :\ h^s_{em}(x) = ||x_e - x_m||^2 - R^2_{safe} \geq 0,\ \forall m\} \\ \mathcal{B}^s(x) &= \{u \in \mathcal{U} :\ \dot{h}^s_{em}(x,u) + \alpha h^s_{em}(x) \geq 0,\ \forall m\} \end{aligned} \tag{3.5}$$

where x_e, x_m are the positions of ego vehicle e and each merging vehicle m with $R_{safe} \in \mathbb{R}$ as the minimum allowed safety distance between pairwise vehicles.

3.2 Control Barrier Function (CBF)-inspired Risk Measurement

In this section, we aim to answer the following question: How can we quantify the accumulated risk an agent faces in multi-agent interactions under motion uncertainty?

Consider a multi-agent system with a total number of agents $N \in \mathcal{N}$, in which every agent has access to observations of all agents' current positions and velocities, but no direct communication is available among agents. As in [11, 12, 13], we consider the particular choice of pairwise safety function $h_{ij}(x)$, safety set $\mathcal{H}_{ij}(x)$, and admissible control space $\mathcal{B}_{ij}(x)$ for each agent pair as follows.

$$\begin{aligned} \mathcal{H}_{ij}(x) &= \{x \in \mathcal{X} :\ h_{ij}(x) = ||x_i - x_j||^2 - R^2_{safe} \geq 0, \forall i \neq j\} \\ \mathcal{B}_{ij}(x) &= \{u \in \mathcal{U} : \dot{h}_{ij}(x,u) \geq -\gamma(h_{ij}(x))\} \end{aligned} \tag{3.6}$$

where $x_i, x_j \in \mathbb{R}^2$ for $i, j \in \{1, ..., N\}$ are the positions of any pairwise agents i and j. $u = \{u_i, u_j\} \in \mathbb{R}^2$ is the joint control input of this particular agent pair, and R_{safe} is the pre-defined safety margin.

Next, to quantify the risk between each pairwise agents from potential collision and motion uncertainty, we draw inspirations from CBF and propose the following pairwise safety loss function $L_{ij}(x,u)$:

$$\begin{aligned} &L_{ij}(x,u) = -\text{CVaR}_\alpha(\dot{h}_{ij}(x,u)) - \gamma h_{ij}(x) + c \\ &= -\text{CVaR}_\alpha(2(x_i - x_j)^T(u_i + \epsilon_i - u_j - \epsilon_j)) \\ &- \gamma(||x_i - x_j||^2 - R^2_{safe}) + c \\ &= -2(x_i - x_j)^T(u_i - u_j) - 2 \cdot \text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) \\ &- \gamma(||x_i - x_j||^2 - R^2_{safe}) + c \\ &\text{(by CVaR Translational Invariance Property [22])} \end{aligned} \tag{3.7}$$

where c as a constant offset is a large number to ensure $L_{ij}(x,u)$ is always positive to prevent unintended cancelling-out when being accumulated later. $u_i, u_j \in \mathbb{R}^2$ are

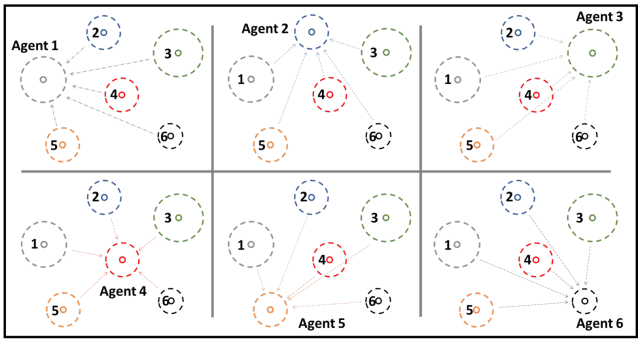


Figure 3.1: Propagated risk evaluation for individuals in a multi-agent system. Each arrow represents the risk posed by the pairwise relative movement under uncertainty of the evaluated agent and its neighboring agent.

the agent's current velocities. $\epsilon_i, \epsilon_j \sim \mathcal{N}(\hat{\epsilon}, \Sigma)$ are random Gaussian variables with known mean $\hat{\epsilon} \in \mathbb{R}^2$ and variance $\Sigma \in \mathbb{R}^{2\times 2}$, representing the uncertainty in each vehicle's motion. γ is the CBF design factor representing how aggressive the pairwise agents are [2]. $\alpha \in (0, 1)$ is the user-defined confidence level for Conditional Value at Risk calculation, e.g., $\alpha = 0.95$. The $\text{CVaR}_\alpha(\cdot) \in \mathbb{R}$ calculation indicates the expected value of $(\cdot)$ under the worst case of 5% probability that something bad happens [31]. The safety loss function $L_{ij}(x, u)$ represents how close the system is to the boundary of the safe set given the user-specified confidence level α, or how easily a safety violation could occur, under the assumption that both agents will move with piecewise-constant velocity.

Now with $L_{ij}(x, u)$ as a handy tool describing the risk agent i faces when interacting with agent j, for a multi-agent system, we define the aggregated risk $R_i \in \mathbb{R}$ agent i faces posed by all surrounding agents as:

$$R_i = \sum_{j=1}^{N} L_{ij}(x, u), \quad \forall j \neq i \tag{3.8}$$

The larger R_i is, the more likely safety violation is to occur. The proposed risk evaluation framework is simple yet effective: 1) R_i grows with the increased number of agents in the system, as the environment becomes more complex and challenging; 2) R_i varies depending on the changes of states, including positions and motion of other agents as we expected, as it is important to tell how much risk agent i is exposed to even when collision has not happened yet. Fig. 3.1 provides an illustrative example of how risk is calculated for individual agents in a multi-agent interaction scenario. For all six robots, the risk each individual agent faces consists of the pairwise risk generated by the surrounding five agents.

Note that the proposed risk evaluation framework does not necessarily require the agents to use Control Barrier Function-based controllers. We understand that

in the real world agents may use different kinds of controllers, yet it does not prevent them from understanding the risk generated from multi-agent interaction via the proposed framework, with the mild but reasonable assumption that information about safety margin, agent states, and uncertainty distribution is known or observable. Even for agents not using CBF-based controllers, it is still possible to learn the parameter γ from observations using machine learning techniques like linear ridge regression [12].

Based on the proposed risk evaluation framework, we now present CBF-inspired Risk Map (CBF-RM) as a visualization tool to better understand how risk is aggregated in multi-agent interactions.

To construct a map such as that shown in Fig. 3.2, we augment the proposed agent-to-agent risk evaluation to agent-to-point risk evaluation, by assuming that for any point $p, \forall p \neq x_i \in \mathbb{R}^2, i \in \{i, ..., N\}$ in the map, there exists a static agent with zero velocity. Then Eq. 3.8 can be used for augmented agent-to-point risk evaluation. Fig. 3.2 provides an example of how risk is aggregated when continuously adding agents with different positions, motion, safety radius, and parameter γ. Color ranging from dark blue to yellow represents the level of risk from mild to severe. Initially there is only one white agent and the highlighted yellow zone represents the collision zone, namely that if we place an agent with zero velocity anywhere inside this zone, collision will happen. As discussed earlier, we actually care more about the area outside the highlighted yellow zone, as that is the space within which robots with initial collision-free configurations navigate. It is observed that the closer the position to the agent, the greater the risk. When the red agent is added to the map, since it has a larger velocity, the risk it brings to the environment also increases. When adding the green agent to the map, driving away from the white and red agents, we observe that the risk in the area between three agents significantly increases. This makes sense as compared to the two-agent plot: if we put an agent at position (25,25), having the green agent on the side definitely brings more risk than without the green agent, even though it is driving away. In the last figure, a new agent is added with a smaller CBF parameter γ, indicating the agent behaves more conservatively with caution, and that's why the size of its collision zone is larger than other agents. Considering white and red agents together, the risks posed by the green and yellow agents are different, even though they share the same speed, the same distance away from white and red agents, and the same safety radius. In this way, we show the proposed risk evaluation framework is generally applicable, as it embeds different factors including agent position, motion, safety radius, and behavior aggressiveness into one unified framework, thanks to the expressivity in the nature of CBF. In the following section, we choose CBF-based controllers as an example to demonstrate how to translate the centralized safe control into a decen-

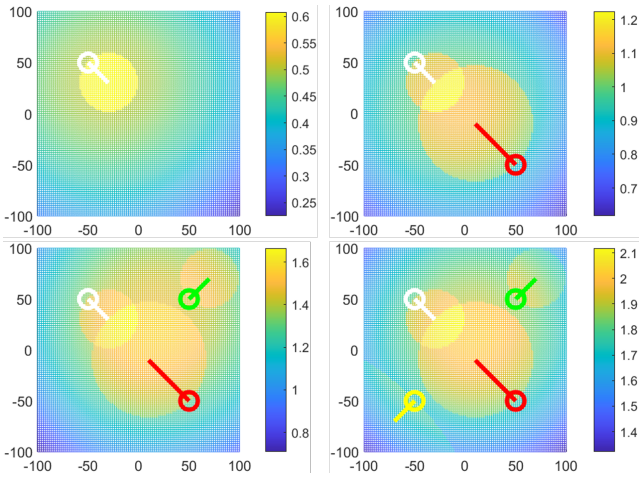


Figure 3.2: Control Barrier Function-inspired Risk Map (CBF-RM) showing how risk is aggregated when considering different numbers of agents in the system with various states and specifications. Agents are marked with different colors of circle and the lines pointing from circles represent agent velocity. The longer the line is, the larger velocity the agent is moving with.

tralized setting, by dynamically allocating responsibility shares for pairwise agents based on CBF-inspired risk measurement.

Chapter 4

Risk-Aware CBF: Decentralized Policy in Action Space

4.1 Problem Formulation

We proved in our previous work [13] that we are able to compose formally provable decentralized CBF-based controllers by assigning responsibility shares ω to any pairwise agents i and j, if we have known information about agent personalities ϕ, where $\omega_i = cos^2(\phi_i)$. In doing so, we partition the admissible control space in the centralized constraint based on the agent personalities. However, without the known information of agent personalities, now the question is, how can we assign such identities to agents, e.g., be a cautious agent or aggressive agent, in other words, how to allocate the responsibility shares ω between pairwise agents based on the proposed risk measurement under motion uncertainty.

For any pairwise agents i and j, the centralized CBF-based safety constraint over agent velocity $u_i, u_j \in \mathcal{R}^2$ is in the linear form of:

$$\mathcal{B}(x_i, x_j) = \{u_i \in \mathcal{U}_i, u_j \in \mathcal{U}_j : A_i(u_i - u_j) \leq b\} \tag{4.1}$$

where $A = -2(x_i - x_j)$ and $b = \gamma h(x_i, h_j) + 2 \cdot \text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j))$.

Theorem 2. *In a multi-agent system, agent safety during an interaction is formally guaranteed at a confidence level α, if for any pair of agents i and j, agent i takes the **Pairwise Responsibility Weight** $\omega_i = \frac{R_j}{R_i+R_j}$, so that the admissible control space in a centralized system in Eq. 4.1 is converted to Eq. 4.2:*

$$\begin{aligned} \mathcal{B}(x) =& \{u_i \in \mathcal{U}_i : A_i u_i \leq \omega_i b_i, A_i = -2(x_i - x_j)^T \in \mathbb{R}^{1\times 2}, \\ & b_i = \gamma h(x_i, x_j) + 2 \cdot \textit{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) \in \mathbb{R}\} \end{aligned}. \tag{4.2}$$

The main idea behind the weight design of ω_i is to compare the relative level of risk each set of pairwise agents is exposed to, so that the agent with lower risk can enjoy a wider admissible control space by taking a larger responsibility portion, compared to the agent with higher risk, which can be understood as its already being in a not-that-safe situation, and thus preferably only proceeding with caution with a tighter safety constraint. This design also reflects the idea of taking the neighbors of your neighbors into account, as for agent i's weight calculation, R_j embeds the information of constraints posed on agent j by its own neighbors.

Therefore, for any pair of agents in multi-agent interaction, the **risk-aware CBF-based decentralized safe controller** is formulated as a quadratic program:

$$\begin{aligned} & \min_{u_i \in \mathcal{U}_i} ||u_i - \bar{u}_i||^2 \\ s.t \quad & u_{min} \leq u_i \leq u_{max} \\ & A_i u_i \leq \omega_i b_i, \quad A_i = -2(x_i - x_j)^T, \omega_i = \frac{R_j}{R_i + R_j} \\ & b_i = \gamma h(x_i, x_j) + 2 \cdot \text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) \end{aligned} \tag{4.3}$$

where $\bar{u}_i \in \mathbb{R}^2$ is the nominal controller input, assumed to be computed by a higher-level task-related planner, for example, a behavior planner. u_{min} and u_{max} are the minimum and maximum allowed velocity. The objective function represents the goal of minimum deviation control, and the problem can be easily solved by a standard QP solver.

Remark 1. *(Solution feasibility) In the presense of bounded input constraint, the above quadratic problem may become infeasible. This can be addressed by co-optimizing parameter γ to enhance the non-emptiness of the solution set with bounded input constraint and CBF safety constraint [11], or by enforcing an additional CBF constraint which characterizes the sufficient condition of solution feasibility [29].*

Remark 2. *(Long-term safety) Despite the fact that the proposed method is a step-wise optimization, the long-term probabilistic safety along an entire time horizon $[0, \tau]$ can be guaranteed if for any pairwise agents, $x_i, x_j \in \mathcal{H}_{ij}(x)$ and $u_i, u_j \in \mathcal{B}(x)$ for all $t \in [0, \tau]$.*

4.2 Theoretical Analysis on Safety Guarantees

In this section, detailed proof is provided to show that the proposed framework in Th. 2 is the sufficient condition of formally provable safety guarantees. In other words, we aim to show that for any pair of agents i and j, Th. 2 ensures the agents will not collide.

Proof. **Step 1:** By applying Th. 2 to agent j, we get the safety constraint over u_j as:

$$\begin{aligned} A_j u_j \leq \omega_j b_j, \quad A_j = -2(x_j - x_i)^T \\ b_j = \gamma h(x_j, x_i) + 2 \cdot \text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) \end{aligned} \tag{4.4}$$

We know $A_j = -2(x_j - x_i)^T = 2(x_i - x_j)^T$. The summation of the left hand sides of the inequalities in Eq. 4.3 and Eq. 4.4 is:

$$A_i u_i + A_j u_j = -2(x_i - x_j)^T u_i + 2(x_i - x_j)^T u_j \tag{4.5}$$

which equals the left hand side of the inequality in Eq. 4.1.

Step 2: By Eq. 3.6, we have $h(x_j, x_i) = h(x_i, x_j)$, $2 \cdot \text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) = 2 \cdot \text{CVaR}_\alpha((x_j - x_i)^T(\epsilon_j - \epsilon_i))$, and therefore $b_i = b_j$. Now we sum the right hand sides of the two inequalities and get:

$$\begin{aligned} &\omega_i b_i + \omega_j b_j = (\omega_i + \omega_j) b_i \\ &= (\frac{R_i}{R_i + R_j} + \frac{R_j}{R_i + R_j}) * (\gamma h(x_i, x_j) \\ &+ 2\text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) \\ &= \gamma h(x_i, x_j) + 2 \cdot \text{CVaR}_\alpha((x_i - x_j)^T(\epsilon_i - \epsilon_j)) \end{aligned} \tag{4.6}$$

which equals the right hand side of Eq. 4.1. Thus the proof is concluded that the proposed risk-aware decentralized CBF in Th. 2 provides sufficient conditions to ensure formal safety guarantees. □

Next, the algorithm of the proposed framework is presented in Algorithm 1. The superscript 0 represents the initial condition at $t = 0$, and the superscript t represents variables at timestep t. At each timestep, we consider all pairwise agents for decentralized safety constraint composition. For every pairwise agent i and j, the pairwise responsibility shares ω_i and ω_j are calculated based on the risks level two agents are exposed to. By allocating the responsibility shares dynamically, we allow the agent with lower risk to enjoy more freedom with a wider admissible control space, and constrain the motion of the agent at high risk with a tighter bound to force it to proceed conservatively with caution. The proposed framework scales up well with a larger number of agents and is highly generally applicable to other real-time robotics applications.

4.3 Simulations & Discussion

In this section, we demonstrate the validity and effectiveness of our proposed risk-aware CBF-based decentralized controller from the following four aspects: 1) safety,

Algorithm 1 Risk-aware CBF-based Decentralized Safe Control Framework

input: $x^0_{1,\dots,N}, \bar{u}^t_{1,\dots,N}, u^t_{1,\dots,N}, \gamma, R_{safe}, \epsilon_{1,\dots,N}$
output: $u^t_{1,\dots,N}$
  for $t = 1 : T$ **do**
    for $i = 1 : N$ **do**
      for $j = 1 : N$ except i **do**
        Perform risk evaluation for agent i and j (Eq. 3.8)
        Compute Pairwise Responsibility Share ω_i (Th. 2)
        Calculate A^t_i, b^t_i for decentralized safety constraint composition (Eq. 4.2)
        Stack A^t_i, b^t_i for all surrounding agent j
      end for
      Minimum Deviation Control: $\min_{u^t_i} ||u^t_i - \bar{u}^t_i||^2$
      s.t $\quad u^t_i \in [u_{min}, u_{max}], \quad A^t_i u^t_i \leq \omega_i b^t_i$
    end for
  end for

2) interpretability, 3) implicit coordination, and 4) task efficiency through two examples.

Example 1: Ramp Merging in Autonomous Driving

The proposed risk-aware CBF-based decentralized controller is deployed on three autonomous vehicles in a ramp merging scenario where no over-the-air communication is available, shown on the left in Fig. 4.1. With Vehicle 1 and 3 on the ramp and Vehicle 2 on the main road, the goal for each vehicle is to pass the merging point as quickly as possible while avoiding collisions with each other. For simplicity, we use V1, V2, and V3 to refer to the three vehicles. The vehicle dynamics are described by double integrators as in [11]: $\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0_{2\times2} & I_{2\times2} \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0_{2\times2} & I_{2\times2} \\ I_{2\times2} & 0_{2\times2} \end{bmatrix} \begin{bmatrix} u \\ \epsilon \end{bmatrix}$, where $x \in \mathbb{R}^2, v \in \mathbb{R}^2$ are the position and velocity of each vehicle. $u \in \mathbb{R}^2$ represents the acceleration control input that is being optimized (Algorithm 1). $\epsilon \in \mathbb{R}^2$ is a random Gaussian variable with known distribution, representing the motion uncertainty in agent velocity.

Safety Validation

To validate the safety performance of our proposed method, we set the confidence level $\eta = 99.9\%$. 50 trials are conducted with randomized initial configurations, including vehicle position, velocity and acceleration. The Euclidean distance between all the $50 \times 3 = 150$ pairwise relationships over time is recorded in Fig. 4.2. The

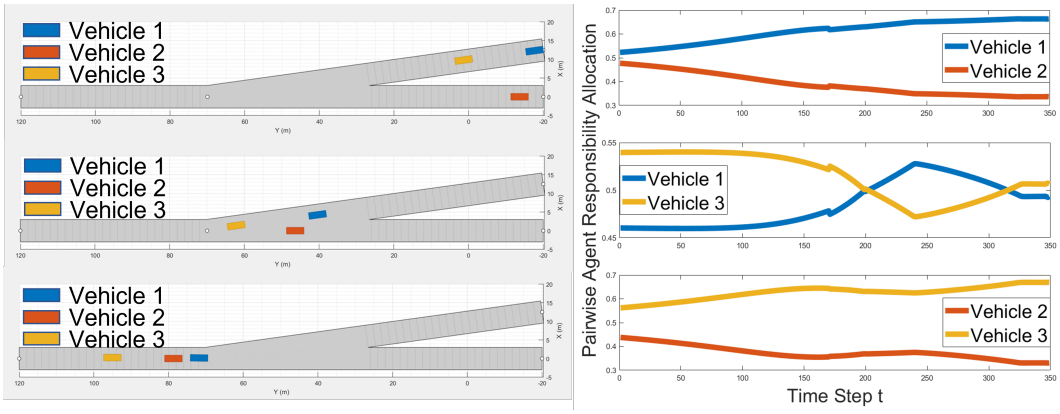


Figure 4.1: Dynamic responsibility allocation for pairwise agents in a ramp merging scenario, considering the risk from the potential collision under agent motion uncertainty in multi-agent interaction. The higher risk to which the agent is exposed, the less responsibility share is allocated to it, meaning the agent motion is more tightly constrained and forced to proceed with caution, compared to the agent facing lower risk.

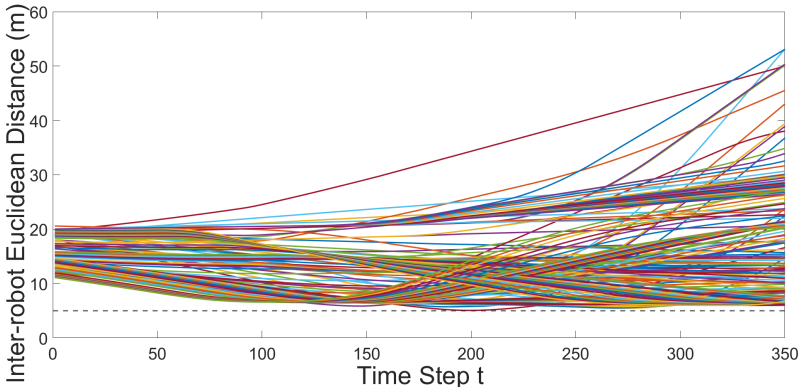


Figure 4.2: Euclidean distance between three pariwise vehicles in 50 trials over time with randomized initial configurations in the ramp merging scenario.

grey dashed line represents the safety margin $R_{safe} = 5$ m. It is observed that the inter-vehicle distance in all trials is staying above the safety margin, meaning that the safety requirement is being satisfied and no collision happens.

Physical Interpretation

One of the advantages of the proposed method is its yielding a measure of relative safety and adaptive conservativeness that is interpretable. The right plot in Fig. 4.1 provides an example in which V2 accelerates to merge into the gap between V1 and V3. The figure depicts how the three vehicles adapt their behavior conservativeness based on the relative difference of risk they receive in the interaction.

The situations in the top subplot and the bottom one are similar. Taking the top subplot as an example, when analyzing the responsibility shares between V1 and V2, the presence of V3 brings an increasingly higher risk to V2 while it's merging into the gap compared to V1. Therefore, V2 is assigned a continuously decaying responsibility share compared to V1, meaning its motion is constrained more tightly.

This aligns with our expectation that since V2 is on the main road at first with no other vehicles in the front or back, it enjoys a lot of freedom. However, as V2 completes the merge, it has to be very careful in the action it chooses to avoid colliding with V1 or V3.

In the middle subplot, we observe that the responsibility shares between V1 and V3 experience two swaps. The first swap happens when V2 is about to merge into the gap. V2's merge brings V3 higher risk compared to V1, as the risk V2 generated is highest along the direction it moves. Therefore, it's not surprising that the two vehicles' responsibility shares swap around $t = 200$. The increasing pressure on V3 posed by V2 makes V3 keep accelerating as its reaction, trying to maintain a further distance from V2. Then with the increased distance between V2 and V3, the relative risk V3 receives decreases significantly compared to V1, causing the second swap.

Example 2: Multi-agent Position Swapping Task

In this example, we have six agents in total with the task of swapping positions with each other. All agents aim to safely navigate to their goal locations, employing a move-to-goal controller $\bar{u} = -k \cdot (x - x_{target})$, where $k \in \mathbb{R}^{>0}$, and $x_{target} \in \mathbb{R}^2$ is the goal position of each agent. A right-hand heuristic rule is used for deadlock resolution, as in [19]. We compare our proposed method with 1) a centralized control baseline approach and 2) a non-risk-aware decentralized baseline approach.

Comparison with Centralized Control

In this section, we compare our proposed decentralized CBF-based control approach with the centralized CBF-based control approach. In the centralized CBF-based control approach, the problem is formulated as a single optimization, trying to calculate the optimal control for all six agents at the same time. The objective is to minimize the overall control deviation from the nominal controller of all agents while respecting all the pairwise safety constraints in Eq. 4.1.

The average control deviation among the agents during the whole task is depicted in Fig. 4.3. The orange line and the green line represent the average control deviation from the nominal controller using our proposed decentralized CBF-based control and the centralized CBF-based control respectively. The shaded regions represent the maximum and minimum individual deviation over time. It is observed that although the maximum deviation of our decentralized method is greater than that of the centralized method, the overall duration of control deviation is about 44% shorter than the centralized one. This means with our method, less intervention is applied to the agents to complete the task safely. Our proposed method also completed the task faster than the centralized method. Their time of task completion is 402s and 617s respectively.

Through this comparison, we can see that in the absence of an explicit global

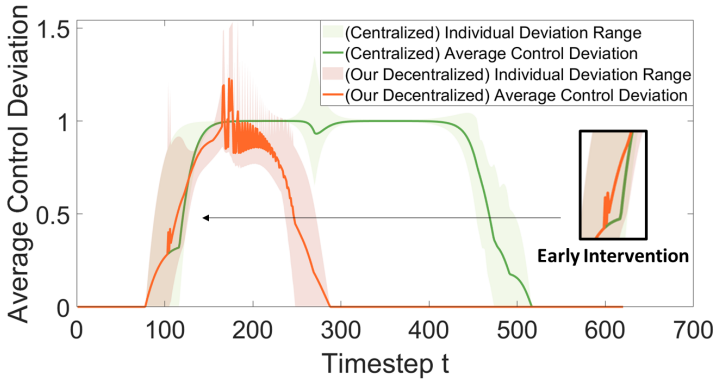


Figure 4.3: Robot control deviation comparison of our proposed decentralized control with centralized control.

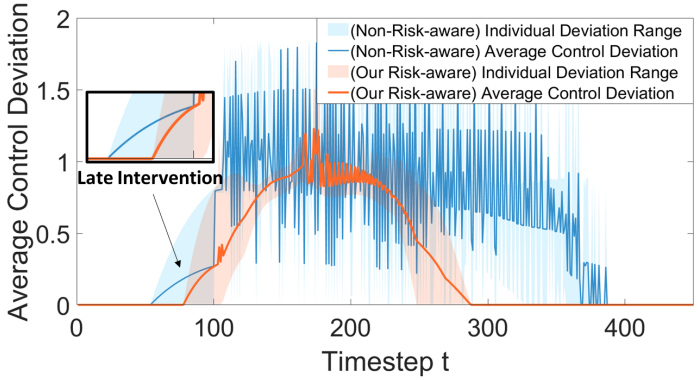


Figure 4.4: Robot control deviation comparison of our proposed risk-aware decentralized control and non-risk-aware decentralized control in robot control deviation.

coordinator, the dynamic responsibility allocation process in the proposed method plays the role of an implicit coordinator among agents. This makes early intervention possible (enlarged in the figure), forcing robots with higher risk to take precautionary measures even when agents are not close enough to collide immediately. This results in heterogeneous behavior among agents that helps to avoid possible deadlock situations and complete the task faster.

Comparison with Non-risk-aware Decentralized Control

We compare our proposed risk-aware decentralized CBF-based control approach with the non-risk-aware decentralized CBF-approach in terms of average control deviation from the nominal controller, the task efficiency as well as the agent behavior observations. The non-risk-aware approach is implemented with the same CBF-based decentralized controller, but with fixed and equal responsibility shares among agents, representing their non-changeable behavior conservativeness regardless of the amount of risk agents receive.

The average control deviation during the whole task is recorded in Fig. 4.4. It is observed that our proposed method has a lower maximum individual control devi-

ation and a shorter overall duration of intervention compared to the non-risk-aware method. In the non-risk-aware method, all agents are assigned fixed and equal responsibility shares, meaning they all have the same tight control constraint, no matter whether the agent is in a risky or extremely safe situation. This leads to unnecessary conservative behavior, for instance, agents have to decelerate significantly even when they are very far away from each other. Thanks to the risk-aware responsibility allocation in our method, late intervention (enlarged in the figure) ensures that nominal controllers are only revised when necessary. Collective safety is achieved at a lower cost of sacrificing control optimality with our method.

Next, comparisons of task efficiency and agent behavior observations are shown in Fig. 4.5 and Fig. 4.6. It is shown that our proposed method improves the overall task efficiency by 4.7% with less time spent.

To better analyze the difference in agent behavior in the two methods, we use the proposed CBF-RM for visualization. Since the initial and final agent configurations are the same, the first two and last two subplots in the two cases are almost the same. Agents come closest to one another in both cases starting in the third subplot, and start to form a rotational formation due to the right-hand heuristic for deadlock resolution. In the baseline method, from $t = 241$ to $t = 268$, since all agents are assigned equal responsibility shares, agents 3 and 6 surrounded in the middle are not additionally constrained compared to others and therefore try to break the formation and escape from being surrounded in the middle. On the other hand, at the same time in our proposed approach, since agents 3 and 6 are recognized as the agents experiencing the highest risk level, they are automatically assigned very small responsibility shares, resulting in a very limited admissible control space. Therefore, they have no choice but to stop where they are and wait for other agents to complete the formation rotation. Surprisingly, having the riskiest agents stop and wait in our method actually brings higher overall task efficiency, as tending to escape from where they are in the baseline method just makes things worse and prevents the agents with higher flexibility in motion from completing their job smoothly. Therefore from this point of view, our proposed method is a promising way to configure heterogeneous robot teams, as the way it assigns responsibility shares implicitly encourages agents to be more cooperative instead of competitive when facing conflict.

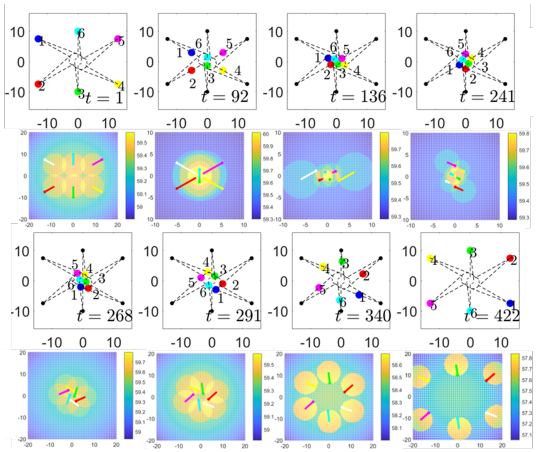


Figure 4.5: Multi-agent position swapping game with fixed and equal responsibility allocation (Baseline method).

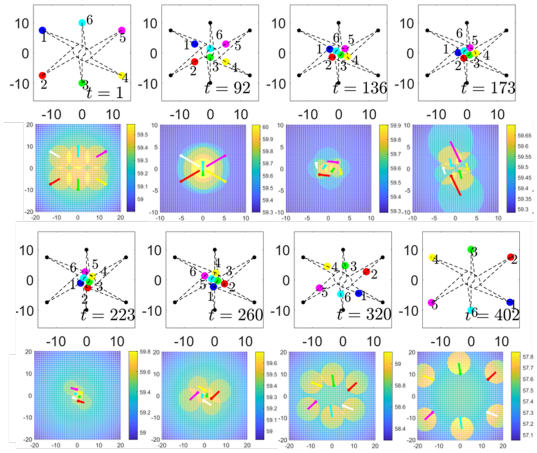


Figure 4.6: Multi-agent position swapping game with dynamic responsibility allocation (proposed method).

Chapter 5

Risk-Aware WBVC: Decentralized Policy in State Space

5.1 Background on Weighted Buffered Voronoi Cell

[*Summarized from* [19]] Consider a bounded, convex environment $Q \subset \mathbb{R}^n$, with individual points $q \in Q$. Within the environment, there are N agents, with the positions of each agent denoted $p_i \in Q$ for $i = \{1, ..., N\}$. Each agent is assumed to have integrator dynamics,

$$\dot{p_i} = u_i \tag{5.1}$$

where u_i is the nominal control input for agent i. The standard Voronoi Partition is defined as:

$$V_i = \{q \in Q | \|q - p_i\|^2 \leq \|q - p_j\|^2, \forall j \neq i, \quad i, j \leq N\} \tag{5.2}$$

where V_i denotes the cell of each agent i. Boundaries between cells are determined based on the Euclidean distance between q and the agents. Therefore, the boundary perpendicularly and equally divides the space between the neighboring agents, leaving them the same distances to the shared boundary.

The Weighted Voronoi Partition is defined as:

$$V_i = \{q \in Q | \|q - p_i\|^2 - \omega_i \leq \|q - p_j\|^2 - \omega_j\} \tag{5.3}$$

where ω_i and ω_j are the weights of each agent, and the Voronoi boundary is moved towards the agent with the larger weight.

The Buffered Voronoi Partition is defined as:

$$V_i = \{q \in Q | \|q - p_i\|^2 \leq \|q - p_j\|^2 - \omega_{ij}\} \tag{5.4}$$

where ω_{ij} defines the cell weightings between agent i and neighbors $j \in \mathcal{N}_i$, creating gaps between agent boundaries as a safety radius buffer. Buffered Voronoi

Cells are usually used to guarantee agents will not collide when situated on their boundaries [3, 30].

5.2 Risk-aware Weighted Buffered Voronoi Cells

To incorporate the risk measurement information in the Voronoi Cells partition, we extend the Control Barrier Function-inspired risk measurement to any individual point q in the state space being evaluated, by assuming that it is occupied by a static agent with zero velocity. We adopt the same calculation of the aforementioned risk assessment, except that we redefine our pairwise safety loss function $L_{ij}(p, u) \in \mathbb{R}$ as:

$$L_{ij}(p, u) = -\dot{h}_{ij}(p, u) - \gamma h_{ij}(p) - c \tag{5.5}$$

and the aggregated risk agent i faces posed by surrounding agents $R_i \in \mathbb{R}$ as:

$$R_i = \phi(\sum_{j=1}^{N} L_{ij}(p, u)), \quad \forall j \neq i \tag{5.6}$$

where c as a constant offset is a very large positive value to ensure $L_{ij}(p, u)$ is always negative to prevent unintended cancel-out when being accumulated later. $\phi(\cdot) \in \mathbb{R} \mapsto \mathbb{R}$ is a mapping function that maps the value of R_i to the range $[0, 1]$. $r_{safe} = r_i + r_j$ as the required minimum pairwise safety distance considering both agent i and j's safety radius. The larger R_i is, the more likely a safety violation is to occur. In this way, Eq. 5.5 and 5.6 can be generalized from the original agent-to-agent risk measurement to agent-to-point risk measurement.

Definition 3. *In a bounded, convex environment $Q \in \mathbb{R}^n$, for any individual point $q \in Q$ and each agent i with $p_i \in Q$ for $i = \{1, ..., N\}$, the cell V_i of agent i is a* ***Risk-aware Voronoi Cell****, if*

$$V_i = \{q \in Q | \|L_{qi}\| \leq \|L_{qj}\|, \forall j \neq i, \quad i, j \leq N\} \tag{5.7}$$

where $L_{(\cdot)}$ is the pairwise safety loss function calculated in Eq.5.5.

Different from traditional Voronoi Cells, we use agent-to-point safety loss function $L_{(\cdot)}$ instead of point-to-point Euclidean Distance, to partition the state space based on the amount of risk agents i and j generate on the individual point q. Therefore, the agent which brings the surrounding environment higher risk has a larger cell.

Definition 4. *In a bounded, convex environment $Q \in \mathbb{R}^n$, for any individual point q with $p_q \in Q$ and each agent i with $p_i \in Q$ for $i = \{1, ..., N\}$, the cell W_i of agent i is a **Risk-aware Weighted Voronoi Cell**, if*

$$\begin{aligned} W_i &= \{q \in Q | \|L_{qi}\| - \omega_i \leq \|L_{qj}\| - \omega_j, \quad \forall j \neq i, \quad i, j \leq N\} \\ \omega_i &= \frac{1}{2}(p_i - p_j)^T(u_i + u_j) + \frac{1}{4}(R_i - R_j) \cdot \|(p_i - p_j)^T(u_i - u_j)\| \\ \omega_j &= -\omega_i \end{aligned} \tag{5.8}$$

where $L_{(\cdot)}$ is the pairwise safety loss function calculated in Eq.5.5 and $R_{(\cdot)}$ is the accumulated risk over the individual agents calculated in Eq.5.6. $p_{(\cdot)}$ and $u_{(\cdot)}$ are agent position and velocity. $\gamma \in \mathbb{R}$ is the CBF design parameter.

By embedding CBF-inspired accumulated risk measurement into the weight design, the direction of the weight bias is decided based on the relative accumulated risk agent i and j received. Compared to agent j, the larger the relative accumulated risk agent i receives from the rest of the group, the larger weight ω_i it has, and the further the boundary between agent i and j is pushed towards j. To take agent safety radius into account, a buffer is added to the cell boundaries to ensure safety while agents are situated on the boundaries:

Definition 5. *In a bounded, convex environment $Q \in \mathbb{R}^n$, for any individual point q with $p_q \in Q$ and each agent i with $p_i \in Q$ for $i = \{1, ..., N\}$, the cell W_i of agent i is a **Risk-aware Weighted Buffered Voronoi Cell(Risk-aware WBVC)**, if*

$$\begin{aligned} W_i &= \{q \in Q | \|L_{qi}\| \leq \|L_{qj}\| - \omega_{ij}, \quad \forall j \neq i, \quad i, j \leq N\} \\ \omega_{ij} &= \|r_i + r_j\| \cdot \|\gamma(p_i - p_j) + (u_i - u_j)\| - (p_i - p_j)^T(u_i + u_j) \\ &+ [1 - \frac{1}{2}(R_i - R_j)] \cdot \|(p_i - p_j)^T(u_i - u_j)\| \end{aligned} \tag{5.9}$$

where $L_{(\cdot)}$ is the pairwise safety loss function calculated in Eq.5.5 and $R_{(\cdot)}$ is the accumulated risk over the individual agents calculated in Eq.5.6. $r_{(\cdot)}$ is agent safety radius, $p_{(\cdot)}$ and $u_{(\cdot)}$ are agent position and velocity. γ is the CBF design parameter.

With the weight design ω_{ij} shown in Eq. 5.9, a gap between cells with a minimum distance of $(r_i + r_j)$ is added between adjacent cells. Note that when CBF parameter $\gamma = 1$ and $u_i = u_j = 0$, the Risk-aware Weighted Buffered Voronoi Cell and weighting degrade to the traditional Voronoi Cell and Voronoi weighting [3, 30], which considers positional information only. Formal safety guarantees are provided in [10], where we prove that for any agents in Risk-aware Weighted Buffered Voronoi Cells, 1) the cell of each agent is guaranteed to be non-empty; 2) the minimum distance between any two cells is guaranteed to satisfy the collision-free configuration requirement; and 3) the cells of Risk-aware WBVC are non-overlapping.

5.3 Collision-free Navigation with Risk-aware WBVC

Now we present our algorithm for multi-agent safe navigation utilizing the proposed Risk-aware Weighted Buffered Voronoi Cells in the following algorithm. $\bar{g}_i$ is the

Algorithm 2 Safe Navigation in Risk-aware WBVC

input: Collision-free $p(t_0), u(t_0), \bar{g}, \quad \forall i \in \{1, ..., N\}$
  while $\|\bar{g}_i - p_i\| > 0, \quad \forall i \in \{1, ..., N\}$ **do**
    Calculate pairwise safety loss function L_{ij} (Eq. 5.5)
    Calculate the accumulated risk R_i (Eq. 3.8)
    Update weights ω_{ij} (Def. 5)
    Update Voronoi Cell Tessellation W_i (Def. 5)
    if $\bar{g}_i \in W_i$ **then**
      Set $g_i^* = \bar{g}_i$
    else
      Find intersection $\hat{g}_i$ of the cell boundary and the connecting line between p_i and $\bar{g}_i$
      Set $g_i^* = \hat{g}_i$
    end if
    Update control policy $\dot{p}_i = -k(g_i^* - p_i)$
  end while

desired goal position of agent i, and $\hat{g}_i$ is the projected goal position if the goal is outside the cell W_i. g_i^* is the goal position fed into a move-to-goal controller or any other kind of controller that drives the robot to move towards the goal. With our proposed algorithm, while the goal has not been reached, the agent will first calculate L_{ij} and R_i with our CBF-inspired risk measurement. By updating the Risk-aware Weighted Buffered Voronoi Cells, the heterogeneity among agents in terms of different levels of risk exposure and risk generation ability is reflected in different sizes and shapes of the cells.

5.4 Simulations & Discussion

Next, we demonstrate the validity and effectiveness of the proposed method in multi-agent swapping games. The goal is for pairwise agents to switch positions with each other without any collision. We set the agent safety radius as $0.2m$ and the right-hand heuristic is applied for deadlock resolution [17, 15]. We demonstrate our method in two scenarios of 6 and 16 agents respectively. The agents have unicycle dynamics, and we employ a nonlinear inversion method [16] to map the desired velocity to the unicycle dynamics of mobile robots without compromising the safety guarantee [8].

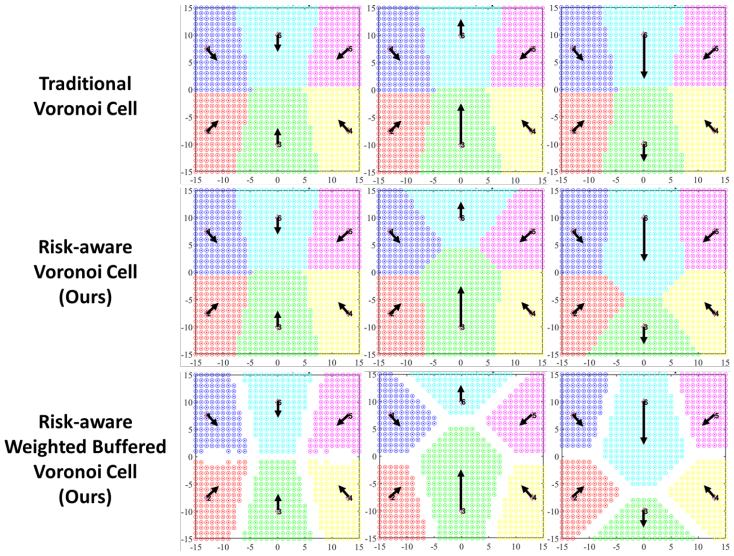


Figure 5.1: Our proposed Risk-aware Weighted Buffered Voronoi Cells v.s. traditional Voronoi Cells.

Comparison with Traditional Voronoi Cells

In the first example, we consider a multi-agent system with six robots. To demonstrate the effectiveness of Risk-aware WBVC, the difference between our proposed tessellation and the traditional Voronoi tessellation is compared on a static configuration. The comparison is shown in Fig. 5.1. The three rows correspond to traditional Voronoi Cells, Risk-aware Voronoi Cells and Risk-aware Weighted Buffered Voronoi Cells. The cells of the six agents are marked in different colors and the black arrows indicate the velocity of the agents with different magnitudes and directions. The agents in all plots share the same positions, but with different velocity settings corresponding to the three columns.

In the first row, we can see that since the tessellation is conducted solely based on positional information and does not take agent motion into account, no matter where the agents are heading and how fast they are moving, the cells are equally partitioned. In the second row, we use the proposed CBF-inspired risk measurement to take not only agent position but also motion into account for tessellation, without any weight or buffer. We can see that the sizes and shapes of the cells change when the agent takes different actions. Since we are not partitioning the space based on the point-to-agent euclidean distance, the shared cell boundaries are no longer perpendicular to the line between neighboring agents. This makes sense, since if we take agent motion into consideration, the generated risk is higher in the direction it moves than in the opposite direction, and it is therefore unnecessary to leave the same large cell space behind the agent. Our proposed Risk-aware WBVC is presented in the third row, and we observe that the lower the risk the agent poses to its

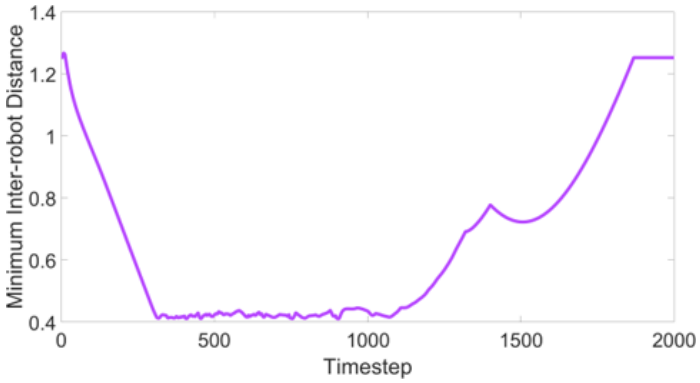


Figure 5.2: The minimum inter-robot distance among 6 robots.

surrounding environment compared to its neighboring agents, the smaller its cell is, warning it of the presence of its dangerous neighbor. The higher the risk the agent is exposed to, the larger cell it is given to grant it additional space for safe maneuvers. We further demonstrate the effectiveness of our Risk-aware WBVC in Fig. 5.5, in which the positions of all agents are fixed and the agent velocities are randomized. From the 10 randomized trials, it is shown that our proposed Risk-aware WBVC can better reflect the risk quantified via the CBF-based risk measurement, compared to the traditional Weighted Buffered Voronoi Cells that will have the same and equal cells.

Safety Performance in Collision Avoidance

Next, we demonstrate the safety performance of our proposed Risk-aware WBVC. We first show the six-agent position swapping game in Fig. 5.6. The numbers in black are agent indices and the numbers in red are target position indices, indicating to which agent they belong. We observe that the agents are able to dynamically compute and update their Voronoi Cells in a risk-aware manner. The inter-robot distance over time is recorded in Fig. 5.2. The y-axis represents the minimum inter-agent distance. Since we set the safety radius of individual agents to be $0.2m$, the minimum inter-agent distance should be larger than or equal to $0.4m$. The plot shows that the safety requirement is satisfied in the six-agent position-swapping game.

In the second example, we consider a more complicated and challenging environment with 16 robots in total. We ran 3 trials with randomly assigned pairwise counterparts for agents to swap positions. Due to the space limit, we only demonstrate the swapping process of one trial in Fig. 5.7. For better visualization, we separately show agent trajectories in Fig. 5.4. The minimum inter-robot distance over time of the 3 trials is shown in Fig. 5.3, with the green line corresponding to the demo case in Fig. 5.7. It is observed that no safety violation happens during the whole process and the agents are able to safely navigate to their target positions.

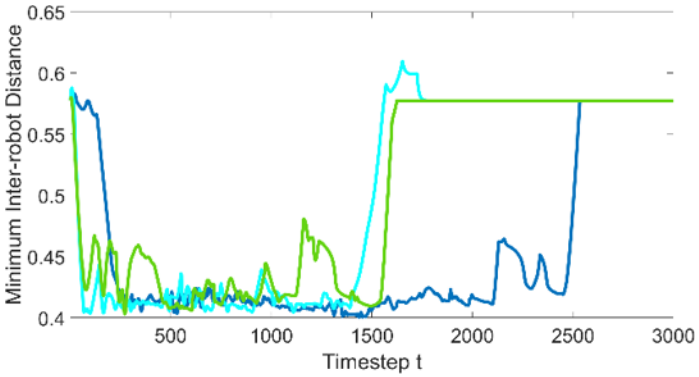


Figure 5.3: The minimum inter-robot distance among 16 robots.

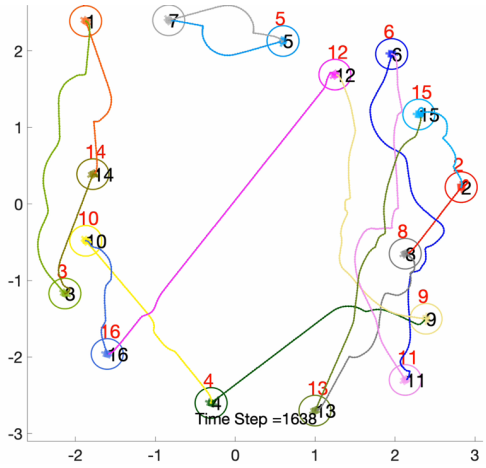


Figure 5.4: Trajectories of sixteen robots.

The risk evaluation is conducted and the risk measurement is leveraged to update the Weighted Buffered Voronoi Cells in a risk-aware manner.

Tunable Overall Risk Sensitivity

Compared to traditional Voronoi Cells that only compute cells based on positional information, by quantifying risk from a CBF perspective, our proposed Risk-aware WBVC also characterizes various levels of the overall risk sensitivity of the multi-agent system.

It is demonstrated in Fig. 5.8 that for a given set of agent positions and velocities, by tuning the CBF design parameter γ, the Risk-aware WBVC is computed with different levels of risk tolerance. The smaller γ is, the more conservative the multi-agent system behaves, and the larger γ is, the more aggressive the overall system is. For higher overall risk sensitivity, extra cautions are taken by the multi-agent system by adaptively enlarging gaps in between the cells for enhanced safety performance.

Note that different levels of system aggressiveness or conservativeness in our

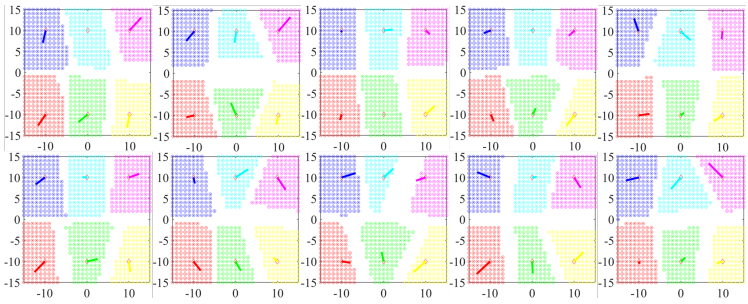


Figure 5.5: Risk-aware WBVC with randomized agent velocities at the same positions. The colorful lines represent the randomized agent velocities with different magnitudes and directions.

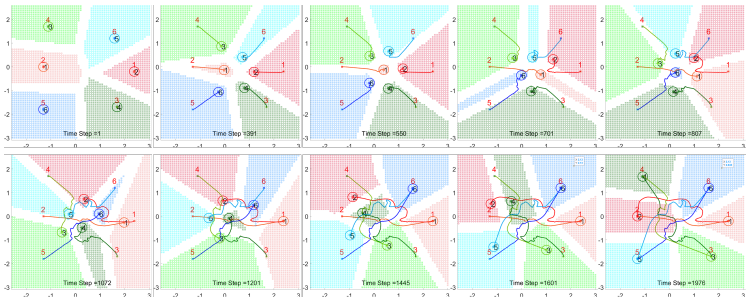


Figure 5.6: Our proposed Risk-aware Weighted Buffered Voronoi Cell in a 6-robot position swapping game.

Risk-aware WBVC are not equivalent to simply using larger or smaller buffers in traditional Weighted Buffered Voronoi Cells (WBVC). In our Risk-aware WBVC, the cells' shapes and sizes are also different, characterizing the possible heterogeneity of the overall risk sensitivity. Mathematically γ serves as a balancing parameter between the influence of the positional information and the motion information in cell partitioning, and the larger γ is, the closer Risk-aware WBVC is to traditional WBVC. As mentioned previously, our Risk-aware WBVC is a generalized version of traditional WBVC, and it degrades to traditional WBVC when $\gamma = 1$ and the agent motion information is set to zero when computing cells, which is also verified in the proof of Lemma 2 that our Risk-aware WBVC provides a tight lower bound on the minimum distance between cells compared to that of traditional WBVC.

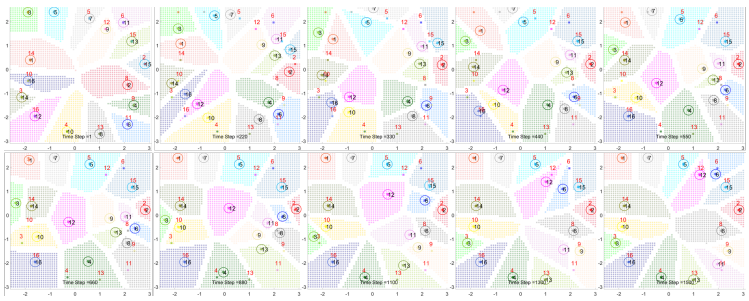


Figure 5.7: Our proposed Risk-aware Weighted Buffered Voronoi Cell in a 16-robot position-swapping game.

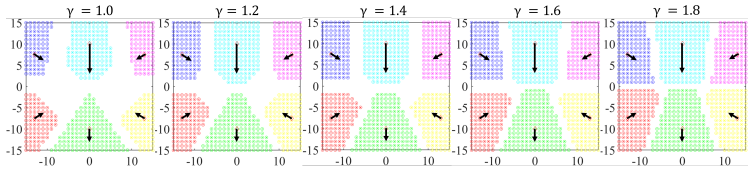


Figure 5.8: Different levels of multi-agent system's sensitivity to risk. The smaller γ is, the more sensitive the multi-agent system is to risk and therefore extra cautions are taken when computing Risk-aware WBVC.

Chapter 6

Deployment Efforts

In this project, to demonstrate the effectiveness of our proposed approach, we deploy our risk assessment module on the F1-Tenth Platform, which are 1:10 scaled autonomous race cars as shown in Fig. 6.1. To show the general applicability of our proposed risk assessment, we integrate the CBF-inspired risk assessment with the existing Model Predictive Path Integral (MPPI)-based controller of the F1-Tenth Racing cars. The overall structure of the Risk-Aware Safety Assured (RASA) MPPI control algorithm is shown in Fig. 6.2.

We run our simulations on an Intel(R) Core(TM) i5-1135G7 with 2.40GHz and 4 cores without any external modern GPU resources, showcasing our optimized computation method that utilizes Python JAX for fast parallel computing. In 6.4, we showcase the risk awareness of our proposed framework by comparing RASA-MPPI to Shield-MPPI in a high-risk scenario as in Fig. 6.3. The vehicle running the Shield-MPPI algorithm violates the safety distance R_{safe} from $t = 76$ to $t = 105$ risking a potential collision, while our proposed approach keeps the safety distance throughout the entire simulation, showcasing its risk awareness and safety assurance 6.4. Overseeing the last fast-approaching opponent vehicle on the left lane, the vehicle operating Shield-MPPI ends up in a CBF-infeasible situation at $t = 73$, the ego-vehicle running RASA-MPPI waits to let this fast-approaching opponent vehicle pass ($t = 10$ until $t = 50$) before overtaking safely ($t = 100$).

We further deploy and validate our approach in hardware testing. To show the efficiency of RASA-MPPI as well as its applicability to real-life scenarios, we run RASA-MPPI on F1Tenth racing cars in real-time. As shown in Fig. 6.5, the ego-vehicle, which starts at the bottom right corner in Fig. 6.5a executes an overtake in a safe and risk-aware manner, merging back to the original lane after surpassing the opponent vehicle while keeping a safety distance throughout the whole maneuver. In this test, the opponent car remained in a static position, because real-life perception is not the main focus of this work and we show in the simulation earlier that

Figure 6.1: F1-Tenth Racing Vehicle

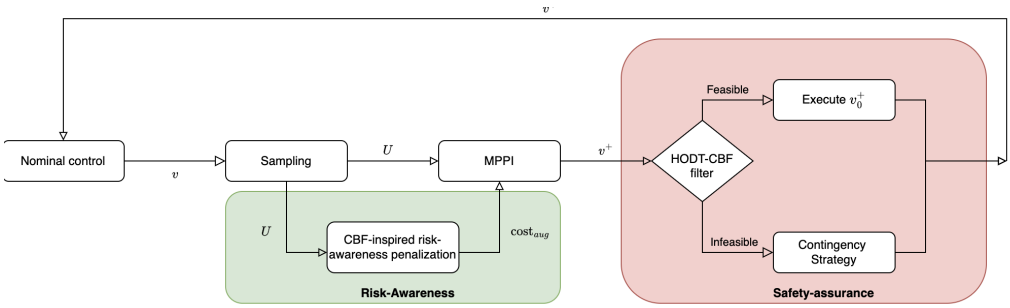
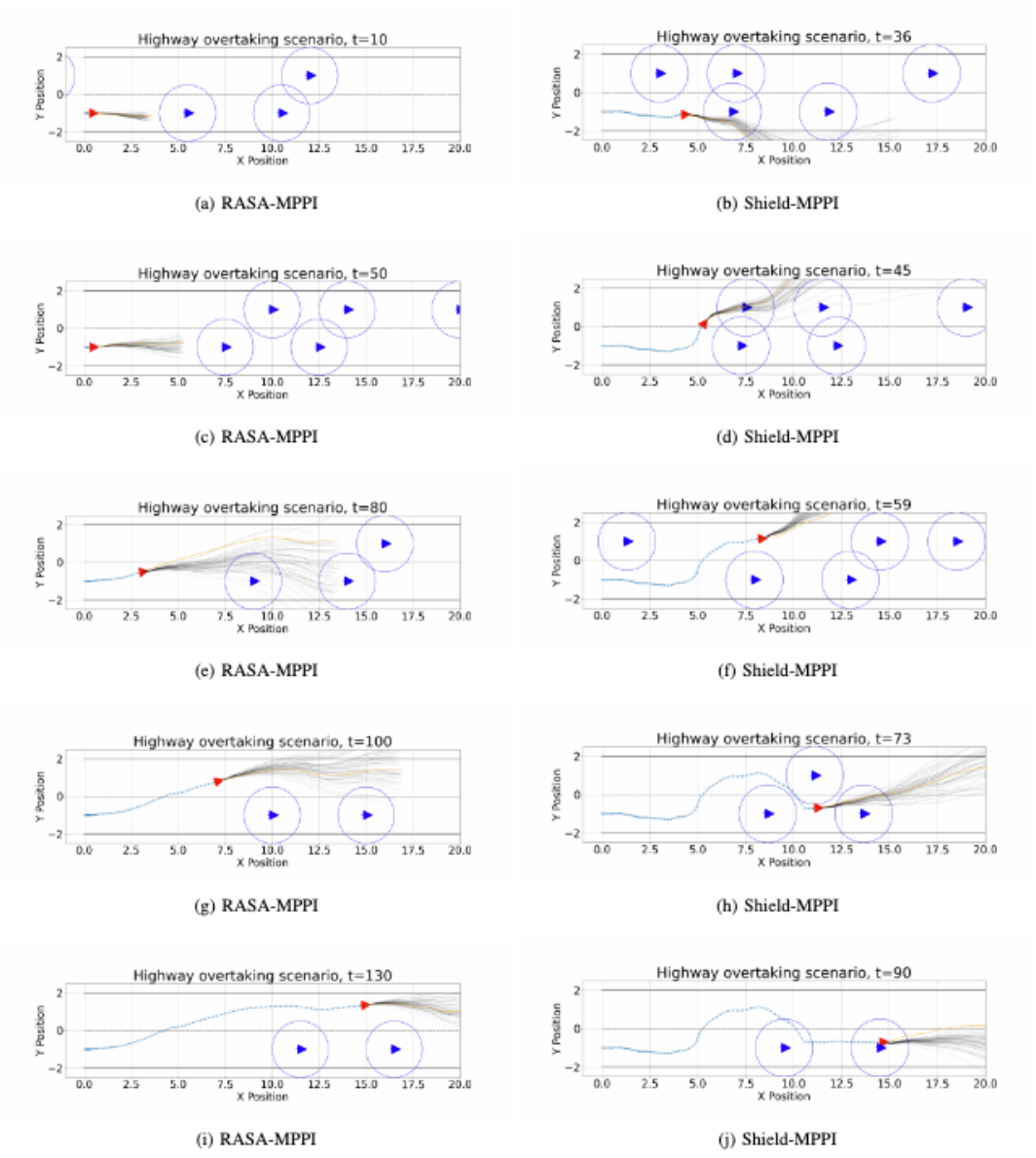


Figure 6.2: RASA-Framework

RASA-MPPI also works with non-static states of the opponent vehicles. Nevertheless, future testing could include adding computer vision to the F1Tenth vehicles to perceive the opponent vehicles and predict their behavior to validate RASA-MPPI in scenarios that are even closer to real-life scenarios.



(a) RASA-MPPI (b) Shield-MPPI

(c) RASA-MPPI (d) Shield-MPPI

(e) RASA-MPPI (f) Shield-MPPI

(g) RASA-MPPI (h) Shield-MPPI

(i) RASA-MPPI (j) Shield-MPPI

Figure 6.3: Comparison of our Risk-Aware Safety Assured-MPPI (1) and the benchmark algorithm Shield-MPPI in a highway overtaking scenario

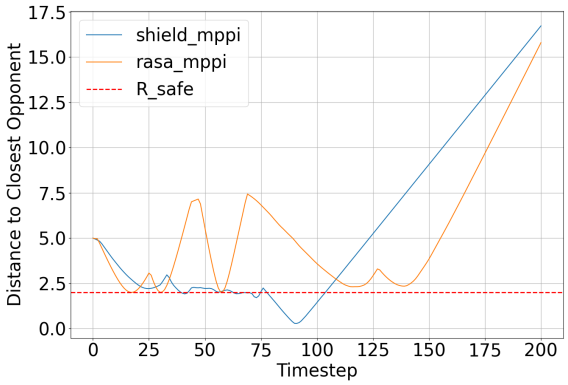
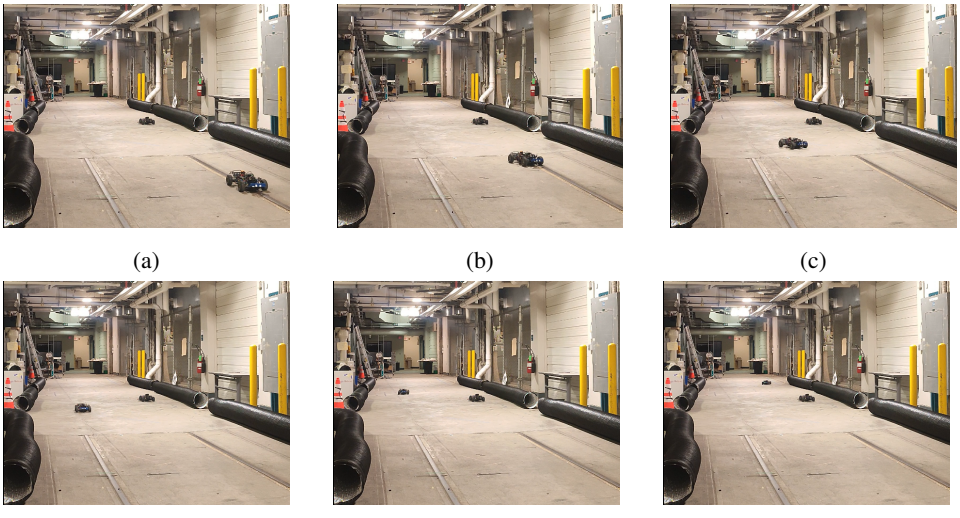


Figure 6.4: Distance to the closest opponent vehicle over time.



(a) (b) (c)

(d) (e) (f)

Figure 6.5: Real-time execution of our Risk-Aware Safety Assured-MPPI on F1Tenth vehicles. The ego-vehicle on the bottom right, starts in the same lane behind the opponent vehicle, and then completes an overtaking scenario, sticking to the reference velocity. The computation is done entirely on the Jetson Nano without connection to an external GPU.

Chapter 7

Conclusion

In this project, we proposed a CBF-inspired risk assessment toolbox, measuring the aggregated risk faced by individual agents due to multi-agent interactions, to help empower the ego robot with a comprehensive understanding of the dynamic environment it is in. We also demonstrate two complementary methodologies of embedding the resulting risk assessment into robot controller design, especially for multi-robot scenarios. We show how to utilize such risk information to perform partition in robots' joint state and action space respectively to construct robot controllers, resulting in the proposed Risk-aware WBVC and Risk-aware decentralized CBF. We further demonstrated the general applicability of our risk assessment in deployment to a physical F1-Tenth testing platform with existing MPPI-based vehicle controllers. A safe and efficient overtaking scenario is demonstrated. In the future, we are interested in extending these methods for safe navigation for self-driving cars with learning-enabled components.

Bibliography

[1] Mohamadreza Ahmadi, Xiaobin Xiong, and Aaron D Ames. Risk-averse planning via cvar barrier functions: Application to bipedal robot locomotion. *arXiv preprint arXiv:2011.01578*, 2020.

[2] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *18th European Control Conference (ECC)*, 2019.

[3] Saptarshi Bandyopadhyay, Soon-Jo Chung, and Fred Y Hadaegh. Probabilistic swarm guidance using optimal transport. In *2014 IEEE Conference on Control Applications (CCA)*, pages 498–505. IEEE, 2014.

[4] Jaskaran Grover, Yiwei Lyu, Wenhao Luo, Changliu Liu, John Dolan, and Katia Sycara. Semantically-aware pedestrian intent prediction with barrier functions and mixed-integer quadratic programming. *IFAC-PapersOnLine*, 55(41):167–174, 2022.

[5] Astghik Hakobyan, Gyeong Chan Kim, and Insoon Yang. Risk-aware motion planning and control using cvar-constrained optimization. *IEEE Robotics and Automation letters*, 4(4):3924–3931, 2019.

[6] Xin Huang, Ashkan Jasour, Matthew Deyo, Andreas Hofmann, and Brian C Williams. Hybrid risk-aware conditional planning with applications in autonomous vehicles. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 3608–3614. IEEE, 2018.

[7] Zhiyu Huang, Haochen Liu, Jingda Wu, and Chen Lv. Differentiable integrated motion prediction and planning with learnable cost function for autonomous driving. *IEEE transactions on neural networks and learning systems*, 2023.

[8] Wenhao Luo, Wen Sun, and Ashish Kapoor. Multi-robot collision avoidance under uncertainty with probabilistic safety barrier certificates. *Advances in Neural Information Processing Systems*, 33:372–383, 2020.

[9] Yiwei Lyu, John M Dolan, and Wenhao Luo. Cbf-inspired weighted buffered voronoi cells for distributed multi-agent collision avoidance. In *2023 American Control Conference (ACC)*, pages 4513–4518. IEEE, 2023.

[10] Yiwei Lyu, John M Dolan, and Wenhao Luo. Decentralized safe navigation for multi-agent systems via risk-aware weighted buffered voronoi cells. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pages 1476–1484, 2023.

[11] Yiwei Lyu, Wenhao Luo, and John M Dolan. Probabilistic safety-assured adaptive merging control for autonomous vehicles. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10764–10770. IEEE, 2021.

[12] Yiwei Lyu, Wenhao Luo, and John M Dolan. Adaptive safe merging control for heterogeneous autonomous vehicles using parametric control barrier functions. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 542–547. IEEE, 2022.

[13] Yiwei Lyu, Wenhao Luo, and John M Dolan. Responsibility-associated multi-agent collision avoidance with social preferences. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 3645–3651. IEEE, 2022.

[14] Yiwei Lyu, Wenhao Luo, and John M Dolan. Risk-aware safe control for decentralized multi-agent systems via dynamic responsibility allocation. In *2023 IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023.

[15] Ahmed Nazeem, Spyros Reveliotis, Yin Wang, and Stéphane Lafortune. Designing compact and maximally permissive deadlock avoidance policies for complex resource allocation systems through classification theory: The linear case. *IEEE Transactions on Automatic Control*, 56(8):1818–1833, 2011.

[16] Daniel Pickem, Paul Glotfelter, Li Wang, Mark Mote, Aaron Ames, Eric Feron, and Magnus Egerstedt. The robotarium: A remotely accessible swarm robotics research testbed. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1699–1706. IEEE, 2017.

[17] Alyssa Pierson and Daniela Rus. Distributed target tracking in cluttered environments with guaranteed collision avoidance. In *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 83–89. IEEE, 2017.

[18] Alyssa Pierson, Wilko Schwarting, Sertac Karaman, and Daniela Rus. Navigating congested environments with risk level sets. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5712–5719. IEEE, 2018.

[19] Alyssa Pierson, Wilko Schwarting, Sertac Karaman, and Daniela Rus. Weighted buffered voronoi cells for distributed semi-cooperative behavior. In *IEEE international conference on robotics and automation (ICRA)*, pages 5611–5617, 2020.

[20] R Tyrrell Rockafellar and Stanislav Uryasev. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 26(7):1443–1471, 2002.

[21] Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. *Active preference-based learning of reward functions*. 2017.

[22] Andrew Singletary, Mohamadreza Ahmadi, and Aaron D Ames. Safe control for nonlinear systems with stochastic uncertainty via risk control barrier functions. *arXiv preprint arXiv:2203.15892*, 2022.

[23] Angel Soriano, Enrique J Bernabeu, Angel Valera, and Marina Vallés. Multi-agent systems platform for mobile robots collision avoidance. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 320–323. Springer, 2013.

[24] Angelos Toytziaridis, Paolo Falcone, and Jonas Sjöberg. A data-driven markovian framework for multi-agent pedestrian collision risk prediction. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 777–782. IEEE, 2019.

[25] L. Wang, A. D. Ames, and M. Egerstedt. Safety barrier certificates for collisions-free multirobot systems. *IEEE Transactions on Robotics*, 33(3):661–674, 2017.

[26] L. Wang, A. D. Ames, and M. Egerstedt. Safety barrier certificates for collisions-free multirobot systems. *IEEE Transactions on Robotics*, 33(3):661–674, 2017.

[27] Li Wang, Aaron Ames, and Magnus Egerstedt. Safety barrier certificates for heterogeneous multi-robot systems. In *American Control Conference (ACC)*, pages 5213–5218, 2016.

[28] Mingyu Wang, Negar Mehr, Adrien Gaidon, and Mac Schwager. Game-theoretic planning for risk-aware interactive agents. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6998–7005. IEEE, 2020.

[29] Wei Xiao, Calin A Belta, and Christos G Cassandras. Sufficient conditions for feasibility of optimal control problems using control barrier functions. *Automatica*, 135:109960, 2022.

[30] Dingjiang Zhou, Zijian Wang, Saptarshi Bandyopadhyay, and Mac Schwager. Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells. *IEEE Robotics and Automation Letters*, 2(2), 2017.

[31] Lifeng Zhou and Pratap Tokekar. Risk-aware submodular optimization for multirobot coordination. *IEEE Transactions on Robotics*, 2022.

[32] Hai Zhu, Bruno Brito, and Javier Alonso-Mora. Decentralized probabilistic multi-robot collision avoidance using buffered uncertainty-aware voronoi cells. *Autonomous Robots*, 46(2):401–420, 2022.

1. Report No. 428	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Risk-Aware Warning and Control for Interactive Traffic Safety		**5. Report Date** July 31, 2024 **6. Performing Organization Code**
7. Author(s) John M. Dolan (PI) (https://orcid.org/0000-0003-2062-100X) Yiwei Lyu (https://orcid.org/0009-0007-0136-5369) Paul Schulte (https://orcid.org/0009-0002-0755-9352)		**8. Performing Organization Report No.**
9. Performing Organization Name and Address The Robotics Institute Carnegie Mellon University 5000 Forbes Avenue Pittsburgh, PA 15213		**10. Work Unit No.** **11. Contract or Grant No.** Federal Grant No. 69A3552344811
12. Sponsoring Agency Name and Address Safety21 University Transportation Center Carnegie Mellon University 5000 Forbes Avenue Pittsburgh, PA 15213		**13. Type of Report and Period Covered** Final Report (July 1, 2023-June 30, 2024) **14. Sponsoring Agency Code** USDOT

15. Supplementary Notes
Conducted in cooperation with the U.S. Department of Transportation.

16. Abstract
In this project, we proposed a CBF-inspired risk assessment toolbox, measuring the aggregated risk faced by individual agents due to multi-agent interactions, to help empower the ego robot with a comprehensive understanding of the dynamic environment it is in. We also demonstrate two complementary methodologies of embedding the resulting risk assessment into robot controller design, especially for multi-robot scenarios. We show how to utilize such risk information to perform partition in robots' joint state and action space respectively to construct robot controllers, resulting in the proposed Risk-aware WBVC and Risk-aware decentralized CBF. We further demonstrated the general applicability of our risk assessment in deployment to a physical F1-Tenth testing platform with existing MPPI-based vehicle controllers. A safe and efficient overtaking scenario is demonstrated. In the future, we are interested in extending these methods for safe navigation for self-driving cars with learning-enabled components.

17. Key Words	18. Distribution Statement
Autonomous driving, safety, risk measurement, Control Barrier Functions, Voronoi Cells	No restrictions.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	39	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized