DISSERTATIONES
TECHNOLOGIAE
UNIVERSITATIS
TARTUENSIS
100

BASANT SHARMA

Sample-efficient risk-aware trajectory optimization leveraging maximum mean discrepancy

DISSERTATIONES TECHNOLOGIAE UNIVERSITATIS TARTUENSIS
100

DISSERTATIONES TECHNOLOGIAE UNIVERSITATIS TARTUENSIS
100

BASANT SHARMA

Sample-efficient risk-aware trajectory optimization leveraging maximum mean discrepancy

UNIVERSITY OF TARTU
Press
1632

Institute of Technology, Faculty of Science and Technology, University of Tartu, Estonia.

The dissertation was accepted for the commencement of the degree of Doctor of Philosophy in Computer Engineering on June 30, 2026, by the Joint Council of the Computer Engineering Specialisation of the Doctoral Programme in Engineering and Technology of the University of Tartu.

Supervisor:	Arun Kumar Singh, PhD Professor of Motion Planning Institute of Technology, University of Tartu, Estonia
Reviewer:	Naveed Muhammad, PhD Associate Professor of Autonomous Driving Institute of Computer Science, University of Tartu, Estonia
Opponent:	Johannes Betz, PhD Professor in Autonomous Systems, Technical University of Munich, Germany
Commencement:	Auditorium 121, Nooruse 1, Tartu, Estonia, at 12.15 on August 12, 2026

Publication of this thesis is granted by the Institute of Technology, Faculty of Science and Technology, University of Tartu.

ISSN 2228-0855 (print)
ISBN 978-9908-57-292-5 (print)
ISSN 2806-2620 (pdf)
ISBN 978-9908-57-291-8 (PDF)

University of Tartu Press
www.tyk.ee

ABSTRACT

Motion planning under uncertainty is a fundamental challenge in robotics, since autonomous systems must make safe decisions despite perception noise, imperfect localization, stochastic dynamics, and the inherently unpredictable behavior of dynamic environments. In such settings, safety is naturally expressed in terms of risk, for example, the probability of collision or violation of safety constraints. While deterministic planning and control methods are computationally efficient, they fail to capture the stochastic nature of real-world environments. As a result, reliable motion planning under uncertainty remains a central problem in autonomous robotics.

This thesis addresses this problem through the framework of risk-aware trajectory optimization, with a particular focus on collision-risk estimation and minimization for robot navigation. Collision risk is defined as the probability of violating safety constraints due to uncertainty in ego dynamics, obstacle motion, or perception. However, for realistic uncertainty distributions, which are often non-Gaussian, multi-modal, and available only through samples from modern learning-based predictors, collision-risk estimation becomes analytically intractable and computationally expensive when approached through conventional chance-constrained formulations or large-scale sampling methods.

To overcome these limitations, this thesis introduces a sample-efficient surrogate for collision risk based on Reproducing Kernel Hilbert Space (RKHS) embeddings and the Maximum Mean Discrepancy (MMD). The proposed formulation enables principled risk estimation directly from samples, with finite-sample convergence guarantees, and without imposing restrictive assumptions on the underlying uncertainty distribution. Building on this formulation, the thesis develops sampling-based, risk-aware trajectory optimizers that explicitly minimize the proposed MMD-based risk surrogate.

A central contribution of this work is an optimization-based reduced-set strategy that selects the most informative samples from the underlying uncertainty distribution while discarding redundant ones. This significantly reduces the number of collision checks required for reliable risk estimation and improves the computational efficiency of the overall planning framework. The proposed methodology is developed and evaluated in three settings: navigation under uncertain obstacle motion, trajectory optimization under stochastic ego dynamics, and monocular vision-based navigation with a learned collision model integrated into risk-aware model predictive control.

Extensive evaluations across all three settings demonstrate that the proposed framework achieves substantially lower collision rates than existing risk surrogates while requiring fewer samples, supporting real-time deployment in uncertain real-world environments.

CONTENTS

LIST OF FIGURES

10. Validation of performance of Alg.1. Fig. (a) shows the progression of outer-level cost (4.13a) across iterations. Fig. (b) validates the efficacy of our optimal *reduced-set* selection. We sampled different random *reduced-sets* and computed the average, minimum and maximum of MMD $\|\hat{\mu}[\tau^{'}] - \hat{\mu}[\tau]\|^2_{\mathscr{H}}$. As shown in Fig. (b), our optimal *reduced-set* leads to an $\|\hat{\mu}[\tau^{'}] - \hat{\mu}[\tau]\|^2_{\mathscr{H}}$ that is very close to what can be achieved by exhaustively trying out different randomly selected *reduced-set*. Fig. (c) validates our hypothesis based on Theorem 1 and insights from [50]. As $\hat{\mu}[\tau^{'}] \rightarrow \hat{\mu}[\tau]$, the MMD $\|\hat{\mu}[\overline{f}] - \hat{\mu}[\overline{f^{'}}]\|^2_{\mathscr{H}} \rightarrow 0$. We can once again observe that our optimal *reduced-set* performs close to the best-case performance than can be hoped to be achieved by trying-out different random *reduced-sets*. Fig. (c) verifies that our optimal *reduced-set* selection ensures minimal loss in estimation of r^{emp}_{MMD} while reducing the sample size. To this end, we computed $\Delta_{r_{MMD}}$ through (4.15) which captures the estimation error between r^{emp}_{MMD} computed over a large set and the smaller *reduced-set*. As can be seen, r^{emp}_{MMD} computed over the optimal *reduced-set* samples perform close to the average performance that can be achieved by exhaustively trying out different random choices of the *reduced-set*. It should be noted that in real-world setting, exhaustive sampling is not possible and our bi-level optimization (4.13a)-(4.13c) provides a one-shot solution that is as good as exhaustive sampling in expectation. 50

11. The figures demonstrate the physical significance of the trajectory samples used to form the optimal *reduced-set*. We generate a synthetic multi-modal trajectory distribution that captures the different lane-change intents of a dynamic obstacle (orange rectangle). More importantly, due to our specific generation procedure, we know the approximate probability of each lane-change intent. When Alg.1 starts, the optimal *reduced-set* in the first iteration consists mostly of the less likely samples due to the random initialization of the Gaussian distribution from which λ is sampled. **For example in the left figure, the *reduced-set* samples are mostly formed by the left lane-change maneuvers which is less likely**. But as Alg.1 makes progress, the composition significantly changes. For example, in the right figure, the majority of the *reduced-set* samples switch from left-lane to the right-lane maneuvers. Thus, by just minimizing MMD between two sets of trajectory samples, Alg.1 is able to implicitly infer which trajectory samples are more likely and accordingly split the composition of the optimal *reduced-set* between the less and more likely samples. 51

LIST OF TABLES

LIST OF ABBREVIATIONS

Acronyms

CCO Chance Constrained Optimization. 34

CEM Cross Entropy Method. 48, 50, 58, 62

CVaR Conditional Value At Risk. 9, 26, 27, 35, 36, 41, 45, 46, 52, 57, 58, 62, 63, 65–67, 82–84

DRO Distributionally Robust Optimization. 37

KL Kullback–Leibler. 22, 44

MMD Maximum Mean Discrepancy. 5, 6, 9, 11, 12, 17, 21–24, 27, 28, 30, 31, 37, 40, 41, 44–46, 49, 52, 57–61, 63, 65, 68–70, 72, 74, 79–81, 83–86

MPC Model Predictive Control. 7, 11, 12, 17, 39, 40, 43, 56, 60, 63–67, 69, 70, 76–78, 81, 83

MPPI Model Predictive Path Integral. 36, 50, 58, 62

NLL Negative Log-Likelihood. 12, 71, 72

PRM Probabilistic Roadmap. 33

QP Quadratic Programming. 48, 62

RKHS Reproducing Kernel Hilbert Space. 5, 6, 9, 17, 21–24, 28–32, 36, 37, 41, 44–46, 58, 61, 68, 80–82, 84, 85, 88

RRT Rapidly-Exploring Random Tree. 33

SAA Sample Average Approximation. 9, 26, 27, 35, 41, 45, 46, 52, 57, 82, 84

LIST OF ORIGINAL PUBLICATIONS

Publications included in the thesis

[I] **Basant Sharma**, Aditya Sharma, K Madhava Krishna, and Arun Kumar Singh. "Hilbert space embedding-based trajectory optimization for multi-modal uncertain obstacle trajectory prediction". In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2023, pp. 7448–7455.

[II] **Basant Sharma** and Arun Kumar Singh. "MMD-OPT: Maximum Mean Discrepancy Based Sample Efficient Collision Risk Minimization for Autonomous Driving". In: *IEEE Transactions on Automation Science and Engineering* 22 (2025), pp. 19051–19068.

[III] **Basant Sharma** and Arun Kumar Singh. "Trajectory Optimization Under Stochastic Dynamics Leveraging Maximum Mean Discrepancy". In: *IEEE Robotics and Automation Letters* 10.6 (June 2025), pp. 6079–6086.

[IV] **Basant Sharma**, Prajyot Jadhav, Pranjal Paul, K.Madhava Krishna, and Arun Kumar Singh. "MonoMPC: Monocular Vision Based Navigation with Learned Collision Model and Risk-Aware Model Predictive Control". In: *IEEE Robotics and Automation Letters* (2025), pp. 1–8.

Other Publications

[V] Vikrant Dewangan, **Basant Sharma**, Tushar Choudhary, Sarthak Sharma, Aakash Aanegola, Arun K Singh, and K Madhava Krishna. "UAP-BEV: Uncertainty Aware Planning Using Bird's Eye View Generated From Surround Monocular Images". In: *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*. IEEE. 2023, pp. 1–8.

[VI] Jatan Shrestha, Simon Idoko, **Basant Sharma**, and Arun Kumar Singh. "End-to-end learning of behavioural inputs for autonomous driving in dense traffic". In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2023, pp. 10020–10027.

[VII] Fatemeh Rastgar, Houman Masnavi, **Basant Sharma**, Alvo Aabloo, Jan Swevers, and Arun Kumar Singh. "PRIEST: Projection guided sampling-based optimization for autonomous navigation". In: *IEEE Robotics and Automation Letters* 9.3 (2024), pp. 2630–2637.

[VIII] Simon Idoko, **Basant Sharma**, and Arun Kumar Singh. "Learning Sampling Distribution and Safety Filter for Autonomous Driving with VQ-VAE and Differentiable Optimization". In: *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2024.

Author's contribution to the publications

The publications listed above are numbered with Roman numerals in this section and correspond to their bibliography entries when cited elsewhere in the thesis.

In Publications I, II, and III [1, 2, 3], the author was responsible for all stages of the research, including formulating the idea and hypotheses, developing the methodology, conducting experiments, and writing the publication.
In Publication IV [4], the author was responsible for writing the formulation of the idea, coding the planner and the deep learning model as well as writing the publication.

1. INTRODUCTION

Autonomous robots operating in the real world must routinely make decisions under uncertainty. Whether navigating cluttered indoor spaces using vision-only perception, planning around obstacles whose future motion is inherently multi-modal, or executing control under imperfect localization and stochastic dynamics, the robot must ensure that its planned trajectories satisfy a variety of safety constraints, for example, collision avoidance. Classical motion planning and MPC algorithms [9, 10, 11] assume that the system state, environment, and dynamics model are all deterministic so that constraint satisfaction can be enforced pointwise in time. While this deterministic assumption enables efficient optimization, it fails to represent the true nature of real-world robotics systems, where uncertainty is pervasive and unavoidable. Uncertainty in robotic systems can stem from various sources:

- Perception uncertainty: Depth estimates from monocular vision models may be biased, noisy, or inconsistent [60, 61, 62] while LiDAR returns may be sparse or occluded.
- Prediction uncertainty: Trajectories of other agents are often multi-modal and high-dimensional, produced by black-box learned models [12, 44, 45, 46].
- Dynamics uncertainty: Process noise, imperfect modeling of vehicle dynamics, actuator variability, and localization drift introduce stochasticity in state evolution [40, 57, 47, 58, 59, 56].

Under such conditions, the constraints can no longer be treated as deterministic; instead, they must be regarded as stochastic. The thesis uses the notion of *risk* r, which naturally measures the probability of constraint violation. Minimizing the risk should yield safer trajectories and more robust controllers. To this end, a motion planning problem that accounts for both safety and performance is more appropriately formulated as a risk-aware trajectory optimization problem. This formulation provides a rigorous and principled framework for enforcing safety by minimizing risk while maintaining optimality under uncertainty.

The structure of the thesis is summarized in the roadmap below.

At a Glance

Thesis roadmap.

- **Chapter 2** - Mathematical preliminaries: risk, RKHS, MMD, reduced-set.
- **Chapter 3** - Related work and positioning.
- **Chapter 4** (Publications I+II) - Uncertain obstacle motion.
- **Chapter 5** (Publication III) - Stochastic ego dynamics.
- **Chapter 6** (Publication IV) - Monocular vision-based navigation.
- **Chapter 7** - Conclusion and future directions.

The central challenge, however, is that reliable risk estimation becomes difficult precisely in the settings where uncertainty-aware planning is most needed. Modern robotic systems increasingly rely on learned perception, neural trajectory predictors, and stochastic simulation models, all of which induce uncertainty distributions that are highly complex, often multi-modal, and frequently available only through samples. As a result, classical deterministic formulations and conventional probabilistic approximations are often inadequate for capturing the true safety implications of uncertainty. This observation motivates the focus of this thesis: developing principled and sample-efficient methods for estimating and minimizing collision risk in realistic robotic systems.

1.1. Core Challenges of Estimating Collision Risk

A central component of risk-aware trajectory optimization is collision avoidance. The robot faces a dual objective: it must achieve its navigation goals, such as trajectory tracking, smoothness, or efficiency, while simultaneously ensuring safe operation by avoiding collisions with obstacles in its environment. Thus, in this thesis, we focus on the risk associated with collision, also known as *collision risk*. However, the approach is valid for any notion of risk modeled as the violation of some specified state-dependent constraints. The concepts introduced in this section are motivated by an example from the autonomous driving domain.

1.1.1. Collision Risk in the Deterministic Setting

In the deterministic setting, both the ego vehicle motion and the obstacle trajectories are assumed to be perfectly known and free from uncertainty (Fig. 1).

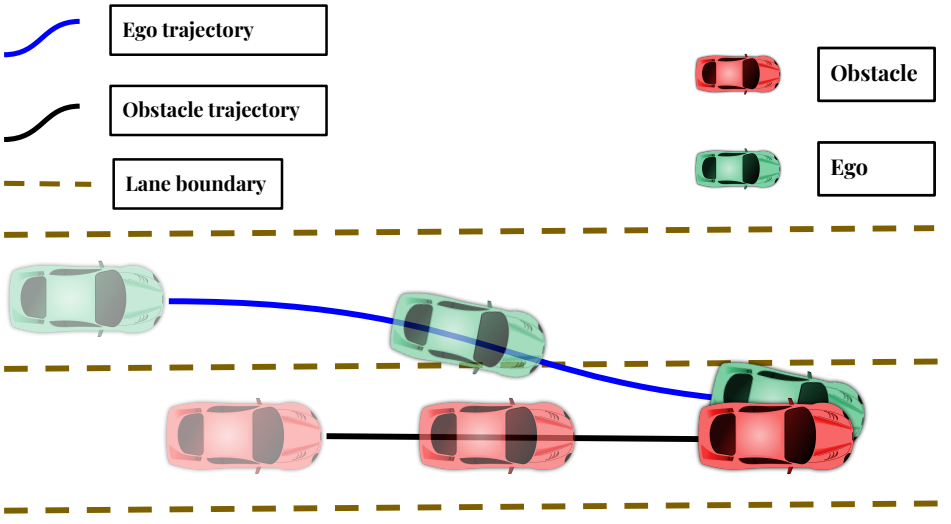


Figure 1: Deterministic collision scenario: the ego vehicle and obstacle trajectories are fully known.

Under these assumptions, collision checking reduces to a deterministic feasibility problem. Let $\mathbf{x} = \{\mathbf{x}_k\}_{k=1}^{T}$ denote the ego trajectory and $\tau = \{\tau_k\}_{k=1}^{T}$ denote the

obstacle trajectory over a finite horizon T. A per-step collision constraint can be expressed as

$$h_k(\mathbf{x}_k, \tau_k) \leq 0, \quad \text{(per-step collision constraint)} \tag{1.1a}$$
$$h(\mathbf{x}, \tau) = \max_k h_k(\mathbf{x}_k, \tau_k), \tag{1.1b}$$
$$h(\mathbf{x}, \tau) \leq 0, \quad \text{(worst-case collision constraint)} \tag{1.1c}$$
$$r = \max(0, h(\mathbf{x}, \tau)), \quad \text{(collision risk over the horizon).} \tag{1.1d}$$

Here, the scalar function $h_k(\cdot)$ measures the signed collision margin at each time step, such that negative values indicate collision-free configurations. Throughout the thesis, the symbol r denotes the deterministic residual $\max(0, h)$ in the deterministic setting and the probability of constraint violation $P(h > 0)$ in the stochastic setting; the intended meaning is clear from context. The worst-case constraint aggregates these per-step evaluations over the entire trajectory, and the resulting collision risk r penalizes any violation of the collision-free condition. In this deterministic formulation, a trajectory is considered safe if and only if $h(\mathbf{x}, \tau) \leq 0$, yielding strictly zero collision risk.

1.1.2. Collision Risk in Stochastic Setting

In the stochastic setting, either $\mathbf{x}$ or τ (or both) may be random variables (Figs. 2, 3).

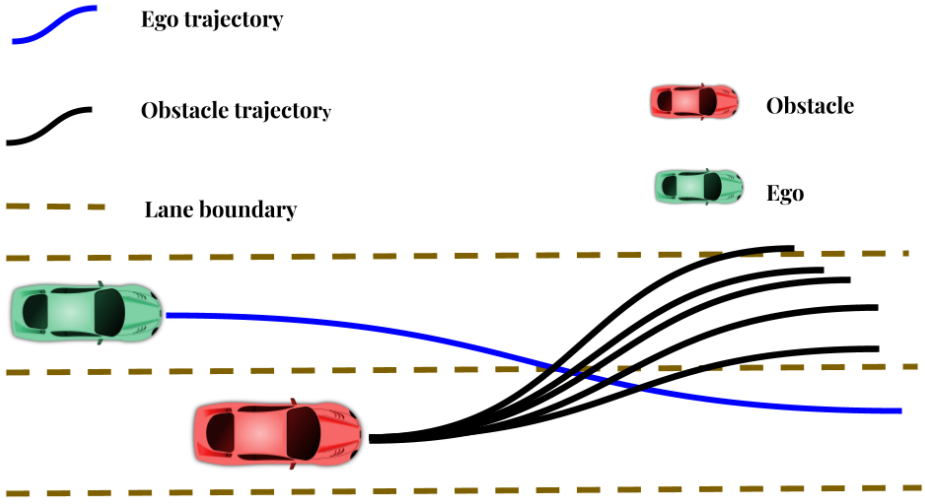


Figure 2: A collision scenario where the obstacle trajectory is stochastic. In other words, the obstacle can move in many unknown ways.

In such a setting, strict deterministic enforcement of $h(\mathbf{x}, \tau) \leq 0$ is no longer appropriate – it can still be achieved by considering the worst-case performance, but it often proves too conservative [106]. Instead, the notion of collision risk provides a probabilistic way to encode safety requirements by demanding that the

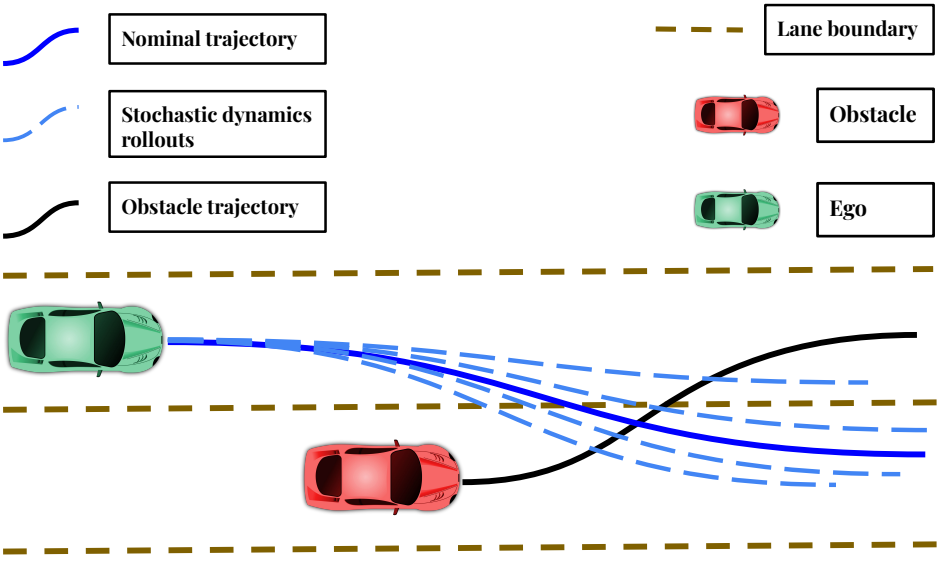


Figure 3: Collision scenario when the ego-vehicle dynamics have uncertainty. The different cyan-colored trajectories show the potential perturbation around the nominal dynamics (blue).

probability of violating the safety constraints must be minimized. We introduce the following general definition of collision risk:

$$r = P(h(\mathbf{x}, \tau) > 0)\,, \tag{1.2}$$

which measures the probability of violating the safety constraints. This leads to the first core challenge that is addressed in the thesis.

Challenge C1 — Tractability

It is generally intractable to obtain an analytical formula for r if the uncertainty in $\mathbf{x}_k$ or τ_k is non-Gaussian. State-of-the-art trajectory predictors [12, 44, 45, 46] characterize predicted trajectory distributions through arbitrarily complex and multi-modal deep generative models. Existing Gaussian chance-constraint methods for collision risk [47, 28, 49, 48] may therefore yield unreliable estimates. Practical risk-aware planning requires *tractable surrogate formulations* that approximate the probability of constraint violation while enabling real-time planning and control.

Example — Gaussian Special Case

Let $\mathbf{x}$ denote the ego trajectory and $\tau \sim \mathcal{N}(\mu, \Sigma)$ [27, 43, 28](Fig.4). For a given $\mathbf{x}$, the collision risk has closed form $r = P(h(\mathbf{x}, \tau) > 0) = \Phi_G(\mathbf{x}, \mu, \Sigma)$, with Φ_G derived from the Gaussian cumulative distribution function. Hence, minimum-risk or specified-risk-level trajectories are analytically computable. This tractability is lost beyond Gaussian uncertainty, motivating the thesis's nonparametric surrogate.

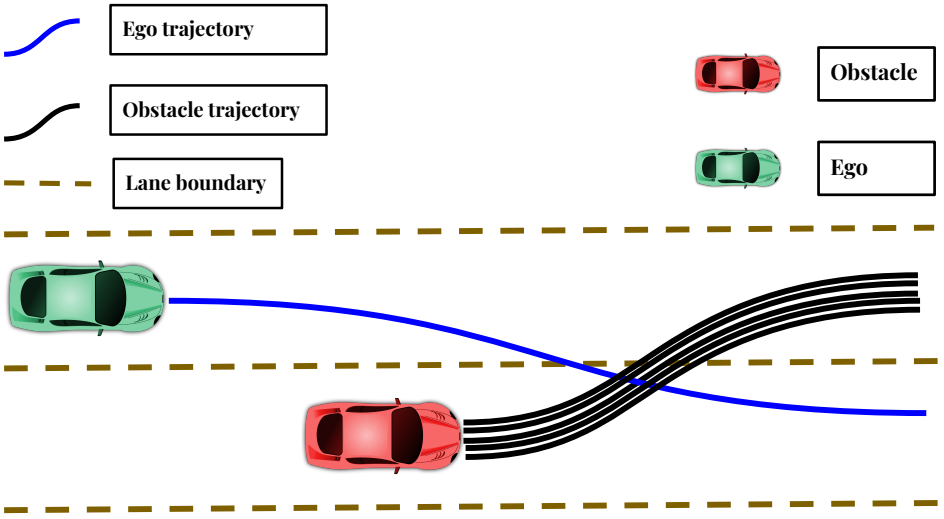


Figure 4: Collision scenario with Gaussian uncertainty over the obstacle trajectory.

1.1.3. Estimating risk via sampling

In the most general case, $\tau \sim p_\tau$, $\mathbf{x} \sim p_\mathbf{x}$ where p_τ, $p_\mathbf{x}$ are either unknown or very complex distributions, but we can draw samples from them. For example, p_τ can be a neural trajectory predictor (e.g. Trajectron++ [12]) and thus, no exact formula for $P(h(\mathbf{x}, \tau) > 0)$ is available which necessitates the need to estimate risk through obstacle trajectory samples. This leads to the second core challenge addressed in the thesis.

Challenge C2 — Sample Efficiency

A good risk-estimation algorithm must be *sample efficient*: the collision-risk cost should capture the true probability of collision using only a few samples from the underlying distribution. Suppose we use N samples ${}^j\tau$ of a predicted obstacle trajectory and the estimated collision risk approaches zero — is the trajectory really risk-free? This question translates directly to the variance of the risk estimate. Sample efficiency is also tied to computational run-time: a larger N means more collision checks between the ego vehicle and obstacles, increasing the computational burden both of estimation and of downstream trajectory optimization.

This thesis demonstrates that MMD-based risk surrogates in RKHS, combined with optimization-based reduced-set construction, provide a principled and sample-efficient solution to both challenges across multiple uncertainty domains. The following section details the objective and contributions of this work.

1.2. Objective and Contributions of the Thesis

The overall objective of this thesis is twofold. First, to develop a sample efficient method for estimating collision risk under arbitrary uncertainty model. Second, leveraging the risk models in downstream risk-aware trajectory optimization and demonstrate improved performance in a variety of applications. While the primary focus of this thesis is robot navigation, the developed methodology is sufficiently general to extend to other safety-critical robotic tasks.

At the core of our approach lies the problem of quantifying the dissimilarity between probability distributions. For special cases such as Gaussian distributions, this can be achieved using closed-form measures such as the Kullback–Leibler (KL) divergence. However, for arbitrary, non-parametric, and multi-modal distributions, such measures are often intractable. To overcome this limitation, we operate in the RKHS, where a distribution can be represented through its kernel mean embedding using only a finite number of samples. We then define a collision-risk surrogate as the MMD between the RKHS embedding of the collision-constraint residual distribution and a Dirac delta distribution. A key advantage of this formulation is that its sample efficiency can be systematically improved through the concept of a *reduced-set*, enabling principled, data-efficient, and computationally tractable risk estimation for motion planning under uncertainty.

Recalling the two core challenges introduced above, C1 denotes tractable risk estimation under non-Gaussian uncertainty and C2 denotes sample-efficient risk estimation under limited computational budgets. The main contributions of this thesis can be summarized as follows:

1. **[C1]** We formulate a novel collision-risk surrogate based on MMD and RKHS embeddings that enables nonparametric risk estimation directly from samples (Chapters 2, 4).
2. **[C2]** We develop an optimization-based reduced-set framework that improves the sample efficiency and computational tractability of collision-risk estimation (Chapters 2, 4, 5).
3. **[C1, C2]** We design risk-aware trajectory optimization methods for navigation under uncertain obstacle motion induced by multi-modal trajectory predictors of surrounding agents (Chapter 4).
4. **[C1, C2]** We extend the framework to trajectory optimization under stochastic ego dynamics, showing how reduced-set ideas can improve risk estimation from perturbed dynamics rollouts (Chapter 5).
5. **[C1, C2]** We further extend the framework to monocular vision-based navigation through a learned collision model integrated into risk-aware model predictive control (Chapter 6).

The remainder of the thesis is organized as follows. Chapter 2 introduces the mathematical preliminaries underlying the proposed framework, including risk formulation, RKHS embeddings, MMD, and reduced-set methods. Chapter 3 reviews existing approaches in deterministic planning, chance-constrained optimization, risk-aware planning, kernel-based stochastic optimization, and monocular vision-based navigation. Chapters 4, 5, and 6 present the main methodological contributions of the thesis in the settings of uncertain obstacle motion, stochastic ego dynamics, and monocular vision-based navigation, respectively. These three chapters correspond to the four peer-reviewed publications on which the thesis is based, with Chapter 4 covering the two publications on uncertain obstacle motion. The thesis concludes with a summary of the main findings and their implications for safe motion planning under uncertainty.

2. MATHEMATICAL PRELIMINARIES

This chapter provides the mathematical background for the remainder of the thesis. We first introduce the notation and formal definition of collision risk used throughout the thesis. We then review the kernel-based concepts that underlie the proposed framework, including RKHS embeddings and MMD. Finally, we present the reduced-set idea, which is central to improving sample efficiency in the planning methods developed in later chapters.

Symbols and Notation

Throughout the thesis, scalars are written as lower-case letters, vectors as bold lower-case letters, and matrices as bold upper-case letters. The constraint residual $\overline{h}(\cdot) = \max(0, h(\cdot))$ will appear repeatedly and is central to the distributional view of risk adopted in this thesis.

Table 1: Notation used throughout the thesis.

Symbol	Meaning
x, $\mathbf{x}$, $\mathbf{X}$	scalar, vector, matrix
$(\cdot)^\top$	transpose
k	time-step index
$p_{(\cdot)}$	probability density
$P(\cdot)$	probability of an event
$h(\cdot)$	safety-constraint function (negative = safe)
$\overline{h}(\cdot) = \max(0, h(\cdot))$	constraint-violation residual
r	collision risk (deterministic residual or $P(h > 0)$)
τ	obstacle trajectory
$\mathbf{x}$	ego state trajectory
ϕ	feature map into the RKHS $\mathscr{H}$
$\langle \cdot, \cdot \rangle_{\mathscr{H}}$	RKHS inner product
$K(\cdot, \cdot)$	positive-definite kernel (Gaussian or Laplace)
$\sigma > 0$	kernel bandwidth
$\mu[u]$, $\hat{\mu}[u]$	true / empirical RKHS embedding of u
N, N'	full sample count, reduced-set size
${}^l\beta$	reduced-set weight, $\sum_l {}^l\beta = 1$

2.1. Algebraic Form of Risk

We begin with a general definition of safety constraints and collision risk that is broad enough to cover the different planning settings considered later in the thesis. Let $h(\cdot) \leq 0$ denote a safety constraint where $h(\cdot)$ is a scalar-valued function. Depending on the application, $h(\cdot)$ can depend on one or more arguments. For example, in later chapters of the thesis based on Publications [1], [2], and [3], we enforce safety constraints at each time step k as follows: Let $h_k(\mathbf{x}_k, \tau_k) \leq 0, \forall k$ denote state and obstacle-dependent safety constraints, where $\mathbf{x}_k$ is the ego state and τ_k the obstacle position at time step k. Let $\mathbf{x}$ and τ be obtained by stacking together the respective values at different time steps. The worst-case safety constraint value over the horizon is

$$h(\cdot) = \max_k \; h_k(\cdot). \tag{2.1}$$

There are several ways to define $h_k(\cdot)$. In the most general setting, it can be a black-box function, for example, one derived from 3D occupancy maps [54]. We can also derive an analytical form for $h_k(\mathbf{x}_k, \tau_k)$ if we approximate the shape of the ego vehicle and the neighboring vehicles in terms of geometric primitives such as ellipses, as done in Publications [1], [2], and [3]. In the stochastic setting, either $\mathbf{x}$ or τ (or both) may be random variables, leading to the following general definition of risk:

$$r(\cdot) = P(h(\cdot) > 0)\,, \tag{2.2}$$

which measures the probability of violating the safety constraints. This formulation is natural and interpretable, but it is also difficult to evaluate in closed form when the uncertainty in $\mathbf{x}_k$ or τ_k is non-Gaussian or available only through samples. One can always linearize $h(\cdot)$ and compute a Gaussian approximation of the uncertainty, but such approximations are often too restrictive for the uncertainty distributions encountered in modern robotic planning. In the following, we describe the state-of-the-art sampling based approaches that can work for arbitrary uncertainty and non-linear constraints.

2.1.1. Constraint Residual and Sample-Based Risk Approximations

We define the constraint residual as:

$$\overline{h}(\cdot) = \max(0, h(\cdot)) \tag{2.3}$$

In the stochastic setting, $\overline{h}$ becomes a random variable with distribution $p_{\overline{h}}$. The residual captures the magnitude of safety-constraint violation: it is zero when the constraint is satisfied and positive otherwise. The following two sample-based approximations of collision risk are widely used.

SAA Risk Estimate. A simple yet effective approximation of collision risk r in (2.2) can be obtained by averaging over samples of the collision constraint residual [31], [69]. In practice, these samples may be derived from predictions of obstacle trajectories (Publications [1] and [2]), from stochastic rollouts of the underlying system dynamics (Publication [3]), or from models that estimate worst-case obstacle clearance (Publication [4]). That is, we have

$$r_{SAA} = \frac{1}{N}\sum_{j=1}^{N} \mathbb{I}\left({}^{j}\overline{h}(\cdot) > 0\right) \tag{2.4}$$

where $\mathbb{I}(\cdot)$ is an indicator function returning 1 when the conditions in the operand are satisfied and 0 otherwise. In essence, (2.4) estimates risk by counting the fraction of sampled realizations that violate the safety constraint. Although conceptually simple, estimates of the form (2.4) have proved useful in the existing literature [31].

CVaR-Based Risk Estimate. An alternative sample approximation of collision risk can be obtained by computing the empirical estimate of CVaR of $\overline{h}$ using the samples $\{{}^{j}\overline{h}\}_{j=1}^{j=N}$ [32, 40, 70]. That is, we define the risk metric as

$$r_{CVaR}^{emp} = CVaR_{\alpha}^{emp}(\overline{h}) \tag{2.5}$$

where α is a tunable parameter associated with CVaR. Unlike SAA, which counts violations, CVaR emphasizes the severity of large violations in the tail of the residual distribution.

Definition 1. Let $(\cdot)$ denote one or more random variables, for example, $\mathbf{x}$ or τ defined previously. Given a parameter $0 < \alpha < 1$, the α-VaR of $\overline{h}(\cdot)$ is ([70] Definition 6.11):

$$VaR_{\alpha}(\overline{h}) := \inf\{c \in \mathbb{R} : P(\overline{h}(\cdot) \leq c) \geq \alpha\} \tag{2.6}$$

Eqn. (2.6) implies that *constraint violations larger than* $VaR_{\alpha}(\overline{h})$ *occur with probability not exceeding* $1-\alpha$. Thus minimizing $VaR_{\alpha}(\overline{h})$ implies reducing the magnitude of risk $r(\cdot)$ defined in (2.2). Note that $VaR_{\alpha}(\overline{h})$ is just the α^{th}-quantile of the distribution of collision constraint represented by $\overline{h}$ and thus its empirical estimate $VaR_{\alpha}^{emp}(\overline{h})$ can be easily computed.

$CVaR_{\alpha}(\overline{h})$ is the more conservative version of $VaR_{\alpha}(\overline{h})$ since $CVaR_{\alpha}(\overline{h})$ is an upper bound on $VaR_{\alpha}(\overline{h})$ [70]. Thus, $CVaR_{\alpha}(\overline{h})$ is a more commonly used risk-surrogate [40], [63]. It is defined as follows.

Definition 2. Given a parameter $0 < \alpha < 1$, the α-CVaR of $\overline{h}(\cdot)$ is ([70] Theorem 6.2)

$$CVaR_{\alpha}(\overline{h}) := \mathbb{E}[\overline{h} : \overline{h} \geq VaR_{\alpha}(\overline{h})] \tag{2.7}$$

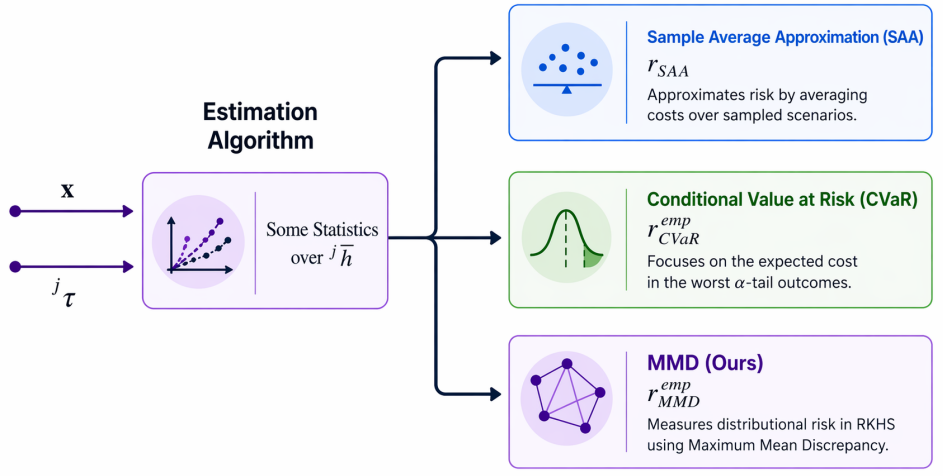


Figure 5: Sample-based collision-risk surrogates used in this thesis. The MMD-based surrogate is defined below and further developed in Chapters 4-6. Here, ego dynamics is deterministic and obstacle motion uncertain. Samples of the constraint residual $^j\bar{h}$, induced by predicted trajectory uncertainty $^j\tau$, are processed into different risk measures. Conventional estimators summarize samples via expectation (SAA) or tail metrics (CVaR), whereas the proposed approach uses MMD to compute a distribution-level discrepancy, enabling richer characterization of sample variability.

The empirical estimate $CVaR^{emp}_{\alpha}(\bar{h})$ then follows naturally: one first computes the empirical quantile $VaR^{emp}_{\alpha}(\bar{h})$ from the samples $\{^j\bar{h}\}_{j=1}^{N}$, identifies the tail subset $\{^{j_k}\bar{h}\}$ for which $^{j_k}\bar{h} \geq VaR^{emp}_{\alpha}(\bar{h})$, and averages over this subset.

Unified Perspective

Throughout this thesis, the core computational object is the sample-wise constraint residual $^j\bar{h}$ of (2.3), evaluated on realizations of the underlying uncertainty (e.g. predicted obstacle trajectories). Every risk surrogate considered here is obtained by computing some *statistic* over the empirical set $\{^j\bar{h}\}_{j=1}^{N}$; the surrogates differ only in which statistic they use:

- SAA (r_{SAA}) [31], [69] takes the empirical mean of the violation indicator over the sample set, i.e. the fraction of samples for which $^j\bar{h} > 0$.
- CVaR (r^{emp}_{CVaR}) [32, 40, 70] averages the largest residuals in the α-tail of the sample set, emphasizing rare but severe violations.
- The MMD-based surrogate (r^{emp}_{MMD}) developed in this thesis follows the same template, but the statistic it computes is different from r_{SAA} and r^{emp}_{CVaR} and is designed to capture the full distributional aspect of collision constraint violation under uncertainty.

Fig. 5 illustrates this common template: the samples $^j\bar{h}$ induced by uncertainty in $^j\tau$ are the shared starting point, and r_{SAA}, r^{emp}_{CVaR}, and r^{emp}_{MMD} are three different statistics computed over them. This view makes it clear why r^{emp}_{MMD} is a natural next step: it stays within the same sample-based framework as the classical estimators but retains richer information about the residual distribution through distribution matching.

2.2. Kernels, MMD and RKHS

This section introduces kernels, feature maps, RKHS embeddings, and MMD, which together form the backbone of the proposed framework. The role of this section is twofold: first, to explain how probability distributions can be represented in RKHS; and second, to show how these representations lead to a non-parametric notion of similarity between distributions. For a detailed exposition of these topics, please refer to [52].

2.2.1. Feature Maps and Kernels

Let $\mathscr{X}$ be a measurable input space. Consider a vector $\mathbf{z} \in \mathscr{X}$ and a non-linear transformation ϕ called a feature map. We can use ϕ to embed $\mathbf{z}$ in the RKHS $\mathscr{H}$ ([52] Section 2.2) as $\phi(\mathbf{z})$. A positive definite real-valued kernel function $K : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ is related to the feature map through the kernel trick, namely $K(\mathbf{z}, \mathbf{z}') = \langle \phi(\mathbf{z}), \phi(\mathbf{z}') \rangle_{\mathscr{H}}$ ([52] Definition 2.1), where $\langle \cdot, \cdot \rangle_{\mathscr{H}}$ is the inner product in $\mathscr{H}$. Intuitively, the kernel evaluates similarity in the feature space without requiring the feature map to be computed explicitly. The kernel trick therefore allows us to work with rich non-linear embeddings while keeping the computations tractable. We primarily use the Laplace kernel in our implementations, which was adopted from Publication [2] onward after initially using the Gaussian kernel in Publication [1]. The Laplace kernel is given by:

$$K(\mathbf{z}, \mathbf{z}') = \exp\left(-\frac{\| \mathbf{z} - \mathbf{z}' \|_1}{\sigma} \right) \tag{2.8}$$

where $\| \cdot \|_1$ denotes the l^1 norm (also known as the *Manhattan* or *Taxicab* norm). Here $\sigma > 0$ is a tunable bandwidth parameter that controls the sensitivity of the kernel to changes in the input.

2.2.2. From data points to probability distributions

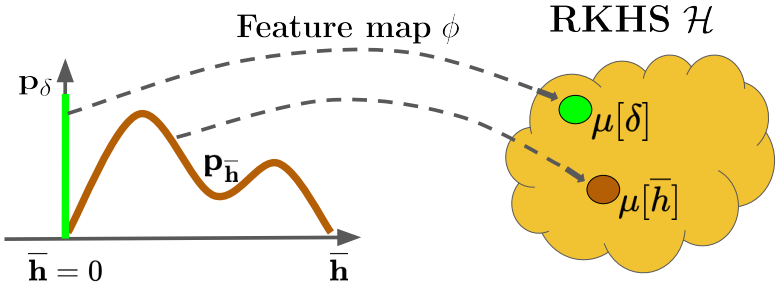


Figure 6: Kernel mean embedding of probability distributions p_δ and $p_{\bar{h}}$ into RKHS $\mathscr{H}$

The same ideas can be generalized from individual data points to probability distributions (see Figure 6). Let u be a random variable with probability distribution p_u. For example, u can denote the obstacle trajectory τ as in Publications [1]

and [2]. The RKHS embedding of u (or equivalently of p_u) is given by the following.

$$\mu[u] = E_{u\sim p_u}[\phi(u)] = E_{u\sim p_u}[K(u,\cdot)] \tag{2.9a}$$

$$= \int K(u,\cdot)dp_u(u) \tag{2.9b}$$

Equations (2.9a)-(2.9b) map a probability distribution to a single element of the RKHS. This is useful because it lets us reason about distributions through objects that live in a Hilbert space. For an arbitrary distribution p_u, however, it is often intractable to compute the right-hand side of (2.9b). In practice, we therefore use an empirical estimate of the RKHS embedding. Let ${}^{j}u$ be samples of u drawn from p_u. The empirical estimate of $\mu[u]$ is given by the following.

$$\hat{\mu}[u] := \sum_{j=1}^{N} \frac{1}{N}\phi({}^{j}u) = \sum_{j=1}^{N} \frac{1}{N}K({}^{j}u,\cdot) \tag{2.10}$$

Functions of Random Variables We can extend the concept of RKHS embedding to functions of random variables [50], [55] as well. This extension is important for the thesis because the quantities of interest in risk-aware planning are rarely the raw uncertainty variables themselves; rather, they are functions of those variables, such as the constraint residual.

Function of a single random variable: We first consider the simpler case in which the quantity of interest depends on a single random variable. Let $g(u) \in \mathscr{X} \to \mathscr{Z}$ be an arbitrary function of a random variable u with p_g denoting the probability distribution of $g(u)$. For example, u can be interpreted as the obstacle trajectory τ as done in Publications [1] and [2], in which case $\overline{h}(\mathbf{x},\tau)$ represents a function of the obstacle trajectory conditioned on a deterministic ego state trajectory $\mathbf{x}$. The RKHS embedding of p_g, denoted by $\mu[g]$, and its empirical estimate $\hat{\mu}[g]$ are given by

$$\mu[g] = E_{u\sim p_u}[\phi(g(u))] = E_{u\sim p_u}[K(g(u),\cdot)] \tag{2.11a}$$

$$= \int K(g(u),\cdot)dp_u(u), \tag{2.11b}$$

$$\hat{\mu}[g(u)] := \sum_{j=1}^{N} \frac{1}{N}\phi(g({}^{j}u)) = \sum_{j=1}^{N} \frac{1}{N}K(g({}^{j}u),\cdot) \tag{2.11c}$$

Note that K in the above expressions (2.11a)-(2.11c) is the same as that used in (2.10).

One of the attractive features of RKHS embeddings is that when constructed with characteristic kernels (Gaussian, Laplacian, etc.), they can capture information about the mean and moments of u and $g(u)$ up to infinite order. Furthermore, $\hat{\mu}[u]$ and $\hat{\mu}[g]$ are consistent estimators, meaning that they converge to the true RKHS embeddings as the sample size N increases. The embeddings $\hat{\mu}[u]$ and $\hat{\mu}[g]$ are also tightly coupled through the following theorem from [50].

Theorem 1. Suppose g is a continuous function and the kernel functions are continuous and bounded. Then, the following holds

$$\text{if } \hat{\mu}[u] \to \mu[u], \text{ then } \hat{\mu}[g] \to \mu[g] \tag{2.12}$$

The proof can be found in the Appendix of [50].

Remark 1. *Practical consequence.* If an approximate (e.g. reduced-set) embedding $\hat{\mu}[u]$ converges to $\mu[u]$, then the embedding of any well-behaved function of u also converges. In later chapters this is applied with $g = \overline{h}$: preserving the embedding of the sample distribution automatically preserves the embedding of the residual distribution needed for r^{emp}_{MMD}.

2.2.3. Maximum Mean Discrepancy

One of the main uses of RKHS embeddings is that they allow us to compare probability distributions directly. In this thesis, that comparison is made through MMD, which quantifies how far two distributions are from each other in RKHS.

Definition 3. Let u_1 and u_2 be random variables denoting two arbitrary distributions. Let u_1 and u_2 have probability distributions p_{u_1} and p_{u_2}, respectively. The MMD between the two distributions is given by

$$\begin{aligned} MMD(p_{u_1}, p_{u_2}) &= \|\mu[u_1] - \mu[u_2]\|^2_{\mathscr{H}} \\ &= \langle \mu[u_1] - \mu[u_2], \mu[u_1] - \mu[u_2] \rangle \end{aligned} \tag{2.13}$$

where $\|(\cdot)\|_{\mathscr{H}}$ is the RKHS norm and can be expressed through inner products, which in turn can be evaluated using the kernel trick. In other words, MMD measures the distance between two distributions by comparing their embeddings in RKHS.

Here, $\mu[u_1]$ is the RKHS embedding of u_1 (or p_{u_1}), defined as

$$\mu[u_1] = \int K(u_1, \cdot)\, dp_{u_1}(u_1), \quad \hat{\mu}[u_1] = \sum_{j=1}^{N} \frac{1}{N} \phi({}^j u_1) \tag{2.14}$$

Similarly, $\mu[u_2]$ is the RKHS embedding of the random variable $u_2 \sim p_{u_2}$. Assuming the underlying kernel is *characteristic* [51], the distributions p_{u_1} and p_{u_2} are identical if and only if the following holds [50, 53].

$$\|\mu[u_1] - \mu[u_2]\|^2_{\mathscr{H}} = 0$$

Remark 2. We can approximate p_δ through a Gaussian $\mathscr{N}(0, \varepsilon)$, with an extremely small covariance ε ($\approx 10^{-5}$).

Finite Sample Guarantees. A key advantage of r^{emp}_{MMD} is that it inherits the convergence and finite-sample guarantees associated with RKHS embeddings and MMD. Specifically, the gap between r^{emp}_{MMD} and the true r_{MMD} can be quantified by the following theorem, which is a direct application of the result presented in [51].

Theorem 2. Let u_1 and u_2 be two arbitrary random variables, respectively, with corresponding probability distributions p_{u_1} and p_{u_2}.
Given i.i.d. samples $X := \{^j u_1\}_{j=1}^N$ and $Y := \{^j u_2\}_{j=1}^N$ drawn from p_{u_1} and p_{u_2}, respectively, and assuming the kernel satisfies $0 \leq K(\xi_1, \xi_2) \leq c_0$ for all ξ_1, ξ_2 and some constant c_0, the following holds:

$$P_{X,Y}\left\{|r^{emp}_{MMD} - r_{MMD}| \leq 2\left(2(c_0/N)^{\frac{1}{2}} + \varepsilon\right)\right\} > 1 - \exp\left(\frac{-\varepsilon^2 N}{4c_0}\right) \tag{2.15}$$

where $P_{X,Y}$ denotes the probability over the samples X and Y, and $\varepsilon > 0$ is a parameter controlling the confidence level. The function $K(\cdot,\cdot)$ denotes the kernel. Theorem 2 ensures that as the sample size N increases, the probability that the absolute difference between r^{emp}_{MMD} and r_{MMD} is small approaches 1. In plain terms, the empirical surrogate becomes more reliable as more samples are used. This is important for planning because it shows that the optimization is not being driven by an arbitrary heuristic, but by a surrogate whose estimation error can be controlled.

This chapter introduced the mathematical tools that underpin the rest of the thesis: a distributional view of collision risk, its MMD-based surrogate in RKHS, finite-sample convergence guarantees, and the reduced-set principle for improving sample efficiency. Chapter 3 reviews related work, after which Chapters 4–6 develop the full planning algorithms that build on these foundations.

3. REVIEW OF EXISTING APPROACHES

To address motion planning under uncertainty, this thesis adopts a risk-aware trajectory optimization methodology. This formulation offers a rigorous, principled framework for enforcing safety through explicit risk minimization while preserving optimality in the presence of uncertainty. Chapters 1 and 2 introduce the general problem setting and mathematical background, while Chapters 4, 5, and 6 develop the concrete formulations used in the three main application settings. Broadly speaking, the literature most relevant to this thesis spans deterministic planning, chance-constrained optimization, risk-aware planning, nonparametric stochastic optimization using RKHS embeddings, and monocular vision-based navigation. These areas are tightly connected: deterministic planners provide the algorithmic foundations, chance-constrained and risk-aware methods introduce probabilistic safety, kernel methods offer nonparametric tools for handling arbitrary uncertainty, and vision-based navigation brings the additional challenge of perception-induced uncertainty. In particular, the review is oriented around the two core challenges identified in Chapter 1: the tractability of risk estimation under non-Gaussian uncertainty (C1) and the sample efficiency of risk surrogates under limited computational budgets (C2). We review these strands next and highlight how they collectively motivate the methods developed in this thesis.

3.1. Deterministic Approaches

Motion planning is one of the most well-studied problems in robotics. A large body of work assumes a deterministic setting in which the robot state, environment geometry, obstacle motion, and system dynamics are sufficiently well known that safety can be enforced point-wise in time. Under this viewpoint, the planning problem reduces to constructing a feasible path or trajectory that optimizes a geometric or control objective while satisfying deterministic constraints. This assumption has historically enabled a broad range of algorithmic advances and remains highly relevant whenever uncertainty is weak or can be neglected.

3.1.1. Graph-Search and Lattice-Based Planning

One major family of deterministic methods is based on graph-search and geometric planning. Classical shortest-path methods such as A* [13] and their later robotic variants [15] operate on discretized maps or lattices and are attractive because they provide explicit search structure and strong completeness properties within the chosen discretization. Such methods remain particularly relevant when the environment can be represented by occupancy grids, road graphs, or state lattices, and when the main objective is to compute globally feasible paths with relatively transparent planning logic.

3.1.2. Sampling-Based Motion Planning

A second major family is sampling-based planning, including Rapidly-Exploring Random Tree (RRT)-style and Probabilistic Roadmap (PRM)-style algorithms [17, 18, 19, 20, 21, 22]. These approaches are especially useful in high-dimensional configuration spaces and for systems with complex geometric constraints, as they avoid explicit discretization of the full state space. Their success has made them a standard baseline in robotic motion planning.

3.1.3. Trajectory Optimization and MPC

A third important family is trajectory optimization and model-predictive planning, where the robot trajectory is optimized directly with respect to smoothness, tracking, control effort, or obstacle avoidance costs [23, 24, 25, 26, 93]. In addition to gradient-based trajectory optimization, stochastic-search methods such as the cross-entropy method [91] have also been used to optimize trajectories and control sequences in complex non-convex settings. These methods are particularly relevant to the present thesis because they naturally support the inclusion of task costs, dynamics, and state/control constraints within a unified optimization problem. In autonomous driving, deterministic planning has also led to practical methods based on potential fields [9], spatio-temporal semantic corridors [10], and batch trajectory optimization in structured road environments [11]. More recently, differentiable and modular control stacks [95] have further bridged classical planning and learning by embedding prediction and optimization modules into a common differentiable pipeline.

Despite their effectiveness, deterministic methods inherit a fundamental limitation: they assume that the information used by the planner is sufficiently accurate to support deterministic collision checking and deterministic constraint satisfaction. In real-world robotic systems, however, obstacle trajectories may be multimodal, perception may be noisy, and system dynamics may be stochastic. In such settings, a trajectory that appears safe under a deterministic model may in fact have a non-negligible probability of collision. This mismatch is precisely what motivates probabilistic formulations of planning under uncertainty.

Gap Identified

These methods provide strong algorithmic foundations for planning and control, but they do not explicitly model the stochasticity induced by uncertain obstacle motion, stochastic dynamics, or noisy perception. This limitation motivates the uncertainty-aware formulations adopted in the thesis.

3.2. Chance-Constrained Optimization (CCO)

In the presence of uncertainty, the constraint function becomes a random variable because it is directly influenced by stochastic effects such as perception noise,

prediction errors, or process disturbances. In such settings, strict deterministic enforcement of the constraints is no longer appropriate. Instead, *chance constraints* provide a probabilistic way to encode safety requirements by demanding that the constraint be satisfied with high probability. Instead of minimizing risk r (as defined in (2.2)), if we add constraints of the form $r \leq \zeta$, for some constant $\zeta \in (0, 1]$, we obtain the setting of Chance Constrained Optimization (CCO). This formulation captures the intuitive idea that occasional constraint violations may be permissible, but should occur only rarely. Chance constraints are conceptually appealing because they retain the semantics of probability of failure while turning the planner into a tool for enforcing explicit safety levels.

3.2.1. Analytical and Parametric Reformulations

The main difficulty is computational. In many robotics applications, the probability appearing in a chance constraint is analytically intractable because the underlying uncertainty is nonlinear, non-Gaussian, multi-modal, or only accessible through samples from learned predictors and simulators [12]. As a result, the literature has largely focused on developing tractable approximations. A first family derives exact or approximate reformulations under restrictive assumptions such as linear dynamics, affine constraints, and Gaussian uncertainty [56, 47, 71, 49, 48]. These methods are attractive because they often lead to convex or optimization-friendly problems. However, their accuracy depends heavily on whether the assumed distributional and structural conditions hold in practice.

3.2.2. Sample-Based and Monte Carlo Approximations

When such assumptions fail, a second family of methods approximates the chance constraint directly using samples. In these approaches, the probability of constraint satisfaction or violation is estimated by propagating sampled uncertainty through the system dynamics and collision constraints. Monte Carlo motion planning and related sample-based chance-constrained methods [30, 29, 28] belong to this class. Extensions to multimodal environmental uncertainty have also been studied [96]. The main advantage of these approaches is generality: in principle, they can accommodate arbitrary uncertainty models without requiring closed-form reformulations. Their drawback is that they can be computationally demanding, particularly when each sample requires an expensive rollout, collision check, or evaluation against multiple dynamic obstacles.

3.2.3. Scenario Approaches

A conceptually different use of samples is found in scenario approaches [33, 34, 35, 36, 37, 38]. Rather than estimating the probability in the chance constraint directly, these methods replace the original probabilistic constraint by a finite collection of deterministic constraints, each corresponding to one sampled realization of the uncertainty. In this way, the stochastic problem is approximated by a de-

terministic optimization problem with multiple sampled constraints. While conceptually straightforward, scenario-based methods often require a large number of enforced scenarios to obtain meaningful probabilistic guarantees, which can render the resulting optimization problem computationally expensive and conservative [39].

3.2.4. Surrogate Relaxations via Sample Average Approximation (SAA) and Conditional Value-at-Risk (CVaR)

Another common approach is to replace the original probabilistic constraint with a tractable surrogate. SAA [31] approximates chance constraints through empirical averages over samples. Closely related methods use coherent risk surrogates such as CVaR either on the constraint function [63, 64, 65, 66] or on the constraint residual [40]. In this section, CVaR appears specifically as a *surrogate or relaxation of the original chance constraint*: the goal is still to approximate or enforce probabilistic constraint satisfaction, but through an optimization-friendly proxy. These formulations can be easier to optimize than the original probability-of-failure constraint and often have stronger numerical behavior. Nevertheless, they inherit the challenge that their quality depends strongly on the informativeness and number of samples used.

Gap Identified

The approaches above illustrate the central trade-off between tractability and fidelity to the true uncertainty distribution. In contrast, the thesis adopts a risk-minimization viewpoint that avoids prescribing a hard chance threshold ζ and does not rely on restrictive distributional assumptions.

3.3. Risk-Aware Planning

Risk-aware motion planning algorithms are designed to explicitly account for uncertainty by minimizing a risk measure associated with undesirable events such as collisions. Such collisions may arise from multiple sources, including errors in robot and obstacle localization, inaccuracies in the robot's dynamic model or in predicting obstacle motion, and unmodeled external disturbances. Compared with chance-constrained optimization, risk-aware planning relaxes the requirement of explicitly satisfying a prescribed probability threshold and instead places a risk term directly in the objective. This makes it a natural framework when exact constraint satisfaction is hard to verify, or when the planner must continuously trade off safety against task performance.

Existing literature differs both in the *choice of risk metric* and in the *optimization method* used to minimize it. At the level of risk formulation, several metrics have been proposed. Time-to-collision and related severity-aware criteria [42] provide intuitive and interpretable surrogates for safety. Other works define risk

through the spatio-temporal overlap of the ego-vehicle and obstacle footprint together with collision severity [41]. A widely used family of approaches employs coherent risk measures such as CVaR, which are attractive because they capture tail behavior more explicitly than simple means or frequencies of violation. For example, [40] estimate risk from samples of the constraint residual, while [57] adapt the level of risk sensitivity online in stochastic-search-based control.

At the level of optimization, these risk measures are commonly embedded into sampling-based controllers and trajectory optimizers. For instance, [40] combine an empirical CVaR-based risk estimate with Model Predictive Path Integral (MPPI) [92] to perform risk-aware control in dynamic environments. Related approaches use stochastic search, model-predictive control, or adversarial formulations to construct trajectories that remain safe under uncertainty. In [90], for example, the environment is modeled as a worst-case opponent, yielding an adversarial view of safe planning for autonomous driving. In contrast to Section 3.2, where CVaR is discussed as a tractable proxy for a chance constraint, here CVaR is used directly as an *objective-level notion of risk* to be minimized by the planner. These works demonstrate that risk-aware formulations can be deployed in realistic planning and control pipelines without abandoning real-time applicability.

However, a central challenge remains under-emphasized in much of the prior literature: *sample efficiency*. In many practical settings, the risk estimate must be computed from a limited number of predicted obstacle trajectories or stochastic rollouts. If the risk metric is noisy under small sample budgets, the downstream planner may become inconsistent, overly conservative, or unsafe. Thus, beyond asking what risk metric to optimize, one must also ask how reliably that risk can be estimated from finite samples. This question is central to the present thesis.

These gaps motivate C2: the present framework treats sample efficiency as a first-class design objective, so the collision-risk surrogate r^{emp}_{MMD} in (4.10) is designed to remain effective under limited sampling budgets as well as to model safety under uncertainty.

3.4. RKHS Embedding for Stochastic Trajectory Optimization

The methods developed in this thesis are built on the RKHS embedding framework for functions of random variables introduced in [50, 55]. The appeal of this framework is that it provides a nonparametric way of representing probability distributions and functions of random variables directly in a reproducing kernel Hilbert space. This is particularly valuable in robotics, where uncertainty may come from black-box predictors, learned policies, stochastic simulators, or complex perception models for which no tractable parametric density is available.

A central idea in this line of work is that distributions can be compared through their kernel mean embeddings rather than through explicit density estimation. When combined with characteristic kernels, this makes it possible to reason about rich, non-Gaussian, and multi-modal uncertainty using only samples. The same

literature also advocates the use of *reduced-set* representations to enhance the expressive capacity of empirical embeddings under finite computational budgets [50, 55]. However, in those works the reduced set is obtained by randomly selecting a subset of the available samples. In contrast, the present thesis treats reduced-set construction itself as an optimization problem.

Several prior works have explored the use of RKHS embeddings and MMD for either chance-constrained optimization or risk minimization. For instance, the authors in [39] use MMD-based rejection sampling to reduce the conservativeness of scenario approximations of chance constraints. Related work in [68], [54], and Publication [1] applies Hilbert-space embeddings to stochastic planning under nonparametric uncertainty. These works demonstrate that kernel methods can provide a powerful bridge between probabilistic modeling and optimization, especially when classical Gaussian assumptions are no longer adequate.

A related but conceptually distinct line of work arises in Distributionally Robust Optimization (Distributionally Robust Optimization (DRO)), where optimization is performed over an ambiguity set of possible probability distributions rather than a single nominal estimate. In that context, MMD can be used to construct ambiguity sets with finite-sample guarantees [67]. In contrast, the present thesis assumes that the uncertainty distribution is sufficiently well estimated and uses MMD directly to quantify risk rather than to define robustness to distributional ambiguity. This distinction is important: the focus here is not on robustness against model misspecification, but on reliable and sample-efficient risk estimation for a given uncertainty model.

Gap Identified

The main distinction of the present work is that MMD is not used only as an auxiliary tool for sampling or robustness, but as the foundation of the risk surrogate itself. This makes the reduced-set structure central to the method. Furthermore, this thesis extends Publication [1], [54], and [68] by jointly optimizing the reduced-set selection, sample weights, and kernel parameters within a unified framework.

3.5. Monocular Vision-Based Navigation Under Perception Uncertainty

Vision-based navigation has gained significant attention as a lightweight alternative to LiDAR- or depth-sensor-based approaches, particularly for platforms with strict size, weight, and power constraints such as micro aerial vehicles and compact ground robots. Vision-only navigation is attractive because it reduces hardware cost, payload, and energy consumption while enabling deployment in environments where dense range sensing may be impractical. The key challenge, however, is that monocular RGB sensing does not provide direct metric depth,

and safe planning must therefore rely on inferred geometry. Recent advances in monocular depth estimation, from the Dense Prediction Transformer (DPT) [99] to vision foundation models such as DepthAnything [60, 61] and ZoeDepth [62], have made this paradigm increasingly viable. At the same time, they highlight a core difficulty for safety-critical planning: even small depth errors can produce large mistakes in free-space reasoning, especially in cluttered scenes and near obstacle boundaries.

3.5.1. End-to-End Policies and Foundation Models

A prominent line of work in vision-based navigation adopts end-to-end learning approaches that map RGB images directly to control commands or waypoints. These methods typically employ supervised learning or imitation learning [74, 75, 76, 79, 101] and offer conceptual simplicity by bypassing explicit geometric reasoning. Their appeal lies in reducing the perception-planning pipeline to a direct policy-learning problem. However, this simplification comes at a cost: such methods often generalize poorly to unseen environments, remain sensitive to perception noise, and lack explicit modeling of uncertainty or safety constraints.

Recent foundation-model-style approaches such as GNM [100], ViNT [88], and NoMaD [83] improve generalization by employing large-scale pretraining and image-conditioned diffusion models to generate trajectories from RGB input. These methods suggest that richer visual priors can help bridge the gap between end-to-end learning and generalizable navigation. Nevertheless, they still rely heavily on high-quality supervision and typically optimize navigation performance rather than explicitly modeling collision uncertainty.

Deep reinforcement learning has also been explored for visual navigation [80, 82, 81]. These methods can learn reactive behaviors directly from visual observations and, in principle, adapt to complex environments without requiring explicit labels for intermediate geometric quantities. In practice, however, they often require extensive training, careful reward design, and large-scale interaction data. More importantly for the present thesis, they provide limited safety guarantees and rarely expose an explicit probabilistic quantity that can be integrated into a risk-aware planning framework.

3.5.2. Map-Based and Geometric Planning from Monocular Input

A related class of methods focuses on learning spatial representations from monocular input for downstream planning. For instance, Active Neural SLAM [77], Neural Topological SLAM [78], and MonoNav [73] construct occupancy maps or local 3D representations from estimated depth and plan over them. This family of methods is attractive because it preserves an explicit notion of geometry and therefore interfaces naturally with conventional planners. Compared with pure end-to-end control, it also offers more interpretability: the planner reasons over maps, obstacles, and candidate trajectories rather than directly over raw pixels.

The weakness of this paradigm is that the reliability of planning becomes tightly coupled to the quality of the estimated geometry. If monocular depth predictions are biased, noisy, or inconsistent, the resulting occupancy map may contain false free space or false obstacles. In cluttered environments, these errors can translate directly into unsafe behavior or excessive conservatism. Thus, while map-based monocular navigation introduces valuable structure, it does not by itself solve the uncertainty problem induced by monocular perception.

3.5.3. Visual Predictive Control and Learned Safety Models

An alternative direction involves learning predictive models of visual dynamics for planning and control. World models [84, 85, 86] learn action-conditioned representations that simulate future states, enabling model-based planning from visual input. Visual MPC approaches [89] output control commands by reasoning over predicted image sequences and subgoal images. This paradigm is structurally closer to classical planning because it retains an explicit predictive model and a downstream optimization step. It therefore offers a promising bridge between learned perception and risk-aware control.

More closely related to the present thesis are learned safety-model approaches in which a network predicts a collision-relevant quantity from observations and candidate actions. For example, [87] predicts collision probabilities from depth images and actions and minimizes these probabilities within an MPC framework. Such methods offer a more structured alternative to end-to-end policies, but their reliability still depends strongly on the quality of the perceptual representation and the expressiveness of the learned safety signal. In particular, directly predicting a binary collision probability may discard useful information about uncertainty in obstacle clearance and near-collision structure.

Gap Identified

From the perspective of this thesis, the key gap in the existing monocular navigation literature is that most methods either plan directly on noisy depth estimates, learn policies without explicit probabilistic safety modeling, or predict a relatively coarse collision quantity. What remains less explored is how to learn a richer probabilistic safety model from monocular input and couple it tightly with a planner that can explicitly optimize a risk-aware objective.
Chapter 6 addresses this gap by retaining an explicit planning layer (unlike end-to-end policies), avoiding direct collision checking on raw monocular depth (unlike depth-to-map pipelines), and learning a probabilistic action-conditioned model over minimum obstacle clearance rather than a binary collision predictor. The resulting uncertainty-aware model is integrated into a risk-aware MPC framework for safe yet non-conservative navigation in dense, cluttered environments.

This chapter surveyed the main strands of related work and identified the gaps that the thesis addresses: the lack of tractable, nonparametric risk surrogates (C1) and the absence of principled mechanisms for sample-efficient risk estimation (C2). The following chapters present the methodological contributions: Chapter 4 develops the MMD-based framework for planning under uncertain obstacle motion, Chapter 5 extends it to stochastic ego dynamics, and Chapter 6 integrates a learned collision model with risk-aware MPC for monocular vision-based navigation.

4. MAXIMUM MEAN DISCREPANCY–BASED COLLISION RISK MINIMIZATION (PUBLICATIONS [1] AND [2])

At a Glance

- **Setting:** Autonomous driving with multi-modal predicted trajectories of surrounding vehicles; ego dynamics deterministic.
- **Challenges addressed:** C1 (tractable non-Gaussian risk) and C2 (sample efficiency under costly collision checks).
- **Method:** MMD-based collision-risk surrogate r^{emp}_{MMD} with a bi-level jointly optimized reduced-set (samples + weights + kernel).
- **Key Result:** Considerably lower collision rate than SAA and CVaR baselines on nuScenes with Trajectron++.

4.1. Introduction

This chapter considers risk-aware trajectory optimization in autonomous driving when nearby vehicles follow multi-modal future trajectories. This setting makes challenges C1 (tractable risk under non-Gaussian predictions) and C2 (sample efficiency under costly collision checks) concrete. The development is based on Publications [1] and [2]: the former introduced an RKHS-based trajectory optimization framework operating on sample-level obstacle predictions, and the latter consolidated it into MMD-OPT by sharpening the risk surrogate and strengthening the reduced-set construction. A representative scenario is shown in Fig. 7.

4.2. Problem Formulation

Ego dynamics are modeled deterministically in the Frenet frame, while obstacle motion is represented by a predictor-generated stochastic trajectory distribution. Thus, the chapter isolates uncertainty in the surrounding agent's behavior.

4.2.1. Motion Model in Frenet Frame

We formulate trajectory planning in the road-aligned Frenet frame, with s as longitudinal displacement and d as lateral offset. We use the bicycle model of [90].

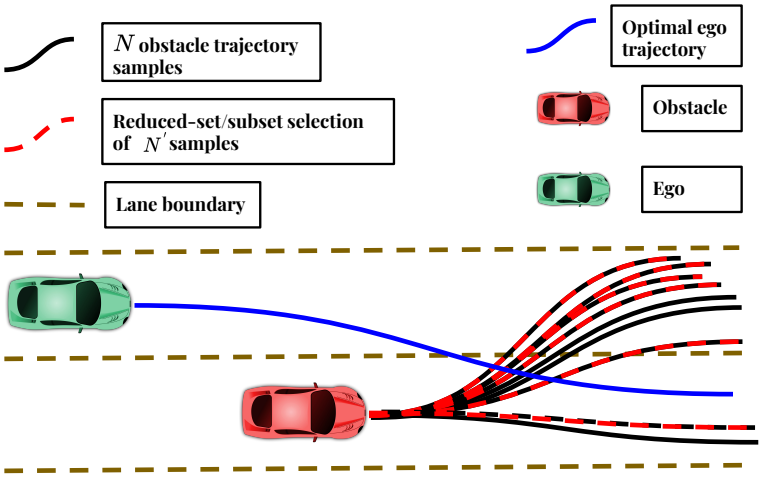


Figure 7: Multi-modal obstacle prediction scenario. The obstacle has two intents (lane-change vs. lane-following) with unequal probabilities. The proposed reduced-set selects the most representative trajectory samples from a black-box distribution, enabling probabilistically safe planning that appropriately discriminates between low- and high-probability maneuvers.

$$s_{k+1} = s_k + \dot{s}_k \Delta t, d_{k+1} = d_k + \dot{d}_k \Delta t \tag{4.1a}$$

$$\psi_{k+1} = \psi_k + \dot{\psi}_k \Delta t, v_{k+1} = v_k + a_k \Delta t \tag{4.1b}$$

$$\dot{s}_{k+1} = \frac{v_k \cos \psi_k}{1 - d_k \kappa(s_k)}, \dot{d}_{k+1} = v_k \sin \psi_k \tag{4.1c}$$

$$\dot{\psi}_{k+1} = \frac{v_k \tan(\theta_k)}{B} - \kappa(s_k) \frac{v_k \cos \psi_k}{1 - d_k \kappa(s_k)} \tag{4.1d}$$

where subscript k denotes the time step, ψ_k is the ego heading, B the wheelbase, a_k, θ_k the longitudinal acceleration and steering inputs, and $\kappa(s_k)$ the curvature at s_k. The state is $\mathbf{x}_k = (s_k, d_k, \psi_k, \dot{\psi}_k, v_k)$, and $\mathbf{x}$ concatenates the states over the horizon.

Differential Flatness The model (4.1a)-(4.1d) is differentially flat, so control inputs can be written as functions of position and its derivatives:

$$v_k = \sqrt{(\dot{s}_k(1 - d_k \kappa(s_k)))^2 + (\dot{d}_k)^2}, a_k = \frac{v_{k+1} - v_k}{\Delta t} \tag{4.2}$$

$$\psi_k = \tan^{-1}(\dot{d}_k, \dot{s}_k), \dot{\psi}_k = \frac{\psi_{k+1} - \psi_k}{\Delta t} \tag{4.3}$$

$$\theta_k = \tan^{-1}\left(\frac{(\dot{\psi}_k + \kappa(s_k)\dot{s}_k)B}{v_k}\right) \tag{4.4}$$

4.2.2. Risk-Aware Trajectory Optimization

Let $\tau \in \mathbb{R}^{2T}$ denote an obstacle trajectory (time-stamped waypoints over horizon T), modeled as a random variable with distribution p_τ and known only through

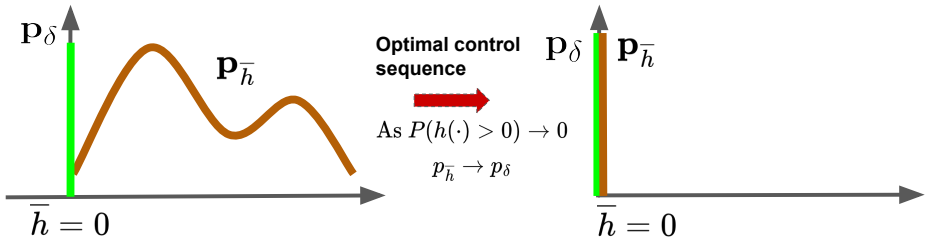


Figure 8: The mass of $p_{\bar{h}}$ lies to the right of $\bar{h} = 0$. The optimal control input is one that leads to the uncertainty distribution for which $p_{\bar{h}}$ resembles a Dirac-Delta distribution.

samples from a black-box predictor. Leveraging differential flatness, we formulate risk-aware trajectory optimization directly in the trajectory space:

$$\min_{\mathbf{x}} \quad c(\mathbf{x}) + r(\mathbf{x}, \tau) \tag{4.5}$$

$$\text{s.t.} \quad \mathbf{f}(\mathbf{x}) = \mathbf{0} \tag{4.6}$$

$$\mathbf{g}(\mathbf{x}) \leq 0, \forall k \tag{4.7}$$

$c(\cdot)$ is a state-dependent cost (any state- or control-dependent term can be rolled into it via differential flatness), $r(\cdot)$ ((2.2)) is the collision risk associated with the uncertain obstacle trajectory, (4.6) enforces boundary conditions on the initial and final ego states, and (4.7) enforces per-step kinematic limits (velocity, acceleration, steering) and lane-boundary constraints.

The per-step collision constraint $h_k(\mathbf{x}_k, \tau_k) \leq 0$ is aggregated over the planning horizon via a worst-case operator, $h(\mathbf{x}, \tau) = \max_k h_k(\mathbf{x}_k, \tau_k)$, so that a single trajectory-level constraint is imposed rather than T per-step constraints. Since τ is a random variable, replacing T chance constraints with one horizon-aggregated constraint avoids joint-risk bookkeeping at each time step, and the induced risk $r(\mathbf{x}, \tau) = P(h(\mathbf{x}, \tau) > 0)$ is a single scalar that the optimizer drives toward zero. The formulation extends to multiple obstacles by computing one h-residual per obstacle and summing (or taking the max of) the corresponding risks, so the single-obstacle presentation above carries over without loss of generality.

Remark 3. p_τ can be arbitrary; we only require samples from it. Sampling is cheap relative to evaluating r on each sample, which is where the computational bottleneck lies and motivates the reduced-set construction developed later in this chapter. The optimizer is used in a receding-horizon MPC loop: at each step, the current ego state and the latest predicted obstacle distribution are fed in, (4.5)–(4.7) is solved, the first control is applied, and the horizon slides forward.

4.3. Main Algorithmic Results

1. **[C1]** We formulate a sample-efficient surrogate for collision risk associated with uncertain obstacle trajectory predictions (samples), based on the concepts of RKHS and the associated MMD.
2. **[C2]** We propose a bi-level optimization that leverages the concepts of RKHS embedding and MMD to reduce the number of collision checks required to estimate the collision risk associated with uncertain obstacle trajectory predictions (samples) - Alg. 1.
3. **[C1, C2]** We propose a custom sampling-based optimizer to minimize the proposed collision risk surrogate - Alg. 2.

4.3.1. Risk as Difference of Distributions

Recall the per-step collision constraint $h_k(\mathbf{x}_k, \tau_k) \leq 0$. Aggregating across time via a worst-case operator yields $h(\mathbf{x}, \tau) = \max_k h_k$ (Eq. (2.1)). The corresponding collision risk is defined as $r = P(h(\mathbf{x}, \tau) > 0)$. For a general trajectory distribution p_τ, this risk does not admit a tractable analytical form (challenge C1).

To address this, we introduce a surrogate based on the collision constraint violation residual:

$$\overline{h}(\mathbf{x}, \tau) = \max(0, h(\mathbf{x}, \tau)).$$

Main Intuition: When τ is modeled as a random variable representing uncertain obstacle trajectories, the residual $\overline{h}(\mathbf{x}, \tau)$ also becomes a random variable. We denote its induced distribution by $\overline{h} \sim p_{\overline{h}}$.

The key insight is that although we do not know the parametric form of $p_{\overline{h}}$, we know that all of its mass lies to the right of $\overline{h} = 0$ (see Fig. 8). Moreover, as $P(h(\cdot) > 0)$ approaches zero, $p_{\overline{h}}$ approaches a Dirac-Delta distribution p_δ concentrated at zero. In other words, a low-risk trajectory is one for which the residual distribution becomes increasingly similar to the zero-risk reference distribution. This motivates the following surrogate view of risk, following [54] and Publication [1]:

$$r(\cdot) \approx \mathscr{L}_{dist}(p_\delta, p_{\overline{h}}), \tag{4.8}$$

where $\mathscr{L}_{dist}$ is any measure that characterizes the difference between two distributions. For example, KL divergence measures distribution similarity but requires analytical forms, making it unsuitable for comparing $p_{\overline{h}}$ and p_δ from samples alone. This thesis adopts MMD as a practical non-parametric $\mathscr{L}dist$, measuring how far the residual distribution $p\overline{h}$ is from the ideal zero-risk Dirac-delta distribution p_δ. Thus, we define the following MMD-based surrogate for collision risk.

$$r_{MMD} = \mathscr{L}_{dist}(p_\delta, p_{\overline{h}}) = \overbrace{\left\| \mu[\overline{h}] - \mu[\delta] \right\|^2_{\mathscr{H}}}^{MMD}. \tag{4.9}$$

As r_{MMD} decreases, $p_{\bar{h}}$ becomes increasingly similar to p_δ. In the limiting case, as $r_{MMD} \to 0$, the distribution $p_{\bar{h}} \to p_\delta$ and consequently $P(h(\mathbf{x}, \tau) > 0) \to 0$.
It is important to clarify the role of r_{MMD} as a surrogate: it is not a direct probability bound on $P(h > 0)$, but rather a distributional distance between the residual distribution $p_{\bar{h}}$ and the ideal zero-risk distribution p_δ. The equivalence $r_{MMD} = 0 \Leftrightarrow p_{\bar{h}} = p_\delta$ requires the use of a *characteristic* kernel [51] (the Gaussian and Laplace kernels used in this thesis are characteristic). The empirical estimate r^{emp}_{MMD} converges to r_{MMD} as $N \to \infty$ (Theorem 1), but at finite N it remains a biased surrogate; its value as a risk cost therefore rests on the distribution-matching interpretation rather than on any pointwise relation to the probability of collision.

Typically, it is intractable to obtain the exact MMD for arbitrary $p_{\bar{h}}$; thus, we resort to an empirical estimate based on sample-level information, as follows. This empirical form is the quantity that is optimized in the planning algorithms developed in the thesis.

$$r^{emp}_{MMD} = \left\| \hat{\mu}[\bar{h}] - \hat{\mu}[\delta] \right\|^2_{\mathscr{H}} \tag{4.10}$$

Unlike SAA or CVaR, this surrogate admits a principled reduced-set mechanism and thus directly addresses C2. Publication [1] used a Gaussian kernel; Publication [2] switched to the Laplace kernel due to its sharper peak under bi-level optimization.

Key Idea

Instead of estimating a collision probability directly, we measure how far the residual distribution $p_{\bar{h}}$ is from the ideal zero-risk Dirac-delta distribution p_δ via MMD. This distributional viewpoint (i) requires only samples, (ii) admits a principled reduced-set mechanism, and (iii) drives the planner toward zero-collision trajectories as $r^{emp}_{MMD} \to 0$.

4.3.2. Improving Sample Efficiency Via Reduced-Set

Consider the setting of Fig.9 where the ego vehicle trajectory is deterministic but obstacle motion is uncertain, represented by a multi-modal distribution, and we have access to a samples drawn from this distribution. Estimating risk, as shown in the previous sections, require computing collision residual $\bar{h}(\mathbf{x}, \tau)$ over all drawn samples. This can be computationally heavy, e.g if the collision checks are expensive. Thus, a key question arises: Is it possible to discard few sampled trajectories without compromising the efficacy of the risk estimation? This subsection provides an answer to this question which is one of the key results of the thesis.

One of the key features of r^{emp}_{MMD} is that it allows us to leverage tools from RKHS embeddings and MMD to systematically improve sample efficiency. This motivates the concept of a *reduced-set*: a carefully selected and possibly reweighted

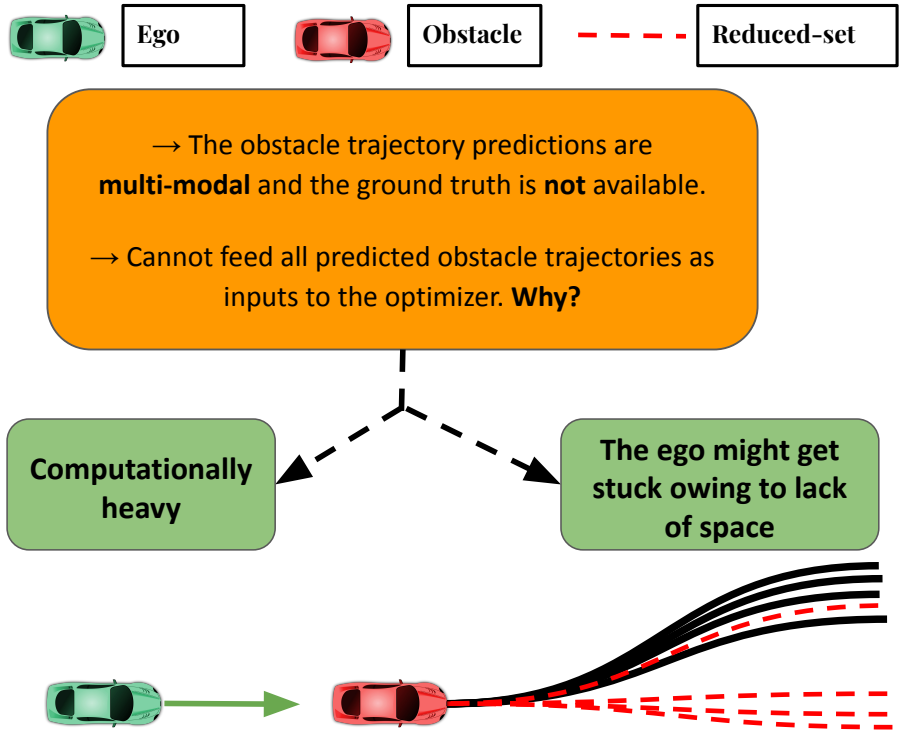


Figure 9: Evaluating collision constraints over all samples from the uncertainty source is the computational bottleneck in risk-aware planning. The reduced-set replaces the full sample set with a smaller, representative subset of samples that captures the maximum information about the total samples, thereby reducing the number of expensive constraint evaluations.

subset of samples that preserves the essential statistical properties of the full sample set in RKHS. Classical risk surrogates such as CVaR or SAA do not naturally admit an analogous reduced-set interpretation, because discarding samples in those settings typically reduces estimator fidelity. In contrast, the reduced-set formulation provides a principled way to retain the most informative samples while respecting the computational limits of planning and MPC. The purpose of this section is to explain the underlying principle; the full optimization formulations are developed later in the thesis for the different application settings.

More formally, let ${}^{l}\tau^{'}, l = 1,2,3,\ldots N^{'}$ be the *reduced-set* formed by subselecting $N^{'}$ samples out of N samples of τ. Moreover, let us attach weights ${}^{l}\beta$ with every ${}^{l}\tau^{'}$ such that $\sum_{l} {}^{l}\beta = 1$. Let ${}^{l}\overline{h}^{'}$ be the samples of $\overline{h}$ evaluated on ${}^{l}\tau^{'}$. The empirical RKHS embeddings of p_{τ} and $p_{\overline{h}}$ computed using ${}^{l}\tau^{'}$ and ${}^{l}\overline{h}^{'}$ are given by :

$$\hat{\mu}[\tau^{'}] = \sum_{l} {}^{l}\beta\,\phi({}^{l}\tau^{'}), \qquad \hat{\mu}[\overline{h}^{'}] = \sum_{l} {}^{l}\beta\,\phi({}^{l}\overline{h}^{'}) \tag{4.11}$$

Now, Theorem 1 from [50] shows that if $\hat{\mu}[\tau^{'}]$ is close to $\hat{\mu}[\tau]$ in the MMD sense, then $\hat{\mu}[\overline{h}^{'}]$ will also be close to $\hat{\mu}[\overline{h}]$ in the same sense. Consequently, r^{emp}_{MMD} com-

puted using $\hat{\mu}[\overline{h'}]$, which requires only $N^{'}$ predicted obstacle trajectories, will remain close to the quantity computed from the larger set of N samples. Equivalently, if we reduce the sample size while keeping $\left\|\hat{\mu}[\tau^{'}]-\hat{\mu}[\tau]\right\|^2_{\mathscr{H}}$ small, then the loss in risk-estimation accuracy will also be small. There are two levers for doing so. The first is a *random reduced-set*: one randomly selects a subset of N' samples and then optimizes only the weights ${}^{l}\beta$ to best approximate the full embedding. The second is a *jointly optimized reduced-set*: one simultaneously optimizes the selection of which samples to retain, the weights ${}^{l}\beta$, and the kernel parameter σ. We present the first approach below. Next, we develop the full jointly optimized formulation.

Random Reduced-Set with Optimal Weights. In this approach, the $N^{'}$ *reduced-set* samples ${}^{l}\tau^{'}$ are simply a random sub-selection from N samples of τ. We then optimize the weights of the ${}^{l}\tau^{'}$ through the following optimization problem.

$$\min_{{}^{l}\beta} \left\| \frac{1}{N}\sum_{i=1}^{i=N}\phi({}^{i}\tau) - \sum_{l=1}^{l=N^{'}} {}^{l}\beta\,\phi({}^{l}\tau^{'}) \right\|^2_{\mathscr{H}} \tag{4.12a}$$

$$\sum_{l} {}^{l}\beta = 1 \tag{4.12b}$$

Using the kernel trick, optimization (4.12a)-(4.12b) can be rephrased as an equality-constrained QP. Conceptually, (4.12a)-(4.12b) distills the information contained in a larger set of predicted obstacle trajectories into the reduced-set by reweighting the retained samples.

Optimal Reduced-Sets, Weights and Kernel Parameters We now present the optimized reduced-set construction central to MMD-OPT. We formulate this as a bi-level optimization, where $\mathbf{O}$ stacks the N samples of τ row-wise:

$$\min_{\lambda,\sigma} \left\| \frac{1}{N}\sum_{i=1}^{i=N}\phi({}^{i}\tau) - \sum_{l=1}^{l=N^{'}} {}^{l}\beta^{*}\phi({}^{l}\tau^{'}) \right\|^2_{\mathscr{H}} \tag{4.13a}$$

$$F_{\lambda}(\mathbf{O}) = ({}^{1}\tau^{'}, {}^{2}\tau^{'}, \ldots, {}^{N'}\tau^{'}) \tag{4.13b}$$

$$\begin{aligned} {}^{l}\beta^{*} = \quad & \arg\min_{{}^{l}\beta} \left\| \frac{1}{N}\sum_{i=1}^{i=N}\phi({}^{i}\tau) - \sum_{l=1}^{l=N^{'}} {}^{l}\beta\,\phi({}^{l}\tau^{'}) \right\|^2_{\mathscr{H}} \\ & \text{s.t.} \sum {}^{l}\beta = 1 \end{aligned} \tag{4.13c}$$

The inner level solves for optimal weights ${}^{l}\beta^{*}$ given a fixed subset and kernel. The outer level searches over the subset selection effected by Function $\mathbf{F}$ (details below), parametrized by parameter λ, and along with it, also finds the optimal

kernel parameter σ. This jointly eliminates the variance of random selection (the fixed-selection baseline of Section 4.3.2) and yields a tighter embedding $\hat{\mu}[\tau']$.
Selection Function F_λ: Let $\lambda \in \mathbb{R}^N$, with $|\lambda_t|$ encoding the value of including the t^{th} row of $\mathbf{O}$ in the reduced-set. F_λ is then defined through:

$$\mathscr{S} = \text{Argsort}\{|\lambda_1|, |\lambda_2|, \ldots, |\lambda_N|\} \tag{4.14a}$$

$$\mathbf{O}^{'} = [\mathbf{O}_{t_1}, \mathbf{O}_{t_2}, \ldots, \mathbf{O}_{t_N}], \forall t_e \in \mathscr{S} \tag{4.14b}$$

$$F_\lambda(\mathbf{O}) = \mathbf{O}^{'}_{t_{N-N'}:t_N} = ({}^1\tau^{'}, {}^2\tau^{'}, \ldots, {}^{N'}\tau^{'}) \tag{4.14c}$$

Rows of $\mathbf{O}$ are sorted by $|\lambda_t|$ and the top N' are chosen, so solving (4.13a)-(4.13c) reduces to finding the right λ. Alg. 1 combines gradient-free Cross Entropy Method (CEM) [91] with equality-constrained Quadratic Programming (QP).

Validating the Optimal Reduced-Set: To empirically demonstrate the importance of the optimal *reduced-set*, we sample 500 obstacle trajectories to form $\mathbf{O}$. From this, N' samples are selected either randomly or via the bi-level optimization (4.13a)–(4.13c).

We evaluate the quality of the reduced set by computing the average, minimum, and maximum of $\left\|\hat{\mu}[\tau^{'}] - \hat{\mu}[\tau]\right\|^2_{\mathscr{H}}$ over multiple random selections, and compare it with the optimal *reduced-set*. As shown in Fig. 10(b), the optimal *reduced-set* achieves performance close to the best-case random selection, with more pronounced gains at smaller N'. Since exhaustive search is computationally prohibitive in practice, (4.13a)–(4.13c) provides an efficient one-shot alternative.

Fig. 10(c) shows that this advantage extends to the embedding of constraint residuals, where $\left\|\hat{\mu}[\overline{f}] - \hat{\mu}[\overline{f'}]\right\|^2_{\mathscr{H}}$ remains close to the best achievable value. Together, these results validate that minimizing $\left\|\hat{\mu}[\tau^{'}] - \hat{\mu}[\tau]\right\|^2_{\mathscr{H}}$ and $\left\|\hat{\mu}[\overline{f}] - \hat{\mu}[\overline{f'}]\right\|^2_{\mathscr{H}}$ leads to minimal loss in estimating r^{emp}_{MMD}.

To quantify this, we compute

$$\Delta_{r_{MMD}} = \left| \left\| \sum_{j=1}^{N} \frac{1}{N} \phi(\overline{f}^j) - \sum_{j=1}^{N} \frac{1}{N} \phi(\delta^j) \right\|^2_{\mathscr{H}} - \left\| \sum_{l=1}^{N'} \beta^l \phi(\overline{f'}^l) - \sum_{l=1}^{N'} \frac{1}{N'} \phi(\delta^l) \right\|^2_{\mathscr{H}} \right|. \tag{4.15}$$

The first term corresponds to r^{emp}_{MMD} computed using all N samples, while the second uses the reduced set. As shown in Fig. 10, the optimal *reduced-set* yields $\Delta_{r_{MMD}}$ close to the average error obtained via exhaustive random sampling.

Physical Manifestation: We now present additional empirical analysis to further understand the optimal *reduced-set*. In particular, we investigate whether the selected trajectory samples carry physical significance.

Consider Fig. 11, which illustrates a synthetic multi-modal obstacle trajectory distribution corresponding to two distinct lane-change intents. This distribution is

Algorithm 1: Reduced-Set Selection, Weight Optimization, and Kernel Tuning using the Cross-Entropy Method (CEM)

e_{cem} = Maximum number of iterations, N= size of $\mathbf{O}$, $N^{'}$ = size of *reduced-set*

Initialize mean and covariance ν_e, Σ_e, respectively, at $e = 0$ for sampling and sorting λ and the kernel σ parameter respectively .

for $e = 1, e \leq e_{cem}, e++$ **do**

  Draw n_{cem} samples $({}^{1}\lambda, {}^{2}\lambda, {}^{m}\lambda, \ldots {}^{n_{cem}}\lambda)$ and $({}^{1}\sigma, {}^{2}\sigma, {}^{m}\sigma, \ldots {}^{n_{cem}}\sigma)$ from $\mathcal{N}(\nu_e, \Sigma_e)$.

  ${}^{m}\lambda = ({}^{m,1}\lambda, {}^{m,2}\lambda, \ldots {}^{m,N}\lambda)$ `// The dimension of` ${}^{m}\lambda$ `is equal to the rows of` $\mathbf{O}$

  Initialize *CostList* = []

  for $m = 1, m \leq n_{cem}, m++$ **do**

    $({}^{1}\tau^{'}, {}^{2}\tau^{'}, {}^{l}\tau^{'}, \ldots, {}^{N^{'}}\tau^{'}) = F_{{}^{m}\lambda}(\mathbf{O})$ `// Reduced-Set selection formulated as sorting of the elements of` ${}^{m}\lambda$

    ${}^{l}\beta^{*} \leftarrow$ Solve optimization (4.13c) with the *reduced-set* formed above and with kernel parameter ${}^{m}\sigma$

    $cost_{mmd} \leftarrow$ Compute MMD cost in (4.13c) using ${}^{l}\beta^{*}$

    append $cost_{mmd}$ to *CostList*

  end

  $EliteSet \leftarrow$ Select top n_{elite} samples of $({}^{m}\lambda)_{m=1}^{m=n_{cem}}$ and $({}^{m}\sigma)_{m=1}^{m=n_{cem}}$ with lowest cost from *CostList*.

  $\nu_{e+1}, \Sigma_{e+1} \leftarrow$ Update distribution based on $EliteSet$ `// Empirical mean and covariance estimate from the EliteSet samples`

end

return $({}^{1}\tau^{'}, {}^{2}\tau^{'}, \ldots, {}^{N^{'}}\tau^{'})$, *weights* ${}^{l}\beta^{*}$ *and optimal kernel hyperparameter* σ *corresponding to lowest cost in the* $EliteSet$

generated by sampling lateral offset set-points (from a binomial distribution) and forward velocities (from a Gaussian mixture model), and then mapping them to trajectories using a Frenet-frame-based planner. Owing to the discrete nature of lateral offsets, the relative likelihood of different lane-change maneuvers can be approximately inferred.

Applying Alg. 1 to construct the optimal *reduced-set*, we visualize the selected samples in Fig. 11. At early iterations, the reduced set is dominated by less likely trajectories. However, as the algorithm progresses and the outer MMD cost (4.13a) decreases, its composition shifts significantly toward more likely trajectories. Eventually, the proportion of samples in the reduced set closely matches the underlying ground-truth distribution.

This behavior is consistently observed across simulations and suggests an im-

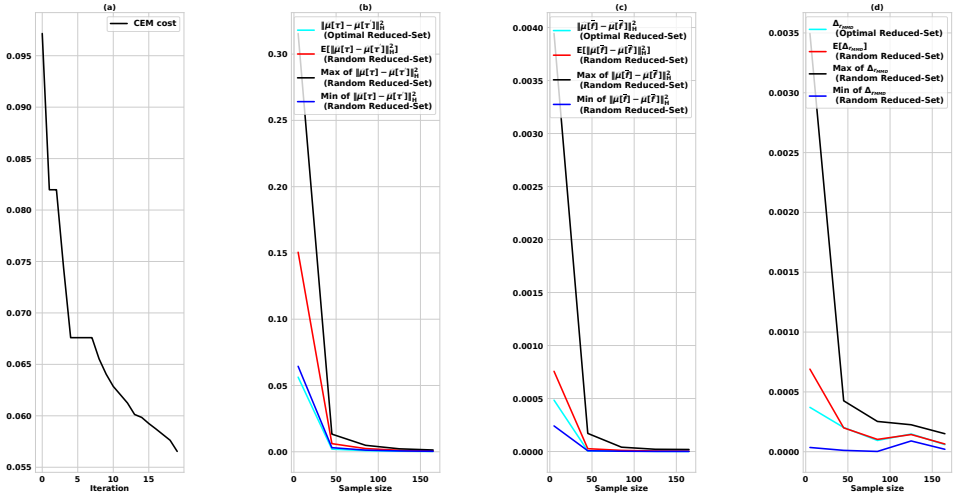


Figure 10: Validation of performance of Alg.1. Fig. (a) shows the progression of outer-level cost (4.13a) across iterations. Fig. (b) validates the efficacy of our optimal *reduced-set* selection. We sampled different random *reduced-sets* and computed the average, minimum and maximum of MMD $\|\hat{\mu}[\tau^{'}]-\hat{\mu}[\tau]\|^2_{\mathscr{H}}$. As shown in Fig. (b), our optimal *reduced-set* leads to an $\|\hat{\mu}[\tau^{'}]-\hat{\mu}[\tau]\|^2_{\mathscr{H}}$ that is very close to what can be achieved by exhaustively trying out different randomly selected *reduced-set*. Fig. (c) validates our hypothesis based on Theorem 1 and insights from [50]. As $\hat{\mu}[\tau^{'}]\rightarrow\hat{\mu}[\tau]$, the MMD $\|\hat{\mu}[\overline{f}]-\hat{\mu}[\overline{f^{'}}]\|^2_{\mathscr{H}}\rightarrow 0$. We can once again observe that our optimal *reduced-set* performs close to the best-case performance than can be hoped to be achieved by trying-out different random *reduced-sets*. Fig. (c) verifies that our optimal *reduced-set* selection ensures minimal loss in estimation of r^{emp}_{MMD} while reducing the sample size. To this end, we computed $\Delta_{r_{MMD}}$ through (4.15) which captures the estimation error between r^{emp}_{MMD} computed over a large set and the smaller *reduced-set*. As can be seen, r^{emp}_{MMD} computed over the optimal *reduced-set* samples perform close to the average performance that can be achieved by exhaustively trying out different random choices of the *reduced-set*. It should be noted that in real-world setting, exhaustive sampling is not possible and our bi-level optimization (4.13a)-(4.13c) provides a one-shot solution that is as good as exhaustive sampling in expectation.

portant emergent property of Alg. 1: although it does not explicitly use trajectory probabilities, minimizing the MMD implicitly biases the selection toward more representative (i.e., higher-probability) samples. Consequently, the resulting r^{emp}_{MMD} computed on the optimal *reduced-set* accurately captures collision risk, leading to safer motion plans.

4.3.3. Trajectory Optimizer

We minimize (4.5) subject to (4.6)–(4.7) using r^{emp}_{MMD} as the risk cost via a sampling-based optimizer that combines elements of CEM [91], MPPI [92], and its variants [93]. At each iteration, a batch of ego trajectories is sampled from a parametric Gaussian distribution, evaluated on both the cost and the risk surrogate, and used to refit the sampling distribution around the low-cost, low-risk modes. Since such sampling-based optimizers are inherently unconstrained, we add a projection step that pushes each sampled trajectory onto the feasible set defined by (4.7) (e.g., lane boundaries, control limits) before the cost is evaluated. This avoids

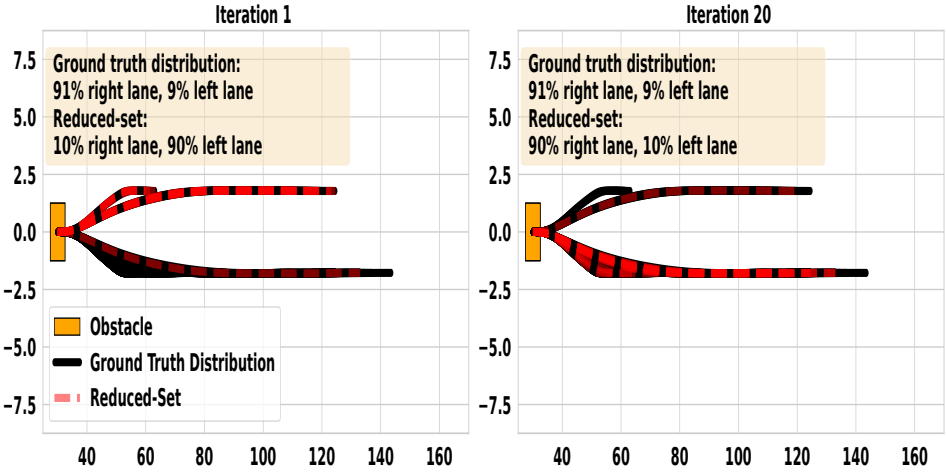


Figure 11: The figures demonstrate the physical significance of the trajectory samples used to form the optimal *reduced-set*. We generate a synthetic multi-modal trajectory distribution that captures the different lane-change intents of a dynamic obstacle (orange rectangle). More importantly, due to our specific generation procedure, we know the approximate probability of each lane-change intent. When Alg.1 starts, the optimal *reduced-set* in the first iteration consists mostly of the less likely samples due to the random initialization of the Gaussian distribution from which λ is sampled. **For example in the left figure, the *reduced-set* samples are mostly formed by the left lane-change maneuvers which is less likely**. But as Alg.1 makes progress, the composition significantly changes. For example, in the right figure, the majority of the *reduced-set* samples switch from left-lane to the right-lane maneuvers. Thus, by just minimizing MMD between two sets of trajectory samples, Alg.1 is able to implicitly infer which trajectory samples are more likely and accordingly split the composition of the optimal *reduced-set* between the less and more likely samples.

penalty-tuning and keeps the sampling distribution concentrated on kinematically consistent trajectories.

The overall procedure is summarized in Alg. 2[1]. The algorithm takes as input the iteration budget e_{max} and the optimal *reduced-set* samples $({}^{1}\tau', \ldots, {}^{N'}\tau')$ together with the kernel parameter σ produced by Alg. 1. Exploiting problem structure, we do not sample in the full state space. Instead, we parametrize ego trajectories in the Frenet frame by a small number of low-dimensional behavioral inputs $\mathbf{b}_i$ (e.g., desired lateral offset and longitudinal-velocity set-points over the horizon), drawn from a Gaussian $\mathcal{N}(\tilde{\nu}_e, \tilde{\Sigma}_e)$ (line 4). Each $\mathbf{b}_i$ is mapped to a full ego trajectory $\overline{\mathbf{x}}_i$ via a Frenet planner [107] (line 6). The raw trajectories $\overline{\mathbf{x}}_i$ are then projected to the feasible set (line 7) using a parallelizable constrained optimizer based on [108, 11], producing projected trajectories $\mathbf{x}_i$. Constraint residuals $\|\max(0, \mathbf{g}(\mathbf{x}_i))\|_2^2$ are evaluated on these projected trajectories (line 8), and the top $n_c (< n)$ samples with the smallest residuals form the $ConstraintEliteSet$ (line 9), ensuring that the risk and cost evaluations spend their budget on (approximately) feasible trajectories. An augmented cost c_{aug}, combining the state-dependent cost $c(\mathbf{x}_i)$, the MMD-based risk surrogate $r^{emp}_{MMD}(\mathbf{x}_i, {}^{l}\tau')$ computed against the reduced-set, and the constraint residual, is computed (line 10) and stored in $CostList$ (lines

[1]Right subscripts denote ego trajectory samples per iteration, while left superscripts denote obstacle trajectory samples.

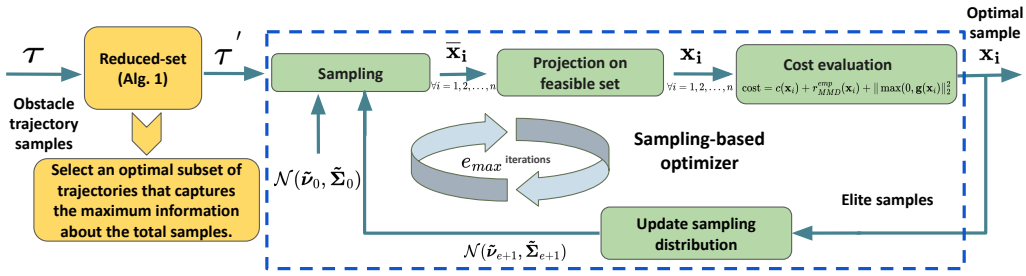


Figure 12: Overview of MMD-OPT. It consists of two main parts. The first part is the so-called *reduced-set* algorithm, which tells us which obstacle trajectories can be discarded without compromising our MMD-based surrogate's ability to capture the collision risk in a given scene. The second part is a sampling-based optimizer that minimizes the MMD-based collision risk cost. The sampling-based optimizer also exploits the problem structure to improve computational efficiency.

11–12). The best $n_{elite}(\leq n_c)$ samples from the *ConstraintEliteSet* define the *EliteSet* (line 13), which is used to update the sampling distribution (line 14) via:

$$\tilde{\nu}_{e+1} = (1-\eta)\tilde{\nu}_e + \eta \frac{\sum_{i=1}^{i=n_{elite}} \omega_i \, \mathbf{b}_i}{\sum_{i=1}^{i=n_{elite}} \omega_i}, \tag{4.16a}$$

$$\tilde{\Sigma}_{e+1} = (1-\eta)\tilde{\Sigma}_e + \eta \frac{\sum_{i=1}^{i=n_{elite}} \omega_i (\mathbf{b}_i - \tilde{\nu}_{e+1})(\mathbf{b}_i - \tilde{\nu}_{e+1})^T}{\sum_{i=1}^{i=n_{elite}} \omega_i} \tag{4.16b}$$

$$\omega_i = \exp\left(\frac{-1}{\gamma} c_{aug}(\mathbf{x}_i)\right) \tag{4.16c}$$

where γ and η denote the temperature and learning rate. The *EliteSet* follows CEM, while the update rules align with MPPI [92, 93]. By parameterizing trajectories via low-dimensional behavioral inputs $\mathbf{b}_i$, the method significantly improves computational efficiency for long-horizon planning.

4.4. Main Qualitative and Quantitative Results

Only a representative subset of the results is presented in this chapter; for the complete analysis the reader is referred to Publications [1, 2].

Key Result

On nuScenes with Trajectron++ predictions, MMD-OPT achieves the lowest collision rate compared to SAA and CVaR baselines in both in-lane and lane-change settings, with $\sim 2\times$ improvement in the lane-change setting at $N' = 25$.

4.4.1. Qualitative Analysis

Premise and setup. To empirically check that minimizing r^{emp}_{MMD} actually pulls the residual distribution $p_{\bar{h}}$ toward the ideal zero-risk distribution p_δ, we run Alg. 2 on a controlled synthetic scene. The ego vehicle must avoid a *static* obstacle whose

Algorithm 2: Sampling-Based Optimizer to Solve (4.5)-(4.7) with r^{emp}_{MMD} as the risk cost

e_{max} = Maximum number of iterations, *reduced-set* samples of obstacle trajectories $({}^1\tau', {}^2\tau', \ldots {}^{N'}\tau')$ and kernel parameter σ from Alg. 1
Initiate mean $\tilde{\nu}_e, \tilde{\Sigma}_e$, at $e = 0$ for sampling Frenet-Frame behavioural inputs $\mathbf{b}$
for $e = 1, e \leq e_{max}, e{+}{+}$ **do**
    Draw n samples $(\mathbf{b}_1, \mathbf{b}_2, \ldots \mathbf{b}_i, \ldots, \mathbf{b}_n)$ from $\mathcal{N}(\tilde{\nu}_e, \tilde{\Sigma}_e)$
    Initialize *CostList* = []
    Query Frenet-planner for $\forall \mathbf{b}_i$: $\bar{\mathbf{x}}_i = \text{Frenet Planner}(\mathbf{b}_i)$
    Project to Constrained Set

$$\mathbf{x}_i = \arg\min_{\mathbf{x}_i} \frac{1}{2}\|\bar{\mathbf{x}}_i - \mathbf{x}_i\|_2^2$$
$$\mathbf{f}(\mathbf{x}_i) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{x}_i) \leq 0$$

    Define constraint residuals: $\|\max(0, \mathbf{g}(\mathbf{x}_i))\|_2^2$
    $ConstraintEliteSet \leftarrow$ Select top n_c samples of $\mathbf{b}_i, \mathbf{x}_i$ with lowest constraint residual norm.
    Define $c_{aug} = c(\mathbf{x}_i) + r^{emp}_{MMD}(\mathbf{x}_i, {}^l\tau') + \|\max(0, \mathbf{g}(\mathbf{x}_i))\|_2^2$
    $cost \leftarrow c_{aug}$, over $ConstraintEliteSet$, $\forall i$
    append each computed *cost* to *CostList*
    $EliteSet \leftarrow$ Select top n_{elite} samples of
    $(\mathbf{b}_i, \mathbf{x}_i)$ with lowest cost from $CostList$.
    $(\tilde{\nu}_{e+1}, \tilde{\Sigma}_{e+1}) \leftarrow$ Update distribution based on
    $EliteSet$
end
return *Frenet parameter* $\mathbf{b}_i$ *and* $\mathbf{x}_i$ *corresponding to lowest cost in the* $EliteSet$

position is uncertain: we model the obstacle centre as a Gaussian distribution, draw N position samples, form the reduced-set via Alg. 1, and then track, iteration by iteration, both the optimized trajectory and the empirical distribution of the constraint residual $\bar{h}(\mathbf{x}, \tau)$ evaluated on those samples. This isolates the effect of the risk surrogate - with the obstacle static and known up to a simple distribution, any residual risk in the solution is attributable to the optimizer, not to the uncertainty model.

Observations. Fig. 13 shows the result. In early iterations, the sampled ego trajectories (top row, red) pass close to the uncertain obstacle region (blue-shaded), and the histogram / KDE of $\bar{h}$ (bottom row) has a long right tail, indicating non-trivial probability of constraint violation. As the CEM update concentrates the sampling distribution on lower-cost behavioural inputs, the trajectory moves clear of the uncertainty cone, and the residual distribution collapses toward a Dirac-delta at

zero. By the final iteration, essentially all the mass of $p_{\overline{h}}$ sits at $\overline{h} = 0$, empirically confirming the claim from (4.9) that driving $r^{emp}_{MMD} \to 0$ is equivalent to driving $p_{\overline{h}} \to p_{\delta}$, and hence the probability of collision to zero.

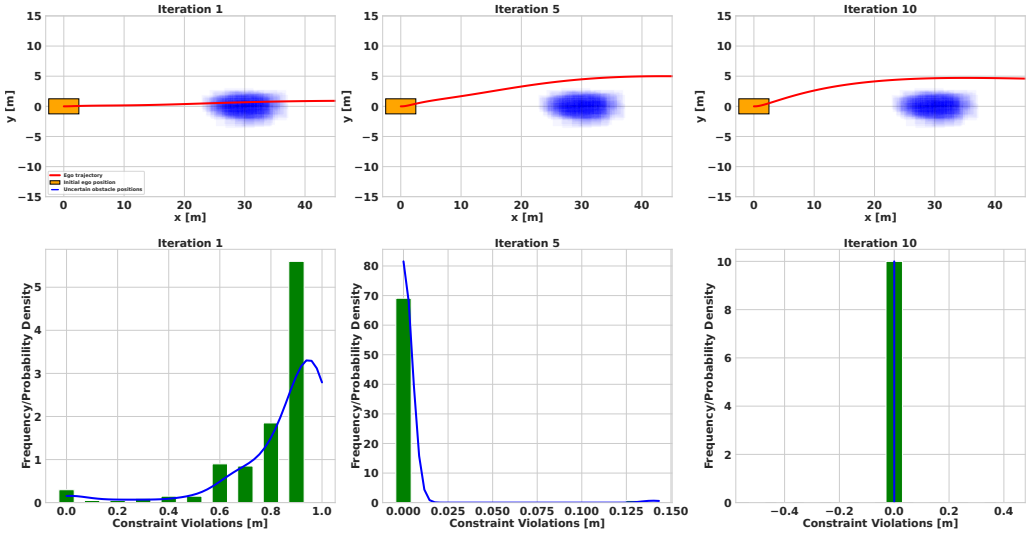


Figure 13: Top: iteration-wise evolution of the optimal trajectory (red) avoiding uncertain static obstacles (blue-shaded region). Bottom: histogram of $\overline{h}$ with kernel density estimate. As iterations progress, $p_{\overline{h}}$ converges to a Dirac delta distribution, confirming risk reduction.

4.4.2. nuScenes Dataset with Trajectron++ Predictor

Premise. We consider the nuScenes [94] benchmark and the state-of-the-art Trajectron++ predictor [97] to provide predictions for potential future motions of the neighboring vehicles. Our goal is to analyze how our MMD based approach performs in this real-world setting. To this end, we run Alg.2 with different risk costs: r^{emp}_{MMD}, r_{SAA} (defined in (2.4)), and r^{emp}_{CVaR} (defined in (2.5)). We individually tune each run to ensure the optimizer converges to a solution with zero risk cost. Thus, in this way, we negate the trajectory optimizer bias and exactly evaluate how well the finite sample approximation of each risk metric truly captures the collision risk

Setup. We evaluate on 160 scenes drawn from the nuScenes validation split. For each scene, Trajectron++ provides a set of predicted future trajectories (samples from the learned predictive distribution) for every nearby agent around the ego vehicle. These samples are fed into the planning pipeline as the obstacle distribution p_{τ}. The 160 scenes are partitioned into two driving settings that stress different aspects of the risk surrogate: (i) an *in-lane* setting, where the ego is constrained to its current lane and must regulate speed/longitudinal spacing in response to the stochastic predicted motion of the lead vehicle; and (ii) a *lane-change* setting, where the ego executes a full overtaking/lane-change maneuver

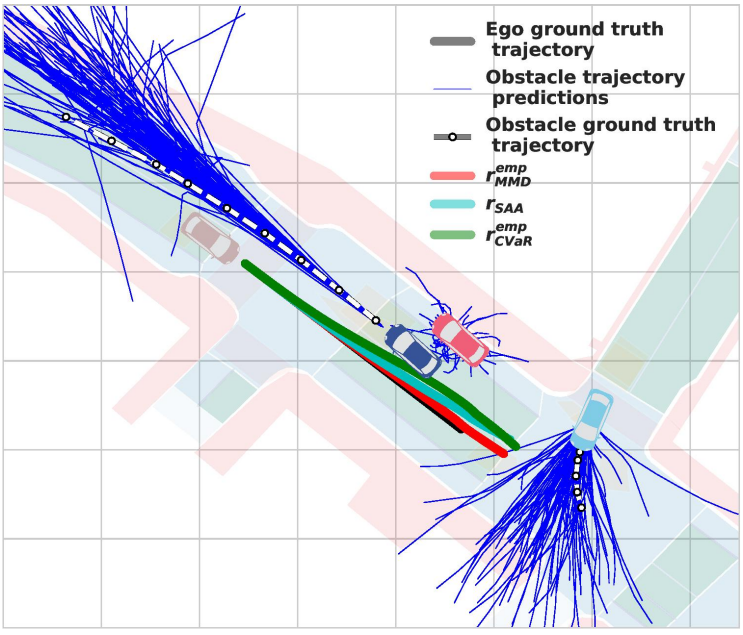


Figure 14: Risk-aware planning at an unprotected intersection. Blue trajectories: multimodal obstacle predictions. The trajectory from r^{emp}_{MMD} (closest to ground-truth) correctly identifies the more likely left-turn intent of the cyan vehicle, while r^{emp}_{CVaR} and r_{SAA} produce overly conservative deviations.

with adjacent-lane traffic, making the risk term substantially more sensitive to the multi-modality of the predictions. For each scene we draw $N = 500$ predicted obstacle samples to form the full set $\mathbf{O}$, and use Alg. 1 to construct a reduced-set of size $N' \in \{5, 10, 15, 20, 25\}$; the trajectory optimizer (Alg. 2) then solves (4.5)–(4.7) with r^{emp}_{MMD} computed over the reduced-set. We compare against two sampling-based baselines under the same optimizer and identical reduced-set budget: r_{SAA} (Sample Average Approximation) and r^{emp}_{CVaR} (empirical CVaR surrogate [40]). To measure collisions, the predicted obstacle trajectories are split into an *optimization set* and a *validation set*. Only the optimization set is used to construct the reduced-set and to evaluate r^{emp}_{MMD}, r_{SAA}, and r^{emp}_{CVaR} inside the trajectory optimizer 2; the validation set is never seen by the planner and is at least two orders of magnitude larger than the optimization set. We then check the computed optimal ego trajectory against each obstacle-trajectory sample in the validation set and report the collision percentage as the fraction of validation-set samples that the ego trajectory collides with.

Results. Fig. 14 shows a representative unprotected-intersection scene. The cyan vehicle has two likely maneuvers: going straight and turning left. Among the three surrogates, r^{emp}_{MMD} produces the trajectory closest to the recorded ground-truth ego motion, because the reduced-set machinery implicitly upweights the more likely left-turn mode. The r_{SAA} and r^{emp}_{CVaR} baselines, lacking an analogous mode-

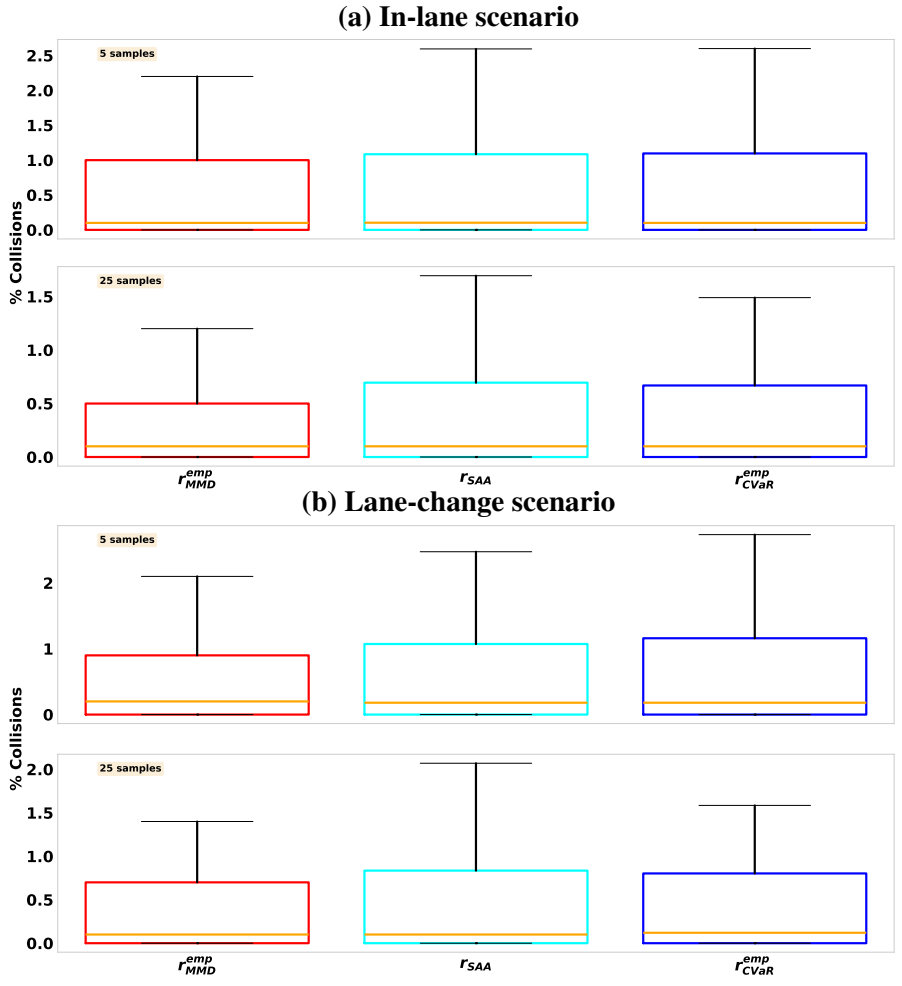


Figure 15: Collision-rate comparison of r^{emp}_{MMD}, r_{SAA}, and r^{emp}_{CVaR} on nuScenes with Trajectron++ [97] predictions. Our surrogate r^{emp}_{MMD} achieves the lowest collision rate in both scenarios.

discrimination mechanism at low sample counts, plan overly conservative detours. Quantitatively (Fig. 15), r^{emp}_{MMD} achieves the lowest collision rate across both the in-lane and lane-change settings and across the full sweep of N', with the margin widening as N' shrinks. At $N' = 25$ in the lane-change setting, r^{emp}_{MMD} attains an approximately $2\times$ reduction in collisions over the best-performing baseline.

4.4.3. Computation Time

On an RTX 3080 laptop GPU (16 GB), the reduced-set optimization (Alg. 1) runs in 0.006–0.007 s, and the downstream trajectory optimizer (Alg. 2) runs in 0.02–0.07 s for $N' \in [10, 50]$. On an RTX 3090 desktop GPU (16 GB), MMD-OPT consumes roughly 5–8 ms for reduced-set construction and 10–23 ms per planning cycle across 1–3 obstacles and $N' \in [5, 50]$, enabling MPC at approximately 50 Hz.

Artefacts and code.. Open-source implementations are available at `https://github.com/arunkumar-singh/RKHS_Stochastic_Traj_Opt` (Publication [1]) and `https://github.com/Basant1861/MMD-OPT` (Publication [2]).

4.5. Conclusion

This chapter addressed collision risk under uncertain obstacle motion. Tractability (C1) is handled via the MMD-based sample surrogate, and sample efficiency (C2) via joint optimization of reduced-set selection, sample weights, and kernel bandwidth. On nuScenes with Trajectron++ predictions, MMD-OPT achieves lower collision rates than SAA and CVaR baselines in low-sample regimes.

Next Chapter

Chapter 5 shifts the uncertainty source from obstacle prediction to stochastic ego dynamics, where the reduced-set mechanism is extended to control-space sampling.

5. TRAJECTORY OPTIMIZATION UNDER STOCHASTIC DYNAMICS LEVERAGING MAXIMUM MEAN DISCREPANCY (PUBLICATION [3])

At a Glance

- **Setting:** Stochastic ego dynamics; nominal acceleration and steering commands are perturbed, inducing a state-trajectory distribution.
- **Challenges addressed:** C1 (non-Gaussian, nonlinear-dynamics-induced risk) and C2 (reducing N^2 rollout-based evaluations of $\bar{h}$).
- **Method:** MMD-based surrogate with a reduced-set that selects N out of N^2 rollouts jointly with sample weights and kernel bandwidth; a CEM+MPPI+convex-optimization trajectory optimizer.
- **Key Result:** On CARLA T5/T10, with both methods evaluated under identical perturbation settings, r^{emp}_{MMD} achieves 0% collision at $N = 2$ in the most challenging Beta-noise case where CVaR reaches up to 45%.

5.1. Introduction

Unlike Chapter 4, where uncertainty originates in obstacle prediction, this chapter considers stochastic ego dynamics: acceleration and steering commands are perturbed, inducing a distribution over state rollouts. Sampling-based optimizers evaluate control candidates by many perturbed rollouts, which is expensive when each rollout requires costly simulation or collision checking. The same RKHS/MMD framework from Chapters 1-2 is specialized here, but the reduced-set now acts on state rollouts rather than obstacle samples.

The pipeline in Fig. 16 draws $\tilde{N}$ stochastic rollouts and evaluates a risk statistic (e.g. CVaR) on safety-distance violations. The question is how to retain reliability when only a very small number of expensive constraint evaluations is affordable.

5.2. Problem Formulation

We reuse the Frenet-frame representation of Chapter 4; the difference is that uncertainty now enters through the ego dynamics. Let s and d denote longitudinal and lateral displacements, and ψ the heading relative to the path. The risk-aware stochastic trajectory optimization problem is given by:

$$\min_{\mathbf{a},\theta} w_1 E[c(\mathbf{x})] + w_2 r(\mathbf{x}) + w_3 \left\| \begin{matrix} \mathbf{a} \\ \theta \end{matrix} \right\|_2^2, \tag{5.1a}$$

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, a_k + \varepsilon_{a,k}, \theta_k + \varepsilon_{\theta,k}), \qquad \mathbf{x}_0 \sim p_0, \tag{5.1b}$$

$$\theta_{min} \leq \theta_k \leq \theta_{max}, a_{min} \leq a_k \leq a_{max} \forall k, \tag{5.1c}$$

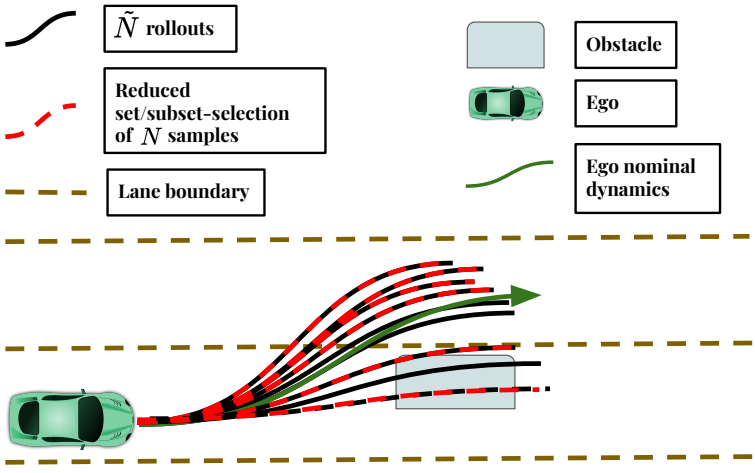


Figure 16: A standard pipeline for risk-aware optimization based on control sampling, along with our improvement. This class of approaches relies on simulating the vehicle's forward dynamics to obtain $\tilde{N}$ samples from the state trajectory distribution, which are then used to estimate risk. Our work provides a novel risk surrogate and a systematic way of estimating it using a reduced number (N) of state trajectory samples (also known as the *reduced-set*).

where $c(.)$ represents the state-dependent cost function. The vector $\mathbf{x}_k = (s_k, d_k, \psi_k, \dot{\psi}_k, v_k)$ represents the vehicle's state at time step k. The vector $\mathbf{x}$ denotes the concatenated state trajectory, with dynamics f adopted from [90] and v_k representing longitudinal velocity. Control inputs a_k (acceleration) and θ_k (steering) are stacked to form $\mathbf{a}$ and θ. Stochasticity is introduced via control perturbations $\varepsilon_{a,k}$ and $\varepsilon_{\theta,k}$, with stacked vectors $\varepsilon_a \sim p_a$ and $\varepsilon_\theta \sim p_\theta$, inducing a trajectory distribution $p_{\mathbf{x}}$ through f. We allow control-dependent noise (e.g., slippage at higher inputs) and make no parametric assumptions, relying solely on sampling and rollouts.

The first term in (5.1a) minimizes the expected state cost, typically addressing path-following errors where average performance suffices. The second term captures risk in the state trajectory $\mathbf{x}$, accounting for higher-order noise characteristics, including variance, skewness, and kurtosis. Intuitively, it represents the probability of an event (e.g., collision) for a given control sequence. The last term in (5.1a) penalizes large control inputs, with weights w_i tuning the ego-vehicle's risk-seeking behavior. Control bounds are enforced through (5.1c).

5.3. Main Algorithmic Results

1. **[C1]** We specialize the MMD-based collision-risk surrogate of Chapter 4 to stochastic ego dynamics, where uncertainty in the state trajectory $\mathbf{x}$ is induced by perturbations $\varepsilon_a, \varepsilon_\theta$ on nominal acceleration and steering inputs.
2. **[C2]** We propose a bi-level reduced-set optimization that selects N out

of N^2 stochastic rollouts, along with the associated weights and kernel parameter, so that collision-constraint evaluation can be performed on the reduced-set without compromising risk-estimation accuracy.

3. **[C1, C2]** We develop a sampling-based trajectory optimizer (Alg. 3) that uses the reduced-set machinery to minimize the proposed risk surrogate in a receding-horizon MPC loop.

5.3.1. Risk as Difference of Distributions

Recall that the per-step collision constraint is $h_k(\mathbf{x}_k) \leq 0$, aggregated across time as $h(\mathbf{x}) = \max_k h_k(\mathbf{x}_k)$ (Eq. (2.1)), and the corresponding collision risk is defined as $r(\mathbf{x}) = P(h(\mathbf{x}) > 0)$. Because $\mathbf{x}$ is now a random variable induced by $\varepsilon_a, \varepsilon_\theta$ through the nonlinear dynamics f, this probability does not admit a tractable analytical form even for simple p_a, p_θ (challenge C1).

Following the distributional viewpoint developed in Chapter 4 (Section 4.3.1), we adopt the constraint-violation residual

$$\overline{h}(\mathbf{x}) = \max(0, h(\mathbf{x})),$$

which is itself a random variable with induced distribution $p_{\overline{h}}$. All mass of $p_{\overline{h}}$ lies to the right of $\overline{h} = 0$, and as $P(h(\mathbf{x}) > 0) \to 0$, $p_{\overline{h}}$ approaches a Dirac-Delta distribution p_δ concentrated at zero. We therefore adopt the MMD-based surrogate from Chapter 4:

$$r_{MMD} = \left\| \mu[\overline{h}] - \mu[\delta] \right\|^2_{\mathscr{H}}, \qquad r^{emp}_{MMD} = \left\| \hat{\mu}[\overline{h}] - \hat{\mu}[\delta] \right\|^2_{\mathscr{H}}. \tag{5.2}$$

The difference from Chapter 4 is that $\hat{\mu}[\overline{h}]$ is now computed from stochastic rollouts of the ego dynamics: ${}^i\varepsilon_a, {}^j\varepsilon_\theta$ are drawn i.i.d., and ${}^{ij}\mathbf{x} = \mathbf{x}({}^i\varepsilon_a, {}^j\varepsilon_\theta)$ are the induced state-trajectory samples. The empirical embedding is

$$\hat{\mu}[\overline{h}] = \sum_{i,j=1}^{i,j=N} \frac{1}{N^2} \phi(\overline{h}({}^{ij}\mathbf{x})), \tag{5.3}$$

so that obtaining r^{emp}_{MMD} requires N^2 rollouts of the stochastic dynamics and as many evaluations of $\overline{h}$. Theorem 1 ensures $r^{emp}_{MMD} \to r_{MMD}$ as $N \to \infty$.

Key Idea

Under stochastic dynamics, the embedding $\hat{\mu}[\overline{h}]$ in (5.3) naively needs N^2 rollouts and N^2 evaluations of $\overline{h}$. The reduced-set reduces this to N evaluations by jointly optimizing *which* rollouts to keep, their weights, and the kernel bandwidth - without loss in risk fidelity, thanks to Theorem 1.

5.3.2. Improving Sample Efficiency Via Reduced-Set

We extend the theory developed in Section 4.3.2 for uncertain obstacle motion to the present setting of stochastic ego dynamics. The embedding (5.3) requires N^2 rollouts of the stochastic dynamics and, more importantly, N^2 evaluations of $\overline{h}$. In driving scenarios with multiple obstacles or high-fidelity collision checks, each $\overline{h}$ evaluation is expensive, so the N^2 scaling is the dominant computational bottleneck.

Let ${}^{l}\mathbf{x}^{'}, l = 1,2,\ldots,N$ be a *reduced-set* of N rollouts chosen out of the N^2 samples ${}^{ij}\mathbf{x}$, with weights ${}^{l}\beta$, $\sum_l {}^{l}\beta = 1$. The empirical RKHS embeddings of the state trajectory and constraint residual distributions over the reduced-set are

$$\hat{\mu}[\mathbf{x}^{'}] = \sum_{l=1}^{l=N} {}^{l}\beta\,\phi({}^{l}\mathbf{x}^{'}), \qquad \hat{\mu}[\overline{h}^{'}] = \sum_{l=1}^{l=N} {}^{l}\beta\,\phi(\overline{h}({}^{l}\mathbf{x}^{'})). \tag{5.4}$$

Theorem 1 ensures that if $\hat{\mu}[\mathbf{x}^{'}]$ is close to $\hat{\mu}[\mathbf{x}]$ in the MMD sense, then $\hat{\mu}[\overline{h}^{'}]$ will be close to $\hat{\mu}[\overline{h}]$ in the same sense. Consequently, r^{emp}_{MMD} computed on N rollouts will remain close to that computed on all N^2 samples. As in Chapter 4, there are two levers for obtaining a small $\left\|\hat{\mu}[\mathbf{x}^{'}] - \hat{\mu}[\mathbf{x}]\right\|^2_{\mathscr{H}}$. The first is a *random reduced-set*: randomly select N out of N^2 rollouts and optimize only the weights ${}^{l}\beta$ against the full embedding. The second is a *jointly optimized reduced-set*: simultaneously optimize the selection of which rollouts to retain, the weights ${}^{l}\beta$, and the kernel parameter σ. We develop the latter here, which is the main algorithmic contribution of this chapter.

Optimal Reduced-Set, Weights and Kernel Parameters The same idea was developed in Section 4.3.2 of Chapter 4 for obstacle-trajectory samples; here we continue it for state rollouts. Let $\mathbf{O}$ stack the N^2 rollouts ${}^{ij}\mathbf{x}$ row-wise. Minimizing $\left\|\hat{\mu}[\mathbf{x}^{'}] - \hat{\mu}[\mathbf{x}]\right\|^2_{\mathscr{H}}$ over subset selection, weights, and kernel parameter yields the bi-level program:

$$\min_{\lambda,\sigma} \left\| \sum_{i,j=1}^{i,j=N} \frac{1}{N^2}\phi({}^{ij}\mathbf{x}) - \sum_{l=1}^{l=N} {}^{l}\beta^{*}\phi({}^{l}\mathbf{x}^{'}) \right\|^2_{\mathscr{H}} \tag{5.5a}$$

$$F_\lambda(\mathbf{O}) = ({}^{1}\mathbf{x}^{'}, {}^{2}\mathbf{x}^{'}, \ldots, {}^{N}\mathbf{x}^{'}) \tag{5.5b}$$

$${}^{l}\beta^{*} = \quad \arg\min_{{}^{l}\beta} \left\| \sum_{i,j=1}^{i,j=N} \frac{1}{N^2}\phi({}^{ij}\mathbf{x}) - \sum_{l=1}^{l=N} {}^{l}\beta\,\phi({}^{l}\mathbf{x}^{'}) \right\|^2_{\mathscr{H}} \tag{5.5c}$$

$$\text{s.t.} \sum_{l} {}^{l}\beta = 1$$

The inner level solves for optimal weights ${}^{l}\beta^{*}$ given a fixed subset and kernel; the outer level searches over subset selection (effected by F_λ as in Section 4.3.2) and

the kernel parameter σ. The program has the same bi-level structure as (4.13a)-(4.13c) in Chapter 4, but acts on state rollouts ${}^{ij}\mathbf{x}$ rather than obstacle-trajectory samples. It is solved with the same CEM-plus-QP procedure used for Alg. 1.

5.3.3. Trajectory Optimizer

Alg. 3 presents our approach for solving (5.1a)-(5.1c) when the risk cost is given by r^{emp}_{MMD}. It combines constrained gradient-free CEM [91], MPPI [93, 92], and convex optimization to iteratively refine low-risk, low-cost control inputs.

The algorithm initializes the sampling distribution (Line 2) and samples longitudinal velocity and lateral offset setpoints (Line 5), which are fed to a Frenet planner (Line 6) to generate trajectories. Using differential flatness [90], these are converted to accelerations and clipped to control bounds. Control perturbations are sampled, and N^2 rollouts of (5.1b) yield state-trajectory samples (Line 7). A *reduced-set* of N trajectories is selected (Line 8), and r^{emp}_{MMD} is estimated (Line 9). The lowest-risk n_c samples form the $ConstraintEliteSet$ (Line 10), from which costs are computed (Line 11) and stored (Line 12). A final $EliteSet$ of n_e samples is selected (Line 14) to update the sampling distribution (Line 15) using the same CEM/MPPI mean–covariance update and exponential weighting as in Chapter 4 (see (4.16a)-(4.16c)), with the only change that the elite samples here are the behavioural inputs $\mathbf{b}_q$ of Alg. 3 rather than those of Alg. 2. By parameterizing long-horizon control sampling through low-dimensional velocity and offset setpoints, Alg. 3 significantly improves computational efficiency.

5.4. Main Qualitative and Quantitative Results

Only a representative subset of the results is presented in this chapter; for the complete analysis the reader is referred to Publication [3].

Key Result

On CARLA T5 and T10 with $N = 2$ rollouts, r^{emp}_{MMD} achieves 0% collisions under Gaussian and Beta perturbations, while r^{emp}_{CVaR} incurs up to 45% collisions under high-noise Beta perturbations.

5.4.1. Baselines and Evaluation Metrics

Baselines. We compare our method with two risk-aware surrogates, evaluated under the same sampling-based optimizer (Alg. 3) and with the same per-step budget of expensive $h(\mathbf{x})$ evaluations, to isolate the effect of the surrogate from that of the optimizer: (i) the empirical CVaR surrogate $r^{emp}_{CVaR} = CVaR^{emp}_{\alpha}(\overline{h}(\mathbf{x}))$ of [40], which estimates the mean over the worst α-tail of the residual distribution; and (ii) a deterministic reference obtained by turning off all noise injection (no perturbations on $(\mathbf{a}, \theta)$), which quantifies the benefit of explicitly modelling stochasticity. Our method rolls out N^2 samples but evaluates h only on the N

Algorithm 3: Sampling-Based Optimizer to Solve (5.1a)-(5.1c)

M = Maximum number of iterations
Initiate mean ${}^{m}\nu, {}^{m}\Sigma$, at iteration $m = 0$ for sampling frenet parameters (velocity and lane-offsets) $\mathbf{b}$
for $m = 1, m \leq M, m{+}{+}$ **do**
  Initialize *CostList* = []
  Draw n samples $(\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_q, ..., \mathbf{b}_n)$ from $\mathcal{N}({}^{m}\nu, {}^{m}\Sigma)$
  Query Frenet Planner $\forall \mathbf{b}_q$: $(\mathbf{a}_q, \theta_q) =$ Frenet Planner$(\mathbf{b}_q), \forall q = (1, 2, \ldots, n)$
  Compute N samples each of $\varepsilon_a, \varepsilon_\theta$ and subsequently N^2 rollouts ${}^{ij}\mathbf{x}_q$ for $(\mathbf{a}_q, \theta_q)$ control trajectory. Repeat this process $\forall q = (1, 2, \ldots, n)$
  Choose N rollouts ${}^{l}\mathbf{x}_q^{'}$ out of N^2 ${}^{ij}\mathbf{x}_q$ through (5.5a)-(5.5c) and compute corresponding ${}^{l}\beta_q$ and kernel parameter σ_q. Repeat this process $\forall q = (1, 2, \ldots, n)$
  Compute $\hat{\mu}[\overline{h}^{'}]$ over the optimal *reduced-set* through (5.4) and subsequently r_{MMD}^{emp}. Repeat this $\forall q = (1, 2, \ldots, n)$
  $ConstraintEliteSet \leftarrow$ Select top n_c batch of $\mathbf{a}_q, \theta_q$, ${}^{ij}\mathbf{x}_q$, $\mathbf{b}_q$ with lowest r_{MMD}^{emp}
  Define $c_q = w_1 \sum_{i,j=1}^{i,j=N} c({}^{ij}\mathbf{x}_q) + w_2 r_{MMD}^{emp} + w_3 \left\| \begin{matrix} \mathbf{a}_q \\ \theta_q \end{matrix} \right\|_2^2$
  $cost \leftarrow c_q, \forall q$ in the $ConstraintEliteSet$
  append each computed *cost* to *CostList*
  $EliteSet \leftarrow$ Select top n_e samples of $(\mathbf{a}_q, \theta_q)$, ${}^{ij}\mathbf{x}_q$, $\mathbf{b}_q$ with lowest cost from *CostList*.
  $({}^{m+1}\nu, {}^{m+1}\Sigma) \leftarrow$ Update distribution based on $EliteSet$
end
return *Control Inputs* $\mathbf{a}_q$ *and* θ_q *corresponding to lowest cost in the* $EliteSet$

reduced-set samples, while the CVaR baseline rolls out N samples and evaluates h on all of them, so both consume the same number of h-evaluations per MPC step and the comparison is fair on the quantity that actually dominates runtime. In this chapter, we focus only on results comparing r_{CVaR}^{emp} with our MMD-based approach.

Evaluation metrics. Safety is captured by the percentage of runs ending in a collision. Dynamic-feasibility and lane-keeping are measured by the percentage of lane-constraint violations, normalized by total path length so that longer runs are not penalized. Progress is measured by the average and maximum longitudinal speed over the run, which together flag the degenerate cases of excessive conservativeness (a surrogate that avoids collisions only by stalling or crawling).

Table 2: (1 dynamic obstacle) Low Gaussian noise: $c_{a,1} = 0.1, c_{\theta,1} = 0.1$. High Gaussian noise: $c_{a,1} = 0.15, c_{\theta,1} = 0.15$. Low Beta noise: $c_{a,1} = 0.1, c_{\theta,1} = 0.005$. High Beta noise: $c_{a,1} = 0.15, c_{\theta,1} = 0.0075$. $c_{a,2} = 0.001$, $c_{\theta,2} = 0.001$

Noise	**Method**	**(% Collisions)**					
		N=2		**N=4**		**N=6**	
		Median	**Worst**	**Median**	**Worst**	**Median**	**Worst**
Low Gaussian	r^{emp}_{MMD}	**4.1**	**41.3**	**0.25**	**9.2**	**0.55**	**9**
	r^{emp}_{CVaR}	6.4	43.1	2.7	20.1	1.5	12.3
High Gaussian	r^{emp}_{MMD}	**7.8**	**49.2**	**1.8**	**11.1**	**1.35**	**11.4**
	r^{emp}_{CVaR}	9	46.3	3.5	19.7	1.6	11.3
Low Beta	r^{emp}_{MMD}	**3.95**	**37.4**	**0.6**	**12**	**0.5**	**8.9**
	r^{emp}_{CVaR}	8.5	55.6	3.5	28.3	1.6	12
High Beta	r^{emp}_{MMD}	**5.1**	**35.2**	**0.65**	**12.9**	**1.4**	**15.2**
	r^{emp}_{CVaR}	17.3	80.4	5.6	36.3	3.5	22.2

5.4.2. Benchmarking in Stand-Alone Trajectory Optimization Setting

Premise. Before closed-loop MPC, we first test single-shot behavior to isolate how each risk surrogate converts a small sample budget into a low-collision trajectory, without replanning effects. We fix the scene, solve (5.1a)–(5.1c) once, and estimate the resulting control sequence's ground-truth collision probability via Monte Carlo rollout. Alg.3 is run with r^{emp}_{MMD} and r^{emp}_{CVaR} from (2.5). Each run is tuned to reach zero risk cost, removing optimizer bias and directly assessing how well each finite-sample risk metric captures collision risk.

Setup. We use two benchmarks: 200 randomly generated static-obstacle scenes with varying sizes and placements, and one dynamic-obstacle scene with a scripted maneuver. In both, nominal acceleration and steering inputs are perturbed by Gaussian and Beta noise families. The Gaussian case follows Eqs. (5.6a)–(5.6b), while the skewed Beta case follows Eqs. (5.7a)–(5.7b) to stress surrogates that assume symmetric uncertainty. For each scene, we sweep $N \in \{2,4,6,8\}$ to assess degradation under shrinking sample budgets.

Observations. Across both benchmarks and noise families, trajectories optimized with r^{emp}_{MMD} achieve lower ground-truth collision rates than r^{emp}_{CVaR}. The gap is largest in the low-sample regime ($N = 2$), the key regime for real-time MPC. Under the skewed Beta noise, r^{emp}_{CVaR} shows a much larger collision spread than r^{emp}_{MMD}, indicating that the tail-mean estimator is more sensitive to the asymmetry of the

noise than the distribution-matching MMD surrogate is. Table 2 reports the median and worst-case collision rates across $N \in \{2,4,6\}$ for the dynamic obstacle benchmark. For more details, the reader can refer to Publication [3].

5.4.3. Benchmarking in MPC Setting using CARLA

Premise. The stand-alone results show that r^{emp}_{MMD} achieves lower collision probability with limited samples. In practice, however, the optimizer runs in a receding-horizon MPC loop, where the surrogate is re-evaluated over a sliding window of stochastic rollouts. Low overall collision rate does not necessarily imply temporally stable behavior: high surrogate variability can induce oscillatory controls. We therefore evaluate closed-loop performance under noisy dynamics in a high-fidelity simulator.

Setup. We conduct closed-loop MPC experiments in CARLA [98] on two maps with different clutter: T10 with 8 obstacles and T5 with 10 obstacles. Each run lasts 40 steps and is repeated over 50 randomized seeds. At every MPC step, nominal acceleration and steering commands are perturbed before simulator execution, making the applied controls stochastic and coupling the surrogate noise model with executed motion. The Gaussian noise distribution has the following form:

$$\varepsilon_{a,k} \sim |c_{a,1}a_k|\mathcal{N}(0,1) + c_{a,2}\mathcal{N}(0,1), \tag{5.6a}$$

$$\varepsilon_{\theta,k} \sim |c_{\theta,1}\theta_k|\mathcal{N}(0,1) + c_{\theta,2}\mathcal{N}(0,1), \tag{5.6b}$$

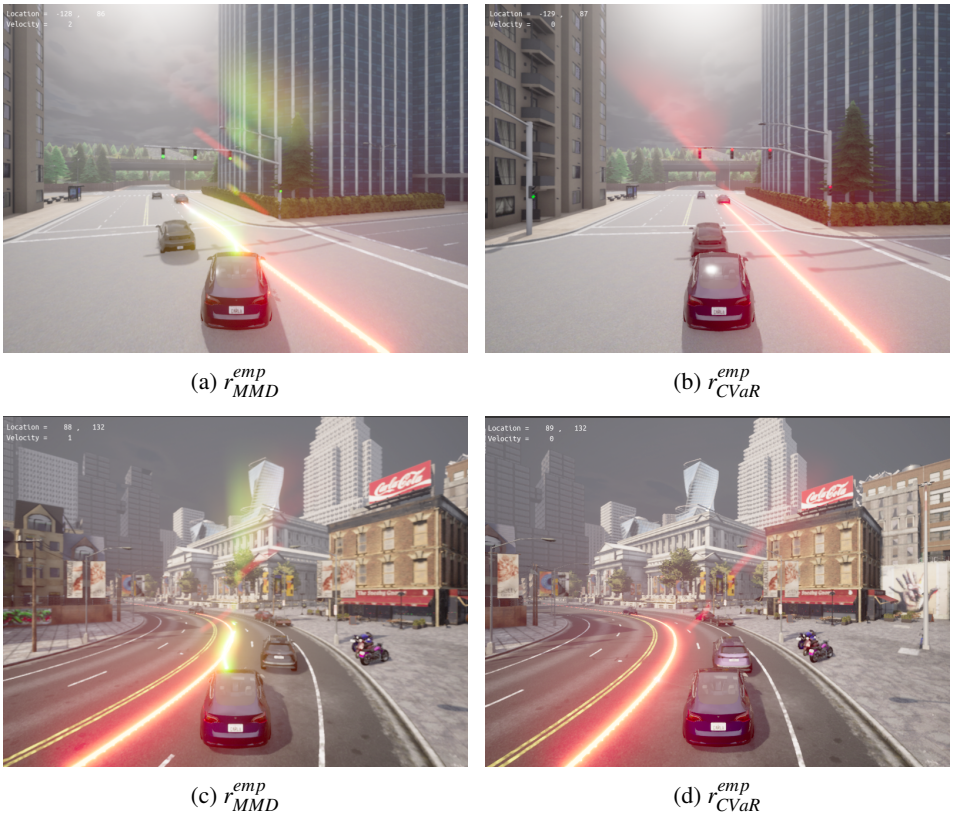
where $c_{a,i}$ and $c_{\theta,i}$ are positive constants. As shown, the noise has a control-dependent component and a constant component. We create different noise settings by varying $c_{a,i}$ and $c_{\theta,i}$. We also consider a setting where the r.h.s of (5.6a), (5.6b) are replaced by Beta distribution with probability density function $g(x;a,b) \propto x^{a-1}(1-x)^{b-1}$ where a,b are control dependent parameters. The Beta noise distribution has the following form:

$$\varepsilon_{a,k} \sim c_{a,1}\mathcal{B}(a,b) + c_{a,2}\mathcal{N}(0,1), \tag{5.7a}$$

$$\varepsilon_{\theta,k} \sim c_{\theta,1}\mathcal{B}(a,b) + c_{\theta,2}\mathcal{N}(0,1), \tag{5.7b}$$

$$a = \{2|a_k|, 2|\theta_k|\}, b = \{5|a_k|, 5|\theta_k|\} \tag{5.7c}$$

Qualitative Results Fig. 17 shows a representative closed-loop run in CARLA under Gaussian noise with the low-sample budget $N = 2$. The left column shows trajectories obtained by minimizing r^{emp}_{MMD}; the ego vehicle smoothly clears the obstacles in both T5 and T10. The right column shows trajectories from minimizing r^{emp}_{CVaR}; here the ego either gets stuck behind the obstacles or collides with them. With only $N = 2$ samples, the empirical CVaR estimate is very unreliable and changes sharply between consecutive MPC steps, so the risk signal used to guide the vehicle becomes inconsistent. The r^{emp}_{MMD} estimate, in contrast, remains much more stable across MPC steps at the same sample budget, providing a consistent signal that allows the ego to make steady forward progress past the obstacles.



(a) r^{emp}_{MMD} (b) r^{emp}_{CVaR}

(c) r^{emp}_{MMD} (d) r^{emp}_{CVaR}

Figure 17: MPC simulations in CARLA T5 (a)-(b) and T10 (c)-(d). The red line is the route/reference path. Trajectories result from minimizing r^{emp}_{MMD} (left column) and r^{emp}_{CVaR} (right column) for $N = 2$. The small number of residual samples yields a highly unreliable estimate of r^{emp}_{CVaR}, which changes sharply between consecutive MPC steps. As a result, the r^{emp}_{CVaR}-driven vehicle often gets stuck behind the obstacle and eventually collides. In contrast, r^{emp}_{MMD} provides a more consistent risk estimate to guide the ego vehicle.

Quantitative Results We summarize the closed-loop CARLA results in two regimes that differ only in the magnitude of the injected control noise. The *lower-noise* regime (Table 3) uses $c_{a,1} = c_{\theta,1} = 0.3$ for Gaussian noise and $c_{a,1} = c_{\theta,1} = 0.01$ for Beta noise, i.e., modest perturbations. Here, r^{emp}_{MMD} achieves 0% collisions in both T5 and T10 under both noise families, while r^{emp}_{CVaR} collides in 7.69% of T5-Gaussian runs and 16.67% of T10-Gaussian runs. The r^{emp}_{MMD} ego also maintains higher average and maximum speeds in every cell of the table. The slightly lower lane-violation numbers reported for r^{emp}_{CVaR} are a side-effect, not a genuine advantage: because the CVaR-driven runs end early in collisions or stalls, there is simply less path over which lane violations can be accumulated.

The *higher-noise* regime (Table 4) uses $c_{a,1} = c_{\theta,1} = 0.3$ for Gaussian noise and $c_{a,1} = c_{\theta,1} = 0.05$ for Beta noise, i.e., the control noise is substantially inflated and in particular the Beta component becomes strongly asymmetric. All methods degrade in this regime, but very unevenly: r^{emp}_{CVaR} jumps to 15% (T5-Gaussian),

45% (T5-Beta), and 17% (T10-Gaussian) collisions, while r^{emp}_{MMD} stays at 0% in three of the four cells and only 4% in the fourth (T10-Gaussian). The Beta-noise column is particularly informative: a 45% vs. 0% collision rate is a near-total breakdown of the tail-mean CVaR estimator under skewed noise at $N = 2$, whereas the MMD surrogate absorbs the asymmetry through its distributional comparison. Together, Tables 3 and 4 demonstrate that r^{emp}_{MMD} degrades much more gracefully than r^{emp}_{CVaR} as either the sample budget shrinks or the noise becomes non-Gaussian.

Table 3: $N = 2$, Rollout horizon 40 time steps, 50 experiments.
Gaussian noise: $c_{a,1} = c_{\theta,1} = 0.3$, Beta noise: $c_{a,1} = c_{\theta,1} = 0.01$.
$c_{a,2} = 0.3, c_{\theta,2} = 0.01$, Gaussian noise in the initial state.

Method	Town	% Collisions		% Lane Constr. Viol.		Avg. Speed (m/s)		Max. Speed (m/s)	
		Gaussian	Beta	Gaussian	Beta	Gaussian	Beta	Gaussian	Beta
r^{emp}_{MMD}	T5	**0**	**0**	2.39	1.18	2.59	2.68	3.69	5.08
r^{emp}_{CVaR}	T5	7.69	**0**	2.25	1.08	2.12	2.49	3.04	4.61
r^{emp}_{MMD}	T10	**0**	**0**	0.47	0.99	3.42	3.63	5.94	5.5
r^{emp}_{CVaR}	T10	16.67	**0**	**0.07**	0.55	2.8	3.02	5.19	5.17

Table 4: $N = 2$, Rollout horizon 40 times steps, 50 experiments.
Gaussian noise: $c_{a,1} = c_{\theta,1} = 0.3$, Beta noise: $c_{a,1} = c_{\theta,1} = 0.05$.
$c_{a,2} = 0.4, c_{\theta,2} = 0.01$, Gaussian noise in the initial state.

Method	Town	% Collisions		% Lane Constr. Viol.		Avg. Speed (m/s)		Max. Speed (m/s)	
		Gaussian	Beta	Gaussian	Beta	Gaussian	Beta	Gaussian	Beta
r^{emp}_{MMD}	T5	**0**	**0**	2.7	3.5	**2.59**	2.02	**4.06**	**4.07**
r^{emp}_{CVaR}	T5	15	45	2.14	**2.46**	2.08	1.53	3.45	3.4
r^{emp}_{MMD}	T10	4	**0**	0.5	5.11	**3.56**	2	7.01	3.73
r^{emp}_{CVaR}	T10	17	4	**0.1**	**3.23**	2.73	1.56	4.55	4.07

5.4.4. Computation Time

On an RTX 3090 desktop GPU, r^{emp}_{CVaR} runs at approximately 0.1 s per MPC step, while the full r^{emp}_{MMD} with reduced-set construction runs at approximately 0.18 s per step, corresponding to closed-loop rates of 5–10 Hz.

Artefacts and code.. An open-source implementation is available at `https://github.com/Basant1861/MPC-MMD`.

5.5. Conclusion

This chapter extended the framework to stochastic ego dynamics by constructing the reduced-set over state rollouts rather than obstacle samples. Stand-alone and CARLA MPC experiments show r^{emp}_{MMD} is more reliable than CVaR in low-sample

regimes, with lower collision rates and better progress. Together with Chapter 4, this confirms that the same RKHS/MMD framework specializes cleanly to two distinct uncertainty sources.

Next Chapter

Chapter 6 extends this framework to monocular vision-based navigation through a learned probabilistic collision model, where uncertainty enters via perception rather than prediction or dynamics.

6. VISION-BASED NAVIGATION AS RISK-AWARE TRAJECTORY OPTIMIZATION (PUBLICATION [4])

At a Glance

- **Setting:** Monocular vision-based navigation in cluttered environments, where depth estimates are noisy and no explicit uncertainty sampler is available.
- **Challenges addressed:** C1 (learned non-parametric uncertainty from perception) and C2 (reliable risk estimation from a small set of learned clearance samples).
- **Method:** A learned probabilistic collision model predicts a distribution over worst-case clearance; the model is co-trained with the downstream MMD-based risk metric (task-aware training) and embedded in risk-aware MPC.
- **Key Result:** Up to 7× fewer collisions than ROSNAV/MonoNav and an order-of-magnitude reduction over NoMaD on real hardware.

6.1. Introduction

Chapter 4 estimated collision risk by assuming access to samples drawn from the predictive distribution describing obstacle motion. This chapter takes a more end-to-end approach. We map raw perception and control sequence directly to collision risk. Subsequently, we embed this predicted collision risk directly into the downstream risk-aware trajectory optimization.

Key Idea

We use noisy depth from vision foundational models [60, 61] as a proxy for highly noisy depth perception. So, the key idea is to map this noisy perception to collision risk in a way that is not too conservative, while also correctly identifying unsafe control sequences.

6.2. Problem Formulation

We frame monocular vision-based collision avoidance and goal reaching as the following trajectory optimization.

$$\min_{\mathbf{u}} w_1 c(\mathbf{x}) + w_2 r(d(\mathbf{u},\mathbf{o})) + w_3 \left\|\mathbf{u}\right\|_2^2, \tag{6.1a}$$

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k),\ \mathbf{u}_{min} \leq \mathbf{u}_k \leq \mathbf{u}_{max} \forall k \tag{6.1b}$$

where the state at time-step k is $\mathbf{x}_k$, and $\mathbf{x}$ denotes the concatenated trajectory. The dynamics are given by f, with control inputs $\mathbf{u}_k$ stacked to form $\mathbf{u}$.

The key difference from Chapters 4 and 5 is the scalar $d(\mathbf{u},\mathbf{o})$, a *learned probabilistic collision model* that, given observations $\mathbf{o}$ (from estimated depth) and controls $\mathbf{u}$, outputs a distribution over worst-case obstacle clearance along the induced trajectory. The first term c in (6.1a) minimizes the state cost, typically addressing path-following errors. The second term r captures collision-risk based on the predictions of d. The last term $\|\mathbf{u}\|_2^2$ in (6.1a) penalizes large control inputs, and the weights w_i tune the robot's risk-seeking behavior. Control bounds are enforced through (6.1b). We construct an MPC feedback loop by solving (6.1a)-(6.1b) in a receding horizon manner from the current state.

6.3. Main Algorithmic Results

1. **[C1]** We develop a learned probabilistic collision model that replaces the explicit obstacle-prediction or dynamics sampler used in Chapters 4 and 5 with a distribution over worst-case clearance conditioned on monocular depth observations.
2. **[C1, C2]** We introduce a *task-aware* training pipeline that co-trains the collision model with the downstream MMD-based risk metric, so that the predicted variance (and kernel bandwidth λ_θ) is shaped by safe-vs-unsafe decision utility rather than pure density fit.
3. **[C1, C2]** We integrate the learned model in a receding-horizon MPC loop (Alg. 4) and demonstrate real-time, low-collision navigation on commodity GPUs and the Jetson Orin.

6.3.1. Collision Risk and MMD-Based Surrogate

With robot footprint d_o, the collision constraint is $h(d(\mathbf{u},\mathbf{o})) = -d(\mathbf{u},\mathbf{o}) + d_o \leq 0$ and $r = P(h > 0)$. As in previous chapters, we adopt the MMD-based surrogate r^{emp}_{MMD} from Eq. (4.10), computed from N residual samples ${}^i\overline{h}$ drawn via d. We use the Laplace kernel K_λ with width λ.

Importance of Kernel Parameter. The kernel parameter λ critically controls the sensitivity of r^{emp}_{MMD}. An overly large λ causes the embeddings to collapse, making them insensitive to collision risk; a λ that is too small makes embeddings artificially distinct, leading to overly conservative behavior. Unlike the previous chapters where λ was optimized through the reduced-set bi-level program, in this chapter, the collision model learns to predict an optimal λ directly from the current observations and control sequences (Section 6.3.3).

6.3.2. Learning the Collision Model: Baseline Approach

The collision model predicts obstacle clearance as $\mathcal{N}(\mu_\theta(\mathbf{u},\mathbf{o}), \sigma_\theta(\mathbf{u},\mathbf{o}))$. The architecture (Fig. 18) encodes a depth-derived point cloud via PointNet++ [102], concatenates with the robot state to form $\mathbf{o}$, and passes $(\mathbf{u},\mathbf{o})$ through an MLP.

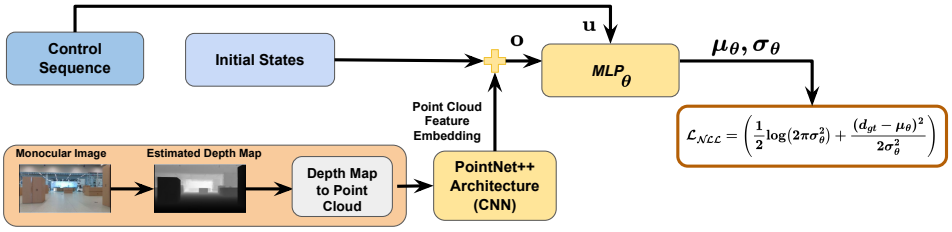


Figure 18: Overview of the baseline learning pipeline for the probabilistic collision model. Given an RGB image and control sequence, geometric features are extracted from the estimated point cloud using a pre-trained depth estimator and PointNet++. Combined with the initial robot state, these form the observation vector, which an MLP uses to predict the mean and variance of obstacle clearance. The learnable components (yellow) are trained end-to-end using Gaussian NLL loss.

Training uses the NLL loss:

$$\mathscr{L}_{\text{NLL}} = \frac{1}{2}\log(2\pi\sigma_\theta^2) + \frac{(d_{gt}-\mu_\theta)^2}{2\sigma_\theta^2}, \tag{6.2}$$

where d_{gt} is the ground-truth worst-case obstacle clearance along the trajectory, obtained from LiDAR measurements during training.

Key Idea

A probabilistic collision model trained with NLL alone captures the data distribution but not the decision it will drive: inflated variance can make safe trajectories look risky, and vice versa. Co-training the model with the downstream r^{emp}_{MMD}-based risk estimate aligns predictive uncertainty (mean μ_θ, variance σ_θ, *and* kernel bandwidth λ_θ) with safe-vs-unsafe classification, not with likelihood fit alone.

6.3.3. Learning the Collision Model: Task-Aware Approach

NLL alone does not supervise how variance should behave in downstream planning: a safe trajectory may be flagged high-risk due to inflated variance, or vice versa. We augment the baseline (Fig. 19) with a risk-estimation head. During training, $\mu_\theta, \sigma_\theta$ generate N clearance samples via the reparameterization trick:

$$^i d = \mu_\theta + \sigma_\theta \cdot {}^i\varepsilon, \quad {}^i\varepsilon \sim \mathcal{N}(0,1), \quad \forall i = 1,\ldots,N \tag{6.3a}$$

$$^i h(^i d) = -{}^i d + d_o, \quad {}^i\overline{h}(^i d) = \max(0, {}^i h(^i d)) \tag{6.3b}$$

The empirical r^{emp}_{MMD} (using predicted kernel width λ_θ) passes through an MLP and softmax to produce $\hat{\mathbf{y}}$. The ground-truth label is

$$\mathbf{y} = \begin{cases} [0,1]^\top, & \text{if } -d_{gt} + d_o \le 0 \\ [1,0]^\top, & \text{if } -d_{gt} + d_o > 0 \end{cases} \tag{6.4}$$

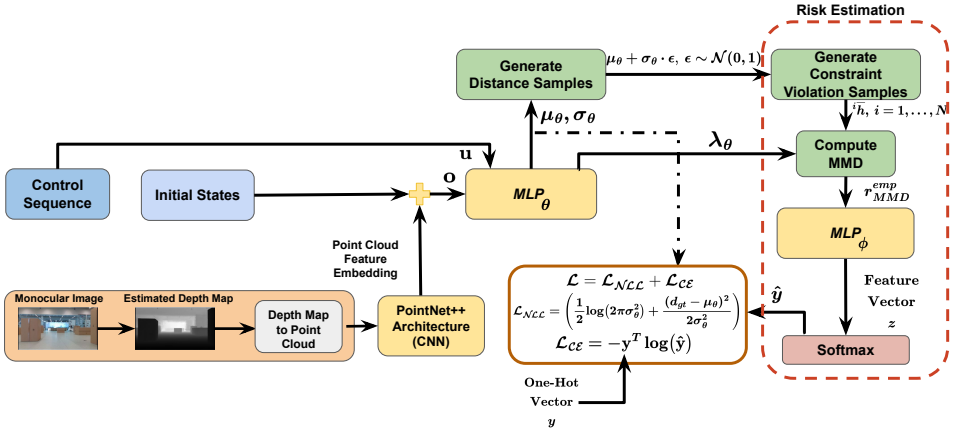


Figure 19: Overview of task-aware learning pipeline. The baseline model (Fig. 18) is augmented with a risk estimation head. The observation vector and control sequence are passed through MLP_θ to predict the mean, variance, and kernel parameter. Using the reparameterization trick, obstacle clearance samples are generated to compute constraint violations, forming an MMD-based risk representation. This is processed by MLP_ϕ followed by a softmax layer. The learnable parts (yellow) are trained end-to-end with Gaussian NLL and cross-entropy loss.

The combined loss is:

$$\mathscr{L} = \mathscr{L}_{\text{NLL}} + \mathscr{L}_{\text{CE}}, \qquad \mathscr{L}_{\text{CE}} = -\mathbf{y}^\top \log(\hat{\mathbf{y}}) \tag{6.5}$$

The cross-entropy term acts as task-aware regularization on σ_θ, shaping variance by downstream safe-versus-unsafe decisions rather than data fit alone.

6.3.4. Trajectory Optimizer

Alg. 4 presents our overall approach, in which we solve (6.1a)-(6.1b) using the trained neural worst-case obstacle clearance model as our collision predictor. We adopt a sampling-based optimizer from [93], in which a batch of control sequences is drawn from a distribution that is gradually refined across iterations.

The algorithm initializes the sampling distribution (Line 2) and samples control inputs (Line 5), which are passed through a trained neural network (Line 6) to predict the mean $\mu_{\theta,q}$ and variance $\sigma_{\theta,q}$ associated with the distribution of obstacle clearances, as well as the kernel hyperparameter $\lambda_{\theta,q}$. These predictions generate N distance samples ${}^i d_q$, constraint violations ${}^i\bar{h}_q$ (Line 7), and subsequently $r^{emp}_{MMD,q}$ (Line 8). State trajectories $\mathbf{x}_q(\mathbf{u}_q)$ are then generated using the robot dynamics (Line 9). The n_c lowest-risk samples form the *ConstraintEliteSet* (Line 10), from which costs are evaluated and stored (Lines 11–13). Finally, the top n_e samples form the *EliteSet* (Line 14) used to update the sampling distribution (Line 15) following [93].

Algorithm 4: Sampling-Based Optimizer to Solve (6.1a)–(6.1b)

M = Maximum number of iterations
Initiate mean ${}^{m}\nu, {}^{m}\Sigma$, at iteration $m = 0$ for sampling control inputs $\mathbf{u}$; Given observation vector $\mathbf{o}$, trained neural network θ (Fig. 19)
for $m = 1, m \leq M, m{+}{+}$ **do**
    Initialize *CostList* = []
    Draw batch of n control sequences $(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_q, \ldots, \mathbf{u}_n)$ from $\mathcal{N}({}^{m}\nu, {}^{m}\Sigma)$
    Query collision model $\forall \mathbf{u}_q$: $(\mu_{\theta,q}, \sigma_{\theta,q}, \lambda_{\theta,q}) = \text{MLP}_\theta(\mathbf{u}_q, \mathbf{o})$, where $\mu_{\theta,q}, \sigma_{\theta,q}, \lambda_{\theta,q}$ are the predicted mean, variance, and kernel parameter for each $\mathbf{u}_q$
    Compute N distance samples ${}^{i}d_q \sim \mathcal{N}(\mu_q, \sigma_q), \forall i = 1, \ldots, N$ and subsequently N constraint violation samples ${}^{i}\bar{h}_q$. Repeat $\forall q = (1, 2, \ldots, n)$
    Compute $\hat{\mu}_q[{}^{i}\bar{h}_q]$ and subsequently $r^{emp}_{MMD,q}$. Repeat $\forall q = (1, 2, \ldots, n)$
    Generate state trajectories $\mathbf{x}_q(\mathbf{u}_q), \forall q = (1, 2, \ldots, n)$
    ConstraintEliteSet $\leftarrow$ Select top n_c batch of $\mathbf{u}_q, \mathbf{x}_q$ with lowest $r^{emp}_{MMD,q}$
    Define $c_q = w_1 c(\mathbf{x}_q) + w_2 r^{emp}_{MMD,q} + w_3 \left\|\mathbf{u}_q\right\|_2^2$
    cost $\leftarrow c_q, \forall q$ in the *ConstraintEliteSet*
    append each computed *cost* to *CostList*
    EliteSet $\leftarrow$ Select top n_e samples of $(\mathbf{u}_q, \mathbf{x}_q)$ with lowest cost from *CostList*.
    $({}^{m+1}\nu, {}^{m+1}\Sigma) \leftarrow$ Update distribution based on *EliteSet*
end
return *Control inputs* $\mathbf{u}_q$ *and* $\mathbf{x}_q$ *corresponding to lowest cost in the EliteSet*

6.4. Main Qualitative and Quantitative Results

Only a representative subset of the results is presented in this chapter; for the complete analysis the reader is referred to Publication [4].

Key Result

On real hardware with a Clearpath Jackal, MonoMPC achieves $\sim 7\times$ fewer collisions than ROSNAV and MonoNav in cluttered environments, and an order-of-magnitude reduction over NoMaD (8.3% vs. 81.6% collisions).

We benchmark against three baseline families spanning the main paradigms for vision-based navigation.

- **ROSNAV** [109] is the classical ROS map-based navigation stack, combining a global planner with a local DWA controller over an occupancy cost-map. Here, the cost-map is built online from monocular-depth point clouds, so ROSNAV treats noisy reconstructed geometry as reliable, isolating the effect of raw perception noise in a classical planner.
- **MonoNav** [73] combines multi-frame depth estimates to build a 3D occupancy map and plan over it. It is a stronger geometric baseline than ROSNAV, but still treats the reconstructed map as fixed geometry without modeling depth uncertainty.
- **NoMaD** [83] is an end-to-end diffusion-based image-to-action policy trained on large-scale teleoperation data. It maps RGB observations directly to action distributions, bypassing the perception–planning split entirely. It represents the learned-policy family of approaches and serves as a reference for how a fully end-to-end alternative behaves on the same hardware.

6.4.1. Implementation Details

Software stack. The trajectory optimizer of Alg. 4 is implemented in JAX [104], exploiting `vmap` and `jit` to batch the n per-iteration control samples, the N worst-case obstacle clearance samples, and the MMD computation onto the GPU. The probabilistic collision model of Section 6.3.3 is built in Equinox [105], and Depth-Anything-V2 [61] is accelerated via TensorRT so that depth estimation runs in parallel with planning.

Hardware and data. Real-hardware experiments use a Clearpath Jackal equipped with a RealSense D435i (used only for its RGB stream; depth is never read off the camera) and a SICK TIM551 2D LiDAR. The LiDAR is used *only* during training to provide ground-truth worst-case clearance d_{gt} for each logged trajectory. At deployment, the robot sees only the RGB image, from which Depth-Anything-V2 produces a noisy depth that conditions the collision model. The training set comprises approximately 6 million observation-control-clearance tuples collected by teleoperating the Jackal through varied cluttered indoor and outdoor scenes, ensuring broad coverage of realistic depth-noise patterns.

6.4.2. Validation of the Learned Collision Model

Premise. Before full navigation results, we evaluate the learned collision model on a held-out validation set. This isolates two questions: (i) whether the predictive distribution is well-calibrated and sharp, and (ii) whether the task-aware cross-entropy term, which shapes predictive variance through the downstream safe-vs-unsafe decision, translates to safer closed-loop behavior.

Metrics. We use four complementary metrics.

- **UR (Unsafe Rate)** = $U/(U+CU)$, the fraction of actually-unsafe configurations that the model classifies as safe. This is the most safety-critical metric, since each counted event is a would-be collision that the planner does not flag.

- **CR (Conservative Rate)** = $C/(C+CS)$, the fraction of actually-safe configurations that the model flags as unsafe. This measures over-conservativeness: a high CR freezes the robot in safe states.
- **ES (Energy Score)** is a strictly proper scoring rule measuring the joint accuracy and sharpness of the predictive distribution; lower is better.
- **ECE over PIT** is the Expected Calibration Error computed over Probability Integral Transform bins; a value near zero indicates that the predictive distribution is perfectly calibrated (quantile coverage matches nominal level).

Results. Table 5 shows that the task-aware augmented model (Fig. 19) surpasses the NLL-only baseline (Fig. 18) on all four axes: UR drops by roughly 50% (2.97% $\rightarrow$ 1.3%), CR drops by roughly 80% (30.88% $\rightarrow$ 5.36%), ES improves by roughly 74% (0.0917 $\rightarrow$ 0.0236), and ECE improves by roughly 20% (0.1142 $\rightarrow$ 0.0893). The CR improvement is especially telling: the baseline model flags almost a third of genuinely safe configurations as unsafe because NLL alone is free to inflate σ_θ in regions of the data where the conditional mean is already accurate, whereas the task-aware loss penalizes exactly this kind of decision-irrelevant variance. The navigation ablation in Table 6 closes the loop: the task-aware model yields a 4$\times$ collision reduction (6.6% vs. 28.3%) over the NLL-only baseline, with essentially the same speed profile, and both learned models outperform the deterministic reference (41.5% collisions) that ignores predictive uncertainty.

Table 5: Validation metrics for the learned collision model.

Method	**UR (%)** (lower is better)	**CR (%)** (lower is better)	**ES** (lower is better)	**ECE over PIT** (lower is better)
Alg. 4 with r^{emp}_{MMD}-based Augmented Model (Fig. 19)	**1.3**	**5.36**	**0.0236**	**0.0893**
Alg. 4 with Baseline Model (Fig. 18)	2.97	30.88	0.0917	0.1142

Table 6: Navigation with and without task-aware training.

Method	**% Collisions**	**% Stuck**	**Avg. Speed (m/s)**	**Max. Speed (m/s)**
Alg. 4 with r^{emp}_{MMD}-based Augmented Model (Fig. 19)	**6.6**	**0**	0.32	**1.02**
Alg. 4 with Baseline Model (Fig. 18)	28.3	**0**	0.35	0.94
Deterministic	41.5	**0**	**0.52**	1.01

6.4.3. Comparison with ROS Navigation Stack

Setup. We deploy ROSNAV [109] at two nominal speeds (0.5 m/s and 1.0 m/s) on the same hardware and in the same cluttered scenes used for our method, so that the only difference is the planner. The DWA local controller consumes a cost-map

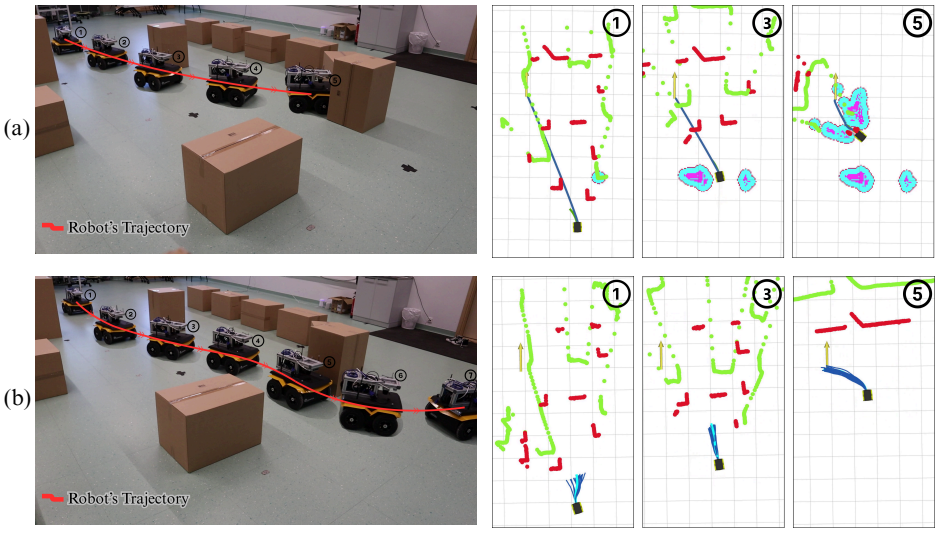


Figure 20: Monocular navigation in cluttered environments using ROSNAV (top) vs. our approach (bottom). ROSNAV constructs cost maps directly from the estimated point cloud (green) generated by DepthAnything[61], which deviates significantly from the ground-truth (red), leading to incorrect free-space detection (e.g., top row, panel 3) and collisions. In contrast, our method treats the estimated point cloud as a conditioning input to a learned probabilistic collision model, integrated with a risk-aware MPC framework. Snapshots across time steps are shown for both methods (corresponding time indices are labeled).

that is built online from the point cloud back-projected out of the Depth-Anything-V2 estimate, making ROSNAV fundamentally dependent on the accuracy of the reconstructed geometry.

Failure modes. ROSNAV fails for two coupled reasons, both visible in Fig. 20 (top row). First, systematic depth offsets produce phantom free space between the ego and real obstacles, so the DWA controller commits to trajectories that the physical robot cannot execute safely. Second, the same noise inflates the cost-map near the goal, which causes the global planner to either oscillate or stall short of the goal. Neither failure mode is fixable by classical cost-map smoothing, because the error is bias in the depth estimate rather than high-frequency noise.

Table 7: Comparison with ROSNAV.

Method	**% Collisions**	**% Stuck**	**Avg. Speed (m/s)**	**Max. Speed (m/s)**
Alg. 4 with r^{emp}_{MMD}-based Augmented Model	**6.6**	**0**	0.32	1.02
ROSNAV (0.5 m/s)	25	51.6	0.38	0.59
ROSNAV (1 m/s)	48.3	48.3	**0.73**	**1.15**

Table 7 quantifies these failure modes. At 0.5 m/s, ROSNAV collides in 25% of runs and stalls in another 51.6%, i.e., only roughly a quarter of the runs end

successfully. At 1.0 m/s, collisions jump to 48.3% while the stuck rate remains near 48.3%, indicating that simply commanding higher speeds converts stalls into collisions without improving progress. Our planner reduces collisions by up to 7× (6.6% vs. 48.3%) and eliminates the stuck behavior entirely (0% vs. 48–52%), while maintaining a higher maximum speed than ROSNAV at 0.5 m/s.

6.4.4. Comparison with MonoNav

Setup. MonoNav [73] plans on 3D reconstructions obtained by fusing per-frame depth estimates over time into an occupancy map. We evaluate on 40 scenes split into two difficulty levels: an *easy* setting with sparse obstacle placement and an *moderately difficult* setting with denser clutter that forces the planner to reason about tight gaps.

Observations. Fig. 21 makes the failure mode visible: depth noise translates into 3D occupancy maps whose reconstructed point clouds do not align with the ground-truth obstacle cuboids (yellow). In easy scenes, the misalignment is small relative to the free space and both methods navigate cleanly. In moderately difficult scenes, the same misalignment collapses the apparent free space between obstacles, so MonoNav either collides or gets stuck. Our method, by contrast, never commits to a single fixed reconstruction: the noisy depth enters only as context to the probabilistic clearance model, whose predictive variance then absorbs the uncertainty into the risk cost of the MPC.

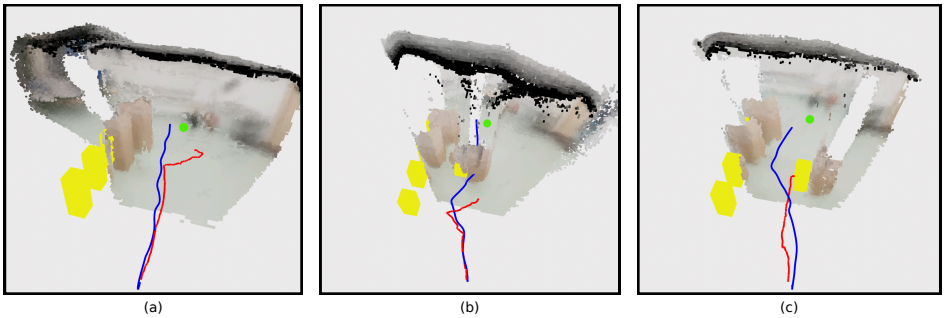


Figure 21: Comparison between our approach (blue) and MonoNav [73] (red); goal in green. Noise in the estimated depth translates to erroneous 3D occupancy maps. The yellow cuboids (ground-truth obstacles) do not align with reconstructed point clouds, causing MonoNav to get stuck or collide (b–c). Our approach avoids obstacles based on noisy estimated depth by reasoning about uncertainty.

Table 8 confirms this qualitative picture quantitatively. In the easy setting both methods achieve 0% collisions and 0% stuck rate. In the moderately difficult setting, MonoNav degrades sharply (20% collisions, 80% stuck), i.e., four out of every five runs fail to complete, while our approach maintains 0% collisions and 0% stuck rate. The gap is explained entirely by how depth noise is handled: MonoNav commits to a single reconstruction that fails in clutter, whereas our probabilistic model converts the same noise into a distributional risk signal the planner can actively reason about.

Table 8: Comparison with MonoNav.

Method	Easy Setting		Moderately Difficult Setting	
	% Collisions	% Stuck	% Collisions	% Stuck
Alg. 4 with r^{emp}_{MMD}-based Augmented Model	**0**	**0**	**0**	**0**
MonoNav [73]	**0**	**0**	20	80

6.4.5. Comparison with NoMaD

Setup. NoMaD [83] is a diffusion-based image-to-action policy. To give the learned baseline its fairest chance, we fine-tune the released checkpoint on approximately 10 k teleoperation tuples collected in a training-similar environment on the same Jackal hardware, then deploy it in the same evaluation scenes used for our method.

Observations. Despite fine-tuning, NoMaD collides in 81.6% of runs (Table 9), far worse than our 8.3%. This aligns with MonoNav [73], which reports more than 50% collision rates for NoMaD even in simpler hallways. We attribute this to a structural limitation: NoMaD lacks an explicit trajectory-level collision objective, relying instead on expert demonstrations, so out-of-distribution failures such as novel clutter or depth noise are not directly penalized. In contrast, our method predicts a *distribution* over clearances, which MPC uses through the risk term to avoid unsafe trajectories.

Table 9: Comparison with NoMaD.

Method	% Collisions	% Stuck	Avg. Speed (m/s)	Max. Speed (m/s)
Alg. 4 with r^{emp}_{MMD}-based Augmented Model	**8.3**	**0**	0.28	**1.05**
NoMaD [83]	81.6	**0**	**0.46**	0.58

6.4.6. Computation Time

On an RTX 3080 laptop, the planning step of Alg. 4 runs in 0.01 s, with an additional 0.02 s for parallel depth estimation using TensorRT-accelerated Depth-Anything-V2. On an NVIDIA Jetson Orin, the planning time increases to 0.065 s, which remains well within the requirements for real-time re-planning on mobile robotic platforms.

Artefacts and code.. Videos, additional hardware results, and accompanying material for MonoMPC are available at the project site `https://sites.google.com/view/monompc`.

6.5. Conclusion

This chapter extended the framework to learned uncertainty from monocular perception. The learned probabilistic collision model is trained jointly with a task-aware MMD-based objective, yielding uncertainty estimates that are better aligned with downstream safe-versus-unsafe decisions. On real hardware, the approach reduces collisions by up to $7\times$ over ROSNAV/MonoNav and by an order of magnitude over NoMaD, while running in real time on a commodity laptop and an NVIDIA Jetson Orin.

Next Chapter

Chapter 7 synthesizes the three main application settings, discusses the empirical evidence for addressing C1 and C2 across them, and outlines directions for future work on distributional risk-aware optimization.

7. CONCLUSION AND FUTURE WORK

7.1. Conclusion

This thesis addressed motion planning under uncertainty through the lens of collision-risk-aware trajectory optimization. The central premise has been that safety in robotic navigation should be represented probabilistically, since real systems must operate under uncertain perception, uncertain obstacle motion, and uncertain dynamics. However, the thesis also showed that this natural probabilistic formulation immediately gives rise to two fundamental difficulties. The first is the tractability challenge identified in Chapter 1: for realistic uncertainty distributions, especially those that are non-Gaussian, multi-modal, or available only through samples, the collision risk $r(\cdot) = P(h(\cdot) > 0)$ does not admit a useful analytical form. The second is the sample-efficiency challenge: even when risk can be approximated from samples, reliable estimation may require a prohibitively large number of collision checks or stochastic rollouts, making downstream trajectory optimization computationally expensive.

The mathematical developments of Chapter 2 provided the foundation for addressing both challenges. Rather than estimating the probability of violation directly, the thesis adopted a distributional view of risk by considering the distribution of the collision-constraint residual $\overline{h}(\cdot) = \max(0, h(\cdot))$. In this view, low-risk behavior corresponds to making the residual distribution resemble a Dirac delta distribution concentrated at zero. This perspective led naturally to a nonparametric surrogate based on RKHS embeddings and MMD. The key advantage of this formulation is that it can operate directly on samples, without imposing restrictive assumptions on the underlying uncertainty model. Chapter 3 then situated this approach relative to deterministic planning, chance-constrained optimization, risk-aware planning, kernel-based stochastic optimization, and monocular vision-based navigation, thereby motivating the thesis contribution as a principled response to the two core challenges C1 and C2.

From this foundation, the thesis developed a unified family of methods across three distinct sources of uncertainty. Although the applications differ substantially, the same conceptual pattern recurs throughout the thesis: represent uncertainty through samples or learned probabilistic models, convert collision risk into a distribution-matching problem in RKHS, and exploit structure in the representation to make risk estimation more sample efficient.

7.1.1. Challenge C1: Tractable Risk Estimation Under Complex Uncertainty

The first major contribution of the thesis is a principled answer to C1, namely the intractability of collision-risk estimation under arbitrary uncertainty. In classical chance-constrained motion planning, exact or approximate probabilistic reformulations typically rely on Gaussian assumptions, linearization, or special geometric

structure. As discussed in Chapter 1, a closed-form expression Φ_G for the collision risk is available only when the uncertain quantity follows a Gaussian distribution. These approaches become inadequate when uncertainty is multi-modal, highly nonlinear, generated by black-box predictors, or induced by learned perception systems. The thesis addressed C1 by replacing direct probability evaluation with an MMD-based surrogate defined on the collision-constraint residual distribution $\overline{h}(\cdot) = \max(0, h(\cdot))$, leveraging the theoretical framework of Chapter 2. Each of the four publications contributed to this response. Publications [1] and [2] (Chapter 4) introduced and refined the MMD-based surrogate for uncertain obstacle motion. Publication [3] (Chapter 5) extended it to stochastic ego dynamics. Publication [4] (Chapter 6) further extended the framework to monocular vision-based navigation through a learned probabilistic collision model.

In Chapter 4, the MMD-based surrogate was first instantiated for planning under uncertain obstacle motion. Publication [1] showed that RKHS embeddings can be used to reason directly over multi-modal trajectory predictions of surrounding agents and to define a tractable surrogate for collision risk from sample-level information, using a Gaussian kernel. Publication [2] refined this idea into the MMD-OPT framework, which replaced analytically intractable probability-of-collision calculations by an MMD-based objective and introduced the Laplace kernel as a more suitable alternative for collision-risk estimation. The shift from Gaussian to Laplace kernel was motivated by the need for sharper sensitivity to constraint residuals near zero, since a heavier-tailed kernel assigns more weight to small but nonzero violations. This kernel choice was retained throughout the subsequent chapters, providing better discrimination between safe and near-collision configurations.

Chapter 5 extended this same logic to a different source of uncertainty: stochastic ego dynamics. Here the uncertainty no longer comes from predicted obstacle trajectories, but from the distribution of state rollouts induced by noisy actuation and model perturbations. Once again, direct computation of collision risk is analytically intractable. The thesis addressed this by applying the same distributional viewpoint to the residuals generated by stochastic rollouts. The resulting MMD-based surrogate preserved the key benefit established earlier: risk can be estimated in a nonparametric manner directly from samples, without requiring closed-form characterization of the rollout distribution. Importantly, this chapter demonstrated that the MMD surrogate remains reliable under both Gaussian and Beta noise distributions, confirming its nonparametric character in a closed-loop MPC setting.

Chapter 6 then showed that the same core idea extends even to monocular vision-based navigation. In this setting, uncertainty arises from noisy depth estimation rather than from explicit stochastic process models or trajectory predictors. Instead of performing brittle collision checking on raw monocular depth, the thesis learned a probabilistic model of worst-case obstacle clearance conditioned on the visual observation and control sequence. This learned clearance distribution was then incorporated into the same MMD-based risk-aware plan-

ning framework. A task-aware training pipeline further ensured that the predicted uncertainty is well-calibrated for the downstream risk surrogate rather than merely statistically plausible. Thus, the thesis showed that the proposed surrogate is not tied to one particular uncertainty source; rather, it provides a common language for risk estimation across prediction uncertainty, dynamics uncertainty, and perception uncertainty.

Taken together, the three application chapters establish that the central response to C1 is the same throughout the thesis: move from direct probability evaluation to nonparametric distribution matching in RKHS. This shift is the main conceptual contribution of the thesis.

7.1.2. Challenge C2: Sample-Efficient Risk Estimation

The second major contribution of the thesis is its response to C2, namely the need to estimate risk reliably from a limited number of samples. This challenge is especially important in robotics, because every additional sample may require a collision check, a trajectory rollout, or an evaluation of a learned model inside an optimization loop. A risk estimator that is theoretically meaningful but sample-hungry is of limited practical value for real-time planning. The central mechanism for addressing C2 is the optimization-based reduced-set framework, which improves both sample efficiency and computational tractability of collision-risk estimation by preserving the RKHS embedding of the full sample distribution using a smaller, optimized subset.

The thesis addressed C2 through the notion of a reduced-set introduced in Chapter 2. The central idea is that not all samples are equally informative for representing the uncertainty relevant to collision risk. If one can preserve the RKHS embedding of the original sample distribution using only a smaller subset, then one can retain the essential structure needed for risk estimation while reducing computation. This idea provided the bridge from elegant theory to practical planning algorithms.

In Chapter 4, reduced-set construction became a concrete algorithmic tool for uncertain obstacle motion. The method selected a compact but informative subset of predicted obstacle trajectories and, in its more mature form (MMD-OPT), optimized not only the subset itself but also the associated sample weights and kernel bandwidth through a bi-level optimization scheme. This reduced the number of collision checks required in planning while preserving the ability of the surrogate to distinguish between low-probability and high-probability obstacle maneuvers. On nuScenes with Trajectron++ predictions, the resulting planner achieved approximately twofold improvement in collision rate over SAA- and CVaR-based baselines in the lane-change setting with $N' = 25$ samples, thereby showing that sample efficiency is not merely a computational convenience but a decisive factor in safety performance.

Chapter 5 extended the reduced-set idea to stochastic dynamics, where the sample-efficiency challenge is arguably even more severe. In that setting, risk estimation depends on a large set of stochastic state rollouts, and each rollout may itself be costly to generate and evaluate. The thesis showed that a carefully selected reduced-set of rollouts can preserve the distributional information needed for MMD-based risk estimation while dramatically reducing the number of constraint evaluations. Most strikingly, the closed-loop MPC experiments in CARLA demonstrated that the MMD-based surrogate with only $N = 2$ reduced-set samples achieves zero collisions in settings where CVaR with the same sample budget incurs collision rates as high as 45%. This result provides the strongest empirical evidence that the reduced-set preserves the distributional structure relevant for safety even under extreme sample compression.

Chapter 6 addressed C2 in a different but closely related way. Here the objective was not to compress explicit external samples from a predictor or simulator, but to learn a compact probabilistic representation of collision-relevant uncertainty directly from monocular observations. The learned collision model amortizes a large portion of the computation that would otherwise be required to reason about noisy visual geometry online. Moreover, the task-aware training strategy ensures that the predicted uncertainty is useful for downstream risk estimation rather than merely statistically plausible. The ablation study in Chapter 6 confirmed the importance of this co-training: replacing the task-aware augmented model with the baseline model that uses only NLL loss increased the collision rate from 6.6% to 28.3%, while removing the probabilistic model entirely (deterministic planning) resulted in a 41.5% collision rate. In this sense, Chapter 6 extends the spirit of C2 from explicit sample selection to learned, decision-oriented uncertainty representations.

Across all three application domains, the thesis therefore developed a coherent answer to C2: risk estimation must be designed not only to be correct in principle, but also to remain informative under restricted sample budgets. The two mechanisms for achieving this, optimization-based reduced-set construction (Chapters 4 and 5) and learned probabilistic models with task-aware training (Chapter 6), are complementary in nature. The former compresses explicit samples into a smaller informative subset, while the latter amortizes the uncertainty estimation into a neural model trained to produce representations that are directly useful for the downstream MMD-based planner. This emphasis on sample efficiency is one of the defining practical contributions of the thesis.

7.1.3. Synthesis Across the Thesis

The main outcome of the thesis is not only a collection of application-specific methods, but a unified viewpoint on uncertainty-aware planning. The common thread is that collision risk should be treated as a distributional object rather than as a scalar probability that must be evaluated exactly. Once this viewpoint is

adopted, RKHS embeddings and MMD provide a flexible mathematical framework for representing uncertainty directly from samples, while reduced-set methods and learned probabilistic models make the resulting computations practical.

The three application chapters show that this viewpoint translates into concrete gains across very different uncertainty sources. Chapter 4 demonstrates that the MMD-based surrogate with optimized reduced-set handles multi-modal obstacle predictions more effectively than SAA and CVaR baselines under tight sample budgets. Chapter 5 shows the same principle extends to stochastic ego dynamics, with r^{emp}_{MMD} remaining reliable where CVaR estimates break down. Chapter 6 shows that the framework further accommodates learned probabilistic uncertainty from monocular perception, enabling safe navigation in cluttered real-world scenes. The framework is therefore not tied to a single robotic modality or a single source of uncertainty.

Chapter Takeaway

Empirical evidence across the three settings.

- **Ch. 4 (uncertain obstacle motion):** Roughly half the collision rate of SAA and CVaR in nuScenes lane-change scenarios.
- **Ch. 5 (stochastic ego dynamics):** Zero collisions for r^{emp}_{MMD} at $N = 2$ on CARLA Towns 5 and 10 under low-noise conditions where r^{emp}_{CVaR} incurs up to 16.67%; the gap widens to 45% under higher noise.
- **Ch. 6 (monocular perception):** Up to 7$\times$ fewer collisions than ROSNAV, zero collisions and zero stuck rate where MonoNav reaches 20%/80%, and an order-of-magnitude improvement over NoMaD (8.3% vs. 81.6%).

A notable technical thread running through the thesis is the progression of the kernel choice. Publication [1] in Chapter 4 initially employed a Gaussian kernel, while the subsequent MMD-OPT framework (Publication [2]) introduced the Laplace kernel, which was then adopted in Chapters 5 and 6. The Laplace kernel's heavier tails provide sharper sensitivity to small but nonzero constraint residuals, which is important for collision-risk estimation where the critical distinction lies near the zero boundary. This choice, though seemingly a technical detail, had consistent empirical impact across all three application domains.

At a broader level, the thesis contributes to the ongoing shift in robotics from deterministic planning toward uncertainty-aware decision making. It argues that practical safety under uncertainty requires more than simply adding a stochastic term to an objective or replacing constraints by loose probabilistic relaxations. Instead, it requires models and optimization procedures that can faithfully represent complex uncertainty while remaining computationally efficient enough for closed-loop planning. The methods developed in this thesis are a step in that direction.

7.2. Future Work

While the thesis addresses both C1 and C2 in several important settings, it also opens multiple directions for future research. The directions below are organized roughly from the most concrete extensions of the current methods to broader open problems.

7.2.1. Joint Treatment of Multiple Sources of Uncertainty

Key Upgrade

Couple perception, prediction, and dynamics uncertainty in a single RKHS/MMD surrogate.

Each of the application chapters isolates one dominant source of uncertainty: obstacle prediction in Chapter 4, stochastic ego dynamics in Chapter 5, and monocular perception in Chapter 6. This decomposition was useful for developing the methods systematically, but real robotic systems often face all of these sources simultaneously. A natural next step is therefore to develop a unified risk-aware planning framework that handles uncertain perception, uncertain prediction, and uncertain dynamics within a single optimization problem. Such a formulation would require coupling multiple learned or sampled uncertainty models and reasoning about their joint effect on collision risk. This is fundamentally a new instance of C1: the joint uncertainty is richer than any individual source, and the compositional structure of the RKHS/MMD framework, where kernel embeddings can be defined over product spaces, provides a natural starting point for extending the surrogate to this fully coupled setting.

7.2.2. Learned Reduced-Set Construction

Key Upgrade

Replace the online bi-level reduced-set optimization with a learned one-shot predictor.

A recurring limitation in Chapters 4 and 5 is that reduced-set construction is itself an optimization problem that must be solved at each planning step. Although this optimization improves sample efficiency (C2), it also introduces additional computational cost that partly offsets the gains. One promising direction is to train a neural network offline that directly predicts the reduced-set, the associated sample weights, or the kernel parameters from the current scene context or rollout statistics. Such a learned predictor would replace the online bi-level optimization with a single forward pass, preserving the reduced-set principle while making it more suitable for real-time deployment. This approach could also provide warm starts for the existing optimization-based formulation, blending learned initialization with online refinement when additional accuracy is needed.

7.2.3. Adaptive Kernels and Stronger Theoretical Guarantees

Key Upgrade

Turn kernel choice from a hyperparameter into a data-driven, theoretically grounded design decision.

The empirical performance of the MMD-based risk surrogate depends on the kernel choice and its bandwidth. The thesis established empirically that the Laplace kernel outperforms the Gaussian kernel for collision-risk estimation, but a rigorous theoretical explanation of this advantage remains open. More broadly, kernel parameters are either optimized or learned in an application-specific manner in this thesis, but a more general theory of adaptive kernel selection for collision-risk estimation is lacking. Future research could study kernel adaptation more systematically, possibly through meta-learning, Bayesian optimization, or data-dependent kernel design. A related theoretical direction is to strengthen finite-sample guarantees for the MMD-based surrogate and to better characterize how reduced-set quality, kernel choice, and risk-estimation error interact in closed-loop planning. Such guarantees would provide formal safety certificates rather than relying solely on empirical validation.

7.2.4. Closed-Loop Interactive Planning in Dense Dynamic Environments

Key Upgrade

Extend the surrogate to settings where surrounding agents react to the ego robot's own plan.

The driving results of Chapter 4 mainly consider the effect of uncertain obstacle motion on a single ego planner. In dense traffic, however, surrounding agents may react to the ego vehicle, leading to interactive multi-agent uncertainty. Extending the proposed framework to interactive prediction-and-planning settings would be especially valuable for urban driving. More generally, future work should study how the proposed nonparametric surrogate behaves in densely coupled environments where the uncertainty distribution evolves as a function of the robot's own actions.

7.2.5. Temporal Models for Vision-Based Navigation

Key Upgrade

Add temporal memory to the learned collision model for stability under cluttered and dynamic scenes.

The main limitation identified in Chapter 6 is the lack of temporal memory in

the current monocular navigation pipeline. Since the learned collision model operates primarily on the current observation, it does not exploit temporal consistency across frames, which can lead to oscillatory behavior in cluttered scenes. A clear future direction is to incorporate temporal information through recurrent architectures, latent-state models, or short-term memory over images and point clouds. Another important extension is to move from largely static environments to dynamic scenes with moving obstacles, where the collision model must reason jointly about perception uncertainty and temporal evolution.

7.2.6. Broader Robotic Domains and Embedded Deployment

Key Upgrade

Transfer the framework to aerial and manipulation platforms and push reduced-set construction to embedded GPUs.

Although the thesis focused on ground-robot navigation, the proposed framework is more general. The same ideas are relevant to aerial navigation, mobile manipulation, and planning in cluttered workspaces where collision checking under uncertainty is a dominant computational bottleneck. Extending the methods to quadrotors, manipulators, and other nonlinear robotic systems is therefore a natural next step. Chapter 6 already demonstrated that the MonoMPC pipeline can run on embedded hardware (Jetson Orin on a Clearpath Jackal robot) using GPU-accelerated inference. Extending this level of embedded deployment to the reduced-set-based methods of Chapters 4 and 5, which currently rely on more expensive online optimization, remains an open challenge that would benefit from the learned reduced-set predictors discussed above.

7.2.7. Learning and Planning Integration

Key Upgrade

Train predictors and reduced-set selectors end-to-end for downstream risk-aware control, not just predictive accuracy.

A final broader direction is to more tightly integrate learning and planning. In Chapter 6, this thesis already moved toward task-aware uncertainty learning by coupling the learned collision model to the downstream planner. Similar ideas could be applied in the other chapters as well: predictors of obstacle motion, stochastic rollout generators, or reduced-set selectors could all be trained not only for predictive accuracy but also for their effect on downstream risk-aware control. Such integration could yield systems whose uncertainty representations are optimized directly for safe decision making rather than for standalone prediction quality.

7.3. Closing Remarks

The thesis began with the observation that uncertainty-aware planning becomes most difficult precisely in the scenarios where safety matters the most: when uncertainty is complex, sample-based, and computationally expensive to reason about. The central contribution of this work has been to show that these difficulties can be addressed in a unified manner through nonparametric risk surrogates in RKHS and through principled methods for improving sample efficiency. Across uncertain obstacle motion, stochastic dynamics, and monocular vision-based navigation, the thesis demonstrated that risk-aware planning can be made both mathematically principled and practically effective. It is hoped that the ideas developed here contribute toward a broader class of robotic systems that can reason about uncertainty explicitly and act safely in complex real-world environments.

BIBLIOGRAPHY

[1] **Basant Sharma**, Aditya Sharma, K Madhava Krishna, and Arun Kumar Singh. "Hilbert space embedding-based trajectory optimization for multi-modal uncertain obstacle trajectory prediction". In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2023, pp. 7448–7455.

[2] **Basant Sharma** and Arun Kumar Singh. "MMD-OPT: Maximum Mean Discrepancy Based Sample Efficient Collision Risk Minimization for Autonomous Driving". In: *IEEE Transactions on Automation Science and Engineering* 22 (2025), pp. 19051–19068.

[3] **Basant Sharma** and Arun Kumar Singh. "Trajectory Optimization Under Stochastic Dynamics Leveraging Maximum Mean Discrepancy". In: *IEEE Robotics and Automation Letters* 10.6 (June 2025), pp. 6079–6086.

[4] **Basant Sharma**, Prajyot Jadhav, Pranjal Paul, K.Madhava Krishna, and Arun Kumar Singh. "MonoMPC: Monocular Vision Based Navigation with Learned Collision Model and Risk-Aware Model Predictive Control". In: *IEEE Robotics and Automation Letters* (2025), pp. 1–8.

[5] Vikrant Dewangan, **Basant Sharma**, Tushar Choudhary, Sarthak Sharma, Aakash Aanegola, Arun K Singh, and K Madhava Krishna. "UAP-BEV: Uncertainty Aware Planning Using Bird's Eye View Generated From Surround Monocular Images". In: *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*. IEEE. 2023, pp. 1–8.

[6] Jatan Shrestha, Simon Idoko, **Basant Sharma**, and Arun Kumar Singh. "End-to-end learning of behavioural inputs for autonomous driving in dense traffic". In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2023, pp. 10020–10027.

[7] Fatemeh Rastgar, Houman Masnavi, **Basant Sharma**, Alvo Aabloo, Jan Swevers, and Arun Kumar Singh. "PRIEST: Projection guided sampling-based optimization for autonomous navigation". In: *IEEE Robotics and Automation Letters* 9.3 (2024), pp. 2630–2637.

[8] Simon Idoko, **Basant Sharma**, and Arun Kumar Singh. "Learning Sampling Distribution and Safety Filter for Autonomous Driving with VQ-VAE and Differentiable Optimization". In: *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2024.

[9] Yadollah Rasekhipour, Amir Khajepour, Shih-Ken Chen, and Bakhtiar Litkouhi. "A potential field-based model predictive path-planning controller for autonomous road vehicles". In: *IEEE Transactions on Intelligent Transportation Systems* 18.5 (2016), pp. 1255–1267.

[10] Wenchao Ding, Lu Zhang, Jing Chen, and Shaojie Shen. "Safe trajectory generation for complex urban environments using spatio-temporal semantic corridor". In: *IEEE Robotics and Automation Letters* 4.3 (2019), pp. 2997–3004.

[11] Vivek K Adajania, Aditya Sharma, Anish Gupta, Houman Masnavi, K Madhava Krishna, and Arun K Singh. "Multi-modal model predictive control through batch non-holonomic trajectory optimization: Application to highway driving". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 4220–4227.

[12] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. "Trajectron++: Dynamically-Feasible Trajectory Forecasting With Heterogeneous

Data". In: *European Conference on Computer Vision (ECCV)*. 2020. URL: https://arxiv.org/abs/2001.03093.

[13] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. "A Formal Basis for the Heuristic Determination of Minimum Cost Paths". In: *IEEE Transactions on Systems Science and Cybernetics* 4.2 (1968), pp. 100–107. DOI: 10.1109/TSSC.1968.300136.

[14] Zhanying Zhang and Ziping Zhao. "A Multiple Mobile Robots Path planning Algorithm Based on A-star and Dijkstra Algorithm". In: *International Journal of Smart Home* 8 (May 2014), pp. 75–86. DOI: 10.14257/ijsh.2014.8.3.07.

[15] Shang Erke, Dai Bin, Nie Yiming, Zhu Qi, Xiao Liang, and Zhao Dawei. "An improved A-Star based path planning algorithm for autonomous land vehicles". In: *International Journal of Advanced Robotic Systems* 17.5 (2020), p. 1729881420962263. DOI: 10.1177/1729881420962263. eprint: https://doi.org/10.1177/1729881420962263. URL: https://doi.org/10.1177/1729881420962263.

[16] E. W. Dijkstra. "A Note on Two Problems in Connexion with Graphs". In: *Edsger Wybe Dijkstra: His Life,Work, and Legacy*. 1st ed. New York, NY, USA: Association for Computing Machinery, 2022, pp. 287–290. ISBN: 9781450397735. URL: https://doi.org/10.1145/3544585.3544600.

[17] James J. Kuffner and Steven M. LaValle. "RRT-Connect: An Efficient Approach to Single-Query Path Planning". In: *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA)*. Vol. 2. IEEE, 2000, pp. 995–1001. DOI: 10.1109/ROBOT.2000.844730.

[18] Samuel Rodriguez, David Hsu, Jean-Claude Latombe, and Kevin Manson. "An Obstacle-Based Rapidly-Exploring Random Tree". In: *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2006, pp. 895–900. DOI: 10.1109/ROBOT.2006.1641791.

[19] Nan Wang and Ricardo G. Sanfelice. "A Rapidly-Exploring Random Trees Motion Planning Algorithm for Hybrid Dynamical Systems". In: *Proceedings of the 2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 2626–2631. DOI: 10.1109/CDC51059.2022.9992779.

[20] Lydia E. Kavraki, Petr Svestka, Jean-Claude Latombe, and Mark H. Overmars. "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces". In: *IEEE Transactions on Robotics and Automation* 12.4 (1996), pp. 566–580. DOI: 10.1109/70.508439.

[21] Roland Geraerts and Mark H. Overmars. "A Comparative Study of Probabilistic Roadmap Planners". In: *Algorithmic Foundations of Robotics V*. Ed. by Dan Halperin and Jacob E. Goodman. Springer, 2004, pp. 43–57. DOI: 10.1007/978-3-540-45058-0_4.

[22] Ashwin Kannan, Pavlos Elinas, and Andres Castano. "Robot Motion Planning Using Adaptive Hybrid Sampling in Probabilistic Roadmaps". In: *Electronics* 5.2 (2016), p. 16. DOI: 10.3390/electronics5020016.

[23] Bharath Gopalakrishnan, Arun Kumar Singh, and K. Madhava Krishna. "Time Scaled Collision Cone Based Trajectory Optimization Approach for Reactive Planning in Dynamic Environments". In: *Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2014, pp. 4169–4176. DOI: 10.1109/IROS.2014.6943222.

[24] Yu Zhao, Hsien-Chung Lin, and Masayoshi Tomizuka. "Efficient Trajectory Optimization for Robot Motion Planning". In: *Proceedings of the 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE, 2018, pp. 260–265. DOI: 10.1109/ICARCV.2018.8581085.

[25] Taylor A. Howell et al. "Trajectory Optimization with Optimization-Based Dynamics". In: *IEEE Robotics and Automation Letters* 7.3 (2022), pp. 6750–6757. DOI: 10.1109/LRA.2022.3186824.

[26] Matt Zucker, Nathan Ratliff, Anca D. Dragan, Mihail Pivtoraiko, Matthew Klingensmith, Christopher M. Dellin, J. Andrew Bagnell, and Siddhartha S. Srinivasa. "CHOMP: Covariant Hamiltonian Optimization for Motion Planning". In: *The International Journal of Robotics Research* 32.9–10 (2013), pp. 1164–1193. DOI: 10.1177/0278364913488805.

[27] Hai Zhu and Javier Alonso-Mora. "Chance-Constrained Collision Avoidance for MAVs in Dynamic Environments". In: *IEEE Robotics and Automation Letters* 4.2 (2019), pp. 776–783. DOI: 10.1109/LRA.2019.2893494.

[28] Lars Blackmore, Masahiro Ono, and Brian C. Williams. "Chance-Constrained Optimal Path Planning With Obstacles". In: *IEEE Transactions on Robotics* 27.6 (2011), pp. 1080–1094. DOI: 10.1109/TRO.2011.2161160.

[29] Lars Blackmore, Masahiro Ono, Askar Bektassov, and Brian C. Williams. "A Probabilistic Particle-Control Approximation of Chance-Constrained Stochastic Predictive Control". In: *IEEE Transactions on Robotics* 26.3 (2010), pp. 502–517. DOI: 10.1109/TRO.2010.2044948.

[30] Lucas Janson, Edward Schmerling, and Marco Pavone. "Monte Carlo Motion Planning for Robot Trajectory Optimization Under Uncertainty". In: *Robotics Research: Volume 2*. Ed. by Antonio Bicchi and Wolfram Burgard. Cham: Springer International Publishing, 2018, pp. 343–361.

[31] Bernardo K Pagnoncelli, Shabbir Ahmed, and Alexander Shapiro. "Sample average approximation method for chance constrained programming: theory and applications". In: *Journal of optimization theory and applications* 142.2 (2009), pp. 399–416.

[32] Thomas Lew, Riccardo Bonalli, and Marco Pavone. "Risk-Averse Trajectory Optimization via Sample Average Approximation". In: *IEEE Robotics Autom. Lett.* 9.2 (2024), pp. 1500–1507. DOI: 10.1109/LRA.2023.3331889. URL: https://doi.org/10.1109/LRA.2023.3331889.

[33] Giuseppe Carlo Calafiore and Marco C Campi. "The scenario approach to robust control design". In: *IEEE Transactions on automatic control* 51.5 (2006), pp. 742–753.

[34] Giuseppe C Calafiore and Lorenzo Fagiano. "Robust model predictive control via scenario optimization". In: *IEEE Transactions on Automatic Control* 58.1 (2012), pp. 219–224.

[35] Heejin Ahn, Colin Chen, Ian M. Mitchell, and Maryam Kamgarpour. "Safe Motion Planning Against Multimodal Distributions Based on a Scenario Approach". In: *IEEE Control Systems Letters* 6 (2022), pp. 1142–1147. DOI: 10.1109/LCSYS.2021.3089641.

[36] Daniele Bernardini and Alberto Bemporad. "Scenario-based model predictive control of stochastic constrained linear systems". In: *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th*

Chinese Control Conference. 2009, pp. 6333–6338. DOI: 10.1109/CDC.2009.5399917.

[37] Georg Schildbach, Lorenzo Fagiano, Christoph Frei, and Manfred Morari. "The scenario approach for Stochastic Model Predictive Control with bounds on closed-loop constraint violations". In: *Automatica* 50.12 (2014), pp. 3009–3018. ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2014.10.035. URL: https://www.sciencedirect.com/science/article/pii/S0005109814004166.

[38] Giuseppe C. Calafiore and Lorenzo Fagiano. "Stochastic model predictive control of LPV systems via scenario optimization". In: *Automatica* 49.6 (2013), pp. 1861–1866. ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2013.02.060. URL: https://www.sciencedirect.com/science/article/pii/S0005109813001507.

[39] Jia-Jie Zhu, Bernhard Schoelkopf, and Moritz Diehl. "A kernel mean embedding approach to reducing conservativeness in stochastic programming and control". In: *Learning for Dynamics and Control*. PMLR. 2020, pp. 915–923.

[40] Ji Yin, Zhiyuan Zhang, and Panagiotis Tsiotras. "Risk-aware model predictive path integral control using conditional value-at-risk". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 7937–7943.

[41] Khaled A Mustafa, Daniel Jarne Ornia, Jens Kober, and Javier Alonso-Mora. "RACP: Risk-Aware Contingency Planning with Multi-Modal Predictions". In: *IEEE Transactions on Intelligent Vehicles* (2024).

[42] Hung Duy Nguyen, Mooryong Choi, and Kyoungseok Han. "Risk-informed decision-making and control strategies for autonomous vehicles in emergency situations". In: *Accident Analysis & Prevention* 193 (2023), p. 107305.

[43] Bharath Gopalakrishnan, Arun Kumar Singh, Meha Kaushik, K. Madhava Krishna, and Dinesh Manocha. "PRVO: Probabilistic Reciprocal Velocity Obstacle for multi robot navigation under uncertainty". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017, pp. 1089–1096. DOI: 10.1109/IROS.2017.8202279.

[44] T. Mitchell Roddenberry, Nicholas Glaze, and Santiago Segarra. "Principled Simplicial Neural Networks for Trajectory Prediction". In: *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 9020–9029. URL: https://proceedings.mlr.press/v139/roddenberry21a.html.

[45] Simone Zamboni, Zekarias Tilahun Kefato, Sarunas Girdzijauskas, Christoffer Norén, and Laura Dal Col. "Pedestrian trajectory prediction with convolutional neural networks". In: *Pattern Recognition* 121 (2022), p. 108252. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2021.108252. URL: https://www.sciencedirect.com/science/article/pii/S0031320321004325.

[46] Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. "GroupNet: Multiscale Hypergraph Neural Networks for Trajectory Prediction With Relational Reasoning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 6498–6507.

[47] Hai Zhu and Javier Alonso-Mora. “Chance-constrained collision avoidance for mavs in dynamic environments”. In: *IEEE Robotics and Automation Letters* 4.2 (2019), pp. 776–783.

[48] Charles Dawson, Ashkan Jasour, Andreas Hofmann, and Brian Williams. “Provably Safe Trajectory Optimization in the Presence of Uncertain Convex Obstacles”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2020, pp. 6237–6244. DOI: 10.1109/IROS45743.2020.9341193.

[49] Manuel Castillo-Lopez, Philippe Ludivig, Seyed Amin Sajadi-Alamdari, Jose Luis Sanchez-Lopez, Miguel A. Olivares-Mendez, and Holger Voos. “A Real-Time Approach for Chance-Constrained Motion Planning With Dynamic Obstacles”. In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 3620–3625. DOI: 10.1109/LRA.2020.2975759.

[50] Carl-Johann Simon-Gabriel, Adam Scibior, Ilya O Tolstikhin, and Bernhard Schölkopf. “Consistent kernel mean estimation for functions of random variables”. In: *Advances in Neural Information Processing Systems* 29 (2016).

[51] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. “A kernel two-sample test”. In: *J. Mach. Learn. Res.* 13.null (2012), pp. 723–773. ISSN: 1532-4435.

[52] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. “Kernel Mean Embedding of Distributions: A Review and Beyond”. In: *Foundations and Trends® in Machine Learning* 10.1–2 (2017), pp. 1–141. ISSN: 1935-8245. DOI: 10.1561/2200000060. URL: http://dx.doi.org/10.1561/2200000060.

[53] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. “A Kernel Method for the Two-Sample-Problem”. In: *Advances in Neural Information Processing Systems*. Ed. by B. Schölkopf, J. Platt, and T. Hoffman. Vol. 19. MIT Press, 2006. URL: https://proceedings.neurips.cc/paper_files/paper/2006/file/e9fb2eda3d9c55a0d89c98d6c54b5b3e-Paper.pdf.

[54] Sudarshan S Harithas, Rishabh Dev Yadav, Deepak Singh, Arun Kumar Singh, and K Madhava Krishna. “Cco-voxel: Chance constrained optimization over uncertain voxel-grid representation for safe trajectory planning”. In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE. 2022, pp. 11087–11093.

[55] Bernhard Schölkopf, Krikamol Muandet, Kenji Fukumizu, Stefan Harmeling, and Jonas Peters. “Computing functions of random variables via reproducing kernel Hilbert space representations”. In: *Statistics and Computing* 25 (2015), pp. 755–766.

[56] Marcello Farina, Luca Giulioni, and Riccardo Scattolini. “Stochastic linear model predictive control with chance constraints–a review”. In: *Journal of Process Control* 44 (2016), pp. 53–67.

[57] Ziyi Wang, Oswin So, Keuntaek Lee, and Evangelos A. Theodorou. “Adaptive Risk Sensitive Model Predictive Control with Stochastic Search”. In: *Proceedings of the 3rd Conference on Learning for Dynamics and Control*. Vol. 144. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 510–522. URL: https://proceedings.mlr.press/v144/wang21b.html.

[58] Kaveh Nazem Tahmasebi, Parthib Khound, and DeJiu Chen. “A Condition-Aware Stochastic Dynamic Control Strategy for Safe Automated Driving”. In: *IEEE Transactions on Intelligent Vehicles* (2024).

[59] Gang Chen, Peng Peng, Peihan Zhang, and Wei Dong. “Risk-aware trajectory sampling for quadrotor obstacle avoidance in dynamic environments”. In: *IEEE Transactions on Industrial Electronics* 70.12 (2023), pp. 12606–12615.

[60] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. “Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data”. In: *CVPR*. 2024.

[61] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. “Depth Anything V2”. In: *Advances in Neural Information Processing Systems*. Ed. by A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang. Vol. 37. Curran Associates, Inc., 2024, pp. 21875–21911. URL: https://proceedings.neurips.cc/paper_files/paper/2024/file/26cfdcd8fe6fd75cc53e92963a656c58-Paper-Conference.pdf.

[62] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. *ZoeDepth: Zero-shot Transfer by Combining Relative and Metric Depth*. 2023. DOI: 10.48550/ARXIV.2302.12288. URL: https://arxiv.org/abs/2302.12288.

[63] Thomas Lew, Riccardo Bonalli, and Marco Pavone. “Risk-averse trajectory optimization via sample average approximation”. In: *IEEE Robotics and Automation Letters* (2023).

[64] Dohyeong Kim and Songhwai Oh. “Efficient Off-Policy Safe Reinforcement Learning Using Trust Region Conditional Value At Risk”. In: *IEEE Robotics and Automation Letters* 7.3 (2022), pp. 7644–7651. DOI: 10.1109/LRA.2022.3184793.

[65] Mohamadreza Ahmadi, Xiaobin Xiong, and Aaron D. Ames. “Risk-Averse Control via CVaR Barrier Functions: Application to Bipedal Robot Locomotion”. In: *IEEE Control Systems Letters* 6 (2022), pp. 878–883. DOI: 10.1109/LCSYS.2021.3086854.

[66] Fernando S. Barbosa, Bruno Lacerda, Paul Duckworth, Jana Tumova, and Nick Hawes. “Risk-Aware Motion Planning in Partially Known Environments”. In: *2021 60th IEEE Conference on Decision and Control (CDC)*. 2021, pp. 5220–5226. DOI: 10.1109/CDC45484.2021.9683744.

[67] Yassine Nemmour, Heiner Kremer, Bernhard Schölkopf, and Jia-Jie Zhu. “Maximum mean discrepancy distributionally robust nonlinear chance-constrained optimization with finite-sample guarantee”. In: *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE. 2022, pp. 5660–5667.

[68] Bharath Gopalakrishnan, Arun Kumar Singh, K Madhava Krishna, and Dinesh Manocha. “Solving chance-constrained optimization under nonparametric uncertainty through hilbert space embedding”. In: *IEEE Transactions on Control Systems Technology* 30.3 (2021), pp. 901–916.

[69] Yuki Shirai, Devesh K. Jha, and Arvind U. Raghunathan. “Covariance Steering for Uncertain Contact-rich Systems”. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2023. DOI: 10.1109/icra48891.2023.10160249. URL: http://dx.doi.org/10.1109/ICRA48891.2023.10160249.

[70] A. Shapiro, D. Dentcheva, and A.P. Ruszczynski. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition*. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2014. ISBN: 9781611973426. URL: https://books.google.ee/books?id=ob4ABAAAQBAJ.

[71] Manuel Castillo-Lopez, Philippe Ludivig, Seyed Amin Sajadi-Alamdari, Jose Luis Sanchez-Lopez, Miguel A Olivares-Mendez, and Holger Voos. "A real-time approach for chance-constrained motion planning with dynamic obstacles". In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 3620–3625.

[72] Jiankun Wang, Max Q-H Meng, and Oussama Khatib. "EB-RRT: Optimal motion planning for mobile robots". In: *IEEE Transactions on Automation Science and Engineering* 17.4 (2020), pp. 2063–2073.

[73] Nathaniel Simon and Anirudha Majumdar. "MonoNav: MAV Navigation via Monocular Depth Estimation and Reconstruction". In: *Symposium on Experimental Robotics (ISER)*. 2023. URL: https://arxiv.org/abs/2311.14100.

[74] Felipe Codevilla, Matthias Müller, Antonio López, Vladlen Koltun, and Alexey Dosovitskiy. "End-to-End Driving Via Conditional Imitation Learning". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 4693–4700. DOI: 10.1109/ICRA.2018.8460487.

[75] Antonio Loquercio, Ana I. Maqueda, Carlos R. del-Blanco, and Davide Scaramuzza. "DroNet: Learning to Fly by Driving". In: *IEEE Robotics and Automation Letters* 3.2 (2018), pp. 1088–1095. DOI: 10.1109/LRA.2018.2795643.

[76] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. "Learning by cheating". In: *Conference on Robot Learning*. PMLR. 2020, pp. 66–75.

[77] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. "Learning To Explore Using Active Neural SLAM". In: *International Conference on Learning Representations (ICLR)*. 2020.

[78] Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. "Neural Topological SLAM for Visual Navigation". In: *CVPR*. 2020.

[79] Fereshteh Sadeghi and Sergey Levine. "CAD2RL: Real Single-Image Flight Without a Single Real Image". In: *Proceedings of Robotics: Science and Systems*. Cambridge, Massachusetts, July 2017. DOI: 10.15607/RSS.2017.XIII.034.

[80] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. "Target-driven visual navigation in indoor scenes using deep reinforcement learning". In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017, pp. 3357–3364. DOI: 10.1109/ICRA.2017.7989381.

[81] Shiwei Lian and Feitian Zhang. "TDANet: Target-Directed Attention Network for Object-Goal Visual Navigation With Zero-Shot Ability". In: *IEEE Robotics and Automation Letters* 9.9 (2024), pp. 8075–8082. DOI: 10.1109/LRA.2024.3440100.

[82] Gregory Kahn, Adam Villaflor, Bosen Ding, Pieter Abbeel, and Sergey Levine. "Self-Supervised Deep Reinforcement Learning with Generalized Computation Graphs for Robot Navigation". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 5129–5136. DOI: 10.1109/ICRA.2018.8460655.

[83] Ajay Sridhar, Dhruv Shah, Catherine Glossop, and Sergey Levine. "NoMaD: Goal Masked Diffusion Policies for Navigation and Exploration". In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. 2024, pp. 63–70. DOI: 10.1109/ICRA57147.2024.10610665.

[84] Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. "Navigation World Models". In: *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*. June 2025, pp. 15791–15801.

[85] Nicklas Hansen, Hao Su, and Xiaolong Wang. "TD-MPC2: Scalable, Robust World Models for Continuous Control". In: *International Conference on Learning Representations (ICLR)*. 2024.

[86] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Ken Goldberg, and Pieter Abbeel. "DayDreamer: World Models for Physical Robot Learning". In: *Conference on Robot Learning* (2022).

[87] Martin Jacquet and Kostas Alexis. "N-mpc for deep neural network-based collision avoidance exploiting depth images". In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2024, pp. 13536–13542.

[88] Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, and Sergey Levine. "ViNT: A Foundation Model for Visual Navigation". In: *7th Annual Conference on Robot Learning*. 2023. URL: https://arxiv.org/abs/2306.14846.

[89] Noriaki Hirose, Fei Xia, Roberto Martín-Martín, Amir Sadeghian, and Silvio Savarese. "Deep Visual MPC-Policy Learning for Navigation". In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 3184–3191. DOI: 10.1109/LRA.2019.2925731.

[90] Alexander Liniger and Luc Van Gool. "Safe Motion Planning for Autonomous Driving using an Adversarial Road Model". In: *Proceedings of Robotics: Science and Systems*. Corvalis, Oregon, USA, July 2020.

[91] Zdravko I Botev, Dirk P Kroese, Reuven Y Rubinstein, and Pierre L'Ecuyer. "The cross-entropy method for optimization". In: *Handbook of statistics*. Vol. 31. Elsevier, 2013, pp. 35–59.

[92] Grady Williams, Paul Drews, Brian Goldfain, James M Rehg, and Evangelos A Theodorou. "Aggressive driving with model predictive path integral control". In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2016, pp. 1433–1440.

[93] Mohak Bhardwaj, Balakumar Sundaralingam, Arsalan Mousavian, Nathan D Ratliff, Dieter Fox, Fabio Ramos, and Byron Boots. "Storm: An integrated framework for fast joint-space model-predictive control for reactive manipulation". In: *Conference on Robot Learning*. PMLR. 2022, pp. 750–759.

[94] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. "nuscenes: A multimodal dataset for autonomous driving". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11621–11631.

[95] Peter Karkus, Boris Ivanovic, Shie Mannor, and Marco Pavone. "Diffstack: A differentiable and modular control stack for autonomous vehicles". In: *Conference on robot learning*. PMLR. 2023, pp. 2170–2180.

[96] Kai Ren, Heejin Ahn, and Maryam Kamgarpour. “Chance-constrained trajectory planning with multimodal environmental uncertainty”. In: *IEEE Control Systems Letters* 7 (2022), pp. 13–18.

[97] Boris Ivanovic and Marco Pavone. “The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 2375–2384.

[98] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. “CARLA: An Open Urban Driving Simulator”. In: *Proceedings of the 1st Annual Conference on Robot Learning*. 2017, pp. 1–16.

[99] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. “Vision Transformers for Dense Prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 12179–12188.

[100] Dhruv Shah, Ajay Sridhar, Arjun Bhorkar, Noriaki Hirose, and Sergey Levine. “GNM: A General Navigation Model to Drive Any Robot”. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 7226–7233.

[101] Antonio Loquercio, Elia Kaufmann, René Ranftl, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. “Learning High-Speed Flight in the Wild”. In: *Science Robotics* 6.59 (2021), eabg5810.

[102] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. “Pointnet++: Deep hierarchical feature learning on point sets in a metric space”. In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017.

[103] Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. “Deep reinforcement learning in a handful of trials using probabilistic dynamics models”. In: *Advances in Neural Information Processing Systems* 31 (2018).

[104] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. *JAX: composable transformations of Python+NumPy programs*. 2018.

[105] Patrick Kidger and Cristian Garcia. “Equinox: neural networks in JAX via callable PyTrees and filtered transformations”. In: *Differentiable Programming workshop at Neural Information Processing Systems* (2021).

[106] Thomas Lew, Riccardo Bonalli, and Marco Pavone. “Chance-Constrained Sequential Convex Programming for Robust Trajectory Optimization”. In: *2020 European Control Conference (ECC)*. 2020, pp. 1871–1878. DOI: 10.23919/ECC51009.2020.9143595.

[107] Junqing Wei, Jarrod M Snider, Tianyu Gu, John M Dolan, and Bakhtiar Litkouhi. “A behavioral planning framework for autonomous driving”. In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE. 2014, pp. 458–464.

[108] Houman Masnavi, Jatan Shrestha, Manav Mishra, PB Sujit, Karl Kruusamäe, and Arun Kumar Singh. “Visibility-Aware Navigation With Batch Projection Augmented Cross-Entropy Method Over a Learned Occlusion Cost”. In: *IEEE Robotics and Automation Letters* 7.4 (2022), pp. 9366–9373.

[109] Steve Macenski, Francisco Martín, Ruffin White, and Jonatan Ginés Clavero. “The Marathon 2: A Navigation System”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 2718–2725.

ACKNOWLEDGEMENTS

A doctoral thesis is never the work of a single individual. Reflecting upon the journey that culminated in this dissertation, I am deeply aware of the many individuals whose support, guidance, and encouragement made this work possible. It is a privilege to acknowledge them here.

First and foremost, I express my sincere and profound gratitude to my supervisor, Prof. Arun Kumar Singh. At a particularly challenging phase in my career, he placed his trust in me, offered me an opportunity, and extended unwavering support throughout my doctoral journey. His guidance, patience, and confidence in my abilities have been instrumental to this work. His keen technical insight, willingness to engage with evolving ideas, and emphasis on intellectual rigor have significantly shaped both this thesis and my development as a researcher. I am especially grateful for the academic freedom he provided, along with the clarity and direction he offered whenever it was most needed.

I am deeply indebted to my family, whose constant encouragement and belief have been the foundation upon which this work rests.

I wish to express my heartfelt gratitude to my parents, Dr. Satish Sharma and Mrs. Shikha Sharma, for their unwavering faith in me and for always supporting my aspirations without hesitation. They instilled in me, from an early age, a deep respect for education and the discipline required to pursue it. The values they imparted-perseverance, integrity, and a commitment to meaningful inquiry-continue to guide my academic and personal life.

I am profoundly grateful to my brother, Dr. Guru Mehar Sharma, who laid the foundation for this journey. His guidance and example played a decisive role in shaping my academic path, including my decision to pursue my Master's at IIT Madras, which ultimately set the stage for this doctoral work. His constant support and candid advice have been invaluable throughout.

I extend my deepest appreciation to my wife, Dr. Tanvi Bhardwaj. Her encouragement was pivotal in my decision to pursue a PhD, and her unwavering support sustained me through its many challenges. She has been a constant source of strength and motivation, sharing in the demands and sacrifices that accompany this journey. This thesis is as much a reflection of her belief in me as it is of my own efforts.

To my son, Tejas, who entered our lives during this journey, I am profoundly grateful for the joy and perspective he has brought into my life. He has been a source of inspiration and a reminder of what truly matters.

Basant Sharma
Tartu, 2026

SISUKOKKUVÕTE

Valimiefektiivne riskiteadlik trajektoorioptimeerimine maksimaalse keskmise lahknevuse abil

Doktoritöö käsitleb riskiteadlikku liikumise planeerimist autonoomsetele robotsüsteemidele ebakindlates keskkondades. Peamised väljakutsed on kokkupõrkeriski arvutuslik hindamine (C1) ja valimiefektiivne riskihindamine (C2), eriti olukordades, kus ebakindlus on mitte-Gausslik, multimodaalne või esitatud üksnes valimitena. Töö keskne idee on käsitleda kokkupõrkeriski jaotuste võrdlemise probleemina. Selleks defineeritakse kokkupõrkepiirangu jääk ning mõõdetakse selle kaugust ohutut olukorda kirjeldavast Diraci delta-jaotusest kasutades RKHS-esitusi ja Maximum Mean Discrepancy (MMD) mõõdikut. Lähenemine võimaldab hinnata riski otse valimitest ilma piiravate jaotuseeldusteta.

Töö esimene rakendus käsitleb autonoomset sõitu ebakindlate takistustrajektooride korral. Välja töötatud MMD-OPT raamistik kasutab trajektoorivalimeid ja reduced-set mehhanismi, et saavutada väikese valimihulga korral madalam kokkupõrkesagedus kui SAA- ja CVaR-põhised meetodid. Teine rakendus laiendab meetodit stohhastilise dünaamikaga trajektoorioptimeerimisele ja mudelipõhisele ennustavale juhtimisele, kus näidatakse paremat ohutust ja väiksemaid piirangurikkumisi. Kolmas rakendus käsitleb monokulaarset nägemispõhist navigatsiooni, kus õpitakse tõenäosuslik kliirensimudel ning ühendatakse see riskiteadliku juhtimisega.

Peamised tulemused on: (1) üldine MMD-põhine riskisurrogaat mitteparameetriliseks riskihindamiseks, (2) reduced-set meetod valimiefektiivsuse parandamiseks, (3) valideerimine kolme erineva ebakindlusallika korral ning (4) järjepidevalt parem ohutus väikese valimihulga režiimis. Töö näitab, et RKHS-il, MMD-l ja reduced-set põhimõttel põhinev raamistik võimaldab ühendada matemaatilise põhjendatuse, arvutusliku efektiivsuse ja praktilise rakendatavuse reaalsetes robotsüsteemides.

PUBLICATIONS

CURRICULUM VITAE

Personal data

Full Name:	Basant Sharma
Date of birth:	11.12.1991
Nationality:	Indian
Email:	basantsharma1990@gmail.com,basant.sharma@ut.ee

Education

2022–Present	PhD in Engineering and Technology, University of Tartu
2015–2017	MSc Mathematics, Indian Institute of Technology Madras, India
2009–2013	B.E. (Hons) Mechanical Engineering, BITS Pilani, K. K. Birla Goa Campus

Employment

2022–2026	Junior Research Fellow in Computer Engineering, University of Tartu
2021–2022	Research Engineer in Robotics, University of Tartu
2019–2020	Mathematics Educator (Online Teaching)
2018	Software Engineer, Bungeni
2017–2018	Software Engineer, AlexaApps

Supervision

2025, MSc Thesis	Kwasi Akuamoah Boateng, Master's Degree, 2025, Arun Kumar Singh; Basant Sharma, *Risk-Aware Planning on Point Clouds*, University of Tartu, Faculty of Science and Technology, Institute of Technology

Teaching

Fall 2023, 2024, 2025	Perception and Control of Quadrotor Drones (Teaching Assistant)
Spring 2023, 2024	Optimization for Robot Motion Planning and Control (Teaching Assistant)

ELULOOKIRJELDUS

Isikuandmed

Täisnimi:	Basant Sharma
Sünniaeg:	11.12.1991
Kodakondsus:	Indian
E-post:	basantsharma1990@gmail.com,basant.sharma@ut.ee

Haridus

2022–praegu	Doktoriõpe inseneeria ja tehnoloogia erialal, Tartu ülikool
2015–2017	Matemaatika magistriõpe, Indian Institute of Technology Madras, India
2009–2013	Masinaehituse bakalaureuseõpe (kiitusega), BITS Pilani, K. K. Birla Goa Campus

Teenistuskäik

2022–2026	Arvutitehnika nooremteadur, Tartu ülikool
2021–2022	Robotitehnika insener-teadur, Tartu ülikool
2019–2020	Matemaatikaõpetaja (veebiõpe)
2018	Tarkvarainsener, Bungeni
2017–2018	Tarkvarainsener, AlexaApps

Juhendamine

2025, magistritöö	Kwasi Akuamoah Boateng, magistrikraad, 2025, Arun Kumar Singh; Basant Sharma, *Risk-Aware Planning on Point Clouds*, Tartu ülikool, loodus- ja täppisteaduste valdkond, tehnoloogiainstituut

Õppetöö

Sügis 2023, 2024, 2025	Quadrotor-droonide taju ja juhtimine (õppeassistent)
Kevad 2023, 2024	Roboti liikumise planeerimise ja juhtimise optimeerimine (õppeassistent)

DISSERTATIONES TECHNOLOGIAE UNIVERSITATIS TARTUENSIS

1. **Imre Mäger**. Characterization of cell-penetrating peptides: Assessment of cellular internalization kinetics, mechanisms and bioactivity. Tartu 2011, 132 p.
2. **Taavi Lehto**. Delivery of nucleic acids by cell-penetrating peptides: application in modulation of gene expression. Tartu 2011, 155 p.
3. **Hannes Luidalepp**. Studies on the antibiotic susceptibility of *Escherichia coli*. Tartu 2012, 111 p.
4. **Vahur Zadin**. Modelling the 3D-microbattery. Tartu 2012, 149 p.
5. **Janno Torop**. Carbide-derived carbon-based electromechanical actuators. Tartu 2012, 113 p.
6. **Julia Suhorutšenko**. Cell-penetrating peptides: cytotoxicity, immunogenicity and application for tumor targeting. Tartu 2012, 139 p.
7. **Viktoryia Shyp**. G nucleotide regulation of translational GTPases and the stringent response factor RelA. Tartu 2012, 105 p.
8. **Mardo Kõivomägi**. Studies on the substrate specificity and multisite phosphorylation mechanisms of cyclin-dependent kinase Cdk1 in *Saccharomyces cerevisiae*. Tartu, 2013, 157 p.
9. **Liis Karo-Astover**. Studies on the Semliki Forest virus replicase protein nsP1. Tartu, 2013, 113 p.
10. **Piret Arukuusk**. NickFects–novel cell-penetrating peptides. Design and uptake mechanism. Tartu, 2013, 124 p.
11. **Piret Villo**. Synthesis of acetogenin analogues. Asymmetric transfer hydrogenation coupled with dynamic kinetic resolution of α-amido-β-keto esters. Tartu, 2013, 151 p.
12. **Villu Kasari**. Bacterial toxin-antitoxin systems: transcriptional cross-activation and characterization of a novel *mqsRA* system. Tartu, 2013, 108 p.
13. **Margus Varjak**. Functional analysis of viral and host components of alphavirus replicase complexes. Tartu, 2013, 151 p.
14. **Liane Viru**. Development and analysis of novel alphavirus-based multifunctional gene therapy and expression systems. Tartu, 2013, 113 p.
15. **Kent Langel**. Cell-penetrating peptide mechanism studies: from peptides to cargo delivery. Tartu, 2014, 115 p.
16. **Rauno Temmer**. Electrochemistry and novel applications of chemically synthesized conductive polymer electrodes. Tartu, 2014, 206 p.
17. **Indrek Must**. Ionic and capacitive electroactive laminates with carbonaceous electrodes as sensors and energy harvesters. Tartu, 2014, 133 p.
18. **Veiko Voolaid**. Aquatic environment: primary reservoir, link, or sink of antibiotic resistance? Tartu, 2014, 79 p.
19. **Kristiina Laanemets**. The role of SLAC1 anion channel and its upstream regulators in stomatal opening and closure of *Arabidopsis thaliana.* Tartu, 2015, 115 p.

20. **Kalle Pärn**. Studies on inducible alphavirus-based antitumour strategy mediated by site-specific delivery with activatable cell-penetrating peptides. Tartu, 2015, 139 p.
21. **Anastasia Selyutina**. When biologist meets chemist: a search for HIV-1 inhibitors. Tartu, 2015, 172 p.
22. **Sirle Saul**. Towards understanding the neurovirulence of Semliki Forest virus. Tartu, 2015, 136 p.
23. **Marit Orav**. Study of the initial amplification of the human papillomavirus genome. Tartu, 2015, 132 p.
24. **Tormi Reinson**. Studies on the Genome Replication of Human Papillomaviruses. Tartu, 2016, 110 p.
25. **Mart Ustav Jr**. Molecular Studies of HPV-18 Genome Segregation and Stable Replication. Tartu, 2016, 152 p.
26. **Margit Mutso**. Different Approaches to Counteracting Hepatitis C Virus and Chikungunya Virus Infections. Tartu, 2016, 184 p.
27. **Jelizaveta Geimanen**. Study of the Papillomavirus Genome Replication and Segregation. Tartu, 2016, 168 p.
28. **Mart Toots**. Novel Means to Target Human Papillomavirus Infection. Tartu, 2016, 173 p.
29. **Kadi-Liis Veiman**. Development of cell-penetrating peptides for gene delivery: from transfection in cell cultures to induction of gene expression *in vivo*. Tartu, 2016, 136 p.
30. **Ly Pärnaste**. How, why, what and where: Mechanisms behind CPP/cargo nanocomplexes. Tartu, 2016, 147 p.
31. **Age Utt**. Role of alphavirus replicase in viral RNA synthesis, virus-induced cytotoxicity and recognition of viral infections in host cells. Tartu, 2016, 183 p.
32. **Veiko Vunder**. Modeling and characterization of back-relaxation of ionic electroactive polymer actuators. Tartu, 2016, 154 p.
33. **Piia Kivipõld**. Studies on the Role of Papillomavirus E2 Proteins in Virus DNA Replication. Tartu, 2016, 118 p.
34. **Liina Jakobson**. The roles of abscisic acid, CO_2, and the cuticle in the regulation of plant transpiration. Tartu, 2017, 162 p.
35. **Helen Isok-Paas**. Viral-host interactions in the life cycle of human papillomaviruses. Tartu, 2017, 158 p.
36. **Hanna Hõrak**. Identification of key regulators of stomatal CO_2 signalling via O_3-sensitivity. Tartu, 2017, 260 p.
37. **Jekaterina Jevtuševskaja**. Application of isothermal amplification methods for detection of *Chlamydia trachomatis* directly from biological samples. Tartu, 2017, 96 p.
38. **Ülar Allas**. Ribosome-targeting antibiotics and mechanisms of antibiotic resistance. Tartu, 2017, 152 p.
39. **Anton Paier**. Ribosome Degradation in Living Bacteria. Tartu, 2017, 108 p.
40. **Vallo Varik**. Stringent Response in Bacterial Growth and Survival. Tartu, 2017, 101 p.

41. **Pavel Kudrin**. In search for the inhibitors of *Escherichia coli* stringent response factor RelA. Tartu, 2017, 138 p.
42. **Liisi Henno**. Study of the human papillomavirus genome replication and oligomer generation. Tartu, 2017, 144 p.
43. **Katrin Krõlov**. Nucleic acid amplification from crude clinical samples exemplified by *Chlamydia trachomatis* detection in urine. Tartu, 2018, 118 p.
44. **Eve Sankovski**. Studies on papillomavirus transcription and regulatory protein E2. Tartu, 2018, 113 p.
45. **Morteza Daneshmand**. Realistic 3D Virtual Fitting Room. Tartu, 2018, 233 p.
46. **Fatemeh Noroozi**. Multimodal Emotion Recognition Based Human-Robot Interaction Enhancement. Tartu, 2018, 113 p.
47. **Krista Freimann**. Design of peptide-based vector for nucleic acid delivery in vivo. Tartu, 2018, 103 p.
48. **Rainis Venta**. Studies on signal processing by multisite phosphorylation pathways of the *S. cerevisiae* cyclin-dependent kinase inhibitor Sic1. Tartu, 2018, 155 p.
49. **Inga Põldsalu**. Soft actuators with ink-jet printed electrodes. Tartu, 2018, 85 p.
50. **Kadri Künnapuu**. Modification of the cell-penetrating peptide PepFect14 for targeted tumor gene delivery and reduced toxicity. Tartu, 2018, 114 p.
51. **Toomas Mets**. RNA fragmentation by MazF and MqsR toxins of *Escherichia coli.* Tartu, 2019, 119 p.
52. **Kadri Tõldsepp**. The role of mitogen-activated protein kinases MPK4 and MPK12 in CO_2-induced stomatal movements. Tartu, 2019, 259 p.
53. **Pirko Jalakas**. Unravelling signalling pathways contributing to stomatal conductance and responsiveness. Tartu, 2019, 120 p.
54. **S. Sunjai Nakshatharan**. Electromechanical modelling and control of ionic electroactive polymer actuators. Tartu, 2019, 165 p.
55. **Eva-Maria Tombak**. Molecular studies of the initial amplification of the oncogenic human papillomavirus and closely related nonhuman primate papillomavirus genomes. Tartu, 2019, 150 p.
56. **Meeri Visnapuu**. Design and physico-chemical characterization of metal-containing nanoparticles for antimicrobial coatings. Tartu, 2019, 138 p.
57. **Jelena Beljantseva**. Small fine-tuners of the bacterial stringent response – a glimpse into the working principles of Small Alarmone Synthetases. Tartu, 2020, 104 p.
58. **Egon Urgard**. Potential therapeutic approaches for modulation of inflammatory response pathways. Tartu, 2020, 120 p.
59. **Sofia Raquel Alves Oliveira**. HPLC analysis of bacterial alarmone nucleotide (p)ppGpp and its toxic analogue ppApp. Tartu, 2020, 122 p.
60. **Mihkel Örd**. Ordering the phosphorylation of cyclin-dependent kinase Cdk1 substrates in the cell cycle. Tartu, 2021, 228 p.
61. **Fred Elhi**. Biocompatible ionic electromechanically active polymer actuator based on biopolymers and non-toxic ionic liquids. Tartu, 2021, 140 p.

62. **Liisi Talas**. Reconstructing paleo-diversity, dynamics and response of eukaryotes to environmental change over the Late-Glacial and Holocene period in lake Lielais Svētiņu using sedaDNA. Tartu, 2021, 118 p.
63. **Livia Matt**. Novel isosorbide-based polymers. Tartu, 2021, 118 p.
64. **Koit Aasumets**. The dynamics of human mitochondrial nucleoids within the mitochondrial network. Tartu, 2021, 104 p.
65. **Faiza Summer**. Development and optimization of flow electrode capacitor technology. Tartu, 2022, 109 p.
66. **Olavi Reinsalu**. Cancer-testis antigen MAGE-A4 is incorporated into extracellular vesiclesand is exposed to the surface. Tartu, 2022, 130 p.
67. **Tetiana Brodiazhenko**. RelA-SpoT Homolog enzymes as effectors of Toxin-Antitoxin systems. Tartu, 2022, 132 p.
68. **Georg-Marten Lanno**. Development of novel antibacterial drug delivery systems as wound scaffolds using electrospinning technology. Tartu, 2022, 175 p.
69. **Liubov Cherkashchenko**. New insights into alphaviral nsP2 functions. Tartu, 2023, 171 p.
70. **Kristina Kiisholts**. Peptide-based drug carriers and preclinical nanomedicine applications for endometriosis treatment. Tartu, 2023, 138 p.
71. **Kai Rausalu**. Alphaviral nsP2 protease: From requirements for functionality to inhibition. Tartu, 2023, 175 p.
72. **Laura Sandra Lello**. Unraveling the intricate nature of the alphavirus RNA replicase. Tartu, 2023, 219 p.
73. **Houman Masnavi**. Visibility Aware Navigation. Tartu, 2023, 180 p.
74. **Kadir Aktas**. Cosmic Ray Tomography based Object Reconstruction and Recognition. Tartu, 2023, 104 p.
75. **Egils Avots**. Brain abnormality detection using statistical analysis of individual structural connectivity networks and EEG signals. Tartu, 2023, 223 p.
76. **Sainan Wang**. Structure-guided insights into the functions of CHIKV nsP2. Tartu, 2024, 154 p.
77. **Anneli Samel**. Unveiling the characteristics of cancer-testis antigen MAGEA10. Tartu, 2024, 136 p.
78. **Ikechukwu Ofodile**. Fault tolerant attitude control for nanosatellites: ESTCube-2 case. Tartu, 2024, 130 p.
79. **Olena Zamora**. Impacts of plant hormones on controlling stomatal conductance. Tartu, 2024, 166 p.
80. **Mariliis Hinnu**. *In vitro* methods for studying the mechanisms of ribosome-targeting antibiotics. Tartu, 2024, 143 p.
81. **Chung-Yueh Yeh**. Characterization of MPK and HT1 kinases in CO_2-induced stomatal movements. Tartu, 2024, 118 p.
82. **Iman Dadras**. Low power neural network-based control and actuation solutions for insect-scale robots. Tartu, 2024, 149 p.
83. **Fatemeh Rastgar**. Towards reliable real-time trajectory optimization. Tartu, 2024, 158 p.

84. **Maria Maloverjan**. Optimizing cell-penetrating peptide-based nanoparticles for delivery of nucleic acid therapeutics. Tartu, 2024, 172 p.
85. **Joonas Merisalu**. Resistive switching in memristor structures with multilayer dielectrics. Tartu, 2024, 149 p.
86. **Siim Laanesoo**. Novel high-performance biomass-based polymers. Tartu, 2024, 117 p.
87. **Henri Ingelman**. Systems-level characterisation and improvement of *Clostridium autoethanogenum* metabolism. Tartu, 2024, 164 p.
88. **Mailis Laht**. Using the One Health approach for mapping the spread of antibiotic resistant bacteria in Estonia. Tartu, 2024, 188 p.
89. **Ingrid Rebane**. Structure-property relationships of moldable silicone foams. Tartu, 2024, 164 p.
90. **Robert Valner**. Design of TeMoto, a software framework for dependable, adaptive, and collaborative autonomous robots. Tartu, 2024, 182 p.
91. **Kristiina Kurg**. Exploring the potential of a liquid biopsy approach for melanoma diagnostics and the role of extracellular vesicles in atherosclerosis development. Tartu, 2025, 201 p.
92. **Rauno Sedrik**. Synthesis and investigation of polymers from different cyclic bio-based monomers. Tartu, 2025, 155 p.
93. **Alina Ismagilova**. Safety assessment of novel bio-based polymers and compounds used in low carbon technologies. Tartu, 2025, 156 p.
94. **Baiba Brūmele**. Uncovering the TRMT112 methyltransferase network and characterising the cellular functions of TRMT112-network member N6AMT1. Tartu, 2025, 135 p.
95. **Ingmar Tulva**. Causes and consequences of stomatal density in relation to atmospheric humidity. Tartu, 2025, 142 p.
96. **Yauheni Sarokin**. Passive and active liquid mediation in natural and synthetic morphing systems. Tartu, 2025, 173 p.
97. **Hans Priks**. Life within 3D printed engineered living materials based on micellar hydrogels. Tartu, 2026, 126 p.
98. **Kaspar Koolmeister**. Stomatal CO_2 regulation pathway and its application for modulating tomato plants. Tartu, 2026, 118 p.
99. **Mart Ernits**. Microfluidic production and characterization of liposomes towards localized payload release. Tartu, 2026, 127 p.