Towards Integrated Perception and Motion Planning with Distributionally Robust Risk Constraints[★]

Venkatraman Renganathan [∗] **Iman Shames** [∗∗]
Tyler H. Summers [∗]

[∗] *Department of Mechanical Engineering,*
University of Texas at Dallas, Richardson, TX 75080 USA
(e-mails: {vrengana, tyler.summers}@utdallas.edu).
[∗∗] *Department of Electrical and Electronic Engineering and Melbourne*
Information, Decision and Autonomous Systems (MIDAS) Laboratory,
University of Melbourne, Parkville, VIC 3010, Australia
(e-mail: ishames@unimelb.edu.au)

Abstract: Safely deploying robots in uncertain and dynamic environments requires a systematic accounting of various risks, both within and across layers in an autonomy stack from perception to motion planning and control. Many widely used motion planning algorithms do not adequately incorporate inherent perception and prediction uncertainties, often ignoring them altogether or making questionable assumptions of Gaussianity. We propose a distributionally robust incremental sampling-based motion planning framework that explicitly and coherently incorporates perception and prediction uncertainties. We design output feedback policies and consider moment-based ambiguity sets of distributions to enforce probabilistic collision avoidance constraints under the worst-case distribution in the ambiguity set. Our solution approach, called Output Feedback Distributionally Robust `RRT`* (`OFDR-RRT`*), produces asymptotically optimal risk-bounded trajectories for robots operating in dynamic, cluttered, and uncertain environments, explicitly incorporating mapping and localization error, stochastic process disturbances, unpredictable obstacle motion, and uncertain obstacle locations. Numerical experiments illustrate the effectiveness of the proposed algorithm.

Keywords: Risk-Bounded Motion Planning, Distributional Robustness, Integrated Perception & Planning in Robotics.

1. INTRODUCTION

More sophisticated motion planning and control algorithms are needed for the robots to operate in increasingly dynamic and uncertain environments to ensure safe and effective autonomous behavior. Many widely used motion planning algorithms have been developed in deterministic settings. However, since motion planning algorithms must be coupled with the outputs of inherently uncertain perception systems, there is a crucial need for more tightly coupled perception and planning frameworks that explicitly incorporate perception uncertainties.

Motion planning under uncertainty has been considered in several lines of recent research Blackmore et al. (2006); Agha-Mohammadi et al. (2014); Luders et al. (2010, 2013); Blackmore et al. (2011); Liu and Ang (2014); Zhu and Alonso-Mora (2019). Many approaches make questionable assumptions of Gaussianity and utilize chance constraints,

ostensibly to maintain computational tractability. However, this can cause significant miscalculations of risk, and the underlying risk metrics do not necessarily possess desirable coherence properties Rockafellar (2007); Majumdar and Pavone (2020). The emerging area of distributionally robust optimization (DRO) shows that stochastic uncertainty can be handled in much more sophisticated ways without sacrificing computational tractability Goh and Sim (2010); Wiesemann et al. (2014). These approaches allow modelers to explicitly incorporate inherent ambiguity in probability distributions, rather than making overly strong structural assumptions on the distribution.

Traditionally, the perception and planning components in a robot autonomy stack are loosely coupled, in the sense that nominal estimates from the perception system may be used for planning, while inherent perception uncertainties are usually ignored. This paradigm is inherited in part from the classical separation of estimation and control in linear systems theory. However, in the presence of uncertainties and constraints, estimation and control should *not* be separated; there are needs and opportunities to explicitly incorporate perception uncertainties into planning, both to mitigate risks of constraint violation Blackmore et al. (2006); Florence et al. (2016); Luders et al. (2010);

Summers (2018); Zhu and Alonso-Mora (2019) and to actively plan paths that improve perception Costante et al. (2016).

Contributions: In this paper, we take steps toward a tighter integration of perception and planning in autonomous robotic systems. Our main contributions are:

- We propose a distributionally robust incremental sampling-based motion planning framework that explicitly and coherently incorporates perception and prediction uncertainties. Our solution approach, called Output Feedback Distributionally Robust RRT* (OFDR-RRT*) (Algorithm 1), produces asymptotically optimal risk-bounded trajectories for robots operating in dynamic, cluttered, and uncertain environments, explicitly incorporating mapping and localization error, stochastic process disturbances, unpredictable obstacle motion, and uncertain obstacle locations. We design output feedback policies and consider moment-based ambiguity sets of distributions to enforce probabilistic collision avoidance constraints under the worst-case distribution in the ambiguity set (Algorithm 2).

- We demonstrate via numerical simulation results that it gives a more sophisticated and coherent risk quantification compared to an approach that accounts for uncertainty using Gaussian assumption, without increasing the computation complexity.

The rest of the paper is organized as follows. The dynamical model of the robot and the uncertainty modeling in the motion planning problem is discussed in section 2. Then, the proposed OFDR-RRT* algorithm for motion planning is explained in section 3. The simulation results using a double integrator model are then presented in section 4. The paper is finally closed in section 5 with a summary of results and with directions for the future research.

2. ROBOT & ENVIRONMENT MODELING

Consider a robot operating in an uncertain environment, $\mathcal{X} \subset \mathbb{R}^n$ cluttered with n_0 obstacles. We denote the set of obstacles as $\mathcal{B} = \{1, \ldots, n_0\}$. The robot and the obstacles are modeled as a stochastic discrete-time linear system

$$x_{t+1} = A x_t + B u_t + G w_{r,t}, \qquad (1)$$
$$\mathcal{O}_{it} = \mathcal{O}_i^0 \oplus c_{it}, \quad i \in \mathcal{B}, \qquad (2)$$
$$\mathcal{X}_{it} = \Phi(\mathcal{O}_{it}), \quad i \in \mathcal{B}, \qquad (3)$$

where $x_t \in \mathbb{R}^n$ is the robot state at time t, $u_t \in \mathbb{R}^m$ is the input at time t, and A and B are the system dynamics matrix and input matrix, respectively. The process noise $w_{r,t} \in \mathbb{R}^n$ is a zero-mean random vector independent and identically distributed across time. The initial condition x_0 is subject to an uncertainty model, with the distribution of x_0 belonging to an ambiguity set, $P_{x_0} \in \mathcal{P}^x$. Moreover, $\mathcal{O}_i^0 \subset \mathbb{R}^n$ represents the shape of obstacle i, $c_{it} \in \mathbb{R}^n$ is a random vector that represents an uncertain obstacle location and motion, not necessarily zero-mean, with unknown distribution $P_{it}^c \in \mathcal{P}_{it}^c$, and $\oplus$ denotes set translation. The obstacles $\mathcal{O}_{it}, i \in \mathcal{B}$ are assumed to be convex polytopes. The state $\mathcal{X}_{it}$ of obstacle i is defined by a set function $\Phi : 2^{\mathbb{R}^n} \to \mathbb{R}^l$ that maps the obstacle set $\mathcal{O}_{it}$ to a finite vector describing the location, motion, and shape of each obstacle relative to the uncertain trajectory c_{it}.

2.1 Integrated Perception & Motion Planning

We concatenate both the robot's state and the obstacle states at time t to form the environmental state

$$\mathcal{Z}_t = \begin{bmatrix} x_t \\ \mathcal{X}_{\mathcal{O}_t} \end{bmatrix} \in \mathbb{R}^{n+ln_0}, \qquad (4)$$

where $\mathcal{X}_{\mathcal{O}_t} = [\mathcal{X}_{1t} \ \mathcal{X}_{2t} \ \ldots \ \mathcal{X}_{n_0 t}]^\top$ represents the concatenated states of all the n_0 obstacles at time t. Then the dynamics of the environmental state can be written as

$$\mathcal{Z}_{t+1} = \underbrace{\begin{bmatrix} A & \mathbf{0}_{n \times ln_0} \\ \mathbf{0}_{ln_0 \times n} & I_{ln_0} \end{bmatrix}}_{A_z} \mathcal{Z}_t + \underbrace{\begin{bmatrix} B \\ \mathbf{0}_{ln_0 \times m} \end{bmatrix}}_{B_z} u_t + G_z \underbrace{\begin{bmatrix} w_{r,t} \\ w_{\mathcal{O},t} \end{bmatrix}}_{w_t}, \qquad (5)$$

where $G_z = \mathrm{diag}(G, I)$ and $w_{\mathcal{O},t} \in \mathbb{R}^n$ is an obstacle process noise and can be derived from c_{it}. The distribution P_{w_t} of w_t is unknown and will be assumed to belong to an ambiguity set $\mathcal{P}^w$ of distributions satisfying

$$\mathcal{P}^w = \{P_{w_t} | \mathbb{E}[w_t] = 0, \mathbb{E}\left[(w_t - \hat{w}_t)(w_t - \hat{w}_t)^\top\right] = \Sigma_{w_t}\}. \qquad (6)$$

At time t, the state of the robot can be extracted from the environmental state $\mathcal{Z}_t$ as

$$x_t = \underbrace{[\mathbf{1}_n \ \mathbf{0}_{ln_0}]}_{C_{xr}} \mathcal{Z}_t. \qquad (7)$$

In an autonomous robot, the environmental state $\mathcal{Z}_t$ must be estimated with a perception system from noisy onboard sensor measurements. We assume that a high-level perception system, such as Semantic SLAM described in Sünderhauf et al. (2017), processes high dimensional raw data $\Theta_t \in \mathbb{R}^N$ to recognize obstacles and produce noisy joint measurements of their state and the robot state. In particular, we define feature vectors $\mathcal{Y}_{xt} \in \mathbb{R}^r$ and $\mathcal{Y}_{it} \in \mathbb{R}^q$ for $i \in \mathcal{B}$ obtained through

$$\mathcal{Y}_{xt} = \Upsilon_x(\Theta_t) \qquad (8)$$
$$\mathcal{Y}_{\mathcal{O}t} = \Upsilon_{\mathcal{O}}(\Theta_t) = [\mathcal{Y}_{1t} \ \mathcal{Y}_{2t} \ \ldots \ \mathcal{Y}_{n_0 t}]^\top, \qquad (9)$$

where $\Upsilon_x : \mathbb{R}^N \to \mathbb{R}^r$ and $\Upsilon_{\mathcal{O}} : \mathbb{R}^N \to \mathbb{R}^{qn_0}$ are mappings defined by the SLAM algorithm to process the raw sensor data. We then represent these features as noisy measurements of the robot and obstacle states using an assumed linear (or linearized) output model

$$y_t = \begin{bmatrix} y_{x,t} \\ y_{\mathcal{O},t} \end{bmatrix} = \mathcal{C}\left(\begin{bmatrix} \mathcal{Y}_{xt} \\ \mathcal{Y}_{\mathcal{O}t} \end{bmatrix}\right) = C \mathcal{Z}_t + H v_t, \qquad (10)$$

$$C = \begin{bmatrix} C_r \\ C_{\mathcal{O}} \end{bmatrix}, H = \begin{bmatrix} H_r \\ H_{\mathcal{O}} \end{bmatrix}, v_t = \begin{bmatrix} v_{r,t} \\ v_{\mathcal{O},t} \end{bmatrix} \sim P_{v_t} \in \mathcal{P}^v, \qquad (11)$$

where $y_t \in \mathbb{R}^{p+s}$, and $y_{x,t} \in \mathbb{R}^p, y_{\mathcal{O},t} \in \mathbb{R}^s$ are the output vectors corresponding to the robot and the obstacles respectively. The matrices $C, C_{\mathcal{O}}, H_x, H_{\mathcal{O}}$ are of appropriate dimensions. The function $\mathcal{C}$ maps the feature vectors $\mathcal{Y}_{xt}, \mathcal{Y}_{\mathcal{O}t}$ to produce outputs $y_{xt}, y_{\mathcal{O}t}$ respectively as a linear function of the environmental state $\mathcal{Z}_t$ with additive measurement noises v_t which is a zero-mean random variable. For simplicity, we assume that w_t and v_t are independent. The distribution P_{v_t} of v_t is assumed to belong to an ambiguity set, $\mathcal{P}^v$ satisfying

$$\mathcal{P}^v = \{P_{v_t} | \mathbb{E}[v_t] = 0, \mathbb{E}\left[(v_t - \hat{v}_t)(v_t - \hat{v}_t)^\top\right] = \Sigma_{v_t}\}. \qquad (12)$$

The robot is nominally subject to constraints on the state and input of the form, $\forall t = 0, \ldots, T - 1$,

$$x_t \in \mathcal{X}_t^{\texttt{free}} = \mathcal{X} \backslash \bigcup_{i \in \mathcal{B}} \mathcal{O}_{it}, \qquad (13)$$

$$u_t \in \mathcal{U}, \qquad (14)$$

where the environment $\mathcal{X} \subset \mathbb{R}^n$, and $\mathcal{U} \subset \mathbb{R}^m$ are assumed to be convex polytopes, The obstacles $\mathcal{O}_{it}, i \in \mathcal{B}$ are described by (2), and the operator $\backslash$ denotes set subtraction. The set $\mathcal{B}$ represent a set of n_0 obstacles in the environment to be avoided.

$$\mathcal{U} = \{u_t \mid A_u u_t \leq b_u\},$$
$$\mathcal{X} = \{\mathcal{Z}_t \mid A_0 C_{xr} \mathcal{Z}_t \leq b_0\}, \qquad (15)$$
$$\mathcal{O}_{it} = \{\mathcal{Z}_t \mid A_i C_{xr} \mathcal{Z}_t \leq b_{it}\}, \quad i \in \mathcal{B}$$

where $b_u \in \mathbb{R}^{n_u}, b_0 \in \mathbb{R}^{n_E}, b_{it} \in \mathbb{R}^{n_i}$, and A_u, A_0, and A_i are matrices of appropriate dimension. The nonconvex obstacle avoidance constraints for obstacle $i \in \mathcal{B}$ can be expressed as the disjunction

$$\neg (A_i C_{xr} \mathcal{Z}_t \leq b_{it}) \Leftrightarrow \bigvee_{j=1}^{n_i} (a_{ij}^\top C_{xr} \mathcal{Z}_t \geq a_{ij}^\top c_{it}), \qquad (16)$$

where $\vee$ denotes disjunction.

2.2 A Distributionally Robust Motion Planning Problem

We seek a dynamic output feedback control policy $\pi = [\pi_0, \ldots, \pi_{T-1}]$ with $u_t = \pi_t(y_{0:t}, u_{0:t-1})$, where $y_{0:t}$ and $u_{0:t-1}$ are the output and input histories available to make control decisions at time t, that produces a feasible and minimum cost trajectory from an initial state x_0 to a goal set $\mathcal{X}_{goal} \subset \mathbb{R}^n$. In particular, we seek to (approximately) solve the distributionally robust constrained stochastic optimal control problem

$$\begin{aligned}
\underset{\pi}{\text{minimize}} \quad & \sum_{t=0}^{T-1} \ell_t(\mathbb{E}[\mathcal{Z}_t], \mathcal{X}_{goal}, u_t) + \ell_T(\mathbb{E}[\mathcal{Z}_T], \mathcal{X}_{goal}) \\
\text{subject to} \quad & \mathcal{Z}_{t+1} = A_z \mathcal{Z}_t + B_z u_t + G_z w_t, \\
& y_t = C \mathcal{Z}_t + H v_t \\
& \mathcal{Z}_0 \sim P_{\mathcal{Z}_0} \in \mathcal{P}^{\mathcal{Z}}, \\
& w_t \sim P_w \in \mathcal{P}^w, \\
& u_t \in \mathcal{U} = \{u_t \mid A_u u_t \leq b_u\}, \\
& \mathcal{X}_t^{\texttt{free}} = \mathcal{X} \backslash \bigcup_{i \in \mathcal{B}} \mathcal{O}_{it}, \\
& \inf_{P_{\mathcal{Z}_t} \in \mathcal{P}^{\mathcal{Z}}} P_{\mathcal{Z}_t}(C_{xr} \mathcal{Z}_t \in \mathcal{X}_t^{\texttt{free}}) \geq 1 - \alpha,
\end{aligned}$$
$$(17)$$

where $\mathcal{P}^{\mathcal{Z}}$ is an ambiguity set of marginal state distributions and $\alpha \in (0, 0.5]$ is a user-prescribed risk parameter. The stage cost functions $\ell_t(\cdot)$ quantify the robot's distance to the goal set and actuator effort, and are assumed to be expressed in terms of the environmental state mean $\mathbb{E}[\mathcal{Z}_t]$, so that all the stochasticity appears in the constraints. Two key features distinguish our problem formulation. First, the state constraints are expressed as *distributionally robust chance constraints*. This means that the nominal constraints $x_t \in \mathcal{X}_t^{\texttt{free}}$ are enforced with probability α under the worst-case distribution in the ambiguity set. Second, since information about the environmental state is obtained only from noisy measurements, we optimize over dynamic output feedback policies. Our proposed solution framework, detailed in the next section, combines a dynamic state estimator with a full-state kinodynamic

motion planning under uncertainty algorithm. This combination and explicit incorporation of state estimation uncertainty into the motion planning and control takes a step toward tighter integration of perception, planning, and control, which are nearly always separated in state-of-the-art robotic systems.

3. OUTPUT FEEDBACK DISTRIBUTIONALLY ROBUST RRT* (OFDR-RRT*)

We propose to use a distributionally robust, kinodynamic variant of the RRT* motion planning algorithm with dynamic output feedback policies. RRT* adds a rewiring operation to RRT to obtain asymptotic optimality. Our proposed algorithm grows trees of state and state estimate distributions, rather than merely trees of states, and incorporates distributionally robust probabilistic constraints to build risk-constrained state trajectories and feedback policies.

3.1 LQG Control Based Steering Law

Sampling based motion planning algorithms require a steering law to steer the robot from a node in the tree to a feasible sampled point in the free space. Since the environment state is not directly observed and must be estimated from noisy output measurements (10), our proposed steering law $\pi = [\pi_0, \ldots, \pi_{T-1}]$ with $u_t = \pi_t(y_{0:t}, u_{0:t-1})$ comprises a combination of dynamic state estimator and state feedback control law. Here we utilize a Kalman filter (which has been used in seminal SLAM algorithms for joint estimation of robot and environmental states Dissanayake et al. (2001)) for state estimation together with a finite horizon optimal linear quadratic state feedback controller. It is also possible within our framework to more sophisticated estimation and control components (e.g., extended/unscented Kalman filters or particle filters and stochastic model predictive controllers), which will be explored in future work.

The output feedback control policy has the form

$$u_t = K_t \tilde{\mathcal{Z}}_t + k_t. \qquad (18)$$

where $\tilde{\mathcal{Z}}_t$ is the Kalman filter estimate of the environmental state, and K_t and k_t are linear and constant feedback gains to be derived with dynamic programming for a finite-horizon LQR problem.

Kalman Filter: The Kalman filter equations with gain $\mathcal{L}_t$ are

$$\mathcal{L}_{t+1} = \Sigma_{\tilde{\mathcal{Z}}_t} C^\top \left(C \Sigma_{\tilde{\mathcal{Z}}_t} C^\top + H \Sigma_{v_t} H^\top \right)^{-1}, \qquad (19)$$

$$\tilde{\mathcal{Z}}_{t+1} = (I - \mathcal{L}_{t+1} C)(A_z \tilde{\mathcal{Z}}_t + B_z u_t) + \mathcal{L}_{t+1} y_{t+1}, \qquad (20)$$

$$\Sigma_{\tilde{\mathcal{Z}}_{t+1}} = (I - \mathcal{L}_{t+1} C)(A_z \Sigma_{\tilde{\mathcal{Z}}_t} A_z^\top + G_z \Sigma_{w_t} G_z^\top). \qquad (21)$$

Together with the control law (18) we can write the combined dynamics for the true unknown state $\mathcal{Z}_t$ and the state estimate $\tilde{\mathcal{Z}}_t$ as

$$\underbrace{\begin{bmatrix} \mathcal{Z}_{t+1} \\ \tilde{\mathcal{Z}}_{t+1} \end{bmatrix}}_{Z_{t+1}} = \underbrace{\begin{bmatrix} A_z & B_z K_t \\ \mathcal{L}_{t+1} C A_z & (I - \mathcal{L}_{t+1} C) A_z + B_z K_t \end{bmatrix}}_{\bar{A}_t} \underbrace{\begin{bmatrix} \mathcal{Z}_t \\ \tilde{\mathcal{Z}}_t \end{bmatrix}}_{Z_t}$$

(22)

$$+ \underbrace{\begin{bmatrix} B_z \\ B_z \end{bmatrix}}_{\bar{B}_t} [k_t] + \underbrace{\begin{bmatrix} G_z & 0 \\ \mathcal{L}_{t+1} C G_z & \mathcal{L}_{t+1} H \end{bmatrix}}_{\bar{G}_t} \underbrace{\begin{bmatrix} w_t \\ v_{t+1} \end{bmatrix}}_{W_t}, \quad (23)$$

Let us define $\bar{Z}_t = Z_t - \hat{Z}_t$, $\bar{W}_t = W_t - \hat{W}_t$. Then,

$$\mathbb{E}\left[\bar{Z}_t \bar{Z}_t^\top\right] = \Pi_t = \tilde{A}_{t-1} \Psi_{t-1} \tilde{A}_{t-1}^\top \qquad (24)$$

$$\mathbb{E}\left[\bar{W}_t \bar{W}_t^\top\right] = \Sigma_{W_t} = \begin{bmatrix} \Sigma_{w_t} & 0 \\ 0 & \Sigma_{v_{t+1}} \end{bmatrix} \qquad (25)$$

where $\tilde{A}_t = \begin{bmatrix} \bar{A}_t & \bar{G}_t W_t \end{bmatrix}$, $\Psi_t = \begin{bmatrix} \bar{Z}_t \bar{Z}_t^\top & \bar{Z}_t \\ \bar{Z}_t^\top & I \end{bmatrix}$. With the initial estimates $\hat{Z}_0 = \begin{bmatrix} \hat{Z}_0 \\ \hat{Z}_0 \end{bmatrix}$, $\Pi_0 = \begin{bmatrix} \Sigma_{Z_0} & 0 \\ 0 & 0 \end{bmatrix}$, $\Sigma_{W_0} = \begin{bmatrix} \Sigma_{w_0} & 0 \\ 0 & \Sigma_{v_0} \end{bmatrix}$ given (or estimated from historical data), the combined state and state estimate mean and covariance evolve as

$$\hat{Z}_{t+1} = \bar{A}_t \hat{Z}_t + \bar{B}_t k_t, \qquad (26)$$

$$\Pi_{t+1} = \bar{A}_t \Pi_t \bar{A}_t^\top + \bar{G}_t \Sigma_{W_t} \bar{G}_t^\top. \qquad (27)$$

For analysis purposes, the unknown environmental state mean and covariance can then be extracted as

$$\mathcal{Z}_{t+1} = \begin{bmatrix} \mathbf{1}^\top & \mathbf{0}^\top \end{bmatrix} \hat{Z}_{t+1}, \qquad (28)$$

$$\Sigma_{\mathcal{Z}_{t+1}} = \begin{bmatrix} \mathbf{I}_n \\ \mathbf{0}_n \end{bmatrix}^\top \Pi_{t+1} \begin{bmatrix} \mathbf{I}_n \\ \mathbf{0}_n \end{bmatrix}. \qquad (29)$$

Optimal Finite-Horizon LQR Control: Define the error $e_t = C_{xr} \tilde{\mathcal{Z}}_t - x_s$, where x_s represents a sample of the free space to be steered to. Then the optimal control gains in (18) can be obtained by minimizing the cost function

$$J = \mathbb{E}\left[\sum_{t=0}^{T-1} \left(e_t^\top Q e_t + u_t^\top R u_t \right) + e_T^\top Q e_T \right], \qquad (30)$$

via dynamic programming with the following backward in time recursion from $t = T, \dots, 1$

$$K_t = -\beta^{-1} B_z^\top P_{t+1} A_z, \qquad (31)$$

$$k_t = -\beta^{-1} B_z^\top q_{t+1}, \qquad (32)$$

$$P_t = Q + A_z^\top P_{t+1} \left[I - B_z \beta^{-1} B_z^\top P_{t+1} \right] A_z, \qquad (33)$$

$$q_t = (A_z + B_z K_t)^\top q_{t+1} + (K_t \beta + A_z^\top P_{t+1} B_z) k_t - Q x_s, \qquad (34)$$

with initial values $P_T = Q$, $q_T = -Q x_s$ and further $\beta = (R + B_z^\top P_{t+1} B_z)$ and Q, R are the state and control cost matrices, respectively.

3.2 Moment-Based Ambiguity Set To Model Uncertainty

Unlike most stochastic motion planning algorithms that often assume a functional form (often Gaussian) for probability distributions to model uncertainties, we will focus here on uncertainty modeling using moment-based ambiguity sets. Based on the ambiguity sets for the primitive random variables (namely, the process noise w in (6), the measurement noise v in (12), and the analogous one for the initial state and state estimate Z_0), and based on the estimator and control law, the combined environmental state and state estimate and covariance propagate according to (26) and (27). Since the primitive distributions are not

assumed to be Gaussian, then neither are the marginal state and state estimate distributions distributions P_{Z_t}. The ambiguity set defining the combined environmental state and state estimate is

$$\mathcal{P}^Z = \left\{ P_{Z_t} \mid \mathbb{E}[Z_t] = \hat{Z}_t, \mathbb{E}[(Z_t - \hat{Z}_t)(Z_t - \hat{Z}_t)^\top] = \Sigma_{Z_t} \right\}, \qquad (35)$$

and the ambiguity set for the true environmental state is

$$\mathcal{P}^{\mathcal{Z}} = \left\{ P_{\mathcal{Z}_t} \mid \mathbb{E}[\mathcal{Z}_t] = \hat{\mathcal{Z}}_t, \mathbb{E}[(\mathcal{Z}_t - \hat{\mathcal{Z}}_t)(\mathcal{Z}_t - \hat{\mathcal{Z}}_t)^\top] = \Sigma_{\mathcal{Z}_t} \right\}. \qquad (36)$$

3.3 Distributionally Robust Collision Check

The control law returned by the steering function should also satisfy the state constraints which are expressed as distributionally robust chance constraints. In particular, the nominal state constraints, $C_{xr} \mathcal{Z}_t \in \mathcal{X}_t^{\texttt{free}}$, are required to be satisfied with probability $1 - \alpha$, under the worst case probability distribution in the ambiguity set. Let the moment-based ambiguity set for obstacle motion be defined using $\mathbf{E}[c_{it}] = \hat{c}_{it}$ and $\mathbf{E}[(c_{it} - \hat{c}_{it})(c_{it} - \hat{c}_{it})^\top] = \Sigma_{it}^c$. Now, under the moment-based ambiguity set defined by (36), a constraint on the worst-case probability of violating the jth constraint of obstacle $i \in \mathcal{B}$

$$\sup_{P_{\mathcal{Z}_t} \in \mathcal{P}^{\mathcal{Z}}} P_{\mathcal{Z}_t}(a_{ij}^\top C_{xr} \mathcal{Z}_t \geq a_{ij}^\top c_{it}) \leq \alpha_i \qquad (37)$$

is equivalent to the linear constraint on the state mean $\hat{\mathcal{Z}}_t$

$$a_{ij}^\top C_{xr} \hat{\mathcal{Z}}_t \geq a_{ij}^\top \hat{c}_{it} + \sqrt{\frac{1 - \alpha_i}{\alpha_i}} \left\| (D_{\hat{x}_t} + \Sigma_{it}^c)^{\frac{1}{2}} a_{ij} \right\|_2, \quad (38)$$

where $D_{\hat{x}_t} = C_{xr} \Sigma_{\mathcal{Z}_t} C_{xr}^\top$ and α_i is the user prescribed risk parameter for obstacle $i \in \mathcal{B}$. Obstacle risks are allocated such that their sum does not exceed the constraint risk α. The scaling constant $\sqrt{\frac{1 - \alpha_i}{\alpha_i}}$ in the deterministic tightening of the nominal constraint is larger than the Gaussian one, leading to a stronger tightening that reflects the weaker assumptions about the uncertainty distributions.

3.4 Sample-Based Motion Planning Algorithm

The OFDR-RRT* tree expansion procedure, similar to the CC-RRT* algorithm developed in Luders et al. (2013), is presented in Algorithm 1. The OFDR-RRT* tree is denoted by $\mathcal{T}$, consisting of $|\mathcal{T}|$ nodes. Each node N of the tree $\mathcal{T}$ consists of a sequence of state distributions, characterized by a distribution mean $\hat{x}$ and covariance D. A sequence of means and covariances is denoted by $\bar{\sigma}$ and $\bar{\Pi}$, respectively. The final mean and covariance of a node's sequence are denoted by $x[N]$ and $D[N]$, respectively. For the state distribution sequence $(\bar{\sigma}, \bar{\Pi})$, the notation $\Delta J(\bar{\sigma}, \bar{\Pi})$ denotes the cost of that sequence. If $(\bar{\sigma}, \bar{\Pi})$ denotes the trajectory of node N with parent N_{parent}, then we denote by $J[N]$, the entire path cost from the starting state to the terminal state of node N, constructed recursively as

$$J[N] = J[N_{parent}] + \Delta J(\bar{\sigma}, \bar{\Pi}). \qquad (39)$$

In the first step, a random sample x_{rand} is taken from the feasible state set, $\mathcal{X}_t^{\texttt{free}}$. Then the tree node, $N_{nearest}$ that is nearest to the sample is selected (line 3 of Algorithm 1) according to an optimal cost-to-go function without the obstacle constraints, in order to efficiently explore the reachable set of the dynamics and increase the likelihood

of generating collision-free trajectories (similar to what is done in Frazzoli et al. (2002)). Attempts are then made to steer the robot from the nearest tree node to the random sample using the steering law explained in subsection 3.1 (line 4). The control policy obtained is then used to propagate the state mean and covariance, and the entire trajectory $(\bar{\sigma}, \bar{\Pi})$ is returned by the steer function. We assume that the terminal cost is selected such that the steering law gives negligible terminal residual error after solving (30); alternatively, it is straightforward to include an equality constraint on the terminal state mean. Each state distribution in the trajectory is then checked for distributionally robust probabilistic constraint satisfaction given by (38) and further the line connecting subsequent state distributions in the trajectory are also checked for collision with the obstacle sets $\mathcal{O}_{it}, i \in \mathcal{B}$. An outline of the DR-Feasible subroutine is shown in Algorithm 2. If the entire trajectory $(\bar{\sigma}, \bar{\Pi})$ is probabilistically feasible, a new node N_{min} with that distribution sequence $(\bar{\sigma}, \bar{\Pi})$ is created (line 7) but not yet added to $\mathcal{T}$. Instead, nearby nodes are identified for possible connections via the NearNodes function (line 8), which returns a subset of nodes $\mathcal{N}_{near} \subseteq \mathcal{T}$, if they are within a search radius ensuring probabilistic asymptotic optimality guarantees specified in Luders et al. (2013)

$$r = \min \left\{ \gamma \left(\frac{\log(|\mathcal{N}_t|)}{\mathcal{N}_t} \right)^{1/d}, \mu_{max} \right\}, \qquad (40)$$

where $\mathcal{N}_t$ refers to the number of nodes in the tree at time t, $\mu_{max} > 0$ is the maximum radius specified by the user, γ refers to the planning constant based on the d dimensional environment. Then we seek to identify the lowest-cost, probabilistically feasible connection from the $\mathcal{N}_{near}$ nodes to x_{rand} (lines 10-14). For each possible connection, a distribution sequence is simulated via the steering law (line 11). If the resulting sequence is probabilistically feasible, and the cost of that node represented as $c_{rand} = J[N_{near}] + \Delta J(\bar{\sigma}, \bar{\Pi})$, is lower than the cost of N_{min} denote by $J[N_{min}]$, then a new node with this sequence replaces N_{min} (line 14). The lowest-cost node is ultimately added to $\mathcal{T}$ (line 15).

Finally, edges are rewired based on attempted connections from the new node N_{min} to nearby nodes $\mathcal{N}_{near}$ (lines 17-22), ancestors excluded (line 17). A distribution sequence is simulated via the steering law from N_{min} to the terminal state of each nearby node $N_{near} \in \mathcal{N}_{near}$ (line 18). If the resulting sequence is probabilistically feasible, and the cost of that node c_{near} is lower than the cost of N_{near} given by $J[N_{near}]$ (line 19), then a new node with this distribution sequence replaces N_{near} (lines 21-22). The tree expansion procedure is then repeated until a node from the goal set is added to the tree. At that point, a distributionally robust feasible trajectory is obtained from the tree root to $\mathcal{X}_{goal}$.

4. SIMULATION RESULTS

We demonstrate our proposed framework using a double integrator robot moving in a bounded and cluttered environment. While the proposed framework can handle dynamic and uncertain obstacles, for simplicity of illustration we assume the obstacles are static and deterministic $(w_{\mathcal{O}_t} = 0, \forall t)$, so that all uncertainty in this example comes

---

Algorithm 1 OFDR-RRT*- Tree Expansion Procedure

1: Inputs: Current Tree $\mathcal{T}$, time step t
2: $x_{rand} \leftarrow$ Sample$(\mathcal{X}_t)$
3: $N_{nearest} \leftarrow$ NearestNode$(x_{rand}, \mathcal{T})$
4: $(\bar{\sigma}, \bar{\Pi}) \leftarrow$ Steer$(\hat{x}[N_{nearest}], D[N_{nearest}], x_{rand})$
5: // Check if sequence $(\bar{\sigma}, \bar{\Pi})$ is DR-Feasible
6: **if** DR-Feasible$(\bar{\sigma}, \bar{\Pi})$ **then**
7:     Create node $N_{min}\{\bar{\sigma}, \bar{\Pi}\}$
8:     $\mathcal{N}_{near} \leftarrow$ NearNodes$(\mathcal{T}, x_{rand}, |\mathcal{T}|)$
9:     // Connect along a minimum-cost path
10:     **for each** $N_{near} \in \mathcal{N}_{near} \backslash N_{nearest}$ **do**
11:         $(\bar{\sigma}, \bar{\Pi}) \leftarrow$ Steer$(\hat{x}[N_{near}], D[N_{near}], x_{rand})$
12:         $c_{rand} \leftarrow$ J$[x_{near}] + \Delta J(\bar{\sigma}, \bar{\Pi})$
13:         **if** DR-Feasible$(\bar{\sigma}, \bar{\Pi})$ & $c_{rand} < J[N_{min}]$ **then**
14:             Replace N_{min} with $N_{min}\{\bar{\sigma}, \bar{\Pi}\}$
15:     Add N_{min} to $\mathcal{T}$
16:     // Re-Wire the Tree
17:     **for each** $N_{near} \in \mathcal{N}_{near} \backslash$ Ancestors(N_{min}) **do**
18:         $(\bar{\sigma}, \bar{\Pi}) \leftarrow$ Steer$(\hat{x}[N_{min}], D[N_{min}], \hat{x}[N_{near}])$
19:         $c_{near} \leftarrow$ J$[N_{min}] + \Delta J(\bar{\sigma}, \bar{\Pi})$
20:         **if** DR-Feasible$(\bar{\sigma}, \bar{\Pi})$ & $c_{near} < J[N_{near}]$ **then**
21:             Delete N_{near} from $\mathcal{T}$
22:             Add new node $N_{new}\{\bar{\sigma}, \bar{\Pi}\}$ to $\mathcal{T}$

Algorithm 2 DR-Feasible Subroutine

    Input: $\mathbf{T}-$time step distribution sequence$(\bar{\sigma}, \bar{\Pi})$
2: **for** $t = 1$ to $\mathbf{T}$ **do**
      $(\hat{x}_t, D_{\hat{x}_t}) \leftarrow t^{\text{th}}$ element in $(\bar{\sigma}, \bar{\Pi})$ sequence
4:     $\mathbb{L} \leftarrow$ Line connecting position block of $\hat{x}_{t-1}$ to $\hat{x}_t$
      for each $i \in \mathcal{B}$ **do**
6:         **if** $(\hat{x}_t, D_{\hat{x}_t})$ dissatisfies (38) or $\mathbb{L} \in \mathcal{O}_{it}$ **then**
            Return false
8: Return true

---

from the unknown initial state, robot process disturbance, and measurement noise. The robot dynamics matrices are

$$A = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} \dfrac{dt^2}{2} & 0 \\ 0 & \dfrac{dt^2}{2} \\ dt & 0 \\ 0 & dt \end{bmatrix}, G = B$$

where $dt = 0.1s$ and the states are the two dimensional position and velocity with two dimensional force inputs. The environmental state dynamics matrices A_z, B_z, G_z are formed accordingly using (5) with the above robot dynamics matrices. We assume the robot to start from the origin with zero initial velocity, and that the initial state and noise covariance matrices are

$$\Sigma_{x_0} = 0.1 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \Sigma_w = 0.1 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}, \Sigma_v = 10^{-3}I.$$

The robot is treated as a point mass without loss of generality, since a known geometry can be easily handled by a fixed tightening of the state constraints. The 2D position of the robot is sampled uniformly within the bounds of the feasible 2D environment whose boundaries are not treated probabilistically. The search radius used in the Algorithm 1 uses a maximum radius of $\mu_{max} = 1$ and the environment based planning constant $\gamma = 20$. The
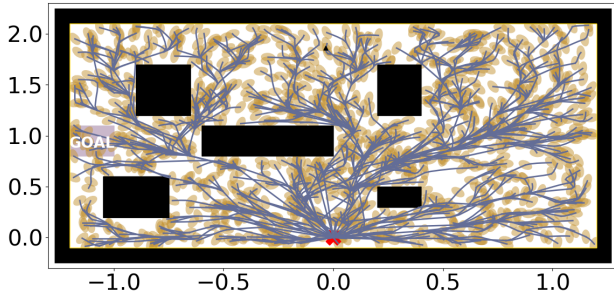
Fig. 1. **No Risk Constraints.** An RRT* tree of 1200 nodes showing nominal states and 1-σ position uncertainty ellipses, generated without any constraint tightening. Since uncertainty is not explicitly incorporated into collision checking, it produces in highly risky trajectories.

output filter dynamics matrices are

$$C_r = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, C_{\mathcal{O}} = I_{s \times ln_0}, H_r = I, H_{\mathcal{O}} = \mathbf{0}. \quad (41)$$

We define the error $e_t = C_{xr}\mathcal{Z}_t - x_s$ for $i = 0, \ldots, T$, with $T = 5$. A dynamic output feedback policy u_t of the form given by (18) that minimizes the cost function

$$J = \mathbb{E}\left[\sum_{t=0}^{T-1} \left(e_t^\top Q e_t + u_t^\top R u_t \right) + e_T^\top Q e_T \right], \quad (42)$$

is computed using dynamic programming to steer the robot from a tree node state x_t to a random feasible sample x_s. The matrices $Q = \begin{bmatrix} 40I & 0 \\ 0 & 2I \end{bmatrix}, R = 0.02I$ are used to penalize the state and control deviations respectively. The distributionally robust state constraints are enforced with probabilistic satisfaction parameter $\alpha = 0.05$. Three different simulations using the above double integrator system are performed namely: 1) deterministic collision check where uncertainties are not accounted for, 2) chance constrained collision check where the system noises are assumed to be Gaussian distributed and, 3) distributionally robust collision check assuming the noises belong to their respective ambiguity sets. For all the simulations, the chance constrained RRT* algorithm with the corresponding collision check procedure is run for 1200 iterations, with 1-σ position uncertainty ellipses from the covariance matrix being drawn at the end of each trajectory.

4.1 Results and Discussion

The tree generated by an RRT* algorithm using a deterministic collision check where uncertainties are not accounted for is shown in Figure 1. It can be seen that highly risky trajectories around the obstacles are generated, since the uncertainty in the state due to the initial localization and system dynamics uncertainties are not explicitly incorporated. Assuming Gaussian noises and using risk parameter $\alpha = 0.05$, the chance constrained variant of RRT* generates more conservative trajectories as shown in Figure 2. However, the actual perception uncertainties in robotic systems often do not match well with a Gaussian assumption, so this approach can still significantly underestimate risks of constraint violation.
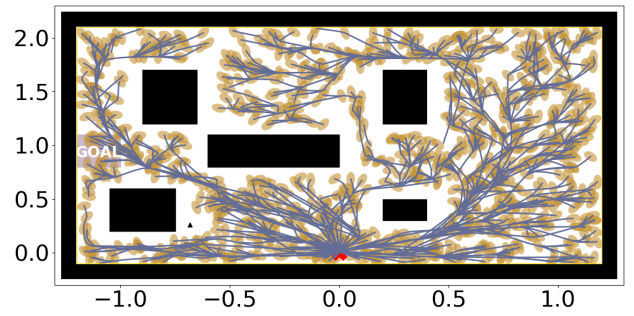

Fig. 2. **Gaussian Risk Constraints.** An RRT* tree of 1200 nodes showing nominal states and 1-σ position uncertainty ellipses, generated with Gaussian chance constraints with $\alpha = 0.05$. Although the trajectories are less risky than when uncertainty is ignored, this approach can still significantly underestimate risks of constraint violation since the actual perception uncertainties in robotic systems often do not match well with a Gaussian assumption.
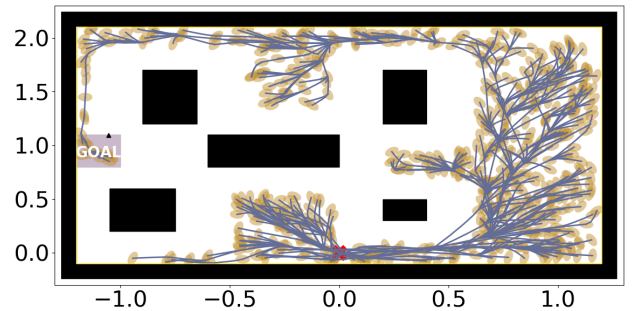

Fig. 3. DR **Risk Constraints.** An RRT* tree of 1200 nodes showing nominal states and 1-σ position uncertainty ellipses, generated with OFDR-RRT* using distributionally robust chance constraints with $\alpha = 0.05$. This approach produces more risk-averse trajectories with a more sophsticated and coherent risk quantification.

The OFDR-RRT* tree as shown in Figure 3 generates more conservative trajectories around the obstacles than the Gaussian chance constrained counterpart, by explicitly incorporating the uncertainty in the state due to the initial localization, system dynamics, and measurement uncertainties in the form of ambiguity sets. It produces trajectories that satisfy the chance constraints under the worst-case distribution in the ambiguity sets. Clearly, the feasible set is smaller with the distributionally robust constraints, and certain nominally feasible paths from the initial state to the goal are deemed too risky in the presence of the uncertainties, e.g., the relatively narrow gaps to the right and below the goal region. These trajectories, with a more sophisticated and coherent quantification of risk, are generated with the same computational complexity as with Gaussian chance constraints.

5. CONCLUSION

In this paper, we presented a methodological framework aimed towards tighter integration of perception and planning in autonomous robotic systems. The environmental state is estimated from sensor data to propagate both estimates and uncertainties of both robot and obstacles. Risk constraints are posed in a meaningful and coherent manner through distributionally robust chance constraints. Using a dynamic output feedback controller together with the distributionally robust risk constraints, a new algorithm called `OFDR-RRT`* is shown to produce risk bounded trajectories with coherent risk assessment. Future research involves studying distribution propagation for nonlinear systems with higher order moments and considering Kalman filter variations (e.g., unscented) along with more sophisticated steering methods to explicitly incorporate the nearby obstacle constraints. Also, the combination of moment- and data-based distribution parameterizations for uncertainty modeling could be used to combine their relative advantages.

REFERENCES

Agha-Mohammadi, A.A., Chakravorty, S., and Amato, N.M. (2014). Firm: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *The International Journal of Robotics Research*, 33(2), 268–304.

Blackmore, L., Li, H., and Williams, B. (2006). A probabilistic approach to optimal robust path planning with obstacles. In *2006 American Control Conference*, 7–pp. IEEE.

Blackmore, L., Ono, M., and Williams, B.C. (2011). Chance-constrained optimal path planning with obstacles. *IEEE Transactions on Robotics*, 27(6), 1080–1094.

Costante, G., Forster, C., Delmerico, J., Valigi, P., and Scaramuzza, D. (2016). Perception-aware path planning. *arXiv preprint arXiv:1605.04151*.

Dissanayake, M.G., Newman, P., Clark, S., Durrant-Whyte, H.F., and Csorba, M. (2001). A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on robotics and automation*, 17(3), 229–241.

Florence, P., Carter, J., and Tedrake, R. (2016). Integrated perception and control at high speed: Evaluating collision avoidance maneuvers without maps. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*.

Frazzoli, E., Dahleh, M.A., and Feron, E. (2002). Real-time motion planning for agile autonomous vehicles. *Journal of guidance, control & dynamics*, 25(1), 116–129.

Goh, J. and Sim, M. (2010). Distributionally robust optimization and its tractable approximations. *Operations research*, 58(4-part-1), 902–917.

Liu, W. and Ang, M.H. (2014). Incremental sampling-based algorithm for risk-aware planning under motion uncertainty. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2051–2058. IEEE.

Luders, B., Kothari, M., and How, J. (2010). Chance constrained rrt for probabilistic robustness to environmental uncertainty. In *AIAA guidance, navigation, and control conference*, 8160.

Luders, B.D., Karaman, S., and How, J.P. (2013). Robust sampling-based motion planning with asymptotic optimality guarantees. In *AIAA Guidance, Navigation, and Control (GNC) Conference*, 5097.

Majumdar, A. and Pavone, M. (2020). How should a robot assess risk? towards an axiomatic theory of risk in robotics. In *Robotics Research*, 75–84. Springer.

Rockafellar, R.T. (2007). Coherent approaches to risk in optimization under uncertainty. In *OR Tools and Applications: Glimpses of Future Technologies*, 38–61. Informs.

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

Sünderhauf, N., Pham, T.T., Latif, Y., Milford, M., and Reid, I. (2017). Meaningful maps with object-oriented semantic mapping. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5079–5085. IEEE.

Wiesemann, W., Kuhn, D., and Sim, M. (2014). Distributionally robust convex optimization. *Operations Research*, 62(6), 1358–1376.

Zhu, H. and Alonso-Mora, J. (2019). Chance-constrained collision avoidance for mavs in dynamic environments. *IEEE Robotics and Automation Letters*, 4(2), 776–783.