Distributionally Robust Risk-Bounded Path Planning Through Exact Spatio-temporal Risk Allocation

Kajsa Ekenberg

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
TFRT-6174
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Tryckeriet i E-huset
Lund 2022

Abstract

Planning safe paths in the presence of uncertainty is considered a central challenge in enabling robots to successfully navigate in real-world environments. Assumptions about Gaussian uncertainty are rarely justifiable based on real data and can lead to serious miscalculations of risk. Lately, it has become increasingly common to consider distributionally robust uncertainty, where the exact distribution of the uncertainty is unknown. Existing motion planning algorithms that consider distributionally robust uncertainty generates more conservative paths then their Gaussian counterparts. The aim of this thesis is to mitigate this conservatism by incorporating non-uniform spatio-temporal risk allocation into existing frameworks for distributionally robust motion planning, specifically the DR-RRT algorithm. To this end, a novel motion planning algorithm called DR-RRT-ERA (DR-RRT with Exact Risk Allocation) is proposed. This is a sampling based motion planning algorithm that builds trees of state distributions while enforcing distributionally robust chance constraints. Instead of allocating the risk uniformly over time and space, the DR-RRT-ERA uses a novel concept called exact risk allocation (ERA). The principle of ERA is to allocate exactly as much risk needed to enforce the distributionally robust risk constraints. Numerical simulations illustrate that this approach leads to less conservative paths compared to when uniform risk allocation is used.

Acknowledgements

First and foremost, I would like to thank my academic supervisor Venkatraman Renganathan. Without his assistance and dedicated involvement in every step of the process, this thesis would not have been possible. I am also thankful to the professors and students within the Department of Automatic Control that gave me valuable feedback on my work and Cem Alptürk who helped me with some technical aspects of the project. I am also grateful to Anders Robertsson and Björn Olofsson for serving as examiners and to Mattias Lundström and Jakob Pettersson for reviewing the thesis and providing valuable feedback.

In addition, I would like to thank my family, partner and friends for their encouragement and support during my years of study and through the process of writing this thesis.

Contents

1

Introduction

1.1 Background and Motivation

The topic of this thesis is spatio-temporal risk allocation in path planning. Consider a robot operating in an uncertain environment cluttered with obstacles. The objective is to steer the robot to a specified goal location while ensuring that the risk for collision with the obstacles does not exceed the user-specified risk bound. The position of the robot and the obstacles are subject to uncertainty. Many motion planning algorithms are based on assumptions of Gaussian uncertainty, but these assumptions are rarely justifiable based on real data and can lead to serious miscalculations of risk [1]. Lately, more and more approaches that utilize a *Distributionally Robust* quantification of the uncertainty have emerged [1–12]. In distributionally robust approaches, the uncertainty is not assumed to belong to a specific distribution, but rather an ambiguity set of distributions. Here we will consider moment-based ambiguity sets that include all distributions with a specified mean and covariance.

Since distributionally robust approaches make weaker assumptions about the uncertainty distribution, they are inherently more conservative than their Gaussian counterparts. In the context of this thesis, conservatism is defined according to how close to the obstacles the robot can venture. When a high degree of conservatism is present, the robot is forced to take large detours around the obstacles. Since we want the robot to take a more direct path to the goal whenever possible, it is beneficial to invent ways to reduce the conservatism of distributionally robust motion planning algorithms. One way to achieve this is through the use of non-uniform spatio-temporal risk allocation. The aim of this thesis is to develop an algorithm for this that can be effectively incorporated into existing path planning algorithms, such as distributionally robust rapidly exploring random tree (DR-RRT) [1].

For risk allocation to be possible, the user-specified risk constraint for the entire path to goal must be decomposed into *individual risk constraints* for all obstacles and timesteps. This can be done with Boole's inequality, as proposed by [8] and [9]. Each obstacle at each timestep is then assigned an individual risk bound. And, as long as the sum of all the individual risk-bounds does not exceed the total allowed risk for the entire path, the risk constraint for the entire path will hold. The individual risk-bounds are referred to as risk allocations and can be fixed in a number of ways. A common approach is to allocate the risk uniformly over all obstacles and timesteps, so that all individual risk-bounds are assigned the same value. However, this approach often results in overly-conservative paths, which is why a non-uniform risk allocation can be preferable. The main objective of this thesis is to find a way to fix the risk allocations in a non-uniform way that reduces the conservatism of the generated paths while still enforcing the distributionally robust risk constraint for the entire path.

1.2 Related Work

Path planning approaches that utilize a non-uniform risk allocation in the presence of *Gaussian* uncertainty have been considered in [13–17]. The majority of these papers have used a two-stage optimization that alternates between optimizing the risk allocation and the feedback control. While this approach reduces the conservatism, it also requires additional computation since the risk allocations need to be updated iteratively through successive optimizations. Luders [17] proposed an alternative approach in which the risk bounds were computed online, effectively removing the need for multiple iterations. In online risk allocation, the exact risk of violating each individual constraint is calculated and used to check the probabilistic feasibility of a path. The Exact Risk Allocation (ERA) method proposed in this thesis uses a similar principle by allocating the exact risk for individual constraint violations. However, there are two major distinctions: ERA allocates risk based on distributionally robust uncertainty whereas online risk allocation was proposed for Gaussian constraints. The other difference lies in how the risk allocations are incorporated into the motion planning algorithm to determine which paths are probabilistically feasible.

1.3 Contribution

The main contribution of this thesis is a novel motion planning algorithm called DR-RRT-ERA (Distributionally Robust Rapidly Exploring Random Tree with Exact Risk Allocation), which is an extension of the DR-RRT algorithm proposed in [1]. We show that DR-RRT-ERA generates less conservative paths while still enforcing the same distributionally robust risk constraints as DR-RRT with uniform risk allocation.

1.4 Thesis Structure

The rest of the thesis is divided into the following chapters: Background, Problem Formulation, DR-RRT with Exact Risk Allocation, Simulation results and Conclusions & Future Outlook.

- 2. Background: Notations, probability theory, DR-RRT and chance constrained path planning with Gaussian and distributionally robust constraints are presented.
- 3. Problem Formulation: Specifies the model dynamics and risk constraints for the path planning problem considered in this thesis and provides a formal problem statement.
- 4. DR-RRT with Exact Risk Allocation: Describes the tree expansion, ERA-procedure and feasibility check for the proposed DR-RRT-ERA algorithm.
- 5. Simulation Results: Numerical simulations of DR-RRT with ERA and uniform risk allocations are shown and the results are discussed.
- 6. Conclusion & Future Outlook: The thesis and main conclusions are summarized, followed by a discussion about limitations and future research topics.

2

Background

In this chapter, notations and probability theory that are used throughout the thesis are presented. Then, chance constrained path planing with Gaussian and distributionally robust constrains are introduced and compared to each other. This is followed by a brief description of the sampling based path planning algorithm DR-RRT. The probability theory covers Gaussian distributions and Cantelli's inequality, which provides an upper bound on the probability that a random variable exceeds a specified threshold for a general probability distribution with known mean and covariance, as well as Fréchet and Boole's inequalities which upper-bound the probability of a conjunction and a disjunction, respectively. The presented theory is then used to convert Gaussian and distributionally robust probabilistic chance constraints into deterministic linear constraints that can be effectively used in path planning.

2.1 Notations

The set of real and natural numbers are denoted by $\mathbb{R}$ and $\mathbb{N}$, respectively. The subset of natural numbers between and including a and b with $a < b$ is denoted by $[a : b]$. The operator $|\cdot|$ denote set cardinality. The operators $\oplus$ and $\setminus$ denote set translation and set subtraction respectively. An identity matrix of dimension n is denoted by I_n. For a vector $x \in \mathbb{R}^n$ and a matrix $P \in \mathbb{S}^n_{++}$ (set of positive definite matrices), let $\|x\|_P = \sqrt{x^\top P x}$. The Euclidean norm of a vector x is denoted as $\|x\|_2$ or simply $\|x\|$. A binary condition being true or false is denoted by $\top$ and $\perp$, respectively. The Borel $\sigma-$algebra on $\mathbb{R}^d$ is denoted by $\mathcal{B}(\mathbb{R}^d)$ and the space of probability measures on $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$ is denoted by $\mathcal{P}(\mathbb{R}^d)$. A probability distribution with mean μ and covariance Σ is denoted by $\mathbb{P}(\mu, \Sigma)$, and specifically $\mathcal{N}_d(\mu, \Sigma)$ if the distribution is normal (Gaussian) in $\mathbb{R}^d$. Given a constant $q \in \mathbb{N}_{\geq 1}$, the set of probability measures in $\mathcal{P}(\mathbb{R}^d)$ with finite $q-$th moment is denoted by $\mathcal{P}_q(\mathbb{R}^d) := \left\{\mu \in \mathcal{P}(\mathbb{R}^d) \mid \int_{\mathbb{R}^d} \|x\|^q \, d\mu < \infty\right\}$.

2.2 Probability Theory

Formally, a probability space is defined using the triple $(\Omega, \mathcal{F}, \mathbb{P})$ consisting of the following elements:

1. A *sample space* Ω which is the set of all possible outcomes.
2. An *event space* $\mathcal{F}$ which is a set of events where an event is a set of outcomes in the sample space.
3. A *probability function* $\mathbb{P}$ that assigns a probability $[0, 1]$ to each event in the event space.

A random variable X is a measurable function $X : \Omega \to E$ that maps a set of possible outcomes Ω to a measurable space E. The probability that X takes on a value in a measurable set $S \subseteq E$ is written as

$$\mathbb{P}(X \in S) = \mathbb{P}\left(\{\omega \in \Omega \mid X(\omega) \in S\}\right). \tag{2.2.1}$$

Any random variable can be described by its cumulative distribution function, which describes the probability that the random variable will be less than or equal to a certain value. The probability distribution of a random variable can be characterized by a small number of parameters with a practical interpretation instead of a known probability distribution. In this thesis, we characterize the distribution of a random variable using its moments, specifically the first and the second central moments (mean and covariance), which can be defined for real-valued functions of random variables. Interested readers are referred to [18] for more details.

Gaussian distributions

Gaussian distributions are abbreviated by $\mathcal{N}(\mu, \Sigma)$, where μ is the mean and Σ is the covariance. The probability density function (pdf) ϕ, the cumulative distribution function (cdf) Φ and the error function (erf) of the standard Gaussian distribution, with $\mu = 0$ and $\Sigma = I$, are given by

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \tag{2.2.2}$$

$$\Phi(b) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{b} e^{-\frac{x^2}{2}}\, dx = \mathbb{P}(x \leq b), \tag{2.2.3}$$

$$\operatorname{erf}(b) = \frac{2}{\sqrt{\pi}} \int_{0}^{b} e^{-\frac{x^2}{2}}\, dx = \mathbb{P}(-b \leq x \leq b). \tag{2.2.4}$$

The cdf Φ and erf are related to each other according to

$$\Phi(b) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{b}{\sqrt{2}}\right)\right]. \tag{2.2.5}$$

Any Gaussian distributed random variable can be expressed in terms of the standard Gaussian distributed random variable (with zero mean and a variance of 1) as follows

$$x = \hat{x} + \sqrt{\Sigma_x} z, \tag{2.2.6}$$

where $x \sim \mathcal{N}(\hat{x}, \Sigma_x)$ and $z \sim \mathcal{N}(0, 1)$.

Cantelli's inequality

Cantelli's inequality is an improved version of Chebyshev's inequality that provides a bound for the one-sided tail of general probability distributions with known mean and covariance. For a random variable x with zero-mean and variance Σ, it holds that

$$\sup \mathbb{P}(x \geq b) \leq \frac{\Sigma}{\Sigma + b^2}. \tag{2.2.7}$$

Fréchet inequalities

Fréchet inequalities are rules that can be used to bound probabilities of logical conjunctions or disjunctions without any assumptions about dependence. For logical propositions or events, A_i, the Fréchet inequalities for logical conjunctions are as follows,

$$\mathbb{P}\left[\bigwedge_{i=1}^{n} A_i\right] \leq \mathbb{P}\left[A_i\right], \ i = 1 : n. \tag{2.2.8}$$

The proof for this is elementary. Since $\mathbb{P}(A \wedge B) = \mathbb{P}(A \mid B)\mathbb{P}(B) = \mathbb{P}(B \mid A)\mathbb{P}(A)$ and $\mathbb{P}(A \mid B)$, $\mathbb{P}(B \mid A) \leq 1$, it follows that $\mathbb{P}(A \wedge B) \leq P(A)$ and $\mathbb{P}(A \wedge B) \leq \mathbb{P}(B)$.

Boole's inequality

For a finite set of events A_i, Boole's inequality states that the probability of at least one of the events occurring is upper bounded by the sum of the individual probabilities of the events according to

$$\mathbb{P}\left[\bigvee_{i=1}^{n} A_i\right] \leq \sum_{i=1}^{n} \mathbb{P}[A_i]. \tag{2.2.9}$$

Boole's inequality can be proven with induction. When $n = 1$ it follows that $\mathbb{P}[A_1] \leq \mathbb{P}[A_1]$. Since $\mathbb{P}(A \vee B) = \mathbb{P}(A) + \mathbb{P}(B) - \mathbb{P}(A \wedge B)$ and $\mathbb{P}(A \wedge B) \geq 0$ (according to the first axiom of probability), it follows that $\mathbb{P}(A \vee B) \leq \mathbb{P}(A) + \mathbb{P}(B)$. Using this inequality and the fact that conjunction is associative, we get

$$\mathbb{P}\left[\bigvee_{i=1}^{n+1} A_i\right] \leq \mathbb{P}\left[\bigvee_{i=1}^{n} A_i\right] + \mathbb{P}[A_{n+1}] \Rightarrow \mathbb{P}\left[\bigvee_{i=1}^{n+1} A_i\right] \leq \sum_{i=1}^{n+1} \mathbb{P}[A_i]. \tag{2.2.10}$$

2.3 Chance Constrained Path Planning

The objective of chance constrained path planning is to find a control sequence that generates a path that minimizes a given cost function while not violating the probabilistic chance constraints specified by the user. Consider a discrete-time linear system with uncertainty, where the states at timestep k, x_k, evolve according to

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{2.3.1}$$

where A is the dynamics matrix, B is the input matrix and u_k is the control input at timestep k. The uncertainty is represented by the process noise w_k. In this section, two types of uncertainties are considered: Gaussian and distributionally robust uncertainty. In the case of Gaussian uncertainty, we assume $w_k \sim \mathcal{N}(0, \Sigma_w)$, whereas in the distributionally robust case, the distribution of w_k is consider unknown. Instead, w_k belongs to a moment-based ambiguity set that includes all distributions with zero mean and covariance Σ_w. If the input is a fixed affine function $u_k = K_k x_k + g_k$, the mean state $\hat{x}$ and the covariance Σ_x evolves according to

$$\hat{x}_{k+1} = (A + BK_k)\hat{x}_k + Bg_k, \tag{2.3.2}$$

$$\Sigma_{x_{k+1}} = (A + BK_k)\Sigma_{x_k}(A + BK_k)^\top + \Sigma_w. \tag{2.3.3}$$

In this thesis, the chance constraints will be expressed in terms of obstacle avoidance, where the risk of colliding with an obstacle $\mathcal{O}$ given an uncertain state x is upper bounded by a risk parameter δ, according to

$$\sup \mathbb{P}_x (x \in \mathcal{O}) \leq \delta. \tag{2.3.4}$$

When the obstacle is a convex polytope, it can be expressed as a conjunction of linear constraints. That is,

$$\mathcal{O} = \{x | \wedge_{j=1}^{n} a_j^\top x \leq b_j\}. \tag{2.3.5}$$

Accordingly, the probabilistic constraint (2.3.4) can be rewritten as

$$\sup \mathbb{P}_x \left(\bigwedge_{j=1}^{n} a_j^\top x \leq b_j \right) \leq \delta. \tag{2.3.6}$$

This constraint is handled by enlarging the obstacle with a tightening parameter $\beta > 0$ (figure 2.1) and requiring that the mean state $\hat{x}$ lies outside of the enlarged obstacle. The tightening parameter β depends on the risk parameter δ and the covariance and is different for Gaussian and distributionally robust uncertainty. Constraint tightening will be explored further in the next section.

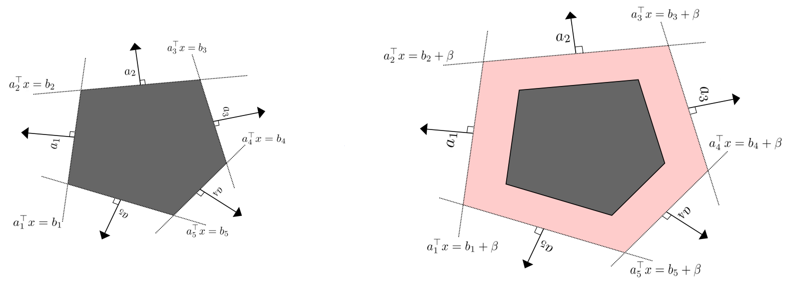

Figure 2.1 Tightening of a convex polytopic obstacle. The obstacle (left) is a conjunction of linear constrains $\wedge_{j=1}^{5} a_j^\top x \leq b_j$ and the tightened obstacle (right) is a conjunction of the tightened linear constraints $\wedge_{j=1}^{5} a_j^\top x \leq b_j + \beta$, where $\beta > 0$.

If the position of the obstacle is also subject to the same type of uncertainty, the probabilistic constraint has to be slightly adjusted. The uncertain obstacle location can be expressed as

$$\mathcal{O} = \mathcal{O}^0 \oplus \hat{c} \oplus c, \tag{2.3.7}$$

where $\mathcal{O}^0$ represents the known shape of the obstacle and $\hat{c}$ represents a known nominal translation. The location uncertainty and unpredictable motion of the obstacle is represented by c. The chance constraint with incorporated obstacle-uncertainty is stated as

$$\sup \mathbb{P}_x \left(\bigwedge_{j=1}^{n} a_j^\top x \leq a_j^\top c_j \right) \leq \delta. \tag{2.3.8}$$

where $c_j = \hat{c}_j + c$ is a point on the jth constraint of obstacle $\mathcal{O}$, with first and second central moments $\hat{c}_j$ and Σ_c. The same kind of uncertainty as the states are considered for obstacle uncertainty, i.e., Gaussian or distributionally robust uncertainty.

2.4 Constraint Tightening

In this section, we will show how the chance constraint in (2.3.8) can be converted into a deterministic linear constraint on the state mean in the case of Gaussian and distributionally robust uncertainty. Then, the tightening parameters for the two cases will be compared.

Gaussian chance constraints

Using obstacle tightening to handle Gaussain chance constraints was proposed and motivated in [19] and this framework was extended in [17] to also include obstacle uncertainty. The presented proof is based on the same principles as mentioned in chapter 3.2 of [17] but the approach was slightly adjusted to achieve coherency with the corresponding proof for distributionally robust constraint tightening.

Theorem 2.1 (Gaussian chance constraint tightening):

The Gaussian chance constraint

$$\mathbb{P}_x \left[\bigwedge_{j=1}^{n} a_j^\top x \le a_j^\top c_j \right] \le \delta, \tag{2.4.1}$$
$$x \sim \mathcal{N}(\hat{x}, \Sigma_x),$$
$$c_j \sim \mathcal{N}(\hat{c}_j, \Sigma_c),$$

is fulfilled if

$$\vee_{j=1}^{n} \left(a_j^\top \hat{x} \ge a_j^\top \hat{c}_j + \beta_{Gauss} \right),$$
$$\beta_{Gauss} = \sqrt{2}\,\mathrm{erf}^{-1}(1-2\delta) \left\| (\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j \right\| = -\Phi^{-1}(\delta) \left\| (\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j \right\|.$$

Proof

Using the Fréchet inequality (2.2.8), the probability in 2.4.1 can be upper bounded according to

$$\mathbb{P}_x \left[\bigwedge_{j=1}^{n} a_j^\top x \le a_j^\top c_j \right] \le \bigvee_{j=1}^{n} \mathbb{P}_x \left[a_j^\top x \le a_j^\top c_j \right].$$

Now we want to express the probabilistic safety constraint

$$\mathbb{P}_x \left[a_j^\top x \le a_j^\top c_j \right] \le \delta$$

as a deterministic linear constraint in terms of mean and covariance. The variables x and c_j can be expressed in terms of their mean and error as $x = \hat{x} + e$ and $c_j = \hat{c}_j + c$, where $e \sim \mathcal{N}(0, \Sigma_x)$ and $c \sim \mathcal{N}(0, \Sigma_c)$. This results in

$$\begin{aligned}\mathbb{P}_x \left[a_j^\top x \le a_j^\top c_j \right] &= \mathbb{P}_x \left[a_j^\top (\hat{x} + e) \le a_j^\top (\hat{c}_j + c) \right] \\ &= \mathbb{P}_x [\underbrace{a_j^\top e - a_j^\top c}_{=y} \le a_j^\top \hat{c}_j - a_j^\top \hat{x}]\end{aligned}$$

A new random variable is defined as $y = a_j^\top e - a_j^\top c$, where $y \sim \mathcal{N}(0, \Sigma_y)$ and $\Sigma_y = a_j^\top(\Sigma_x + \Sigma_c)a_j$. Since y is Gaussian, it can be represented in terms of the standard Gaussian distribution as

$$y = \hat{y} + \sqrt{\Sigma_y} z = \sqrt{\Sigma_y} z, \;\; z \sim \mathcal{N}(0,1)$$

The probability can then be expressed as

$$\begin{aligned}
\mathbb{P}_x\left[\sqrt{\Sigma_y} z \le a_j^\top \hat{c}_j - a_j^\top \hat{x}\right] &= \mathbb{P}_x\left[z \le \frac{a_j^\top \hat{c}_j - a_j^\top \hat{x}}{\sqrt{\Sigma_y}}\right] \\
&= \Phi\left(\frac{a_j^\top \hat{c}_j - a_j^\top \hat{x}}{\sqrt{\Sigma_y}}\right) \\
&= \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{a_j^\top \hat{c}_j - a_j^\top \hat{x}}{\sqrt{2\Sigma_y}}\right)\right].
\end{aligned}$$

Since the probability should be upper bounded by δ, we get the following deterministic linear constraints:

$$\begin{aligned}
\frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{a_j^\top \hat{c}_j - a_j^\top \hat{x}}{\sqrt{2\Sigma_y}}\right)\right] &\le \delta \\
-\operatorname{erf}\left(\frac{a_j^\top \hat{c}_j - a_j^\top \hat{x}}{\sqrt{2\Sigma_y}}\right) &\ge 1 - 2\delta \\
\frac{a_j^\top \hat{x} - a_j^\top \hat{c}_j}{\sqrt{2\Sigma_y}} &\ge \operatorname{erf}^{-1}(1 - 2\delta) \\
a_j^\top \hat{x} &\ge a_j^\top \hat{c}_j + \sqrt{2\Sigma_y}\operatorname{erf}^{-1}(2\delta - 1).
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\Phi\left(\frac{a_j^\top \hat{c}_j - a_j^\top \hat{x}}{\sqrt{\Sigma_y}}\right) &\le \delta \\
a_j^\top \hat{c}_j - a_j^\top \hat{x} &\le \sqrt{\Sigma_y}\Phi^{-1}(\delta) \\
a_j^\top \hat{x} &\ge a_j^\top \hat{c}_j - \sqrt{\Sigma_y}\Phi^{-1}(\delta).
\end{aligned}$$

Inserting $\sqrt{\Sigma_y} = \left\|(\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j\right\|$, we get

$$\begin{aligned}
&\vee_{j=1}^{n}\left(a_j^\top \hat{x} \ge a_j^\top \hat{c}_j + \sqrt{2}\operatorname{erf}^{-1}(1 - 2\delta)\left\|(\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j\right\|\right), \\
&\vee_{j=1}^{n}\left(a_j^\top \hat{x} \ge a_j^\top \hat{c}_j - \Phi^{-1}(\delta)\left\|(\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j\right\|\right).
\end{aligned}$$ □

Distributionally robust chance constraints

Here, we present a theorem for handling distributionally robust risk constraints with obstacle tightening, which was used in [1]. The approach is a modification of Theorem 3.1 in [20] tailored to risk bounded path planning in the presence of state and obstacle-uncertainty.

Theorem 2.2 (Distributionally robust chance constraint tightening):

The Distributionally robust chance constraint

$$\sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x \left[\bigwedge_{j=1}^{n} a_j^\top x \leq a_j^\top c_j \right] \leq \delta, \tag{2.4.2}$$

$$x \sim \mathbb{P}_x \in \mathcal{P}^x = \left\{ \mathbb{P}_x \mid \mathbb{E}[x] = \hat{x}, \mathbb{E}[(x-\hat{x})(x-\hat{x})^\top] = \Sigma_x \right\}$$

$$c_j \sim \mathbb{P}_{c_j} \in \mathcal{P}^c = \left\{ \mathbb{P}_{c_j} \mid \mathbb{E}[c_j] = \hat{c}_j, \mathbb{E}[(c_j-\hat{c}_j)(c_j-\hat{c}_j)^\top] = \Sigma_c \right\}$$

is fulfilled if

$$\vee_{j=1}^{n} \left(a_j^\top \hat{x} \geq a_j^\top \hat{c}_j + \beta_{robust} \right),$$

$$\beta_{robust} = \sqrt{\frac{1-\delta}{\delta}} \left\| (\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j \right\|.$$

PROOF
Using the Fréchet inequality (2.2.8), the probability in 2.4.2 can be upper bounded according to

$$\sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x \left[\bigwedge_{j=1}^{n} a_j^\top x \leq a_j^\top c_j \right] \leq \bigvee_{j=1}^{n} \sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x \left[a_j^\top x \leq a_j^\top c_j \right].$$

Now we want to express the probabilistic safety constraint

$$\sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x \left[a_j^\top x \leq a_j^\top c_j \right] \leq \delta$$

as a deterministic linear constraint in terms of mean and covariance. The variables x and c_j can be expressed in terms of their mean and error as $x = \hat{x} + e$ and $c_j = \hat{c}_j + c$, where e and c are zero-mean random variables with covariance Σ_x and Σ_c, respectively. This results in

$$\begin{aligned} \sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x \left[a_j^\top x \leq a_j^\top c_j \right] &= \sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x \left[a_j^\top (\hat{c}_j + c) \geq a_j^\top (\hat{x} + e) \right] \\ &= \sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x [\underbrace{a_j^\top c - a_j^\top e}_{=y} \geq a_j^\top \hat{x} - a_j^\top \hat{c}_j]. \end{aligned}$$

A new random variable is defined as $y = a_j^\top c - a_j^\top e$, where y has zero mean and covariance $\Sigma_y = a_j^\top(\Sigma_x + \Sigma_c)a_j$. Since y is zero-mean, Cantelli's inequality (2.2.7) can be used to upper bound the probability according to

$$\sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_x\left[y \geq a_j^\top \hat{x} - a_j^\top \hat{c}_j\right] \leq \frac{\Sigma_y}{\Sigma_y + (a_j^\top \hat{x} - a_j^\top \hat{c}_j)^2}.$$

Since the probability should be upper bounded by δ, we get the following deterministic linear constraint:

$$\begin{aligned}
\frac{\Sigma_y}{\Sigma_y + (a_j^\top \hat{x} - a_j^\top \hat{c}_j)^2} &\leq \delta \\
\Sigma_y + (a_j^\top \hat{x} - a_j^\top \hat{c}_j)^2 &\geq \frac{\Sigma_y}{\delta} \\
(a_j^\top \hat{x} - a_j^\top \hat{c}_j)^2 &\geq \Sigma_y\left(\frac{1}{\delta} - 1\right) \\
a_j^\top \hat{x} - a_j^\top \hat{c}_j &\geq \sqrt{\Sigma_y}\sqrt{\frac{1-\delta}{\delta}} \\
a_j^\top \hat{x} - a_j^\top \hat{c}_j &\geq \sqrt{a_j^\top(\Sigma_x + \Sigma_c)a_j}\sqrt{\frac{1-\delta}{\delta}} \\
a_j^\top \hat{x} &\geq a_j^\top \hat{c}_j + \sqrt{\frac{1-\delta}{\delta}}\left\|(\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j\right\|.
\end{aligned}$$

□

Comparison: Gaussian and distributionally robust tightening

As previously mentioned, the tightening parameter for Gaussian and distributionally robust chance constraints are

$$\beta_{Gauss} = \sqrt{2}\,\mathrm{erf}^{-1}(1-2\delta)\left\|(\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j\right\|, \tag{2.4.3}$$

$$\beta_{robust} = \sqrt{\frac{1-\delta}{\delta}}\left\|(\Sigma_x + \Sigma_c)^{\frac{1}{2}} a_j\right\|. \tag{2.4.4}$$

Figure 2.2 shows a comparison between the Gaussian scaling constant $\sqrt{2}\,\mathrm{erf}^{-1}(1-2\delta)$ and the distributionally robust scaling constant $\sqrt{\frac{1-\delta}{\delta}}$ for different values of δ. Note that both scaling constants are a decreasing function of the risk parameter δ, which means that a higher risk leads to a smaller tightening of the obstacle and vice versa. Aside from the scaling constants, the tightening parameter is the same for both Gaussian and distributionally robust chance constraints. The larger scaling constant for distributionally robust uncertainty leads to a stronger tightening of the obstacle and therefore a higher degree of conservatism when the same risk parameter is used.

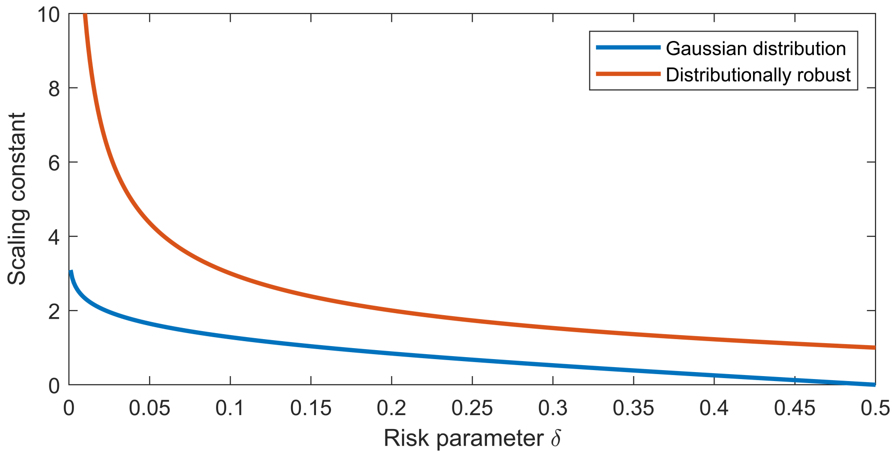

Figure 2.2 Scaling constants for Gaussian and distributionally robust constraint tightening for different values of δ.

2.5 DR-RRT

DR-RRT (Distributionally Robust Rapidly Exploring Random Tree) [1] is a sampling-based motion planning algorithm that builds trees of state distributions while enforcing distributionally robust chance constraints. The tree is expanded by attempting to steer from an existing point in the tree to a randomly sampled point x_s in a specified number of timesteps. The generated path is then checked for distributionally robust constraint satisfaction before being added to the tree. Figure 2.3 illustrates the principle of DR-RRT with an example using individual risk bounds $\delta = 0.05$ for all obstacles and timesteps. The lines show the trajectory of the mean state $\hat{x}$ while the ellipses represent the covariance Σ_x at the end of a generated path. The steering is obtained from finite horizon linear quadratic dynamic programming and accordingly, the generated trajectories can have some curvature. This happens when the robot is steered from a point in the tree where the velocity goes in another direction than towards the sampled point x_s. In that case, the steering must turn the robot towards the sample which leads to curved trajectories.

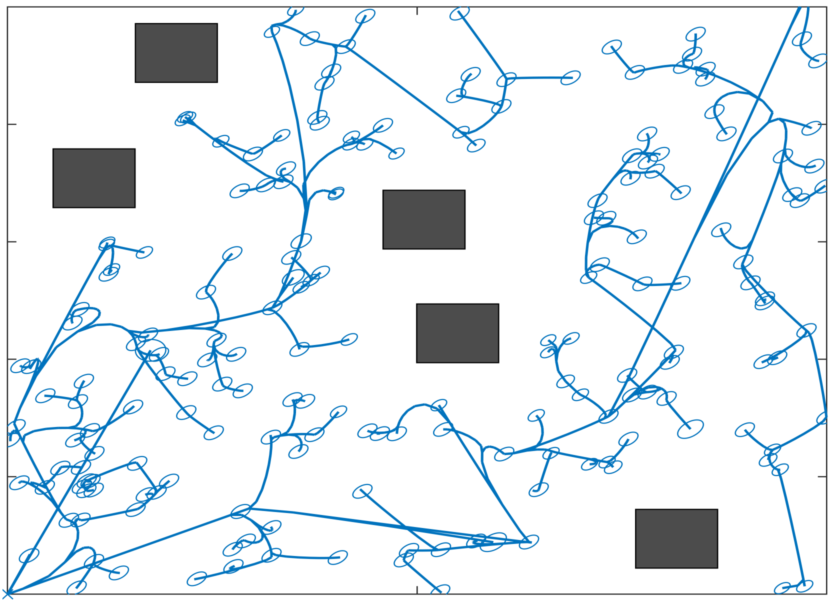

Figure 2.3 DR-RRT with individual risk bounds $\delta = 0.05$ in an $[0, 1]^2$ environment with 5 randomly located deterministic obstacles.

3

Problem Formulation

3.1 Introduction

In this chapter, we formulate the chance constrained path planning problem considered in this thesis. The chance constraints are distributionally robust and expressed in terms of obstacle avoidance. The main objective is to find a control sequence that generates a path that minimizes the finite-horizon cost function while not exceeding the total risk budget specified by the user. Allocating the risk for different timesteps and obstacles in a non-uniform way has the potential to minimize the cost function and generate less conservative paths. We begin by specifying the model dynamics considered. Then, a formal problem statement with a specified risk budget for the entire path from start to goal is formulated. Using Boole's inequality, the chance constraint is decomposed into individual chance constraints for each obstacle and timestep in the path, enabling risk allocation.

3.2 Model Dynamics

Consider a robot operating in an uncertain environment, $\mathcal{X} \subseteq \mathbb{R}^n$, cluttered with obstacles. The set of obstacles is denoted as $\mathcal{B}$ with $|\mathcal{B}| = N$. The robot is modeled as a stochastic discrete-time linear time invariant system where the state of the robot evolves according to

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{3.2.1}$$

where $x_k \in \mathbb{R}^n$ and $u_k \in \mathbb{R}^m$ is the system state and input at timestep k, respectively. A is the dynamics matrix and B is the input matrix. The process noise $w_k \in \mathbb{R}^m$ is a random vector with zero-mean that is independent and identically distributed across all timesteps. The distribution of w_k, $\mathbb{P}_{w_k}$, is unknown but belongs to a moment-based ambiguity set of distributions, $\mathcal{P}^w$, defined as

$$\mathcal{P}^w = \left\{ \mathbb{P}_{w_k} \mid \mathbb{E}[w_k] = 0, \mathbb{E}[w_k w_k^\top] = \Sigma_w \right\}. \tag{3.2.2}$$

The initial state x_0 is subject to a similar uncertainty model as the noise, with the distribution belonging to a moment-based ambiguity set, $\mathbb{P}_{x_0} \in \mathcal{P}^{x_0}$, where

$$\mathcal{P}^{x_0} = \left\{ \mathbb{P}_{x_0} \mid \mathbb{E}[x_0] = \hat{x}_0, \mathbb{E}[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^\top] = \Sigma_{x_0} \right\}. \tag{3.2.3}$$

The shape and orientation of the obstacles are known, but their position is subject to uncertainty, such that

$$\mathcal{O}_{ik} = \mathcal{O}_i^0 \oplus \hat{c}_{ik} \oplus c_{ik}, \quad \forall i \in \mathcal{B}, \tag{3.2.4}$$

where $\mathcal{O}_{ik}$ denotes obstacle $i \in \mathcal{B}$ at timestep k. The known shape of the obstacle is represented by $\mathcal{O}_i^0 \subset \mathbb{R}^n$ while $\hat{c}_{ik}$ represents a known nominal translation. The location uncertainty and unpredictable motion of obstacle $i \in \mathcal{B}$ is represented by $c_{ik} \in \mathbb{R}^n$ which is a random vector with unknown distribution $\mathbb{P}_{c_{ik}}$ that belongs to a moment-based ambiguity set $\mathcal{P}^{c_{ik}}$, defined as

$$\mathcal{P}^{c_{ik}} = \left\{ \mathbb{P}_{c_{ik}} \mid \mathbb{E}[c_{ik}] = \hat{c}_{ik}, \mathbb{E}[(c_{ik} - \hat{c}_{ik})(c_{ik} - \hat{c}_{ik})^\top] = \Sigma_{c_{ik}} \right\}. \tag{3.2.5}$$

Collision with the obstacles should be avoided. The state and input are nominally subject to the following constraints:

$$x_k \notin \bigcup_{i \in \mathcal{B}} \mathcal{O}_{ik}, \quad u_k \in \mathcal{U}. \tag{3.2.6}$$

The environment $\mathcal{X} \subset \mathbb{R}^n$, obstacles $\mathcal{O}_{ik} \subset \mathbb{R}^n$ and input constraints $\mathcal{U} \subset \mathbb{R}^m$ are assumed to be convex polytopes and can therefore be represented by a conjunction of linear inequalities:

$$\mathcal{U} = \left\{ u_k \mid \bigwedge_{j_u=1}^{n_u} a_{uj_u}^\top x_k \leq b_{ukj_u} \right\}, \tag{3.2.7}$$

$$\mathcal{X} = \left\{ x_k \mid \bigwedge_{j_0=1}^{n_0} a_{0j_0}^\top x_k \leq b_{0kj_0} \right\}, \tag{3.2.8}$$

$$\mathcal{O}_{ik} = \left\{ x_k \mid \bigwedge_{j=1}^{n_i} a_{ij}^\top x_k \leq b_{ikj} \right\}. \tag{3.2.9}$$

Note that the environmental bounds are not considered in the state constraint as they will not be treated probabilistically. The initial state x_0 is subject to uncertainty but assumed safe and will therefore not be included in the probabilistic chance constraint. Because distributionally robust constraints are of infinite dimension, solving the chance constrained problem exactly is practically impossible which is why we will seek an approximate solution.

3.3 Problem Statement

Problem 1. We seek to (approximately) solve the following chance constrained path planning problem. Given an uncertain initial state $x_0 \sim \mathbb{P}_{x_0}$ and a set of goal locations $\mathcal{X}_{goal} \subset \mathbb{R}^n$, find a feedback control policy $\pi = [\pi_0, ..., \pi_{T-1}]$ such that applying the control inputs $u_k = \pi_k(x_k), \forall k = 0, \ldots, T-1$ yields a probibalistically feasible path from the initial state to the goal that minimizes the finite-horizon cost function. The distributionally robust control problem is formulated as

$$\underset{\pi}{\text{minimize}} \quad \sum_{k=0}^{T-1} \ell_t(\hat{x}_k, \mathcal{X}_{goal}, u_k) + \ell_T(\hat{x}_T, \mathcal{X}_{goal}) \tag{3.3.1a}$$

$$\text{subject to} \quad x_{k+1} = Ax_k + Bu_k + w_k, \tag{3.3.1b}$$

$$x_0 \sim \mathbb{P}_{x_0} \in \mathcal{P}^x, \tag{3.3.1c}$$

$$w_t \sim \mathbb{P}_w \in \mathcal{P}^w, \tag{3.3.1d}$$

$$u_k \in \mathcal{U}, \tag{3.3.1e}$$

$$\sup_{\mathbb{P}_{x_k} \in \mathcal{P}^x} \mathbb{P}_{x_k} \left[\bigvee_{k=1}^{T} x_k \in \bigcup_{i \in \mathcal{B}} \mathcal{O}_{ik} \right] \leq \Delta, \tag{3.3.1f}$$

$$\mathcal{O}_{ik} = \mathcal{O}_i^0 \oplus \hat{c}_{ik} \oplus c_{ik}, \quad \forall i \in \mathcal{B}, \tag{3.3.1g}$$

$$c_{ik} \sim \mathbb{P}_{c_{ik}} \in \mathcal{P}_{c_{ik}}, \tag{3.3.1h}$$

where $\ell_t(.)$ is the stage cost function that quantifies the distance to the goal set and actuator effort. The stage cost is expressed in terms of the mean state of the robot, $\hat{x}_k$, so that all uncertainty comes from the constraints. Δ represents the user-prescribed risk constraint for the entire planning horizon from $k = [1 : T]$, such that the worst-case probability of colliding with any of the N obstacles over the planning horizon should be at most Δ. The chance constraint in Problem 1 (3.3.1f) can be decomposed into individual chance constraints for each obstacle at each timestep. The individual risk bound for obstacle i at timestep k is then fixed to a value δ_{ik}, such that

$$\sum_{k=1}^{T} \sum_{i=1}^{N} \delta_{ik} \leq \Delta. \tag{3.3.2}$$

For convenience, we define a vector of all individual risk bounds, $\delta = [\delta_{11}, ..., \delta_{NT}]$. By fixing δ in a non-uniform way, the conservatism can be reduced while still enforcing the same chance constraint (3.3.1f). Accordingly, Problem 1 can be reduced to Problem 2 (proof in Section 3.4).

Problem 2. We seek to (approximately) solve the following distributionally robust path planning problem with individual risk bounds:

$$\underset{\pi,\,\delta}{\text{minimize}} \quad \sum_{k=0}^{T-1} \ell_t(\hat{x}_k, \mathcal{X}_{goal}, u_k) + \ell_T(\hat{x}_k, \mathcal{X}_{goal}) \tag{3.3.3a}$$

$$\text{subject to} \quad (3.3.1c) - (3.3.1e), (3.3.1g) - (3.3.1h), \tag{3.3.3b}$$

$$\sup_{\mathbb{P}_{x_k} \in \mathcal{P}^x} \mathbb{P}_{x_k}(x_k \in \mathcal{O}_{ik}) \leq \delta_{ik}, \quad {}^{\forall i \in \mathcal{B},}_{\forall k \in [1:T]}, \tag{3.3.3c}$$

$$\sum_{k=1}^{T} \sum_{i=1}^{N} \delta_{ik} \leq \Delta. \tag{3.3.3d}$$

Note that the only difference between Problems 1 and 2 is that Problem 2 is expressed with individual chance constraints and that the risk allocations, δ, are used to minimize the conservatism alongside the control policy π.

3.4 Decomposition of Chance Constraint

In this section, we seek to prove that Problem 1 can in fact be reduced to Problem 2 by decomposing the chance constraint for the entire planning horizon into individual chance constraints using Boole's inequality, as proposed in [8, 9]. Hence, we seek to prove the following theorem:

Theorem 3.1 (Decomposition of chance constraint in Problem 1): The chance constraint in Problem 1 (3.3.1f) can be decomposed into individual chance constraints for all obstacles i at all timesteps k (3.3.3c, 3.3.3d), i.e.,

$$\left.\begin{aligned} \sup_{\mathbb{P}_{x_k}\in\mathcal{P}^x} \mathbb{P}_{x_k}(x_k \in \mathcal{O}_{ik}) \leq \delta_{ik}, \\ \forall i \in \mathcal{B}, \forall k \in [1:T], \\ \sum_{k=1}^{T}\sum_{i=1}^{N} \delta_{ik} \leq \Delta. \end{aligned}\right\} \implies \sup_{\mathbb{P}_{x_k}\in\mathcal{P}^x} \mathbb{P}_{x_k}\left[\bigvee_{k=1}^{T} x_k \in \bigcup_{i\in\mathcal{B}} \mathcal{O}_{ik}\right] \leq \Delta.$$

Proof

The constraint $x_k \in \bigcup_{i\in\mathcal{B}} \mathcal{O}_{ik}$ is equivalent to $\bigvee_{i=1}^{N} x_k \in \mathcal{O}_{ik}$. Accordingly, the chance constraint in (3.3.1f) can be written as

$$\sup_{\mathbb{P}_{x_k}\in\mathcal{P}^x} \mathbb{P}_{x_k}\left[\bigvee_{k=1}^{T}\bigvee_{i=1}^{N} x_k \in \mathcal{O}_{ik}\right] \leq \Delta.$$

where $x_k \in \mathcal{O}_{ik}$ is the event of colliding with obstacle i at timestep k. According to Boole's inequality (2.2.9), the probability of *at least one* of the events $x_k \in \mathcal{O}_{ik}$ occurring is no greater than the sum of the individual probabilities for the events. This means that the worst-case probability of at least one collision occurring is upper-bounded by the sum of the worst-case probabilities of the individual collision-events, i.e.,

$$\sup_{\mathbb{P}_{x_k}\in\mathcal{P}^x} \mathbb{P}_{x_k}\left[\bigvee_{k=1}^{T}\bigvee_{i=1}^{N} x_k \in \mathcal{O}_{ik}\right] \leq \sum_{k=1}^{T}\sum_{i=1}^{N} \sup_{\mathbb{P}_{x_k}\in\mathcal{P}^x} \mathbb{P}_{x_k}(x_k \in \mathcal{O}_{ik}).$$

Applying (3.3.3c) and (3.3.3d) on the right hand side yields

$$\sum_{k=1}^{T}\sum_{i=1}^{N} \underbrace{\sup_{\mathbb{P}_{x_k}\in\mathcal{P}^x} \mathbb{P}_{x_k}(x_k \in \mathcal{O}_{ik})}_{\leq\delta_{ik}} \leq \sum_{k=1}^{T}\sum_{i=1}^{N} \delta_{ik} \leq \Delta. \qquad \square$$

3.5 Risk Treatment

Since the obstacles $\mathcal{O}_{ik}$ are convex polytopes they can be represented by the intersection of n_i halfspaces defined by hyperplanes in the form $a_{ij}^\top x = b_{ikj}$, where $j = 1, \ldots, n_i$. Collision with obstacle i at timestep k occurs if the position of the robot lies inside the obstacle, $x_k \in \mathcal{O}_{ik}$. This can be expressed as a conjunction of n_i linear constraints on the robots position;

$$\bigwedge_{j=0}^{n_i} a_{ij}^\top x_k < b_{ikj}. \tag{3.5.1}$$

According to the individual chance constraints (3.3.3c), the worst-case probability of colliding with obstacle i at timestep k should be at most δ_{ik}. This can be stated as

$$\sup_{\mathbb{P}_x \in \mathcal{P}^x} \mathbb{P}_{x_k} \left[\bigwedge_{j=1}^{n_i} a_{ij}^\top x_k < a_{ij}^\top c_{ikj} \right] \leq \delta_{ik}, \tag{3.5.2}$$

where $c_{ikj} = \hat{c}_{ikj} + c_{ik}$ is a point on the jth constraint of obstacle $\mathcal{O}_{ik}$, with first and second central moments $\hat{c}_{ikj}$ and $\Sigma_{c_{jk}}$. Using Theorem 2.4, the distributionally robust chance constraint in (3.5.2) can be handled by a disjunction of linear constraints on the state mean $\hat{x}_k$, such that

$$\vee_{j=1}^{n_i} \left(a_{ij}^\top \hat{x}_k \geq a_{ij}^\top \hat{c}_{ikj} + \sqrt{\frac{1-\delta_{ik}}{\delta_{ik}}} \left\| (\Sigma_{x_k} + \Sigma_{c_{jk}})^{\frac{1}{2}} a_{ij} \right\| \right). \tag{3.5.3}$$

This represents a deterministic constraint tightening where the mean position of the robot is required to lie outside of the tightened obstacle in order to fulfill the distributionally robust chance constraint. For convenience, we denote the tightening of the jth constraint of obstacle $\mathcal{O}_{ik}$ as

$$\beta_{ikj} = \sqrt{\frac{1-\delta_{ik}}{\delta_{ik}}} \left\| (\Sigma_{x_k} + \Sigma_{c_{jk}})^{\frac{1}{2}} a_{ij} \right\|. \tag{3.5.4}$$

We define Boolean variables $\mathbf{h}_{ikj}$ and $\mathbf{h}_{ik}$ that represents the mean state being outside the tightened jth constraint of $\mathcal{O}_{ik}$ and outside the tightened obstacle $\mathcal{O}_{ik}$, respectively.

$$\mathbf{h}_{ikj} = \begin{cases} \top, & a_{ij}^\top \hat{x}_k \geq a_{ij}^\top \hat{c}_{ikj} + \beta_{ikj} \\ \bot, & \text{otherwise} \end{cases} \tag{3.5.5}$$

$$\mathbf{h}_{ik} = \begin{cases} \top, & \vee_{j=1}^{n_i} \mathbf{h}_{ikj} \\ \bot, & \text{otherwise} \end{cases} \tag{3.5.6}$$

The distributionally robust chance constraint for obstacle i at timestep k in (3.5.2) is then fulfilled if $\mathbf{h}_{ik} = \top$. Now the question becomes how to fix the risk allocations δ_{ik} in order to minimize the conservatism while still enforcing the chance constraints (3.3.3c–3.3.3d).

3.6 Summary

In this chapter, the distributionally robust chance constrained path planning problem considered in this thesis was formulated. The problem is solved by finding a feedback control policy that steers a robot to its goal in a way that minimizes the finite-horizon cost function while not violating the user-specified risk constraints. The risk constraints are expressed in terms of obstacle avoidance in the presence of distributionally robust uncertainty, where the distribution belongs to a moment-based ambiguity set. First, the problem was formulated with a chance constraint for the entire planning horizon. Then, using Boole's inequality, the chance constraint was decomposed into individual chance constraints for each obstacle and timestep. The problem statement was then reformulated to consider the individual chance constraints instead. Since the obstacles are assumed to be convex polytopes, the individual chance constraints for obstacle avoidance can be handled by requiring that the mean state of the robot lies outside of the tightened obstacle, where the tightening is a deterministic function of the allocated risk and the covariance. The question then becomes how to allocate risk to the individual chance constraints in a way that reduces the conservatism while still assuring probabilistic constraint fulfillment. A potential solution to this problem will be explored in the next chapter.

4

DR-RRT with Exact Risk Allocation

4.1 Introduction

In this chapter, Distributionally Robust Rapidly Exploring Random Tree (DR-RRT) with Exact Risk Allocation (ERA) is described. DR-RRT is a sampling-based algorithm that grows trees of state distributions while enforcing distributionally robust chance constraints. Usually, uniform risk allocation, where each obstacle and timestep are assigned the same risk $\delta_{ik} = \frac{\Delta}{T \cdot N}$, is used to check the probabilistic feasibility of the generated path. In this case, the sum of all risk allocations δ_{ik} over the time horizon $k = [1 : T]$ is equal to Δ, thus fulfilling the inequality in (3.3.3d). Each timestep in the path is then checked for probabilistic feasibility according to the constraint in (3.5.3). With ERA, the problem is tackled in the opposite direction by *first* assigning risks δ_{ik} that fulfill the chance constraints in (3.5.3) and *then* checking if the sum of all δ_{ik} exceeds the specified risk budget.

The rest of the chapter is structured as follows. First, the tree expansion algorithm is described, then the ERA and feasability check procedures are outlined. This is followed by a motivation of why ERA leads to less conservative paths being generated.

4.2 Tree Expansion

Algorithm 1 outlines the DR-RRT tree expansion with Exact Risk Allocation incorporated. First, a sample x_s is taken randomly from the feasible state set. The nearest $M \geq 1$ tree nodes, $N_{\mathbf{near}}$, are then identified according to a specific distance metric. A distance metric based on dynamic-control, such as the optimal cost-to-go function, is often better than one based solely on geometric distance [21]. Trajectories from all the near nodes to the sample are generated using a steering-function. In this case, the steering-function uses an unconstrained optimal feedback control policy obtained from finite horizon linear quadratic dynamic programming. The steering is done in a specified number of timesteps $T_{\mathbf{steer}} \leq T$. The state mean and covariance matrix is propagated using the control policy, $u_k = K_k x_k + g_k$, and thus evolves according to (2.3.2) and (2.3.3). The entire path, with both mean states and covariance matrices, from the near node to the sample, is returned by the steering-function. Note that the generated path does not depend on the risk allocations δ_{ik}. In the next step, Exact Risk Allocation is applied to the generated path, as outlined in Algorithm 2. The ERA-function returns risk allocations δ_{ik} for all obstacles i at all timesteps k in the path. The risk allocation is done so that the linear constraint in (3.5.3) is fulfilled. The ERA-procedure will be described further in Section 4.3.

In the next step, the total risk leading up to each timestep is calculated. This is done by summing up all risk allocations δ_{ik} up to a certain timestep, denoted as k^*. The entire path from $N_{\mathbf{near}}$ up to timestep $T_{\mathbf{steer}}$ is then checked for distributionally robust feasibility, as outlined in Algorithm 3. If the path is feasible, the total cost J and the residual risk δ_{res} is calculated and used to assign a score to the path from node $N_{\mathbf{near}}$. When paths from all near nodes that are DR-feasible have been assigned a score, the path with the best score is chosen and a new node and edge is added to the tree. The exact construction of the scoring algorithm is not covered in the scope of this thesis, but the reasoning behind it will be discussed in Section 6.2. The residual risk δ_{res} is also added to the node, which can in turn be reallocated when steering from this node to a new sample. The reallocation to future paths will be described in Section 4.4. Feasible *portions* of the path is also added to the tree in the same manner.

4.3 Exact Risk Allocation

The purpose of Exact Risk Allocation (ERA) is to allocate as little risk δ_{ik} as possible for all obstacles i at all timesteps k that fulfills the chance constraint in (3.3.3c). As previously mentioned, the chance constraint for obstacle i at time step k is fulfilled if $\mathbf{h}_{ik} = \top$ (3.5.5). Accordingly, we wish to solve the following problem for all obstacles and timesteps in the path:

Problem 3.

$$\begin{aligned} \text{minimize} \quad & \delta_{ik} \\ \text{subject to} \quad & \mathbf{h}_{ik} = \top \end{aligned} \tag{4.3.1}$$

Note that Problem 3 cannot be solved if the mean state $\hat{x}_k$ is inside the obstacle. Such paths will therefore immediately be dismissed as non-feasible.

Since $\mathbf{h}_{ik}$ is a disjunction of Boolean variables, $\vee_{j=1}^{n_i}\mathbf{h}_{ikj}$, we formulate a new problem for the jth constraint:

Problem 4.

$$\begin{aligned} \text{minimize} \quad & \delta_{ik} \\ \text{subject to} \quad & \mathbf{h}_{ikj} = \top \end{aligned} \tag{4.3.2}$$

Then, the minimum solution to Problem 4 for $j = 1, \ldots, n_i$ is also the solution to Problem 3. Note that Problem 4 can only be solved when the mean state $\hat{x}_k$ lies on the outside of the *untightened* jth constraint, i.e., when

$$a_{ij}^\top \hat{x}_k \geq a_{ij}^\top \hat{c}_{ikj}. \tag{4.3.3}$$

Thus, we wish to solve Problem 4 for all constraints $j = 1, \ldots, n_i$ that fulfills (4.3.3) and then identify the minimum of these solutions. For $\mathbf{h}_{ikj} = \top$, the following condition must be fulfilled:

$$a_{ij}^\top \hat{x}_k \geq a_{ij}^\top \hat{c}_{ikj} + \beta_{ikj}. \tag{4.3.4}$$

Theorem 4.1:

The solution to Problem 4 is found from

$$\beta_{ikj} = a_{ij}^\top \hat{x}_k - a_{ij}^\top \hat{c}_{ikj}. \tag{4.3.5}$$

Proof

Since Σ_{x_k} and $\Sigma_{c_{jk}}$ are known and $\sqrt{\frac{1-\delta_{ik}}{\delta_{ik}}}$ is a decreasing function of δ_{ik}, maximizing β_{ikj} minimizes δ_{ik}. The solution to Problem 4 is thus found from maximizing β_{ikj} while still enforcing $\mathbf{h}_{ikj} = \top$, i.e., from (4.3.5). □

Rearranging (4.3.5) gives the solution to Problem 4 as

$$\delta_{ik} = \left(1 + \left(\frac{a_{ikj}^\top \hat{x}_k - a_{ikj}^\top \hat{c}_{ikj}}{\left\|(\Sigma_k + \Sigma_{jk}^c)^{\frac{1}{2}} a_{ikj}\right\|_2}\right)^2\right)^{-1}. \tag{4.3.6}$$

Accordingly, the ERA-procedure allocates risk δ_{ik} given by the minimum solution to Problem 4 and subsequently the solution to Problem 3, for all obstacles i at all timesteps k in the path.

4.4 Feasibility Check

The feasibility check is based on the total risk allocated up to timestep k; $\delta_{tot}(k)$. The risk constraints (3.3.3c–3.3.3d) has to hold for the entire planning horizon T and not just over the steering horizons $T_{\textbf{steer}}$. To assure this is the case, we begin by distributing the total risk budget Δ uniformly over all steering horizons according to

$$\Delta_{\textbf{steer}} = \frac{\Delta \cdot T_{\textbf{steer}}}{T}, \tag{4.4.1}$$

where $\Delta_{\textbf{steer}}$ is the risk budget for each steering horizon $T_{\textbf{steer}}$. An entire path, from a near node to a sample, is deemed feasible if the total risk allocated over the steering horizon $\delta_{tot}(T_{\textbf{steer}})$ fulfill

$$\delta_{tot}(T_{\textbf{steer}}) \leq \Delta_{\textbf{steer}}. \tag{4.4.2}$$

A similar reasoning can be applied to assure the feasibility of a portion of the steered path, from a near node up to a certain timestep k. Then, the total risk allocated up to that time step, $\delta_{tot}(k)$, has to fulfill

$$\delta_{tot}(k) \leq \Delta_k, \tag{4.4.3}$$

where Δ_k is the uniformly allocated risk budget up to time step k, such that

$$\Delta_k = \frac{k \cdot \Delta_{\textbf{steer}}}{T_{\textbf{steer}}}. \tag{4.4.4}$$

This means that a path, or a portion of a path, is considered feasible only when the total allocated risk (using ERA) does not exceed the corresponding total uniformly allocated risk.

While this method has less conservatism than uniform risk allocation, there are still a lot of conservatism present from allocating the total risk budget uniformly over all steering horizons. This conservatism can be mitigated by reallocating the residual risk of a horizon to the subsequent steering. If the entire risk budget $\Delta_{\textbf{steer}}$ or Δ_k is not used, such that $\delta_{tot}(T_{\textbf{steer}}) < \Delta_{\textbf{steer}}$ or $\delta_{tot}(k) < \Delta_k$, a residual for the newly generated node at timestep k or $T_{\textbf{steer}}$ can be created according to

$$\delta_{res} = \Delta_{\textbf{steer}} - \delta_{tot}(T_{\textbf{steer}}) \text{ or} \tag{4.4.5}$$

$$\delta_{res} = \Delta_k - \delta_{tot}(k). \tag{4.4.6}$$

The residual risk can then be reallocated to new paths generated from this node. When a new point x_s is sampled, the residual of the near node $\delta_{res}[N_{\mathbf{near}}]$ can be allocated to the path generated by steering from $N_{\mathbf{near}}$ to the new sample. The total risk budget for the new path or path portion is then $\Delta_{\mathbf{steer}} + \delta_{res}[N_{\mathbf{near}}]$ or $\Delta_k + \delta_{res}[N_{\mathbf{near}}]$, respectively. The constraints in (4.4.2) and (4.4.3) are relaxed to

$$\delta_{tot}(T_{\mathbf{steer}}) \leq \Delta_{\mathbf{steer}} + \delta_{res}[N_{\mathbf{near}}] \text{ or} \tag{4.4.7}$$

$$\delta_{tot}(k) \leq \Delta_k + \delta_{res}[N_{\mathbf{near}}], \tag{4.4.8}$$

and the residual of $N_{\mathbf{near}}$ is added to the residual of newly created nodes originating from $N_{\mathbf{near}}$. That is,

$$\delta_{res} = \Delta_{\mathbf{steer}} + \delta_{res}[N_{\mathbf{near}}] - \delta_{tot}(T_{\mathbf{steer}}) \text{ or} \tag{4.4.9}$$

$$\delta_{res} = \Delta_k + \delta_{res}[N_{\mathbf{near}}] - \delta_{tot}(k). \tag{4.4.10}$$

Note that even when reallocating residual risks to future paths, the method still has some conservatism. First of all, the decomposition of the chance constraint in (3.3.1f) into individual chance constraints has inevitable conservatism. Secondly, the paths that are not feasible according to (4.4.9) are immediately dismissed, even though some of these path could potentially be made feasible by reallocating the residual risk from *future* paths. This conservatism could easily be reduced by also storing infeasible nodes in the hope that they become feasible when new paths are generated from the infeasible node. In this case, the infeasible paths would not be added to the tree until they are connected with new paths such that the *combination* of the paths becomes feasible. However, this would increase the computational expense since a number of infeasible branches would be generated and stored. It could also hinder the generation of feasible branches when infeasible nodes are identified as near nodes to a sample instead of feasible ones. Because of this, we decided not to use this approach and only reallocate risk to future horizons. However, in situations where computational expense is not an issue and M is large, reallocation from future paths could be used.

In the online risk allocation approach proposed in [17], the risk budget for the entire planning horizon was used to check the feasibility of a generated path. Incorporating this solution for distributionally robust chance constraints with the ERA-method could reduce the conservatism in a similar way as reallocating risk from future horizons. However, this would also increase the computational expense in the same way since early paths that use up a lot of the total risk budget, and is therefore much harder to expand, would be stored.

Algorithm 1 DR-RRT: Tree Expansion

Inputs: current tree $\mathcal{T}$, timestep k
x_s =sample($\mathcal{X}_k^{free}$)
$N_{\mathbf{near}}$ =NearestNodes($x_s, \mathcal{T}, M$)
for all $N_{\mathbf{near}}$ **do**
  ($\hat{x}_{path}, \Sigma_{path}$) =steer($N_{\mathbf{near}}, x_s, T_{\mathbf{steer}}$)
  δ_{ik} =ExactRiskAllocation($\hat{x}_{path}, \Sigma_{path}$)
  $\delta_{tot}(k^*) = \sum_{k=1}^{k^*} \sum_{i=1}^{N} \delta_{ik}$
  if DRFeasible($\delta_{tot}(T_{\mathbf{steer}}), \delta_{res}[N_{\mathbf{near}}]$) **then**
    $J = J[N_{\mathbf{near}}] + J(\hat{x}_{path}, \Sigma_{path})$
    $\delta_{res} = \delta_{res}[N_{\mathbf{near}}] + \Delta_{\mathbf{steer}} - \delta_{tot}(T_{\mathbf{steer}})$
    score($N_{\mathbf{near}}$) =assignScore(J, δ_{res})
Select path ($\hat{x}_{path}, \Sigma_{path}$) from $N_{\mathbf{near}}$ with best score
$\mathcal{T}$.AddNode($\hat{x}_{path}(T_{\mathbf{steer}}), \Sigma_{path}(T_{\mathbf{steer}})$)
$\mathcal{T}$.AddEdge($N_{\mathbf{near}}, \hat{x}_{path}(T_{\mathbf{steer}})$)
$\mathcal{T}$.AddResidual(δ_{res})
for $k = 1 : T_{\mathbf{steer}} - 1$ **do**
  if DRFeasible($\delta_{tot}(k), \delta_{res}[N_{\mathbf{near}}]$) **then**
    $\delta_{res} = \delta_{res}[N_{\mathbf{near}}] + \Delta_k - \delta_{tot}(k)$
    $\mathcal{T}$.AddNode($\hat{x}_{path}(k), \Sigma_{path}(k)$)
    $\mathcal{T}$.AddEdge($N_{\mathbf{near}}, \hat{x}_{path}(k)$)
    $\mathcal{T}$.AddResidual(δ_{res})

Algorithm 2 ExactRiskAllocation

Inputs: Path $\hat{x}_{path}, \Sigma_{path}$
Output: Risk allocation matrix $\delta_{ik} \in \mathbb{R}^{N \times T_{\mathbf{steer}}}$
for $k = 1 : T_{\mathbf{steer}}$ **do**
  for $i = 1 : N$ **do**
    Assign δ_{ik} according to (4.3.6)
`return` δ_{ik}

Algorithm 3 DRFeasible

Inputs: total risk $\delta_{tot}(k)$ and residual of $N_{\mathbf{near}}$, $\delta_{res}[N_{\mathbf{near}}]$
Output: true if DR-feasible, otherwise false
if $\delta_{tot}(k)$ satisfies (4.4.8) **then**
  `return` true
else
  `return` false

4.5 Reducing Conservatism with ERA

In this section, we prove that using ERA generates less conservative paths than using uniform risk allocation. Accordingly, we wish to prove the following:

1. All paths that are feasible with uniform risk allocation are also feasible with ERA.
2. Not all paths that are feasible with ERA are feasible with uniform risk allocation.

Theorem 4.2:

If a path x with T_{path} timesteps, mean $\hat{x}$, N obstacles and risk budget Δ_{path} is feasible with uniform risk allocation it is also feasible with ERA.

PROOF
With ERA, the risk allocations δ_{ik} are set such that

$$\vee_{j=1}^{n_i} \; (a_{ij}^\top \hat{x}_k - a_{ij}^\top \hat{c}_{ikj} = \beta_{ikj}), \quad {}_{\forall k \in [1:T_{path}].}^{\forall i \in [1:N],}$$

With uniform risk allocation, all risk allocations are assigned the same value $\delta_{uni} = \frac{\Delta_{path}}{N \cdot T_{path}}$. Since the path is feasible with uniform risk allocation it must fulfill:

$$\vee_{j=1}^{n_i} \; (a_{ij}^\top \hat{x}_k - a_{ij}^\top \hat{c}_{ikj} \geq \beta_{uni}), \quad {}_{\forall k \in [1:T_{path}].}^{\forall i \in [1:N],}$$

Accordingly, the following holds for all $i \in [1:N]$ and $k \in [1:T_{path}]$:

$$\beta_{uni} \leq \beta_{ikj} \implies \sqrt{\frac{1-\delta_{uni}}{\delta_{uni}}} \leq \sqrt{\frac{1-\delta_{ik}}{\delta_{ik}}} \implies \delta_{ik} \leq \delta_{uni}.$$

The sum of all exact risk allocations can be upper bounded according to

$$\sum_{k=1}^{T_{path}} \sum_{i=1}^{N} \delta_{ik} \leq \sum_{k=1}^{T_{path}} \sum_{i=1}^{N} \delta_{uni} \leq \sum_{k=1}^{T_{path}} \sum_{i=1}^{N} \frac{\Delta_{path}}{N \cdot T_{path}} \leq \Delta_{path}.$$

Which means that the path is also feasible with ERA. □

Theorem 4.3:

If a path x with T_{path} timesteps, mean $\hat{x}$, N obstacles and risk budget Δ_{path} is feasible with ERA it can still be infeasible with uniform risk allocation.

Proof

With ERA, the risk allocations δ_{ik} are set such that

$$\vee_{j=1}^{n_i} \left(a_{ij}^\top \hat{x}_k - a_{ij}^\top \hat{c}_{ikj} = \beta_{ikj}\right), \quad {}_{\forall k \in [1:T_{path}]}^{\forall i \in [1:N],}.$$

Since the path is feasible with ERA it must fulfill:

$$\sum_{k=1}^{T_{path}} \sum_{i=1}^{N} \delta_{ik} \leq \Delta_{path}.$$

For the path to be infeasible with uniform risk allocation, the following statement must hold for at least one obstacle i and timestep k:

$$\wedge_{j=1}^{n_i} \left(a_{ij}^\top \hat{x}_k - a_{ij}^\top \hat{c}_{ikj} > \beta_{uni}\right).$$

Since these statements do not contradict each other, it is possible for a path to be feasible with ERA but not with uniform risk allocation. □

4.6 Summary

In this chapter, the proposed path planning algorithm DR-RRT-ERA was described. This is an extension of DR-RRT that grows trees of state distributions while enforcing distributionally robust chance constraints and incorporating exact risk allocation (ERA). The tree is expanded by taking random samples and generating paths to them, but before the path is added to the tree it must be checked for distributionally robust feasibility. This is where ERA comes in. The ERA procedure allocates exactly as much risk that is needed to fulfill the individual chance constraints for all obstacles and timesteps. The sum of the individual risk allocations can then be used to determine the feasibility of the generated path. The ERA approach has been proven to generate less conservative paths compared to when uniform risk allocation is used. In the next chapter, the reduction in conservatism will be illustrated further using simulation examples.

5

Simulation Results

To evaluate the effectiveness of Exact Risk Allocation (ERA), we present simulation examples of Distributionally Robust RRT (DR-RRT) with ERA and uniform risk allocation. First, the principle is illustrated with a simulation of a single steering horizon where the tightened obstacles are plotted. This simulation will depict a path that is feasible with ERA but not with uniform risk allocation. Next, we illustrate how residual risk from one path can be reallocated to the next when using ERA. The path from the previous simulation is expanded by steering to a new sample in the environment and thus generating a second path. While the new path is not feasible in itself, it is made feasible by reallocating the residual risk from the previous path. Finally, simulations of entire DR-RRT-trees with uniform and exact risk allocations are shown.

5.1 Simulation Setup

In all simulations, a unit-mass robot with discrete-time stochastic double integrator dynamics is considered. The robot moves in a two-dimensional environment cluttered with obstacles. The set of obstacles is denoted by $\mathcal{B}$ with $|\mathcal{B}| = N$ and the dynamics and input matrices are

$$A = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} \frac{dt^2}{2} & 0 \\ 0 & \frac{dt^2}{2} \\ dt & 0 \\ 0 & dt \end{bmatrix} \tag{5.1.1}$$

where $dt = 0.1s$. The state of the robot is a two-dimensional position and velocity with two-dimensional force inputs. The covariance matrices of the initial state x_0 and the disturbance w are

$$\Sigma_{x_0} = 10^{-3}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}, \qquad \Sigma_w = 10^{-3}\begin{bmatrix}0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 2 & 1\\ 0 & 0 & 1 & 2\end{bmatrix}. \tag{5.1.2}$$

From Σ_{x0}, we can see that only the *position* of the initial state is subject to uncertainty and not the velocity. For Σ_w, it is the opposite as only the *velocity* of the disturbance is subject to uncertainty and not the position.

All obstacles are static and treated as deterministic, so that all uncertainty comes from the unknown state of the robot. The robot is treated as a point mass (without loss of generality as a known geometry can be easily handled by adding a fixed tightening to all obstacles) and the environmental bounds are not treated probabilistically.

The steering from a near node to a sample x_s is done by solving a discrete-time linear quadratic optimal control problem to compute the affine state feedback policy that minimizes the cost function

$$\sum_{k=0}^{T_s-1} (\hat{x}_k - x_s)^\top Q(\hat{x}_k - x_s) + u_k^\top R u_k + (x_{T_s} - x_s)^\top Q(\hat{x}_{T_s} - x_s) \tag{5.1.3}$$

$$= \sum_{k=0}^{T_s-1} \|\hat{x}_k - x_s\|_Q^{\ 2} + \|u_k\|_R^{\ 2} + \|\hat{x}_{T_s} - x_s\|_Q^{\ 2} \tag{5.1.4}$$

where $T_s = T_{steer}$, $Q = \begin{bmatrix}40I & 0\\ 0 & 40I\end{bmatrix}$ and $R = 0.1I$. The quadratic optimal cost-to-go function is also used as the distance metric in the selection of the nearest tree nodes. In all simulations, the trajectories of the mean state $\hat{x}_k$ is represented by lines and the uncertainty is represented by ellipses of one standard deviation, derived from the covariance Σ_{x_k}. Note that in the simulations of entire DR-RRT-trees, the ellipses are too small to be visible.

5.2 Simulation of a Single Steering Horizon

In this section, a simulation of a single steering horizon from a node N to a new sample x_s is shown. The steering is done in $T_{steer} = 4$ timesteps and the risk budget for the steering horizon is $\Delta_{steer} = 0.1$. The environment $[0, 1.5]^2$ contains two obstacles; Obstacle 1 (in the bottom left corner) and Obstacle 2 (in the top right corner). In this example, we imagine that the robot has previously been steered to the node N in three steering horizons, as the covariance is more unstable in the first few timesteps. Accordingly, the covariance of the node N, denoted as Σ_N, is approximately

$$\Sigma_N \approx 10^{-3} \begin{bmatrix} 0.23 & 0.12 & 0.67 & 0.34 \\ 0.12 & 0.23 & 0.34 & 0.67 \\ 0.67 & 0.34 & 6.22 & 3.11 \\ 0.34 & 0.67 & 3.11 & 6.22 \end{bmatrix} \tag{5.2.1}$$

Furthermore, we assume that the residual of node N is zero, meaning that there are no residual risk to reallocate to the generated path. The mean state and covariance of the path from N to x_s generated by the steering-function is shown in Figure 5.1. Before this path can be added to the (hypothetical) tree, it must be checked for distributionally robust feasibility. It is in this step that risk allocation comes in. Figure 5.2 shows a timestep-by-timestep comparison between exact and uniform risk allocation for the path in Figure 5.1. At each timestep, the tightened obstacles are plotted using different shades of red, where a darker shade indicates a higher risk allocation δ_{ik}. When uniform risk allocation is used, all δ_{ik} have the same value: $\delta_{ik} = \frac{\Delta_{steer}}{T_{steer} \cdot N}$, which is why all tightened obstacles at all timesteps have the same shade of red. Note that in this case, the slight variation in the size of the tightened obstacles is due to covariance and not risk allocation. When ERA is used, the value of δ_{ik} differs for the obstacles and timesteps, leading to different shades of red and varying sizes of the tightened obstacles.

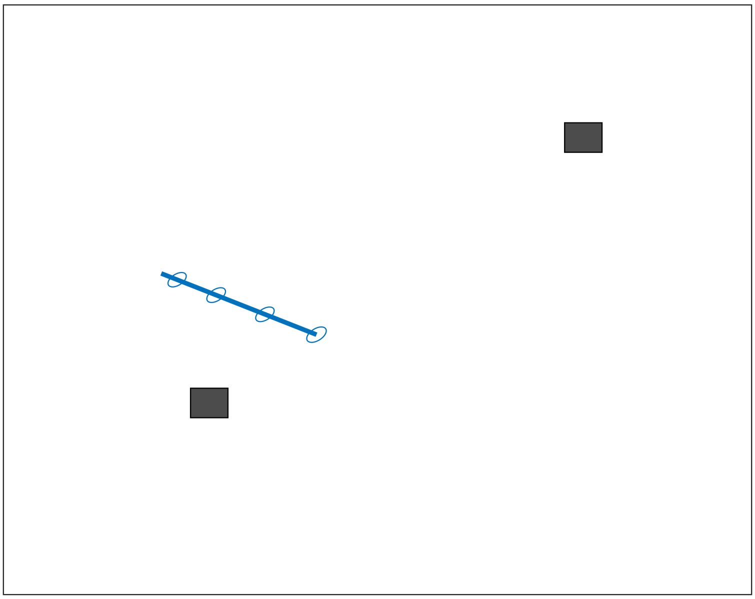

Figure 5.1 Simulation of a single steering horizon, $T_{steer} = 4$, from a node N to a sample x_s with risk budget $\Delta_{steer} = 0.1$ in a $[0, 1.5]^2$ environment.

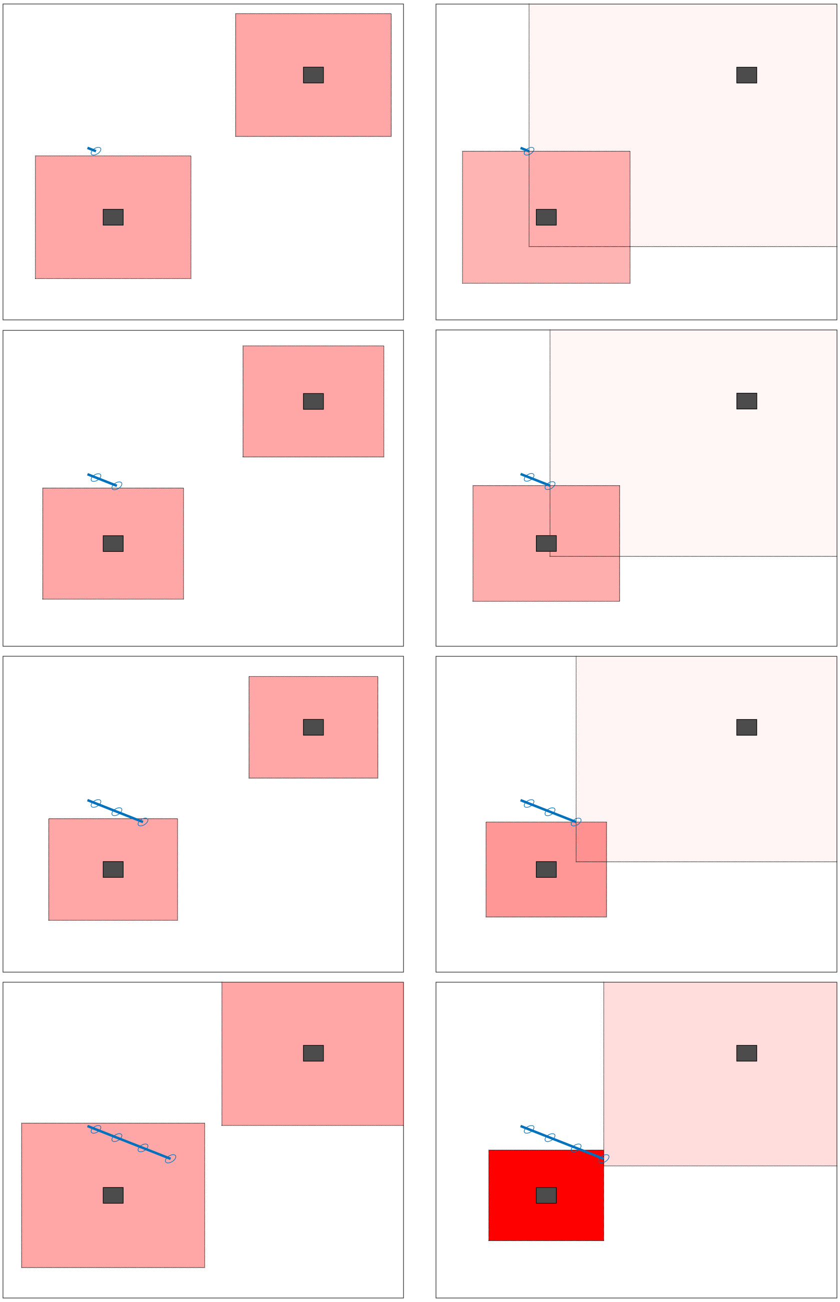

Figure 5.2 Timestep-by-timestep comparison of uniform (left) and exact (right) risk allocation for the same path with risk budget $\Delta_{steer} = 0.1$. The shades of the tightened obstacles correspond to the risk allocations δ_{ik}.

For the path to be feasible, the following conditions must be fulfilled:

Condition 1. $\sum_{k=1}^{T_{steer}} \sum_{i=1}^{N} \delta_{ik} \leq \Delta_{steer}$

Condition 2. $\mathbf{h}_{ik} = \top, \quad \forall i \in \mathcal{B}, \quad \forall k = 1 : T_{steer}$ (see 3.5.6)

When uniform risk allocation is used, the first condition is automatically fulfilled, meaning that the distributionally robust feasibility check is based on fulfillment of the second condition. For Condition 2 to be fulfilled, the mean state of the robot has to lie outside of the tightened obstacles at all timesteps. As can be seen in Figure 5.2, the mean state is outside of all tightened obstacles in the 1st and 2nd timestep but not in the 3rd and 4th timestep. This means that the entire path would *not* be considered feasible with uniform risk allocation and would therefore not be added to the tree.

When exact risk allocation is used, the risk allocations are made so that the second condition is always fulfilled. Accordingly, the feasibility check is based on fulfilling the first condition instead. The fulfillment of the first condition cannot be seen in the figure and has to be explored numerically. Table 5.1 shows the risk allocations δ_{ik} for all obstacles and timesteps in the path for both exact and uniform risk allocation as well as the sum of all risk allocations $\sum_{k=1}^{T_{steer}} \sum_{i=1}^{N} \delta_{ik}$. As can be seen in the table, the sum of all risk allocations in the steering horizon is $= 0.0816$. Since this is less than the risk budget for the steering horizon, $\Delta_{steer} = 0.1$, the first condition is fulfilled and thus, the path is feasible with exact risk allocation and would be added to the tree. If the sum had exceeded the risk budget, the path would not have been feasible even with exact risk allocation.

Table 5.1 Risk allocations for all obstacles and timesteps in the path when using exact and uniform risk allocation as well as the sum of all risk allocations over the steering horizon.

Timestep k	Exact Risk Allocation δ_{ik}		Uniform Risk Allocation δ_{ik}	
	Obstacle 1	Obstacle 2	Obstacle 1	Obstacle 2
1	0.0106	0.0013	0.0125	0.0125
2	0.0114	0.0013	0.0125	0.0125
3	0.0147	0.0015	0.0125	0.0125
4	0.0361	0.0047	0.0125	0.0125
Sum:	0.0816		0.1000	

We can conclude that the same path that was not feasible with unifrom risk allocation was feasible with ERA.

In the tree expansion algorithm (see Algorithm 1), feasible *portions* of the path is also added to the tree. As described in Section 4.4, the path-portion up to a timestep k is feasible if the sum of risk allocations up to that timestep, $\delta_{tot}(k)$ does not exceed the risk budget Δ_k. This condition can be formally stated as

$$\delta_{tot}(k) \leq \Delta_k \tag{5.2.2}$$

where

$$\delta_{tot}(k) = \sum_{k^*=1}^{k} \sum_{i=1}^{N} \delta_{ik^*} \tag{5.2.3}$$

$$\Delta_k = \frac{k \cdot \Delta_{\textbf{steer}}}{T_{\textbf{steer}}}. \tag{5.2.4}$$

Table 5.2 shows the total allocated risk $\delta_{tot}(k)$ and risk budget Δ_k for all timesteps in the path. Since the condition in (5.2.2) is fulfilled for all timesteps, all path-portions are feasible and can be added to the tree.

Table 5.2 Total allocated risk and risk budget for all timesteps in the path and the feasibility of the path-portions up to that timestep when using ERA.

Timestep k	Total risk $\delta_{tot}(k)$	Risk budget Δ_k	DR-feasible $\delta_{tot}(k) \leq \Delta_k$
1	0.0119	0.0250	$\top$
2	0.0247	0.0500	$\top$
3	0.0408	0.0750	$\top$
4	0.0816	0.1000	$\top$

5.3 Simulation of Two Steering Horizons

In this simulation, the path in Figure 5.2 is expanded by steering to a new sample x_s in $T_{steer} = 4$ timesteps. The environment and risk budget $\Delta_{steer} = 0.1$ is the same as in the previous simulation. The newly generated path has to be checked for distributionally robust feasibility in the same way.. Figure 5.3 shows the exact risk allocations for all timesteps in the new path. Just as before, the shades of the tightened obstacles indicate the risk allocation δ_{ik}.

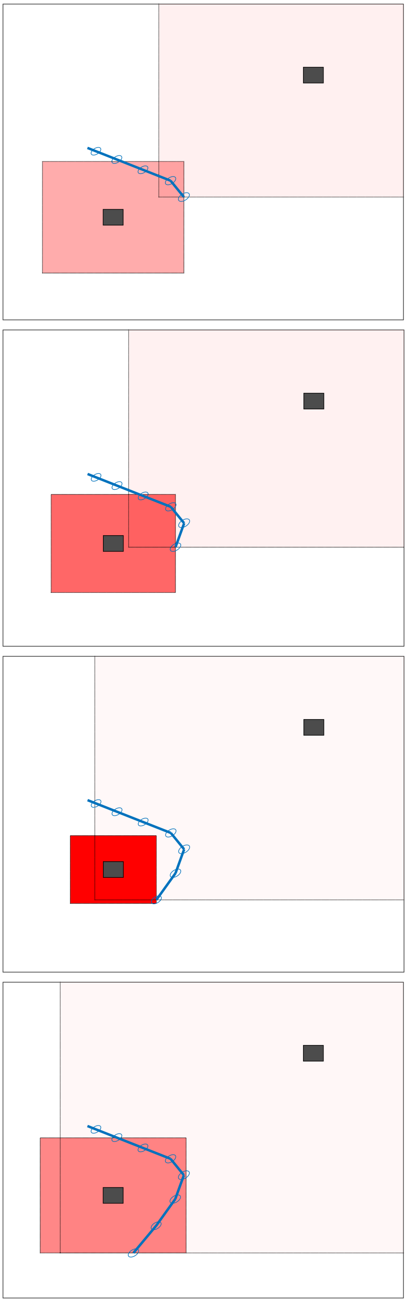

Figure 5.3 Exact risk allocations for each timestep in the new path with risk budget $\Delta_{steer} + \delta_{res} = 0.1184$. The shades of the tightened obstacles correspond to the risk allocations δ_{ik}.

Table 5.3 shows the risk allocations δ_{ik} for all obstacles and timesteps in the new path as well as the sum of all risk allocations up to a certain timestep.

Table 5.3 Exact risk allocations for all obstacles and timesteps in the new path as well as the sum of all risk allocations over the steering horizon.

Timestep k	Exact Risk Allocation δ_{ik}	
	Obstacle 1	Obstacle 2
1	0.0144	0.0026
2	0.0264	0.0024
3	0.0441	0.0012
4	0.0207	0.0014
Sum:	0.1132	

In this case, the sum is 0.1132 which exceeds $\Delta_{steer} = 0.1$. Because of this, the new path is not feasible on its own. However, as described in Section 4.4, residual risk from previous steering horizons can be reallocated to a subsequent steering. Since the entire risk budget Δ_{steer} for the previous path (see Figure 5.2, Table 5.1) was not used, we get a residual

$$\delta_{res} = \Delta_{\textbf{steer}} - \delta_{tot}(T_{\textbf{steer}}) = 0.1000 - 0.0816 = 0.0184 \tag{5.3.1}$$

which can be reallocated to the new path. The risk budget for the new path then becomes $\Delta_{steer} + \delta_{res} = 0.1184$, which means the new path has become feasible (when combined with the previous path). The feasibility of the combined path can also be motivated by summing up the risk allocations for both paths: $0.0816 + 0.1132 = 0.1948 \leq 2 \cdot \Delta_{steer}$.

The next step is to determine which *portions* of the new path are feasible. When we have a residual risk, the path-portion up to timestep k is feasible if the following condition is fulfilled:

$$\delta_{tot}(k) \leq \Delta_k + \delta_{res}. \tag{5.3.2}$$

Table 5.4 shows the total allocated risk $\delta_{tot}(k)$ and risk budget $\Delta_k + \delta_{res}$ for all timesteps in the new path. Since the condition in (5.3.2) is fulfilled for all timesteps, all path-portions are feasible and can be added to the tree.

Table 5.4 Total allocated risk and risk budget for all timesteps in the new path and the feasibility of the path-portions up to that timestep when using ERA.

Timestep k	Total risk $\delta_{tot}(k)$	Risk budget $\Delta_k + \delta_{res}$	DR-feasible $\delta_{tot}(k) \leq \Delta_k + \delta_{res}$
1	0.0170	0.0434	$\top$
2	0.0458	0.0684	$\top$
3	0.0910	0.0934	$\top$
4	0.1132	0.1184	$\top$

5.4 Simulation of DR-RRT

In these simulations, we consider an environment $[0, 50]^2$ cluttered with $N = 10$ randomly located rectangular obstacles. The initial position is $[0, 0]$ and the initial velocity is zero. The planning horizon is $T = 1000$ and the steering horizon is $T_{steer} = 10$. The risk budget for the entire planning horizon T is denoted as Δ. Three trees with 1000 samplings each and $M = 1$ are simulated:

- DR-RRT with ***Uniform Risk Allocation*** and risk budget $\Delta = 0.1$ (Figure 5.4)
- DR-RRT with ***Exact Risk Allocation*** and risk budget $\Delta = 0.1$ (Figure 5.5)
- DR-RRT with ***Exact Risk Allocation*** and risk budget $\Delta = 0.02$ (Figure 5.6)

Besides from the risk allocation and risk budget, everything in the trees and environment are exactly the same, including the random sampling points. This is to get a fair comparison of the different trees.

With uniform risk allocation, the same risk is allocated for all obstacles and timesteps, such that $\delta_{ik} = \frac{\Delta}{T \cdot N} = \frac{0.1}{1000 \cdot 10} = 10^{-5}$.

With exact risk allocation, the risk budget for a steering horizon is $\Delta_{steer} + \delta_{res}$, where δ_{res} is the residual of the node from which the steering is done and

$$\Delta_{\mathbf{steer}} = \frac{\Delta \cdot T_{\mathbf{steer}}}{T} = \frac{0.1 \cdot 10}{1000} = 10^{-3} \quad \text{(with } \Delta = 0.1\text{)} \quad \text{or} \tag{5.4.1}$$

$$\Delta_{\mathbf{steer}} = \frac{\Delta \cdot T_{\mathbf{steer}}}{T} = \frac{0.02 \cdot 10}{1000} = 2 \cdot 10^{-4} \quad \text{(with } \Delta = 0.02\text{).} \tag{5.4.2}$$

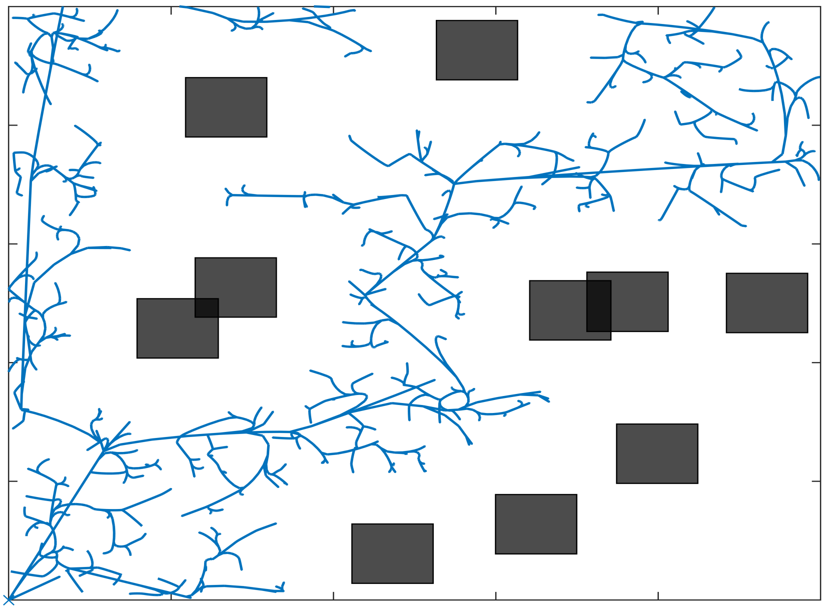

Figure 5.4 DR-RRT with ***Uniform Risk Allocation*** and risk budget $\Delta = 0.1$.

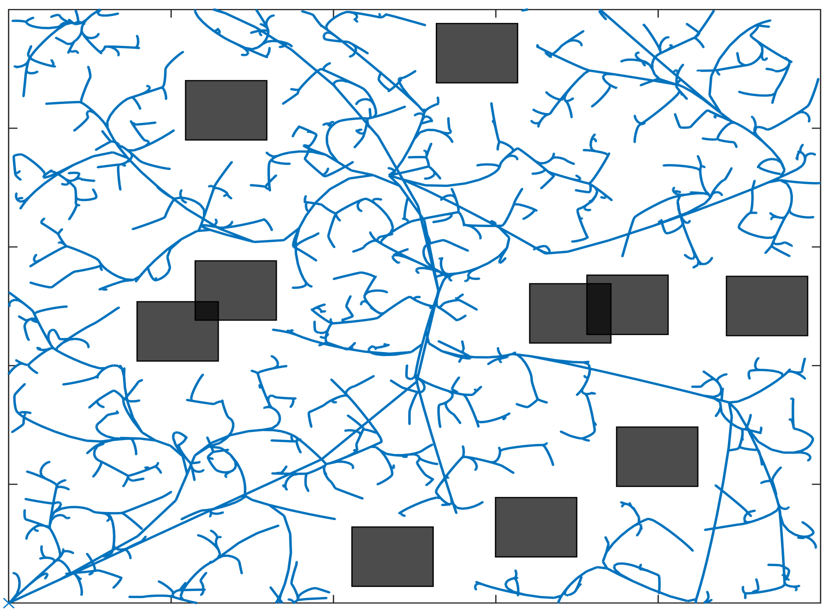

Figure 5.5 DR-RRT with ***Exact Risk Allocation*** and risk budget $\Delta = 0.1$.

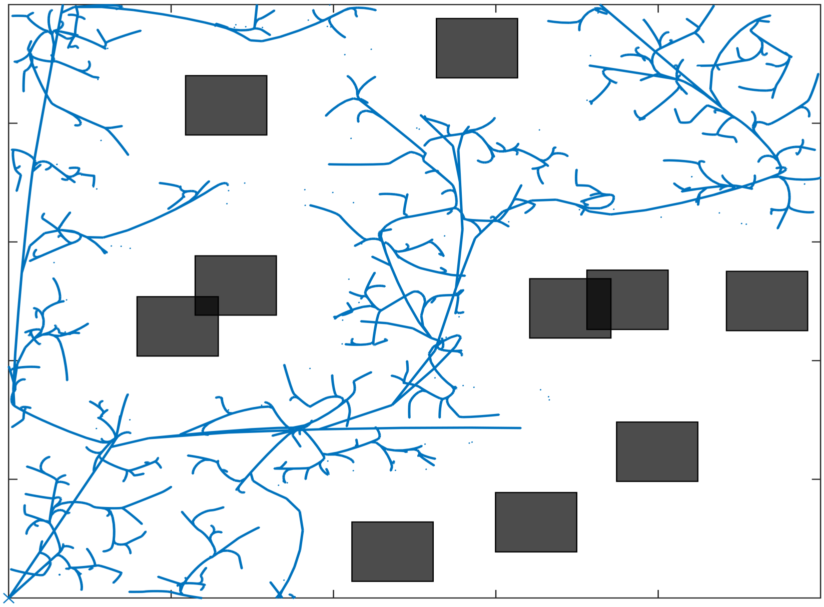

Figure 5.6 DR-RRT with ***Exact Risk Allocation*** and risk budget $\Delta = 0.02$.

5.5 Results and Discussion

From Figures 5.4 and 5.5 it can easily be seen that DR-RRT with ERA generates less conservative paths than DR-RRT with uniform risk allocation when the same risk budget $\Delta = 0.1$ is used. This coincides with the numerical proof presented in Section 4.5. Additionally, Figure 5.6 illustrates how DR-RRT with ERA can be used with lower risk budget $\Delta = 0.02$ and still generate paths with a similar degree of conservatism as DR-RRT with uniform risk allocation and a higher risk budget $\Delta = 0.1$. This means that by switching from uniform risk allocation to ERA it is possible to give stronger risk guarantees while maintaining a similar level of conservatism.

6

Conclusion & Future Outlook

This chapter will begin with a summary of the thesis and the main conclusions, followed by a discussion about limitations and future outlooks.

6.1 Summary and Conclusions

The purpose of this thesis was to design a method for spatio-temporal risk allocation that reduces the conservatism of the distributionally robust path planning algorithm DR-RRT. For risk allocation to be possible, the user-specified risk constraint for the entire path must first be decomposed into individual risk constraints for all obstacles and timesteps. The individual risk bounds can then be assigned different values with the sole requirement that their sum does not exceed the total risk budget. A common approach is to use uniform risk allocation where all obstacles and timesteps are assigned the same risk, but this approach often results in a high level of conservatism. In this thesis, a new approach called exact risk allocation (ERA) was introduced. The principle of ERA is to allocate exactly as much risk that is needed to fulfill the distributionally robust chance constraints. Accordingly, the risk allocations are only set after a path-suggestion has been generated and not in advance like with uniform risk allocation. ERA can be effectively incorporated into the DR-RRT framework to form a new motion planning algorithm called DR-RRT-ERA. Here, the risk budget for a generated path is first established by allocating the total risk budget uniformly over all steering horizons. Then, whenever a new path is generated, ERA is used to assign risk allocations that fulfills the individual chance constraints. The sum of all risk allocations is then compared to the risk budget to determine if the generated path is feasible. In cases where the entire risk budget for the generated path is not used, the residual risk can be reallocated to future paths.

The use of ERA has been proven to generate less conservative paths compared to when uniform risk allocation is used and the effectiveness in reducing the conservatism has been illustrated by simulation examples. Accordingly, we can conclude that the proposed motion planning algorithm DR-RRT-ERA has the potential to generate less conservative paths than its predecessor while still enforcing the same distributionally robust chance constraints.

6.2 Limitations and Future Outlook

Below we list the main limitations of the proposed DR-RRT-ERA algorithm and suggestions on how variations that treat these limitations can be pursued in future research. Some of the variations are straightforward to implement.

Planning horizon: In order to distribute the risk over all steering horizons, the planning horizon T has to be set in advance. The risk constraints are only specified for the planning horizon, which represents the number of timesteps we are allowed to use to get to the specified goal location. The selection of T has a strong influence on the conservatism since choosing a large T means that the risk budget must be distributed over a higher number of timesteps. At the same time, selecting a small T could make it impossible to reach the goal in the specified number of timesteps while maintaining distributionally robust constraint fulfillment. The use of ERA could enable an expansion of the steering horizon while still maintaining the same risk constraint, since any residual risk at timestep T could be reallocated to an expanded time horizon. Accordingly, ERA could enable a more flexible approach where a small planning horizon can be selected initially and expanded if necessary. This variation has the potential to further reduce the conservatism and can be explored in future research.

Probabilistic completeness: The DR-RRT-ERA algorithm is *not* probabilistically complete. This means that even if there exists a path that would be feasible with a certain risk allocation, the algorithm is not guaranteed to find that path even as the number of samples approaches infinity. A feasible (theoretical) path that requires a lot of risk in its early steering horizons and less risk in its later steering horizons may not be generated if the first steering horizons are deemed unfeasible. The algorithm could likely be made probabilistically complete by storing infeasible paths in the hope that they could later become feasible by reallocating risk from future steering horizons, but as previously mentioned, this would require a lot more computation. The algorithm is however guaranteed to find all solutions that are feasible with uniform risk allocation since the standard DR-RRT algorithm is probabilistically complete and all paths that are feasible with uniform risk allocation are also feasible with exact risk allocation, as proven in Section 4.5.

Assigning scores: The algorithm for assigning scores to paths based on their cost J and residual risk δ_{res} (see Algorithm 1) can be composed in a number of ways. It should, however, always be constructed in a way that rewards low cost and high residual. Normally, when performing tree expansion with uniform risk allocation, the minimum-cost path from a near node to the sample is selected. This would be equivalent to a scoring algorithm that only depends on the cost. In the case of ERA, the path selection becomes more complicated since residual risk is involved. Selecting a low-cost path with a low residual could prevent the generation of future low-cost paths that could have been feasible with a higher residual risk. In cases where the same path has the lowest cost and highest residual, the choice is simple. In other cases, the benefit of a low-cost path has to be weighed against the future prospect of generating paths with low cost. What should be prioritized will likely differ for different environments and risk budgets. In highly risky environments it could make sense to prioritize the residual and vice versa. Designing a universal scoring algorithm that always makes the (statistically) best choice is therefore seemingly impossible. In the simulation examples, there was no need to consider the scoring function since the simulations were done with $M = 1$, meaning that only the nearest node was selected. As previously mentioned, the exact composition of the scoring algorithm is not covered in the scope of this thesis, but could be a topic for further research.

Asymptotic optimality: Just as the standard RRT and DR-RRT algorithms, DR-RRT-ERA is *not* asymptotically optimal. This means that a feasible solution generated by the algorithm will in general not minimize the cost function even as the number of samples approaches infinity. An asymptotically optimal version of DR-RRT called DR-RRT* can be obtained by rewiring the tree to find paths with lower costs [2]. A similar approach could be taken to create a DR-RRT*-ERA algorithm. Achieving asymptotic optimally when ERA is used could however be challenging and may require storing infeasible paths in the hope that they could later become feasible by reallocating residual risk from future steering horizons.

Propagating higher moments: In this thesis, we have used moment-based ambiguity sets based on first and second moments. Propagating higher moments can sharpen the risk estimates and lead to less conservatism than only using first and second moments [1].

Environmental constraints: In this thesis, the environmental bounds are not treated probabilistically. The algorithm could however easily be adjusted to consider a problem formulation with a chance constraint that also incorporates the environmental bounds.

Steering methods: ERA can only be applied *after* a path has been generated and therefore requires that the steering can be done without prior knowledge of the risk allocations. The DR-RRT-ERA algorithm would, in its current state, not work when more sophisticated steering methods that explicitly incorporate the constraints are used. Model Predictive Control (MPC) is an example of such a steering method. In future variations it could be beneficial to use a more advanced steering that incorporates mild constraints, for instance a small deterministic tightening of the obstacles. The paths generated by this steering algorithm would then have to be checked for distributionally robust feasibility using the real constraints, for which ERA could be used.

References

[1] T. Summers, "Distributionally robust sampling-based motion planning under uncertainty," in, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 6518–6523.

[2] V. Renganathan, I. Shames, and T. H. Summers, "Towards integrated perception and motion planning with distributionally robust risk constraints," *IFAC-PapersOnLine*, **53**:no. 2, pp. 15 530–15 536.

[3] W. Wiesemann, D. Kuhn, and M. Sim, "Distributionally robust convex optimization," *Operations Research*, **62**:no. 6, pp. 1358–1376.

[4] A. Hakobyan and I. Yang, "Wasserstein distributionally robust motion planning and control with safety constraints using conditional value-at-risk," in, *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 490–496.

[5] L. Lindemann, M. Cleaveland, Y. Kantaros, and G. J. Pappas, "Robust motion planning in the presence of estimation uncertainty," *arXiv preprint arXiv:2108.11983*.

[6] A. Dixit, M. Ahmadi, and J. W. Burdick, "Distributionally robust model predictive control with total variation distance," *arXiv preprint arXiv:2203.12062*.

[7] E. Delage and Y. Ye, "Distributionally robust optimization under moment uncertainty with application to data-driven problems," *Operations research*, **58**:no. 3, pp. 595–612.

[8] L. Blackmore, H. Li, and B. Williams, "A probabilistic approach to optimal robust path planning with obstacles," in, *2006 American Control Conference*, IEEE, 2006, 7–pp.

[9] S. Safaoui, B. J. Gravell, V. Renganathan, and T. H. Summers, "Risk-averse RRT* planning with nonlinear steering and tracking controllers for nonlinear robotic systems under uncertainty," in, *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2021, pp. 3681–3688.

[10] A. Hakobyan, G. C. Kim, and I. Yang, "Risk-aware motion planning and control using CVaR-constrained optimization," *IEEE Robotics and Automation Letters*, **4**:no. 4, pp. 3924–3931.

[11] A. Zolanvari and A. Cherukuri, "Data-driven distributionally robust iterative risk-constrained model predictive control," *arXiv preprint arXiv:2111.12977*.

[12] Y. K. Nakka, A. Liu, G. Shi, A. Anandkumar, Y. Yue, and S.-J. Chung, "Chance-constrained trajectory optimization for safe exploration and learning of nonlinear systems," *IEEE Robotics and Automation Letters*, **6**:no. 2, pp. 389–396.

[13] M. Ono and B. C. Williams, "Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint," in, *2008 47th IEEE Conference on Decision and Control*, IEEE, 2008, pp. 3427–3432.

[14] M. P. Vitus and C. J. Tomlin, "On feedback design and risk allocation in chance constrained control," in, *2011 50th IEEE Conference on Decision and Control and European Control Conference*, IEEE, 2011, pp. 734–739.

[15] J. Pilipovsky and P. Tsiotras, "Covariance steering with optimal risk allocation," *IEEE Transactions on Aerospace and Electronic Systems*, **57**:no. 6, pp. 3719–3733.

[16] L. Blackmore, M. Ono, and B. C. Williams, "Chance-constrained optimal path planning with obstacles," *IEEE Transactions on Robotics*, **27**:no. 6, pp. 1080–1094.

[17] B. D. Luders, *Robust sampling-based motion planning for autonomous vehicles in uncertain environments*, Ph.D. dissertation, Massachusetts Institute of Technology, 2014.

[18] A. N. Kolmogorov and A. T. Bharucha-Reid, *Foundations of the theory of probability: Second English Edition.* Courier Dover Publications, 2018.

[19] L. Blackmore, H. Li, and B. Williams, "A probabilistic approach to optimal robust path planning with obstacles," in, *2006 American Control Conference*, IEEE, 2006, 7–pp.

[20] G. C. Calafiore and L. E. Ghaoui, "On distributionally robust chance-constrained linear programs," *Journal of Optimization Theory and Applications*, **130**:no. 1, pp. 1–22.

[21] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *Journal of guidance, control, and dynamics*, **25**:no. 1, pp. 116–129.

Lund University
Department of Automatic Control
Box 118
SE-221 00 Lund Sweden

Document name
MASTER'S THESIS

Date of issue
August 2022

Document Number
TFRT-6174

Author(s)
Kajsa Ekenberg

Supervisor
Venkatraman Renganathan, Dept. of Automatic Control, Lund University, Sweden
Anders Robertsson, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle
Distributionally Robust Risk-Bounded Path Planning Through Exact Spatio-temporal Risk Allocation

Abstract

Planning safe paths in the presence of uncertainty is considered a central challenge in enabling robots to successfully navigate in real-world environments. Assumptions about Gaussian uncertainty are rarely justifiable based on real data and can lead to serious miscalculations of risk. Lately, it has become increasingly common to consider distributionally robust uncertainty, where the exact distribution of the uncertainty is unknown. Existing motion planning algorithms that consider distributionally robust uncertainty generates more conservative paths then their Gaussian counterparts. The aim of this thesis is to mitigate this conservatism by incorporating non-uniform spatio-temporal risk allocation into existing frameworks for distributionally robust motion planning, specifically the DR-RRT algorithm. To this end, a novel motion planning algorithm called DR-RRT-ERA (DR-RRT with Exact Risk Allocation) is proposed. This is a sampling based motion planning algorithm that builds trees of state distributions while enforcing distributionally robust chance constraints. Instead of allocating the risk uniformly over time and space, the DR-RRT-ERA uses a novel concept called exact risk allocation (ERA). The principle of ERA is to allocate exactly as much risk needed to enforce the distributionally robust risk constraints. Numerical simulations illustrate that this approach leads to less conservative paths compared to when uniform risk allocation is used.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-54

Recipient's notes

Security classification