Motion Planning with Environment Uncertainty

Antony Thomas, Fulvio Mastrogiovanni and Marco Baglietto

***Abstract*—As robots are becoming more pervasive and are increasingly used in close proximity to humans and other objects in factories, living spaces, elderly care, and robotic surgery, planning for collision free trajectories in real-time is imperative for safe and efficient operation. Yet, in the presence of noisy sensors, both the robot and world state cannot be estimated precisely. In this paper we investigate the problem of motion planning under environment uncertainty. To this end, we first incorporate object uncertainties in Belief Space Planning (BSP) and derive the resulting Bayes filter in terms of the Extended Kalman Filter (EKF) update equations. We further formulate the collision constraint as a quadratic form in random variables, thus computing an exact expression for collision probability. We further validate our approach using numerical integration and provide a comparison to other approaches.**

***Index Terms*—Collision Probability, Belief Space Planning**

I. Introduction

Belief Space Planning (BSP) and decision making under uncertainty has been researched extensively in the past with applications spanning a variety of areas including autonomous navigation, multi-modal planning, and active *Simultaneous Localization and Mapping* (SLAM) [1], [6], [10], [11], [15], [17], [20], [22], [23]. Uncertainties often arise due to insufficient knowledge about the environment, inexact robot motion or imperfect sensing. Planning is therefore done in the *belief* space, which corresponds to the set of all probability distributions over possible robot states. Further, the nature of uncertain environments are such that they often preclude the existence of collision free trajectories [2]. As such, for safe navigation, both the robot state uncertainty and the uncertainty in obstacle estimates need to be considered while computing collision probabilities. [13], [16] truncate the Gaussian state distributions to compute bounded collision-free trajectories. [5] and [14] compute the collision probability by marginalizing the joint distribution between the robot pose and the obstacle. However, since there is no closed-form solution to this formulation, an approximation is assumed. In [12], an approximation is computed using *Monte Carlo Integration* (MCI), which is nonetheless computationally intensive. Another relevant method that uses Monte Carlo approach and is real-time compatible is *Monte Carlo Motion Planning* (MCMP) [8]. In [2], a *Gaussian Process* (GP) based approach is used to learn motion patterns (a mapping from states to trajectory derivatives) to identify possible future obstacles trajectories. [3] focus exclusively on obstacle uncertainty. Uncertain obstacles are modeled as polytopes with Gaussian-distributed faces in [19]. Risk contours map are considered in [7], [9] to obtain safe paths with bounded risks. Formal verification methods have also been used to construct safe plans [4], [18]. Yet, most approaches leverage's Boole's inequality to compute collision probability along a path by summing or multiplying the probabilities along different waypoints in the path. Such approaches tend to be overly conservative. Moreover, in most approaches, the collision probability computed along each waypoint is an approximation of the true value. On the one hand, such approximations can overly penalize paths and could gauge all plans to be infeasible. On the other hand some approximations can be lower than the true collision probability values and can lead to synthesizing unsafe plans.

Localization is also a key aspect for safe and efficient navigation. However, most approaches assume that landmarks are known with high certainty. For example, given the map of the environment, while planning for future actions the standard Markov localization does not take into account the map uncertainty (that is, landmark locations are assumed to be perfect). However, this might not be true in practice. For example, landmarks locations obtained from a SLAM session may be uncertain. This landmark uncertainty directly translates to the fact that the viewpoints from which the landmark can be observed are uncertain. Therefore, one can only reason in terms of the probability of observing the object from the considered pose or the viewpoint. This results in a probability distribution function for the viewpoints, whose mean corresponds to observing the object with highest probability. Consequently, not considering this uncertainty can wrongly localize the robot, leading to inefficient plans causing catastrophes. We will use the term *object uncertainty* to refer to the notion of uncertainty in landmark location.

II. Object Uncertainty

We define the *object space* $\mathcal{O} = \{O^i | O^i \text{ is an object, and } 1 \leq i \leq |\mathcal{O}|\}$ to be the set of all objects in the environment. Given an initial distribution $p(\mathbf{x}_0)$, and the motion and observation models $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k)$ and $p(\mathbf{z}_k|\mathbf{x}_k, O^i_k)$, the posterior probability distribution at time k is the *belief* $b[\mathbf{x}_k]$ and can be written as $p(\mathbf{x}_k|\mathbf{z}_k, O^i_k, \mathbf{z}_{0:k-1}, \mathbf{u}_{0:k-1})$, where O^i_k is the object observed at time k, $\mathbf{z}_{0:k-1} \doteq \{\mathbf{z}_0, ..., \mathbf{z}_{k-1}\}$ is the sequence of measurements up to $k-1$ and $\mathbf{u}_{0:k-1} \doteq \{\mathbf{u}_0, ..., \mathbf{u}_{k-1}\}$ is the sequence of control up to $k-1$. Using Bayes rule and theorem of total probability, $b[\mathbf{x}_k]$ can be expanded as

$$p(\mathbf{x}_k|\mathbf{z}_k, O^i_k, \mathbf{z}_{0:k-1}, \mathbf{u}_{0:k-1}) =$$
$$\eta p(\mathbf{z}_k|\mathbf{x}_k, O^i_k) p(O^i_k|\mathbf{x}_k) \int_{\mathbf{x}_{k-1}} p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) b[\mathbf{x}_{k-1}] \quad (1)$$

where $\eta = 1/p(\mathbf{z}_k|\mathbf{z}_{0:k-1}, \mathbf{u}_{0:k-1})$ is the normalization constant. The term $p(O^i_k|\mathbf{x}_k)$ denotes the probability of observing

The authors are with the Department of Informatics, Bioengineering, Robotics, and Systems Engineering, University of Genoa, Via All'Opera Pia 13, 16145 Genoa, Italy. email:antony.thomas@dibris.unige.it, fulvio.mastrogiovanni@unige.it, marco.baglietto@unige.it

the object O_k^i from the pose $\mathbf{x}_k$ and models the object uncertainty. This term can be modeled given the environment and since we consider a Gaussian parametrization of the belief, in this work we model the object distribution as a Gaussian $p(O_k^i|\mathbf{x}_k) \sim \mathcal{N}(\boldsymbol{\mu}_{O_k^i}, \Sigma_{O_k^i})$, where $\boldsymbol{\mu}_{O_k^i}$ is the mean viewpoint/pose that corresponds to the maximum probability of observing O_k^i and $\Sigma_{O_k^i}$ is the associated covariance. The EKF is used for state estimating and the posterior mean and covariance can be computed as (notations and complete derivation can be found in [21])

$$\boldsymbol{\mu}_k = \bar{\boldsymbol{\mu}}_k + K_{k+1}\left(\mathbf{z}_k - h\left(\bar{\boldsymbol{\mu}}_k\right)\right) + \Sigma_k \Sigma_{O_k^i}^{-1}(\boldsymbol{\mu}_{O_k^i} - \bar{\boldsymbol{\mu}}_k) \quad (2)$$

$$\Sigma_k = (I - K_k H_k)\,\bar{\Sigma}_k(\bar{\Sigma}_k + \Sigma_{O_k^i})^{-1}\Sigma_{O_k^i} \quad (3)$$

We note that when no object uncertainty is considered the update step of the standard EKF gives $\boldsymbol{\mu}_k = \bar{\boldsymbol{\mu}}_k + K_k\left(\mathbf{z}_k - h\left(\bar{\boldsymbol{\mu}}_k\right)\right)$ and $\Sigma_k = (I - K_k H_k)\,\bar{\Sigma}_k$. The additional term in (2) rightly adjusts the mean $\boldsymbol{\mu}_k$ accounting for the fact that the object location is uncertain. Similarly, the additional terms in (3) account for the object uncertainty and scale the posterior covariance accordingly.

III. Collision Probability

Let $\mathcal{R}$ represents the collection of points that form the rigid-body robot. Similarly, let $\mathcal{S}$ represent the set of all points occupied by a rigid-body obstacle. Thus, a collision occurs if $\mathcal{R} \cap \mathcal{S} \neq \{\phi\}$. In this work we assume spherical geometries for $\mathcal{R}$ and $\mathcal{S}$ with radii r_1 and s_1, receptively and we denote the corresponding center of masses as by $\mathbf{x}_k$ and $\mathbf{s}_k$, receptively. By abuse of notation we will use $\mathbf{x}_k$ and $\mathbf{s}_k$ equivalently to $\mathcal{R}$ and $\mathcal{S}$. The collision condition will be written in terms of the center of mass as $\mathcal{C}_{\mathbf{x}_k,\mathbf{s}_k} : \mathcal{R} \cap \mathcal{S} \neq \{\phi\}$. Let us now consider an obstacle at any given time instant, distributed according to the Gaussian $\mathbf{s}_k \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{s}_k}, \Sigma_{\mathbf{s}_k}\right)$, where $\boldsymbol{\mu}_{\mathbf{s}_k}$ represents the mean and $\Sigma_{\mathbf{s}_k}$ the uncertainty in the estimation of the object. Then the probability of collision is given by

$$P\left(\mathcal{C}_{\mathbf{x}_k,\mathbf{s}_k}\right) = \int_{\mathbf{x}_k}\int_{\mathbf{s}_k} I_c(\mathbf{x}_k, \mathbf{s}_k) p(\mathbf{x}_k, \mathbf{s}_k) \quad (4)$$

where I_c is an indicator function that evaluates to 1 when $\mathcal{R} \cap \mathcal{S} \neq \{\phi\}$ and otherwise 0.

[5], [14] approximate the integral in (4) as $Vp(\mathbf{x}_k, \mathbf{s}_k)$, where V is the volume occupied by the robot. For computing $p(\mathbf{x}_k, \mathbf{s}_k)$, they first assume a distribution centered around the obstacle with the covariance being the sum of the robot and obstacle location uncertainties. Then the density $p(\mathbf{x}_k, \mathbf{s}_k)$ is computed by assuming a constant robot location. Du Toit and Burdick uses the robot center, whereas in [14] the maximum density on the surface of the robot is used to obtain an upper bound. However, the approximation is valid only when the robot radius is negligible and the obstacle size is relatively small compared to their location uncertainties.

To do away with this approximation, we formulate the above problem by considering an alternative approach. Since the robot and the obstacles are assumed to be spherical objects, the collision constraint can then be written as

$$\left\|\mathbf{x}_k - \mathbf{s}_k\right\|^2 \leq (r_1 + s_1)^2 \quad (5)$$

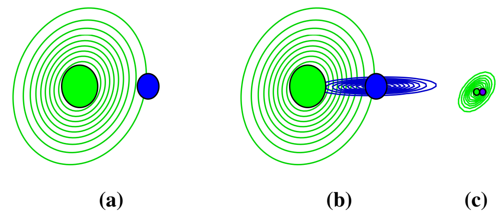

Fig. 1: (a) The robot state is known perfectly, however the obstacle location is uncertain. (b) Robot state uncertainty is considered (contours in blue). (c) A point-like robot and obstacle are considered.

where (as before) $\mathbf{x}_k$ and $\mathbf{s}_k$ are the random vectors that denote the robot and obstacle pose, respectively. Let us denote by $\boldsymbol{w} = \mathbf{x}_k - \mathbf{s}_k$ the difference between the two random variables. Then we know that $\boldsymbol{w}$ is also a Gaussian, distributed as $\boldsymbol{w} \sim \mathcal{N}\left(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{\mathbf{s}_k}, \Sigma_k + \Sigma_{\mathbf{s}_k}\right)$. The collision constraint in (5) can now be written as

$$\boldsymbol{y} = \|\boldsymbol{w}\|^2 = \boldsymbol{w}^T\boldsymbol{w} \leq (r_1 + s_1)^2 \quad (6)$$

where $\boldsymbol{y}$ is a random vector distributed according to the squared L_2-norm of $\boldsymbol{w}$. Now, given the probability density function (pdf) of $\boldsymbol{y}$, the collision constraint reduces to solving the integral

$$P\left(\mathcal{C}_{\mathbf{x}_k,\mathbf{s}_k}\right) = \int_0^{(r_1+s_1)^2} p(y) \quad (7)$$

where $p(y) = P_{\boldsymbol{y}}(\boldsymbol{y} = y)$ is the pdf of $\boldsymbol{y}$. It is noteworthy that the above integral is the cumulative distribution function (cdf) of $\boldsymbol{y}$, that is, $P\left(\mathcal{C}_{\mathbf{x}_k,\mathbf{s}_k}\right) = F_{\boldsymbol{y}}(y)$, where $F_{\boldsymbol{y}}(y)$ denotes the cdf. Moreover, the left hand side of (6), is a quadratic form in the random variables of $\boldsymbol{w}$ and the cdf is computed as

$$F_{\mathbf{y}}(y) = p(\mathbf{y} \leq y) = \sum_{k=0}^{\infty}(-1)^k c_k \frac{y^{\frac{n}{2}+k}}{\Gamma\left(\frac{n}{2}+k+1\right)} \quad (8)$$

The above infinite series converges within the first few terms and we refer the readers to [21] for the definition of a quadratic form in random variables, its cdf and the proof of convergence. Further, it can be shown that to achieve a truncation error $E \leq \delta$, for an $\epsilon-$safe configuration[1], $k = O\left(\log \frac{\delta\rho}{y(1-\epsilon)}\right)$ iterations are required[2]. We note that for each obstacle, the complexity is increased by this factor. As shown in the next section, the computation time is of the order of few milliseconds and hence the approach can be used for fast online planning.

IV. Comparison to Other Approaches

In this section we compare our approach for collision probability computations with that of Park *et al.* [14] and [5]. Numerical integration of (4) gives the exact value and we use the same to validate our approach. Three different cases are considered (see Fig. 1). The solid green circle denotes an obstacle of radius 0.5m and its corresponding uncertainty contours are shown as green circles. The solid blue circle

[1] A robot configuration $\mathbf{x}_k$ is an $\epsilon-$safe configuration with respect to an obstacle s, if the probability of collision is such that, $p\left(\mathcal{C}_{\mathbf{x}_k,\mathbf{s}}\right) \leq 1 - \epsilon$.

[2] $y = (r_1 + s_1)^2$, ρ depends on robot and obstacle state covariance.

Case	Algorithm	CP	Computation time (s)	Feasible
(a)	Numerical integral	4.62%	0.8896 ± 0.0356	Yes
	[5]	5.84%	0.0026 ± 0.0003	Yes
	[14]	33.26%	0.2367 ± 0.2081	No
	Our approach	4.61%	0.0232 ± 0.0024	Yes
(b)	Numerical integral	8.25%	1.2309 ± 0.0298	Yes
	[5]	14.20%	0.0021 ± 0.0001	No
	[14]	36.31%	0.2108 ± 0.3067	No
	Our approach	8.22%	0.0208 ± 0.0021	Yes
(c)	Numerical integral	14.82%	1.2450 ± 0.0301	No
	[5]	0.46%	0.0019 ± 0.0004	Yes
	[14]	0.61%	0.3145 ± 0.4610	Yes
	Our approach	14.83%	0.0271 ± 0.0087	No

TABLE I: Comparison of collision probability (CP) approaches.

denotes a robot of radius 0.3m with the blue circles showing the Gaussian contours. We define a collision probability threshold of 0.1 and the collision probability values and the computation times are provided in Table I. In Fig. 1(a), the robot position known with high certainty and our approach computes collision probability as 4.61% and hence the given configuration is feasible. The numerical integral provides the actual value and as seen in Table I, it is computed to be 4.62%, thus proving the exactness of our method. However, the collision probability computed as given in [14] is 33.26% (almost seven times our value), predicting the configuration to be unfeasible. The approach in [5] also gave a feasible value of 5.84%. In Fig. 1(b), there is robot uncertainty along the horizontal axis and the collision probability computed using our approach is 8.22%. The actual value is computed to be 8.25%. The values computed using the approaches in [14], [5] are 36.31% (4.5 times our value) and 14.20%, respectively.

The approach of Park *et al.* [14] and [5] assumes that the robot radius is very small. We also compute the collision probabilities for a robot and an obstacle with radius 0.05m each, where the robot and the obstacle are touching each other (Fig. 1(c)). The obstacle location is also much more certain, with the uncertainty reduced by 97% as compared to cases in Fig. 1(a),(b). Actual value obtained using numerical integral is 14.82%. The probability of collision computed using our approach is 14.83%, rightly predicting it to be unfeasible. The approach in [14] computed the value as 0.61% and using [5] a value of 0.46% is obtained. Thus, using the approaches in [5], [14] would lead to collision as it predicts the configuration to be feasible. Our approach computes the exact probability of collision and outperforms the approaches in [5], [14].

V. Conclusion

We present a BSP framework that incorporate object uncertainties while localizing the robot and derive the EKF update equations for the same. Unlike previous approaches that compute approximate upper bounds we derive an exact expression for computing the collision probability. We validate our approach using numerical integration and provide comparison to other approaches.

References

[1] Ali-Akbar Agha-Mohammadi, Suman Chakravorty, and Nancy M Amato. FIRM: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *The International Journal of Robotics Research*, 33(2):268–304, 2014.

[2] Georges S Aoude, Brandon D Luders, Joshua M Joseph, Nicholas Roy, and Jonathan P How. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Autonomous Robots*, 35(1):51–76, 2013.

[3] Brian Axelrod, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Provably safe robot navigation with obstacle uncertainty. *The International Journal of Robotics Research*, 37(13-14):1760–1774, 2018.

[4] Xu Chu Ding, Alessandro Pinto, and Amit Surana. Strategic planning under uncertainties via constrained markov decision processes. In *IEEE International Conference on Robotics and Automation*, pages 4568–4575, 2013.

[5] Noel E Du Toit and Joel W Burdick. Probabilistic collision checking with chance constraints. *IEEE Transactions on Robotics*, 27(4):809–815, 2011.

[6] Caelan Reed Garrett, Chris Paxton, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Dieter Fox. Online replanning in belief space for partially observable task and motion problems. *arXiv preprint arXiv:1911.04577*, 2019.

[7] Astghik Hakobyan, Gyeong Chan Kim, and Insoon Yang. Risk-Aware Motion Planning and Control Using CVaR-Constrained Optimization. *IEEE Robotics and Automation Letters*, 4(4):3924–3931, 2019.

[8] Lucas Janson, Edward Schmerling, and Marco Pavone. Monte Carlo motion planning for robot trajectory optimization under uncertainty. In *Robotics Research*, pages 343–361. Springer, 2018.

[9] Ashkan M Jasour and Brian C Williams. Risk contours map for risk bounded motion planning under perception uncertainties. *Robotics: Science and Systems*, 2019.

[10] Leslie Pack Kaelbling and Tomás Lozano-Pérez. Integrated task and motion planning in belief space. *The International Journal of Robotics Research*, 32(9-10):1194–1227, 2013.

[11] Hanna Kurniawati and Vinay Yadav. An Online POMDP Solver for Uncertainty Planning in Dynamic Environment. In *Robotics Research: The 16th International Symposium ISRR*, pages 611–629, 2016.

[12] Alain Lambert, Dominique Gruyer, and Guillaume Saint Pierre. A fast Monte Carlo algorithm for collision probability estimation. In *10th IEEE International Conference on Control, Automation, Robotics and Vision*, pages 406–411, 2008.

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

[14] Chonhyon Park, Jae S Park, and Dinesh Manocha. Fast and bounded probabilistic collision detection for high-DOF trajectory planning in dynamic environments. *IEEE Transactions on Automation Science and Engineering*, 15(3):980–991, 2018.

[15] Shashank Pathak, Antony Thomas, and Vadim Indelman. A unified framework for data association aware robust belief space planning and perception. *The International Journal of Robotics Research*, 37(2-3):287–315, 2018.

[16] Sachin Patil, Jur Van Den Berg, and Ron Alterovitz. Estimating probability of collision for safe motion planning under Gaussian motion and sensing uncertainty. In *IEEE International Conference on Robotics and Automation*, pages 3238–3244, 2012.

[17] Samuel Prentice and Nicholas Roy. The belief roadmap: Efficient planning in belief space by factoring the covariance. *The International Journal of Robotics Research*, 28(11-12):1448–1465, 2009.

[18] Dorsa Sadigh and Ashish Kapoor. Safe control under uncertainty with probabilistic signal temporal logic. *Robotics: Science and Systems*, 2016.

[19] Luke Shimanuki and Brian Axelrod. Hardness of 3D Motion Planning Under Obstacle Uncertainty. *Workshop on Algorithmic Foundations of Robotics*, 2018.

[20] Antony Thomas, Fulvio Mastrogiovanni, and Marco Baglietto. Task-Motion Planning for Navigation in Belief Space. In *The International Symposium on Robotics Research*, 2019.

[21] Antony Thomas, Fulvio Mastrogiovanni, and Marco Baglietto. ———————————. In *Intelligent Service Robotics (submitted)*, 2020.

[22] Antony Thomas, Fulvio Mastrogiovanni, and Marco Baglietto. Towards Multi-Robot Task-Motion Planning for Navigation in Belief Space. In *European Starting AI Researchers' Symposium*. CEUR, 2020.

[23] Jur Van Den Berg, Sachin Patil, and Ron Alterovitz. Motion planning under uncertainty using iterative local optimization in belief space. *The International Journal of Robotics Research*, 31(11):1263–1278, 2012.