Estimating Probability of Collision for Safe Planning under Gaussian Motion and Sensing Uncertainty

Sachin Patil Jur van den Berg Ron Alterovitz

***Abstract*— We present a fast, analytical method for estimating the probability of collision of a motion plan for a mobile robot operating under the assumptions of Gaussian motion and sensing uncertainty. Estimating the probability of collision is an integral step in many algorithms for motion planning under uncertainty and is crucial for characterizing the safety of motion plans. Our method computes an accurate, yet conservative, estimate of the probability of collision based on a priori distributions of the robot state along a given plan. We specifically account for the fact that the probabilities of collision at each stage along the plan are conditioned on the previous stages being collision free, by using a novel method to truncate the a priori state distributions with respect to obstacles and correctly propagating them forward in time. Our method can be directly applied within a variety of existing motion planners to improve their performance and the quality of computed plans. We apply our method to a car-like mobile robot with second order dynamics and to a steerable medical needle in 3D and demonstrate that our method for estimating the probability of collision is orders of magnitude faster than naïve Monte-Carlo sampling methods and reduces estimation error by more than 25% compared to prior methods.**

I. Introduction

For many applications ranging from autonomous vehicles to steerable medical needles operating in the human body [20], the motion plan chosen for execution should be as safe as possible such that there is minimal risk that the robot will collide with obstacles in the environment. Real-world uncertainties arise because the motion of the robot may deviate unpredictably from the assumed dynamics model and because sensors might provide imperfect information about the robot state due to noisy and incomplete measurements. Estimating the probability of collision of a motion plan before actual execution is a critical step in many motion planning algorithms that consider and compensate for the impact of uncertainty on task performance.

In this work, we present a fast, analytical method to estimate the probability of collision for a mobile robot executing a given motion plan under Gaussian models of motion and sensing uncertainty. The speed of our algorithm (requiring only milliseconds of computation time) enables its use in applications that require real-time performance. Our algorithm also ensures that the estimate is conservative; our method never underestimates the probability of collision in order to ensure that safety requirements are guaranteed.

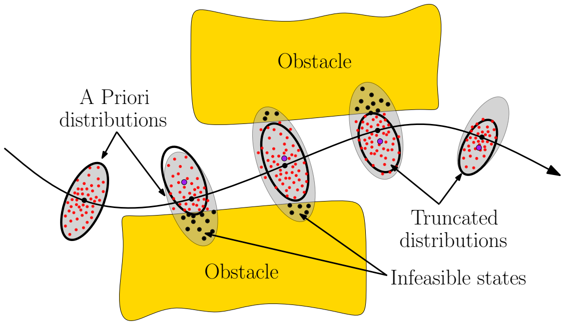


Fig. 1. We estimate the probability of collision based on a priori probability distributions of the robot state. The probabilities of collision at each stage of the plan are conditioned on the previous stages being collision free. We truncate the a priori distributions with respect to obstacles to discount plan executions that collide with obstacles (black disks). Propagating the truncated distributions (black ellipses) accounts for only the collision free samples (red disks), resulting in accurate estimation of the probability of collision. Using the unconditional distributions (gray ellipses) to estimate the collision probability results in a very conservative estimate.

Prior work on motion planning under uncertainty has used both sampling-based and analytical approaches to estimating probability of collision. Naïve Monte-Carlo sampling strategies can estimate the probability of collision by computing the ratio of the number of simulated executions that are collision free. This approach requires a large number of simulations to obtain a reliable estimate, which requires more computation time than analytical approaches. Monte Carlo sampling also offers no guarantee that it will not underestimate the probability of collision, resulting in violation of safety requirements. Under the assumption of Gaussian motion and sensing uncertainty, probability of collision can be estimated quickly based on a priori probability distributions of the robot state [19], [2], [21]. However, prior methods typically "approximate" the collision probability of a plan by assuming the probabilities of collision at stages along the plan are independent. Formally speaking, let $\mathbf{x}_t \in \mathcal{X}$ denote the state of the robot at stage t along the plan, and $\mathcal{X}_F \subset \mathcal{X}$ denote the feasible space not occupied by obstacles. Prior methods assume that the probability that a plan consisting of ℓ stages is collision free is given by $p(\bigwedge_{t=0}^{\ell} \mathbf{x}_t \in \mathcal{X}_F) \approx \prod_{t=0}^{\ell} p(\mathbf{x}_t \in \mathcal{X}_F)$. This yields an overly conservative estimate of the probability of collision (see Fig. 1), which might result in overly conservative motion plans and, depending on the safety required by the motion planner, may result in failure to find a feasible plan even if one exists.

We propose an analytic approach to estimating the probability of collision that accounts for the fact that the distribution of the state at each stage along the plan is *conditioned* on the previous stages being collision free, i.e.,

Sachin Patil and Ron Alterovitz are with Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA. {sachin, ron}@cs.unc.edu
Jur van den Berg is with the School of Computing, University of Utah, Salt Lake City, UT, USA. berg@cs.utah.edu

the probability that a plan is collision free is given by $p(\bigwedge_{t=0}^{\ell} \mathbf{x}_t \in \mathcal{X}_F) = \prod_{t=0}^{\ell} p(\mathbf{x}_t \in \mathcal{X}_F \mid \bigwedge_{i=0}^{t-1} \mathbf{x}_i \in \mathcal{X}_F)$. This amounts to propagating the a priori distributions forward in time in such a way that instances that collide with obstacles are discounted from the propagation (Fig. 1). For this we propose a novel method to truncate the a priori distributions with respect to obstacles, approximate the truncated distributions by Gaussians, and propagate the truncated distributions forward in time. This results in an accurate estimate of the conditional distributions, and consequently, enables accurate estimation of the collision probability.

Our method can be used to quantify the safety of a plan [19], [12], to improve quality of estimation of collision chance constraints [2], [21], or to elegantly account for hard state constraints imposed by obstacles in optimization based [4] or inference based [18] planning methods. Our truncation approach is also directly applicable to the important problem of optimal state estimation with hard state constraints [15].

We present simulation-based results for two scenarios with with stochastic dynamics and partial, noisy state feedback: (1) a car-like robot with second-order dynamics, and (2) a nonholonomic bevel-tip flexible needle. Our method was orders of magnitude faster than naïve sampling based methods and computed a significantly more accurate estimate of collision probability compared to prior analytical methods.

II. Previous Work

Motion planners that consider uncertainty have been developed for a variety of applications [1], [8], [6], [14], [13], [19], [12], [2], [21]. A key step of many motion planning methods that consider uncertainty is to estimate the probability of collision of a motion plan. Efficient Monte-Carlo sampling strategies have been used to accurately estimate this probability [9], [3]. Other methods characterize the uncertainty by estimating the a priori probability distributions of the robot state along a given plan. One approach is to check for collisions between these distributions and obstacles to compute an upper bound for the collision probability for use as a metric to evaluate plan safety [19], [12]. Another approach is to use these distributions to compute a conservative probability bound using Boole's inequality [21], [2]. However, these methods incorrectly assume that the probabilities of collision are independent, which results in overly conservative plans.

The use of truncated Gaussian distributions [7] has been previously explored in the context of optimal state estimation with state constraints [15], but this work does not consider motion uncertainty. Greytak [5] provides an analytical method to compute the probability of collision using truncated Gaussians but does not consider sensing uncertainty. Toussaint [18] uses truncated Gaussians in an expectation-propagation framework for Bayesian inference, but the truncation result is dependent on the order in which constraints are processed, which leads to problems with convergence of the algorithm [17]. In contrast, we propose a novel order-independent algorithm for truncating Gaussian distributions with respect to hard state constraints.

III. Estimating Probability of Collision

A. Problem Statement

We consider a robot operating in an environment that may contain obstacles. The stochastic dynamics of the robot follow the given discrete-time model:

$$\mathbf{x}_t = \mathbf{f}[\mathbf{x}_{t-1}, \mathbf{u}_{t-1}, \mathbf{m}_t], \;\; \mathbf{m}_t \sim \mathcal{N}[\mathbf{0}, M_t] \tag{1}$$

where $\mathbf{x}_t \in \mathcal{X} \subset \mathbb{R}^{n_x}$ is the state of the robot at stage t, $\mathbf{u}_t \in \mathbb{R}^{n_u}$ is the applied control input, and $\mathbf{m}_t$ is zero-mean Gaussian noise with variance M_t that models the motion uncertainty. During execution of a plan, partial and noisy sensor measurements of the robot state are obtained:

$$\mathbf{z}_t = \mathbf{h}[\mathbf{x}_t, \mathbf{n}_t], \;\; \mathbf{n}_t \sim \mathcal{N}[\mathbf{0}, N_t] \tag{2}$$

where $\mathbf{z}_t \in \mathbb{R}^{n_z}$ is the measurement obtained at time t and $\mathbf{n}_t$ is the zero-mean Gaussian noise with variance N_t that models the sensing uncertainty.

Since our method serves as an evaluation metric, we assume the existence of a nominal plan computed by a motion planner. The nominal plan is defined by $[\mathbf{x}_0^\star, \mathbf{u}_0^\star, \ldots, \mathbf{x}_\ell^\star, \mathbf{u}_\ell^\star]$ where $\mathbf{x}_t^\star = \mathbf{f}[\mathbf{x}_{t-1}^\star, \mathbf{u}_{t-1}^\star, \mathbf{0}]$ for $0 < t \leq \ell$, where ℓ is the number of discrete stages in the plan.

During actual execution of the plan, the robot will likely deviate from the nominal plan due to motion uncertainty and inaccurate estimation of the robot state due to sensing uncertainty. To compensate for uncertainty, we assume the robot executes the plan in a closed-loop fashion using a feedback controller and state estimator framework [16]. We assume the existence of a linear feedback control law that operates on the estimate of the robot state and aims to keep the robot close to the nominal plan. We also assume that a Kalman filter is used for state estimation during execution. The objective of our method is to then estimate the probability of collision of a given plan.

B. A Priori State Distributions

Since the robot will be controlled to stay close to the nominal plan during execution, we linearize the nonlinear dynamics and measurement models around the plan, and express them in terms of deviation from the true state $\bar{\mathbf{x}}_t = (\mathbf{x}_t - \mathbf{x}_t^\star)$, control input deviation $\bar{\mathbf{u}}_t = (\mathbf{u}_t - \mathbf{u}_t^\star)$, and deviation from the actual measurement $\bar{\mathbf{z}}_t = (\mathbf{z}_t - \mathbf{h}[\mathbf{x}_t^\star, \mathbf{0}])$, as:

$$\bar{\mathbf{x}}_t = A_t \bar{\mathbf{x}}_{t-1} + B_t \bar{\mathbf{u}}_{t-1} + V_t \mathbf{m}_t, \quad \mathbf{m}_t \sim \mathcal{N}[\mathbf{0}, M_t] \tag{3}$$

$$\bar{\mathbf{z}}_t = H_t \bar{\mathbf{x}}_t + W_t \mathbf{n}_t, \quad \mathbf{n}_t \sim \mathcal{N}[\mathbf{0}, N_t]. \tag{4}$$

where the Jacobians matrices of $\mathbf{f}$ and $\mathbf{h}$ are given by:

$$A_t = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}[\mathbf{x}_{t-1}^\star, \mathbf{u}_{t-1}^\star, \mathbf{0}], \; B_t = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}[\mathbf{x}_{t-1}^\star, \mathbf{u}_{t-1}^\star, \mathbf{0}], \tag{5}$$

$$V_t = \frac{\partial \mathbf{f}}{\partial \mathbf{m}}[\mathbf{x}_{t-1}^\star, \mathbf{u}_{t-1}^\star, \mathbf{0}], \; H_t = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}[\mathbf{x}_t^\star, \mathbf{0}], \; W_t = \frac{\partial \mathbf{h}}{\partial \mathbf{n}}[\mathbf{x}_t^\star, \mathbf{0}].$$

The true state $\mathbf{x}_t$, and hence the true state deviation $\bar{\mathbf{x}}_t$, is not available during actual execution. We use a Kalman filter to keep track of an estimate of the state deviation $\hat{\mathbf{x}}_t = \mathrm{E}[\bar{\mathbf{x}}_t]$. The estimate of the state deviation evolves according to:

$$\hat{\mathbf{x}}_t = K_t \bar{\mathbf{z}}_t + (I - K_t H_t)(A_t \hat{\mathbf{x}}_{t-1} + B_t \bar{\mathbf{u}}_{t-1}), \tag{6}$$

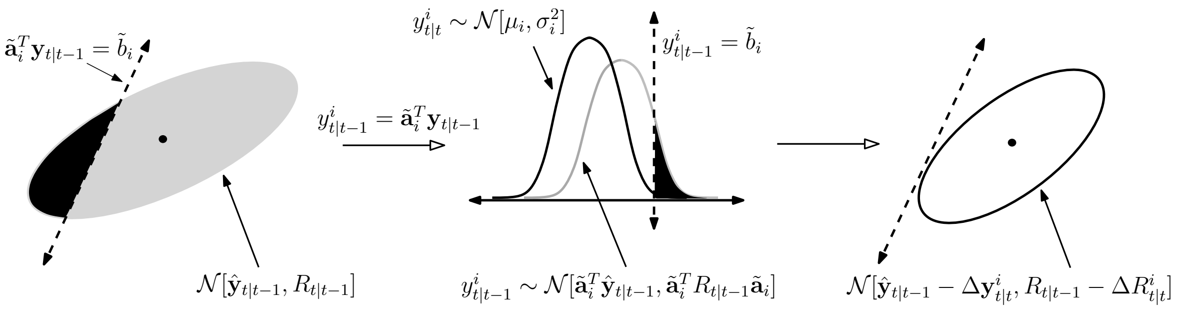

$\tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1} = \tilde{b}_i$
$y^i_{t|t} \sim \mathcal{N}[\mu_i, \sigma_i^2]$
$y^i_{t|t-1} = \tilde{b}_i$
$y^i_{t|t-1} = \tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1}$
$\mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$
$y^i_{t|t-1} \sim \mathcal{N}[\tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1}, \tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i]$
$\mathcal{N}[\hat{\mathbf{y}}_{t|t-1} - \Delta \mathbf{y}^i_{t|t}, R_{t|t-1} - \Delta R^i_{t|t}]$

Fig. 2. The joint conditional distribution $\mathbf{y}_{t|t-1} \sim \mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$ (left), is truncated with respect to the i^{th} constraint $\tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1} \leq \tilde{b}_i$, in $\mathbb{R}^{2n_x}$. Applying an affine transformation, $y^i_{t|t-1} = \tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1}$, transforms the distribution to a 1D Gaussian $y^i_{t|t-1} \sim \mathcal{N}[\tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1}, \tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i]$ (middle). The area under the 1D Gaussian that lies beyond the constraint $y^i_{t|t-1} = \tilde{b}_i$ (shaded in black), gives the probability of collision of the robot with the i^{th} constraint. We estimate the truncated distribution in $\mathbb{R}^{2n_x}$ by conditioning on the truncated 1D Gaussian $y^i_{t|t} \sim \mathcal{N}[\mu_i, \sigma_i^2]$ (right). The mean $(\hat{\mathbf{y}}_{t|t-1} - \Delta \mathbf{y}^i_{t|t})$, and variance $(R_{t|t-1} - \Delta R^i_{t|t})$, of the distribution $\mathbf{y}_{t|t}$ after truncation are obtained by accumulating the effects of truncation with respect to all constraints (order independent).

where K_t is the Kalman gain matrix [15]. To compensate for the uncertainty, the robot is controlled using a linear feedback policy related to the estimate of the state deviation as:

$$\bar{\mathbf{u}}_t = L_{t+1} \hat{\mathbf{x}}_t, \tag{7}$$

where L_t is the control gain matrix determined by the choice of feedback controller [16].

Under the given assumptions, the probability distributions of the robot state can be characterized a priori i.e., before execution. Combining Eqns. (3), (4), (6), and (7), the true state deviation $\bar{\mathbf{x}}_t$, and the estimate $\hat{\mathbf{x}}_t$, jointly evolve as [19]:

$$\begin{bmatrix} \bar{\mathbf{x}}_t \\ \hat{\mathbf{x}}_t \end{bmatrix} = \begin{bmatrix} A_t & B_t L_t \\ K_t H_t A_t & A_t + B_t L_t - K_t H_t A_t \end{bmatrix} \begin{bmatrix} \bar{\mathbf{x}}_{t-1} \\ \hat{\mathbf{x}}_{t-1} \end{bmatrix} + \begin{bmatrix} V_t & 0 \\ K_t H_t V_t & K_t W_t \end{bmatrix} \begin{bmatrix} \mathbf{m}_t \\ \mathbf{n}_t \end{bmatrix}, \begin{bmatrix} \mathbf{m}_t \\ \mathbf{n}_t \end{bmatrix} \sim \mathcal{N}[\mathbf{0}, \begin{bmatrix} M_t & 0 \\ 0 & N_t \end{bmatrix}]. \tag{8}$$

We can write this equation in shorthand (for appropriate definitions of $\mathbf{y}_t$, $\mathbf{q}_t$, F_t, G_t, and Q_t) as:

$$\mathbf{y}_t = F_t \mathbf{y}_{t-1} + G_t \mathbf{q}_t, \; \mathbf{q}_t \sim \mathcal{N}[\mathbf{0}, Q_t]. \tag{9}$$

The mean $\hat{\mathbf{y}}_t \in \mathbb{R}^{2n_x}$ and associated variance $R_t = \text{Var}[\mathbf{y}_t]$, propagate according to:

$$\hat{\mathbf{y}}_t = F_t \hat{\mathbf{y}}_{t-1}, \qquad \hat{\mathbf{y}}_0 = \mathbf{0}, \tag{10}$$

$$R_t = F_t R_{t-1} F_t^T + G_t Q_t G_t^T, \quad R_0 = \begin{bmatrix} \text{Var}[\bar{\mathbf{x}}_0] & 0 \\ 0 & 0 \end{bmatrix}. \tag{11}$$

The *unconditional* a priori distribution of the state $\mathbf{x}_t$ at stage t is then given by the marginal $\mathbf{x}_t \sim \mathcal{N}[(\mathbf{x}_t^\star + \Lambda \hat{\mathbf{y}}_t), \Lambda R_t \Lambda^T]$, where $\Lambda = [I \;\; 0]$.

To accurately estimate the probability of collision, we need to estimate the a priori state distributions at each stage along the plan that are *conditioned* on the previous stages being collision free, i.e. the distributions $(\mathbf{x}_t \mid \bigwedge_{i=0}^{t-1} \mathbf{x}_i \in \mathcal{X}_F)$. To this end, we pursue a recursive approach similar as above to propagate the conditional distributions.

Let $\mathbf{y}_{t|s}$ denote the joint distribution of the true state deviation and its estimate at time t conditioned on the state being collision free for all stages $0, \ldots, s$:

$$\mathbf{y}_{t|s} = (\begin{bmatrix} \bar{\mathbf{x}}_t \\ \hat{\mathbf{x}}_t \end{bmatrix} \mid \bigwedge_{i=0}^{s} \mathbf{x}_i \in \mathcal{X}_F). \tag{12}$$

We then repeatedly, for each stage t of the plan, carry out the following steps. Assume we are given the joint conditional distribution $\mathbf{y}_{t|t-1}$ as approximated by a Gaussian distribution $\mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$. We then approximate the distribution $\mathbf{y}_{t|t} \sim \mathcal{N}[\hat{\mathbf{y}}_{t|t}, R_{t|t}]$ of all collision-free states at stage t by truncating the distribution $\mathbf{y}_{t|t-1}$ against the obstacles in the environment. Truncating the distribution effectively discounts all colliding states from the distribution (Fig. 1), and results in a shift of the mean and variance by $\Delta \mathbf{y}_t$ and ΔR_t (as detailed in Sec. III-C), respectively:

$$\hat{\mathbf{y}}_{t|t} = \hat{\mathbf{y}}_{t|t-1} - \Delta \mathbf{y}_t \tag{13}$$

$$R_{t|t} = R_{t|t-1} - \Delta R_t \tag{14}$$

Using Eqns. (10) and (11), the conditional mean and variance are then propagated according to:

$$\hat{\mathbf{y}}_{t+1|t} = F_{t+1} \hat{\mathbf{y}}_{t|t}, \tag{15}$$

$$R_{t+1|t} = F_{t+1} R_{t|t} F_{t+1}^T + G_{t+1} Q_{t+1} G_{t+1}^T. \tag{16}$$

The recursion then continues. The initial conditions are set by defining $\hat{\mathbf{y}}_{0|-1} = \hat{\mathbf{y}}_0 = \mathbf{0}$ and $R_{0|-1} = R_0 = \begin{bmatrix} \text{Var}[\bar{\mathbf{x}}_0] & 0 \\ 0 & 0 \end{bmatrix}$.

At each stage of the recursion, the marginal $\mathbf{x}_{t|t-1} \sim \mathcal{N}[(\mathbf{x}_t^\star + \Lambda \hat{\mathbf{y}}_{t|t-1}), \Lambda R_{t|t-1} \Lambda^T]$ of the joint distribution $\mathbf{y}_{t|t-1}$ gives the a priori distribution of the robot state $\mathbf{x}_t$, given that all the previous states $[\mathbf{x}_0, \ldots, \mathbf{x}_{t-1}]$ are collision free.

C. Truncating A Priori Distributions

At each stage t of the plan, we approximate the distribution of the feasible robot states with a truncated Gaussian distribution [7]. For the sake of brevity, we assume that the feasible region containing the state at each stage t is convex and is described by the conjunction of k linear inequality constraints as $\bigcap_{i=0}^{k} \mathbf{a}_i \mathbf{x}_t \leq b_i$. We later extend this analysis in Sec. IV to non-convex regions by constructing a locally convex feasible region around the robot state.

Since the true state deviation and its estimate are correlated (Eqn. 8), it is important to truncate the joint conditional distribution $\mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$ in $\mathbb{R}^{2n_x}$, with respect to the k constraints. The i^{th} linear constraint is then represented in $\mathbb{R}^{2n_x}$ as $\tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1} \leq \tilde{b}_i$, where $\tilde{\mathbf{a}}_i = \begin{bmatrix} \mathbf{a}_i \\ \mathbf{0} \end{bmatrix}$, and $\tilde{b}_i = (b_i - \mathbf{a}_i^T \mathbf{x}_t^\star)$.

We truncate the joint conditional distribution with respect to each constraint in a sequential manner and then accumulate the effect of truncation over all the constraints. In contrast to prior methods that use truncated distributions [15], [17], we propose a novel truncation method that does not depend on the order in which the constraints are processed. Given the i^{th} constraint $\tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1} \leq \tilde{b}_i$, we apply an affine transformation $y^i_{t|t-1} = \tilde{\mathbf{a}}_i^T \mathbf{y}_{t|t-1}$ to transform the conditional distribution $\mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$, to a 1D Gaussian $\mathcal{N}[\tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1}, \tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i]$ along an axis normal to the constraint (as shown in Fig. 2). The problem now simplifies to truncating the 1D Gaussian distribution at a specified upper bound given by $y^i_{t|t-1} = \tilde{b}_i$, which is well-known from standard statistical literature [7]. The mean, μ_i, and variance, σ_i^2, of the truncated 1D Gaussian $y^i_{t|t}$ is given by:

$$\mu_i = \tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1} + \lambda(\alpha_i) \sqrt{\tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i}, \tag{17}$$

$$\sigma_i^2 = \tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i (1 - \lambda(\alpha_i)^2 + \alpha_i \lambda(\alpha_i)), \tag{18}$$

where

$$\alpha_i = \frac{(\tilde{b}_i - \tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1})}{\sqrt{\tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i}}, \quad \lambda(\alpha_i) = \frac{\text{pdf}(\alpha_i)}{\text{cdf}(\alpha_i)}. \tag{19}$$

$\lambda(\alpha_i)$ is the ratio of the standard Gaussian (mean 0 and variance 1) probability distribution function and the standard Gaussian cumulative distribution function evaluated at α_i. Note that $(1 - \text{cdf}(\alpha_i))$ is the area under the Gaussian that lies beyond the constraint (shaded in black in Fig. 2), and is the probability that the robot lies in the infeasible region corresponding to the i^{th} constraint.

The mean and variance of the truncated distribution $\mathbf{y}_{t|t}$ are found by conditioning the joint distribution $(\mathbf{y}_{t|t-1}, y^i_{t|t-1})$, on the truncated 1D distribution $y^i_{t|t}$: $\mathbf{y}_{t|t} = (\mathbf{y}_{t|t-1} | y^i_{t|t-1} = y^i_{t|t})$ (see appendix). The shift in the mean due to truncation with respect to the i^{th} constraint is given by:

$$\Delta \mathbf{y}^i_t = \frac{R_{t|t-1} \tilde{\mathbf{a}}_i}{\tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i} (\tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1} - \mu_i), \tag{20}$$

and the shift in variance is given by:

$$\Delta R^i_t = \frac{R_{t|t-1} \tilde{\mathbf{a}}_i}{\tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i} (\tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i - \sigma_i^2) \frac{\tilde{\mathbf{a}}_i^T R_{t|t-1}}{\tilde{\mathbf{a}}_i^T R_{t|t-1} \tilde{\mathbf{a}}_i}. \tag{21}$$

Given k constraints, the cumulative shift in the mean due to truncation is then given by $\Delta \mathbf{y}_t = \sum_{i=0}^k \Delta \mathbf{y}^i_t$, and the cumulative shift in variance is given by $\Delta R_t = \sum_{i=0}^k \Delta R^i_t$. The mean and variance of the truncated conditional distributions are then propagated recursively using Eqs. (15) and (16).

D. Estimating the Probability of Collision

We use the truncated conditional distributions to estimate the overall probability of collision of the given plan, based on the conditional probabilities of collisions at each stage along the plan. Given the joint conditional distribution at stage t, $\mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$, and the set of k linear constraints that define the locally convex region of free space containing the robot, we compute a lower bound for the probability of the robot being collision free using Boole's inequality, as [21]:

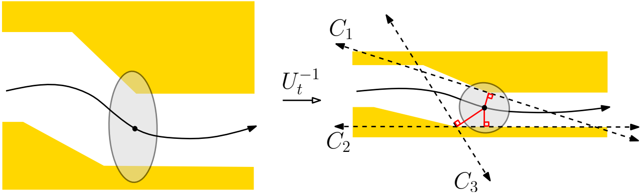


Fig. 3. We transform the environment such that the distribution of the robot position (left) is converted to a unit sphere (right). We then sequentially process the obstacle geometry in increasing order of distance from the origin. The linear constraints that define a locally convex region of the free space are determined by the normal to the vector of closest approach (shown in red). The locally convex region constructed using our approach for this example is defined by three constraints determined in order of their indices: C_1, C_2, and C_3.

$$\begin{aligned} p(\mathbf{x}_{t|t-1} \in \mathcal{X}_F) &\geq 1 - p\Big(\bigvee_{i=0}^{k} \tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1} > \tilde{b}_i\Big) \\ &\geq 1 - \sum_{i=0}^{k} (1 - \text{cdf}(\alpha_i)). \end{aligned} \tag{22}$$

The overall probability that the robot does not collide with any obstacle for the duration ℓ of the plan, is given by:

$$p\Big(\bigwedge_{t=0}^{\ell} \mathbf{x}_t \in \mathcal{X}_F\Big) = \prod_{t=0}^{\ell} p(\mathbf{x}_{t|t-1} \in \mathcal{X}_F), \tag{23}$$

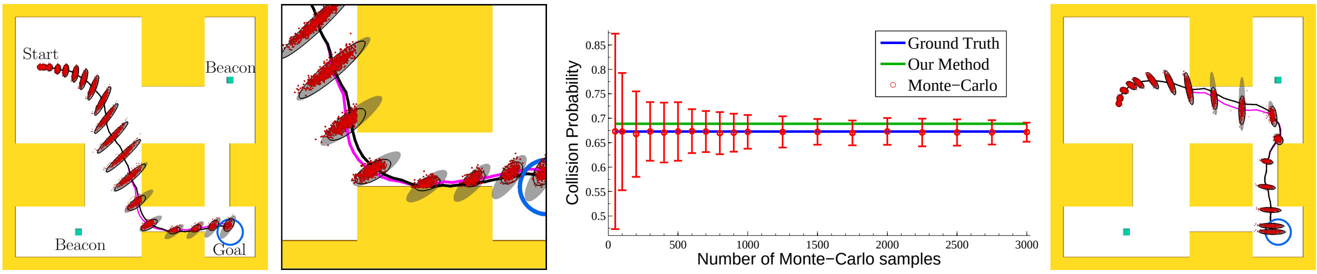
and the overall probability of collision is provided by the complement $(1 - p(\bigwedge_{t=0}^{\ell} \mathbf{x}_t \in \mathcal{X}_F))$.

IV. Local Convexification of Free Space

We extend our analysis to non-convex regions by truncating the joint conditional distributions with respect to linear constraints that define a locally convex region of free space containing the robot. For the sake of simplicity, we assume that only the robot position is relevant for collision detection. Given the conditional distribution at stage t, $\mathcal{N}[\hat{\mathbf{y}}_{t|t-1} + \begin{bmatrix}\mathbf{x}_t^\star \\ \mathbf{x}_t^\star\end{bmatrix}, R_{t|t-1}]$, the distribution of the robot position is given by the marginal $\mathcal{N}[\hat{\mathbf{p}}_{t|t-1}, \Sigma_{t|t-1}]$, computed over the state dimensions that describe the robot position. We outline a greedy method that computes a locally convex region of free space such that the probability that the distribution $\mathcal{N}[\hat{\mathbf{p}}_{t|t-1}, \Sigma_{t|t-1}]$ lies beyond the convex region is minimal.

Adopting the approach suggested in [19], we linearly transform the environment geometry by applying the transform U_t^{-1}, where $\Sigma_t = U_t U_t^T$ is the Cholesky decomposition. This transforms the uncertainty distribution of the robot position to a Gaussian distribution with zero mean and unit variance, which is a unit sphere in Euclidean space centered at the origin. The spherical symmetry simplifies the task of constructing a nonconservative convex region of free space around the distribution of the position of the robot.

We construct the convex region using a sequential process. We consider the closest point on the obstacle geometry from the origin. The linear truncation constraint $\mathbf{a}_i^T \mathbf{p}_{t|t-1} \leq b_i$, is defined by the normal to the vector of closest approach to



(a) Test case for car-like robot (b) Zoom into narrow passage (c) Comparison with Monte-Carlo simulations (d) Different start location

Fig. 4. Car-like robot with second order dynamics: (a) The conditional distributions (black ellipses) computed using our method provide an accurate estimate of the distribution of the collision free robot states along the plan (Monte-Carlo samples shown by red dots), while the unconditional distributions (solid gray ellipses) are overly conservative. The probability of collision estimated by our method for this example is 68.87%, while the ground truth probability determined by Monte-Carlo simulations is 67.28%. (b) Zoomed in view of the conditional distributions in the narrow corridor in the environment. The mean of the conditional distribution (magenta) deviates from the nominal plan due to the truncation of the distributions due to obstacle collisions. (c) Comparison of our method to Monte-Carlo simulations. (d) Conditional and unconditional distributions computed along a second plan. The probability of collision estimated by our method for this example is 44.78%, while the ground truth probability is 42.32%.

the obstacle. We then prune away all geometry that lies in the infeasible half space $\mathbf{a}_i^T \mathbf{p}_{t|t-1} > b_i$ of the constraint, and continue the process by considering the closest point on the remaining obstacle geometry to the origin. This procedure is repeated until all geometry has been pruned away. It is important to note that our convexification method works in a greedy fashion and is not guaranteed to find the least conservative convex bounding region.

V. Results

We present simulation results for two scenarios: a car-like robot with second order dynamics and a nonholonomic bevel-tip flexible needle, navigating with stochastic dynamics under the guidance of partial and noisy state measurements. In each case, we initialize our method with a nominal plan computed using an RRT planner [10]. We tested our C++ implementation on a 3.33 GHz Intel® i7™ PC.

We validate our method by comparing the estimated collision probability with the ground truth probability computed using a million Monte-Carlo simulations (considered as ground truth) of the given motion plan and counting the ratio of collision free simulations. Each execution is simulated in a closed-loop fashion using the given linear feedback controller and a Kalman filter, and with artificially generated motion and measurement noise.

A. Car-like Robot with Second Order Dynamics

We consider a nonholonomic car-like robot with second order dynamics, navigating in a 2D environment with obstacles (Fig. 4). The state of the robot, $\mathbf{x} = [x, y, \theta, v]^T \in \mathbb{R}^4$, consists of its position $[x, y]$, its orientation θ, and its speed v. The control input $\mathbf{u} = [a, \phi]^T \in \mathbb{R}^2$, consists of the acceleration a, and steering angle ϕ, corrupted by motion noise $\mathbf{m} = [\tilde{a}, \tilde{\phi}]^T \sim \mathcal{N}[\mathbf{0}, M]$. This gives the following stochastic dynamics model:

$$\mathbf{f}[\mathbf{x}, \mathbf{u}, \mathbf{m}] = \begin{bmatrix} x + \tau v \cos\theta \\ y + \tau v \sin\theta \\ \theta + \tau v \tan(\phi + \tilde{\phi})/d \\ v + \tau(a + \tilde{a}) \end{bmatrix}, \tag{24}$$

where τ is the time step, and d is the length of the car.

The robot localizes itself using noisy signal measurements from two beacons b_1 and b_2, placed in the environment at locations $[\check{x}_1, \check{y}_1]$ and $[\check{x}_2, \check{y}_2]$ respectively. The strength of the signal decays quadratically with the distance to the beacon. The robot also measures its current speed using an on-board speedometer. This gives us the following stochastic measurement model:

$$\mathbf{h}[\mathbf{x}, \mathbf{n}] = \begin{bmatrix} 1/((x - \check{x}_1)^2 + (y - \check{y}_1)^2 + 1) \\ 1/((x - \check{x}_2)^2 + (y - \check{y}_2)^2 + 1) \\ v \end{bmatrix} + \mathbf{n}. \tag{25}$$

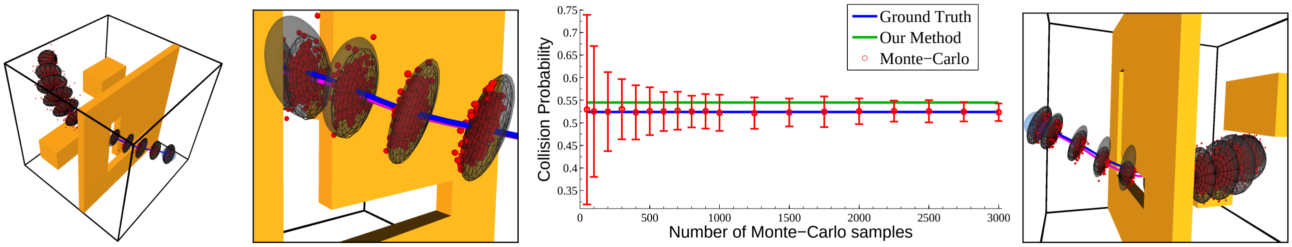
where the observation vector $\mathbf{z} \in \mathbb{R}^3$, consists of two readings of signal strengths from the beacons and a speed measurement, corrupted by sensing noise $\mathbf{n} \sim \mathcal{N}[\mathbf{0}, N]$.

Fig. 4(b) shows the discrepancy between the unconditional and conditional distributions in the presence of obstacles. The conditional distributions computed using our method provide an accurate estimate of the distribution of the collision free robot states along the plan, thus providing an accurate estimate of the probability of collision. Fig. 4(d) shows how the mean of the conditional distribution can deviate significantly in the close vicinity of obstacles. Interestingly, the conditional and unconditional distributions become identical towards the end of the plan in the absence of obstacles.

B. Nonholonomic Bevel-tip Flexible Needle

We also apply our method to a nonholonomic bevel-tip flexible needle [20], navigating in a 3D environment with obstacles (Fig. 5(a)). This class of needles offers improved mobility, enabling access to previously inaccessible targets while maneuvering around sensitive or impenetrable areas.

The state of the needle $\mathbf{x}$, is described by the 4×4 matrix $X = \left[\begin{smallmatrix} R & \mathbf{p} \\ \mathbf{0} & 1 \end{smallmatrix}\right] \in SE(3)$, where $\mathbf{p} \in \mathbb{R}^3$ is the position of the needle tip and $R \in SO(3)$ is the rotation matrix that encodes the orientation of the needle tip relative to a world coordinate frame. The needle naturally moves along constant curvature paths when inserted into tissue, but the curvature of the needle motion can be varied by duty cycled spinning of the needle during insertion. Under these modeling assumptions [20], the control input $\mathbf{u} = [v, w, \kappa]^T \in \mathbb{R}^3$, consists of the



(a) 3D test case

(b) Zoom into narrow corridor

(c) Comparison with Monte-Carlo simulations

(d) Different plan

Fig. 5. Nonholonomic bevel-tip steerable needle: (a) Unconditional distributions (solid gray ellipsoids) provide an overly conservative approximation of the uncertainty. Our method computes conditional distributions (black wireframe ellipsoids), which provide an accurate estimate of the probability distributions of the feasible robot states (shown in red). The collision probability estimated by our method for the given plan is 54.5%, while the ground truth probability is 52.4%. (b) Zoomed in view of the conditional distributions in the narrow corridor in the environment. (c) Comparison of our method to Monte-Carlo simulations. (d) The probability of collision estimated by our method for a second plan is 43.9%, while the ground truth probability is 42.2%.

insertion speed v, rotation speed applied at the base of the needle w, and the curvature κ.

It is convenient to describe the dynamics of the needle tip in terms of the instantaneous twist $U \in se(3)$ expressed in a local coordinate frame attached to the needle tip, given by:

$$U = \begin{bmatrix} [\mathbf{w}] & \mathbf{v} \\ \mathbf{0} & 0 \end{bmatrix}, \ \mathbf{w} = \begin{bmatrix} v\kappa & 0 & w \end{bmatrix}^T, \ \mathbf{v} = \begin{bmatrix} 0 & 0 & v \end{bmatrix}^T, \quad (26)$$

where the notation $[\mathbf{s}]$ for a vector $\mathbf{s} \in \mathbb{R}^3$ refers to the 3×3 skew-symmetric cross-product matrix. The instantaneous twist $\tilde{U}$ that encodes the additive motion noise $\mathbf{m} = [\tilde{\mathbf{v}}^T \ \tilde{\mathbf{w}}^T]^T \sim \mathcal{N}[\mathbf{0}, M]$, can be similarly expressed.

Given a time step τ, the stochastic discrete-time dynamics of the needle tip is given by the following model:

$$\mathbf{f}[\mathbf{x}, \mathbf{u}, \mathbf{m}] = X \exp(\tau(U + \tilde{U})). \quad (27)$$

We also assume that we receive partial, noisy feedback on only the position of the needle tip $\mathbf{p}$, and not its orientation. This is a reasonable assumption since current medical imaging technologies such as ultrasound do not allow for measuring the full state of the needle tip (as the imaging resolution is often too low to infer its orientation). The noise in the sensor measurement is modeled as $\mathbf{n} \sim \mathcal{N}[\mathbf{0}, N]$. This gives the following stochastic measurement model:

$$\mathbf{h}[\mathbf{x}, \mathbf{n}] = \mathbf{p} + \mathbf{n}. \quad (28)$$

We follow the approach in [20] to approximate the given nonlinear dynamics and measurement models with local linearizations around the nominal plan.

C. Analysis

Table I compares the probability of collision estimated by our method against the ground truth probability computed using Monte-Carlo simulations for the scenarios considered above. Our estimate lies within 5% of the ground truth value. It is important to note that Monte-Carlo simulations provide an unbiased estimate of the probability of collision, and can underestimate the probability if a sufficiently large number of samples are not considered. In contrast, our method provides a conservative estimate of the probability.

For the test case depicted in Fig. 4(a), each Monte-Carlo simulation takes 0.36 milliseconds, while our method requires a total computation time of 0.94 milliseconds. Fig. 4(d) shows the variance in the probability estimates computed using Monte-Carlo simulations with much fewer samples (computed over 100 trials). As expected, the variance decreases as the number of Monte-Carlo samples increases. Even neglecting the fact that Monte-Carlo simulations underestimate the collision probability, it still takes 3000 simulations to arrive within the accuracy bounds of our method. This corresponds to over a second of computation time just to estimate the collision probability, which is undesirable for real-time motion planning under uncertainty.

Similarly, for the test case considered in Fig. 5(a), each Monte-Carlo simulation takes 0.69 milliseconds while our method takes 7.4 milliseconds. It takes 2000 simulations to arrive within the accuracy bounds of our method (Fig. 5(d)), which corresponds to 1.4 seconds of computation time. Our method provides accurate, yet conservative, estimates of the collision probability while incurring negligible computational overhead. This makes it especially suitable for online planning algorithms that explicitly consider uncertainty.

We compare our method to prior methods that rely on a priori state distributions to estimate the collision probability. We generated a set of 100 plans using the RRT planner using randomly initialized start states. For each plan, we estimated the collision probability using our method, applying Boole's inequality to the unconditional distributions [21], and LQG-MP [19]. We use the mean error as a metric to compare the probability estimates to the ground truth probability. As summarized in Table I, the estimate computing using our method reduces the estimation error by more than 25% as compared to the collision quality metric provided by LQG-MP [19] and the collision probability computed using the unconditional distributions directly [21]. It is important to note that all these estimation methods, including ours, provide a conservative bound for the collision probability.

Robot	Our method		Unconditional [21]		LQG-MP [19]	
	MAE (%)	Avg. Time (ms)	MAE (%)	Avg. Time (ms)	MAE (%)	Avg. Time (ms)
car	3.0 (± 2)	9	28.0 (± 15)	6	52.2 (± 15)	4
needle	5.0 (± 3)	14	20.7 (± 7)	12	61.7 (± 12)	10

TABLE I

COMPARISON OF OUR METHOD WITH PRIOR METHODS OVER 100 PLANS IN TERMS OF MEAN ABSOLUTE ERROR (MAE) FROM GROUND TRUTH PROBABILITY. STANDARD DEVIATIONS PROVIDED IN PARENTHESES.

VI. Conclusion and Future Work

We have presented an analytical method to estimate a priori the probability of collision for a mobile robot operating under Gaussian motion and sensing uncertainty. We have shown that it is necessary to consider the correlations between the a priori probability distributions of the robot state, to accurately estimate the true distributions and consequently, the probability of collision. We have also proposed a novel method for approximating the distribution of feasible states with truncated Gaussian distributions. Our method is directly applicable to a variety of motion planning under uncertainty methods [19], [21], [2], [4], [18] to improve the quality and safety of the computed plans.

We assume that the robot operates under the assumptions of Gaussian motion and sensing uncertainty, which might not be an acceptable approximation in some applications where multi-modal beliefs are expected to appear. However, the class of problems where Gaussian distributions are applicable is large, as is proven by the widespread use of the extended Kalman filter for state estimation, for instance in mobile robotics. In future work we plan to apply our method to real-world problems that involve complex dynamics and that would benefit from a fast planner that considers uncertainty, including autonomous quadrotor flight, medical needle steering, and manipulation of deformable tissue [12]. We also plan to extend our method to handle non-point robots and to incorporate other sources of uncertainty such as imprecise sensing of obstacles in the environment [6]. Another interesting avenue for future research is to look at applications with unilateral contacts where the distribution of feasible states is no longer Gaussian [4].

References

[1] R. Alterovitz, T. Simeon, and K. Goldberg, "The Stochastic Motion Roadmap: A sampling framework for planning with Markov motion uncertainty," in *Proc. Robotics: Science and Systems (RSS)*, 2007.

[2] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 723–730.

[3] N. E. du Toit and J. W. Burdick, "Probabilistic collision checking with chance constraints," *IEEE Trans. Robotics*, vol. 27, pp. 809–815, 2011.

[4] T. Erez and W. D. Smart, "A scalable method for solving high-dimensional continuous POMDPs using local approximation," in *Conf. on Uncertainty in Artificial Intelligence*, 2010, pp. 160–167.

[5] M. Greytak, "Integrated motion planning and model learning for mobile robots with application to marine vehicles," Ph.D. dissertation, Massachusetts Institute of Technology, 2009.

[6] L. Guibas, D. Hsu, H. Kurniawati, and E. Rehman, "Bounded uncertainty roadmaps for path planning," *Algorithmic Foundation of Robotics VIII*, pp. 199–215, 2009.

[7] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous univariate distributions*. John Wiley & Sons, 1994, vol. 1.

[8] H. Kurniawati, D. Hsu, and W. Lee, "SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces," in *Proc. Robotics: Science and Systems (RSS)*, 2008.

[9] A. Lambert, D. Gruyer, and G. S. Pierre, "A fast Monte Carlo algorithm for collision probability estimation," in *Int. Conf. on Control, Automation, Robotics and Vision (ICARV)*, 2006, pp. 406–411.

[10] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006, Available at http://planning.cs.uiuc.edu.

[11] J. R. Movellan, "Discrete Time Kalman Filters and Smoothers," 2011, MPLab Tutorials, Univ. California at San Diego.

[12] S. Patil, J. van den Berg, and R. Alterovitz, "Motion planning under uncertainty in highly deformable environments," in *Proc. Robotics: Science and Systems (RSS)*, 2011.

[13] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observations," in *Proc. Robotics: Science and Systems (RSS)*, 2010.

[14] S. Prentice and N. Roy, "The Belief Roadmap: Efficient planning in belief space by factoring the covariance," *Int. Journal of Robotics Research*, vol. 28, no. 11–12, pp. 1448–1465, 2009.

[15] D. Simon, *Optimal State Estimation: Kalman, H-infinity, and Nonlinear Approaches*. John Wiley & Sons, 2006.

[16] R. F. Stengel, *Optimal Control and Estimation*. Dover Publications, 1994.

[17] M. Toussaint, "Pros and cons of truncated Gaussian EP in the context of approximate inference control." in *NIPS Workshop on Probabilistic Approaches for Robotics and Control*, 2009.

[18] ——, "Robot trajectory optimization using approximate inference," in *Proc. Int. Conf. on Machine Learning*, 2009, pp. 1049–1056.

[19] J. van den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," in *Int. Journal of Robotics Research*, vol. 30, no. 7, 2011, pp. 895–913.

[20] J. van den Berg, S. Patil, R. Alterovitz, P. Abbeel, and K. Goldberg, "LQG-based planning, sensing, and control of steerable needles," in *Proc. Workshop Algorithmic Foundations of Robotics (WAFR)*, 2010, pp. 373–389.

[21] M. P. Vitus and C. J. Tomlin, "Closed-loop belief space planning for linear, Gaussian systems," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 2152–2159.

Appendix
Truncated Gaussian Distributions

Given a distribution $Y' \sim \mathcal{N}[\mu'_Y, \Sigma'_Y]$ and a joint Gaussian distribution (X,Y):

$$(X,Y) \sim \mathcal{N}\left[\begin{bmatrix}\mu_X \\ \mu_Y\end{bmatrix}, \begin{bmatrix}\Sigma_X & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_Y\end{bmatrix}\right], \tag{29}$$

the conditional distribution $(X|Y=Y')$ can be derived using the law of iterated expectations and the law of total variances, and is given by [11]:

$$(X|Y=Y') \sim \mathcal{N}[\mu_X - L(\mu_Y - \mu'_Y), \Sigma_X - L(\Sigma_Y - \Sigma'_Y)L^T] \tag{30}$$

where $L = \Sigma_{XY}\Sigma_Y^{-1}$.

We can construct the joint distribution of the conditional distribution $\mathbf{y}_{t|t-1} \sim \mathcal{N}[\hat{\mathbf{y}}_{t|t-1}, R_{t|t-1}]$, and the transformed 1D distribution $y^i_{t|t-1} \sim \mathcal{N}[\tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1}, \tilde{\mathbf{a}}_i^T R_{t|t-1}\tilde{\mathbf{a}}_i]$, according to Eqn. (29) as:

$$(\mathbf{y}_{t|t-1}, y^i_{t|t-1}) \sim \mathcal{N}\left[\begin{bmatrix}\hat{\mathbf{y}}_{t|t-1} \\ \tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1}\end{bmatrix}, \begin{bmatrix}R_{t|t-1} & R_{t|t-1}\tilde{\mathbf{a}}_i \\ \tilde{\mathbf{a}}_i^T R_{t|t-1} & \tilde{\mathbf{a}}_i^T R_{t|t-1}\tilde{\mathbf{a}}_i\end{bmatrix}\right]. \tag{31}$$

Using Eqn. (30), we reconstruct the truncated mean and variance of the joint distribution by conditioning on the truncated 1D distribution $y^i_{t|t} \sim \mathcal{N}[\mu_i, \sigma_i^2]$ (Eqn. 17, 18), according to:

$$\begin{aligned}&(\mathbf{y}_{t|t-1} | y^i_{t|t-1} = y^i_{t|t}) \sim \\ &\mathcal{N}[\hat{\mathbf{y}}_{t|t-1} - L(\tilde{\mathbf{a}}_i^T \hat{\mathbf{y}}_{t|t-1} - \mu_i), R_{t|t-1} - L(\tilde{\mathbf{a}}_i^T R_{t|t-1}\tilde{\mathbf{a}}_i - \sigma_i^2)L^T],\end{aligned} \tag{32}$$

where $L = \frac{R_{t|t-1}\tilde{\mathbf{a}}_i}{\tilde{\mathbf{a}}_i^T R_{t|t-1}\tilde{\mathbf{a}}_i}$, which is exactly as in Eqns. (20, 21).