Planar Uncertainty Propagation and a Probabilistic Algorithm for Interception

Andrew W. Long, Kevin C. Wolfe, and Gregory S. Chirikjian

Abstract. Many robotics applications involve motion planning with uncertainty. In this paper, we focus on path planning for planar systems by optimizing the probability of successfully arriving at a goal. We approach this problem with a modified version of the Path-of-Probability (POP) algorithm. We extend the POP algorithm to allow for a moving target and to optimize the number of steps to reach the goal. One tool that we develop in this paper to increase efficiency of the POP algorithm is a second order closed-form uncertainty propagation formula. This formula is utilized to quickly propagate the mean and covariance of nonparametrized distributions for planar systems. The modified POP algorithm is demonstrated on a simple rolling disc example with a moving goal.

1 Introduction

A fundamental problem in robotics is the question *what is the best way to get from here to there?* This problem is known as *motion planning*, in which one tries to optimize the path for a robot to arrive at a goal. Motion planning is important for many applications such as autonomous cars, unmanned aerial vehicles, manufacturing robots, domestic robots and robotic surgery manipulators. If we had perfect knowledge of the robot and its environment, we could simply integrate the velocity commands from a known starting point to obtain an exact position of the robot at some future time. However, uncertainty in actuation, models and the environment will inevitably produce errors between the actual position and this integration making it difficult to follow a deterministic path. As a result, a more pertinent question is *what path has the greatest probability of reaching the goal?* A strategy that we

Andrew W. Long · Kevin C. Wolfe · Gregory S. Chirikjian
Johns Hopkins University,
Baltimore, MD
e-mail: {awlong,kevin.wolfe,gregc}@jhu.edu

E. Frazzoli et al. (Eds.): *Algorithmic Foundations of Robotics X*, STAR 86, pp. 459–474.
DOI: 10.1007/978-3-642-36279-8_28

employ in this paper is to represent all possible poses (position and orientation) with probability density functions (pdfs). The key then is to maintain and update these pdfs as the robot moves along a path. With this strategy, we can then begin to choose paths with higher probability of reaching a desired goal pose.

The main focus of this paper is with regards to motion planning for planar systems with uncertainty. Generally, planners try to minimize the execution time or minimize the final covariance. In this paper, we are concerned with maximizing the probability of successfully arriving at a goal. This objective has been considered by [1] and [2] with the stochastic motion road map (SRM). In our approach, we utilize a modified version of the Path-of-Probability (POP) algorithm to maximize this probability of success. In previous POP algorithms ([10], [18]), there was an assumption that the goal pose was fixed in time and that the planner knew how many discrete steps should be taken to reach the goal. We modify this algorithm to allow for a moving target and to optimize the number of steps.

Our algorithm relies on the ability to efficiently propagate uncertainty along a path. A tool that we develop in this paper is a set of closed-form recursive formulas for planar systems that can be utilized to quickly propagate the mean and covariance of nonparameterized distributions. Our formulas are nonparametric since they do not assume a specific distribution. The formulas are different than those of the Kalman methods because we use the theory of Lie algebras and Lie groups and propagate the error to second order.

The remainder of this section consists of related work and an outline of the paper.

1.1 Related Work

In the last two decades, there has been significant research into motion planning with uncertainty. Motion planning in general has been studied extensively as demonstrated in [14] and [9]. Several techniques that specifically incorporate uncertainty include back propagation [15], Linear Quadratic Gaussian (LQG) motion planning [22], stochastic dynamic programming [4], [3] and partially observable Markov decision processes (POMDPs) [21]. A bottleneck of these approaches is the computation time, especially as the length of the path increases. Another approach uses an extended space consisting of poses and covariances, allowing utilization of standard search techniques such as A^* or sampling-methods [6], [13], [19]. In the area of sampling-based methods, the well-known rapidly exploring random trees (RRTs) has been extended to the belief space in a search algorithm known as RRBT [5].

One problem that must be considered in these algorithms is how to propagate the uncertainty. Small errors can be propagated for linear systems with the Kalman filter [12] or for nonlinear systems with Jacobian-based methods such as in the classic work of [20]. For larger errors with unknown distributions, the standard technique is the particle filter [21], which consists of randomly selecting a large number of particles from the distribution and propagating these particles in time. The robot's true position has a higher probability of occurring in the denser areas. However, the particle filter may require a large number of samples and can become computationally expensive. Recently, Wang and Chirikjian derived second-order propagation formulas for the Euclidean motion group $SE(3)$ [23] which can be used to efficiently propagate uncertainty without sampling. The propagation work presented in this paper is similar to their work but is focused on the planar motion group $SE(2)$.

The POP algorithm first appeared for manipulator arms to solve inverse kinematics problems by representing the reachable workspace with probability densities [10]. The idea of using reachable-state density functions was then applied to trajectory planning problems using the theory of Lie groups in [16]. A problem with the approach in [16] is that it is computationally expensive and is difficult to implement in real-time. With the closed-form propagation formulas for $SE(3)$ from [23], the POP algorithm was implemented more quickly and was applied to needle-steering in [18]. In this paper, we take a similar approach as [18] but remove the requirement of a fixed goal pose and a known number of steps to reach the goal.

1.2 Outline

In Sec. 2, we review important concepts and definitions from rigid-body motions, probability and statistics. We derive the second-order error propagation formulas for the planar motion group in Sec. 3. We describe the modified POP algorithm in Sec. 4. In Sec. 5, we demonstrate the propagation formulas and the POP algorithm on a simple rolling disc example. Our conclusions are discussed in Sec. 6.

2 Terminology and Notation

2.1 Rigid-Body Motions

The elements of the planar special Euclidean group, $SE(2)$, are the semidirect product of the plane, $\mathbb{R}^2$ with the special orthogonal group, $SO(2)$. $G = SE(2)$ is an example of a *matrix Lie group* where G is a set of square 3 x 3 square matrices and the group operation $\circ$ is matrix multiplication. For a full review of Lie groups see [17] and [7]. The elements of $SE(2)$ and their inverses are given respectively as

$$g = \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \quad \text{and} \quad g^{-1} = \begin{pmatrix} R^T & -R^T\mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \tag{1}$$

where R is the rotational part, $\mathbf{t}$ is the translational part and $\mathbf{0}$ is a zero-vector.

In this paper, we also make use of $se(2)$, the Lie algebra associated with $SE(2)$. For a vector $x = [v_1, v_2, \alpha]^T$, we define the $\wedge$ and $\vee$ operators for $se(2)$ by

$$\hat{\mathbf{x}} = X = \begin{pmatrix} 0 & -\alpha & v_1 \\ \alpha & 0 & v_2 \\ 0 & 0 & 0 \end{pmatrix} \quad \text{and} \quad X^\vee = \mathbf{x}, \tag{2}$$

to allow us to map from $\mathbb{R}^3$ to $se(2)$ and back. We can obtain the group elements of $SE(2)$ by applying the matrix exponential $\exp(\cdot)$ to elements of the Lie algebra to give the following [7]

$$g(\mathbf{x}) = \exp(X) = \begin{pmatrix} \cos\alpha & -\sin\alpha & t_1 \\ \sin\alpha & \cos\alpha & t_2 \\ 0 & 0 & 1 \end{pmatrix}, \quad \text{where} \tag{3}$$

$$\mathbf{t} = \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} = \frac{1}{\alpha} \begin{pmatrix} \sin\alpha & -(1-\cos\alpha) \\ (1-\cos\alpha) & \sin\alpha \end{pmatrix} \mathbf{v}, \tag{4}$$

and where $\mathbf{v} = [v_1, v_2]^T$. Since we use the matrix exponential to obtain the elements of $SE(2)$, we refer to the vector $\mathbf{x}$ as exponential coordinates. With the matrix logarithm $\log(\cdot)$, we can obtain the vector $\mathbf{x}$ from a group element $g \in SE(2)$ with

$$\mathbf{x} = (\log(g))^\vee. \tag{5}$$

We define two adjoint operators $Ad(g)$ and $ad(X)$ to satisfy

$$Ad(g)\mathbf{x} = \left(\log(g \circ \exp(X) \circ g^{-1})\right)^\vee, \quad \text{and} \quad ad(X)\mathbf{y} = ([X,Y])^\vee \tag{6}$$

where $g \in SE(2)$, $(X,Y) \in se(2)$ and $[X,Y] = XY - YX$ is the Lie bracket. These two adjoint operators are related by

$$Ad(\exp(X)) = \exp(ad(X)). \tag{7}$$

The adjoint matrices for $SE(2)$ and $se(2)$ are given explicitly as [7]

$$Ad(g) = \begin{pmatrix} R & M\mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}, \quad \text{and} \quad ad(X) = \begin{pmatrix} -\alpha M & M\mathbf{v} \\ \mathbf{0}^T & 0 \end{pmatrix} \quad \text{where} \quad M = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \tag{8}$$

The special Euclidean group $SE(2)$ is a connected unimodular matrix Lie group [17]. As a result, for any function $f: G \to \mathbb{R}$ for which the integral $\int_G f(g)dg$ exists the following properties hold for any fixed $h \in G$

$$\int_G f(g)dg = \int_G f(h \circ g)dg = \int_G f(g \circ h)dg = \int_G f(g^{-1})dg. \tag{9}$$

For a discussion of integration on groups see [7]. These properties of unimodular matrix Lie groups will be used extensively for changing coordinates in this paper.

2.2 Means, Covariances and Gaussians

A probability density function (pdf) for a Lie group $(G, \circ)$ can be defined by

$$f(\mathbf{g}) \geq 0 \; \forall \, \mathbf{g} \in G \quad \text{and} \quad \int_G f(\mathbf{g}) dg = 1.$$

The definitions of the mean and covariance can be naturally extended to matrix Lie groups as in [23]. Given a group G, the mean $\mu \in G$ of a pdf $f(g)$ satisfies

$$\int_G \left[\log^\vee \left(\mu^{-1} \circ g\right)\right] f(g) dg = \mathbf{0}. \tag{10}$$

The covariance about the mean is defined as

$$\Sigma = \int_G \log^\vee(\mu^{-1} \circ g)[\log^\vee(\mu^{-1} \circ g)]^T f(g) dg. \tag{11}$$

A Gaussian in these coordinates is then given by

$$f(g; \mu, \Sigma) = \frac{1}{c(\Sigma)} \exp\left(-\tfrac{1}{2}[\log^\vee(\mu^{-1} \circ g)]^T \Sigma^{-1} [\log^\vee(\mu^{-1} \circ g)]\right) \tag{12}$$

where $c(\Sigma)$ is a normalizing factor. Although there is no closed-form formula for $c(\Sigma)$ in this Gaussian, when $||\Sigma||$ is small

$$c(\Sigma) \approx (2\pi)^{n/2} |\det \Sigma|^{\frac{1}{2}}. \tag{13}$$

We use this normalizing factor to allow us to use a closed-form formula for (12).

3 Propagation of Mean and Covariance of Pdfs on SE(2)

In this section, we are interested in developing closed-form formulas that can be used to propagate the mean and covariance of pdfs on the special Euclidean group $SE(2)$. Imagine that a robot has the option of performing Action 1 which has a pdf $f_1(g; \mu_1, \Sigma_1)$ or Action 2 which has a pdf $f_2(g; \mu_2, \Sigma_2)$. Given only the means (μ_1, μ_2) and the covariances (Σ_1, Σ_2) of these two pdfs, we seek the mean and covariance of the resulting distribution of performing Action 1 then Action 2. This new distribution can be represented with the convolution of the two pdfs.

The convolution of two pdfs on groups is defined as [8]

$$(f_1 * f_2)(g) = \int_G f_1(h) f_2(h^{-1} \circ g) dh. \tag{14}$$

Let $\rho_i(g)$ be a unimodal pdf with mean at the identity. Then $\rho_i(\mu_i^{-1} \circ g)$ is a distribution with the same shape centered at μ_i. We can then rewrite the definition of the convolution in (14) as

$$(f_1 * f_2)(g) = \int_G \rho_1(\mu_1^{-1} \circ h)\rho_2(\mu_2^{-1} \circ h^{-1} \circ g)dh. \quad (15)$$

If we make the change of coordinates $h \to \mu_1 \circ \mu_2 \circ k \circ \mu_2^{-1}$, then

$$(f_1 * f_2)(g) = \int_G \rho_1^{\mu_2}(k)\rho_2(k^{-1} \circ \mu_2^{-1} \circ \mu_1^{-1} \circ g)dk \quad (16)$$

$$= (\rho_1^{\mu_2} * \rho_2)(\mu_2^{-1} \circ \mu_1^{-1} \circ g), \quad (17)$$

where $\rho_1^{\mu_2}(g) = \rho_1(\mu_2 \circ g \circ \mu_2^{-1})$ is a transformed version of $\rho_1(g)$ with its argument conjugated by μ_2. It can be shown that this pdf $\rho_1^{\mu_2}(g)$ has its mean at the identity and covariance given by

$$\Sigma_{\rho_1^{\mu_2}} = Ad(\mu_2)^{-1}\Sigma_{\rho_1}Ad(\mu_2)^{-T}. \quad (18)$$

In [23], 'order' is defined to be the number of terms that were kept in the BCH expansion (see Appendix). We seek the second-order error propagation formulas using this same definition. No assumptions are made here about $f_i(g)$ except that

$$||\Sigma_i|| = O(\nu) \quad \text{where } \nu << 1 \text{ and that } ||\log \mu_i|| = O(1). \quad (19)$$

The condition in (19) means that the density is "highly focused."

Theorem 1. *If $f_i(g)$ is a pdf on $SE(2)$ that has mean μ_i and covariance Σ_i about the mean for $i = 1,2$, then to second order, the mean and covariance of $(f_1 * f_2)(g)$ are*

$$\mu_{1*2} = \mu_1 \circ \mu_2 \quad \textit{and} \quad (20)$$

$$\Sigma_{1*2} = A + B + F(A,B) \quad \textit{where} \quad (21)$$

$$F(A,B) = \frac{1}{4}C(A,B) + \frac{1}{12}\left[A''B + (A''B)^T + B''A + (B''A)^T\right],$$
$$A = Ad(\mu_2^{-1})\Sigma_1 Ad^T(\mu_2^{-1}), \quad \textit{and} \quad B = \Sigma_2,$$

where the double-prime operator $''$ takes a matrix A and rearranges it to obtain

$$A'' = \begin{pmatrix} -a_{33} & 0 & a_{31} \\ 0 & -a_{33} & a_{32} \\ 0 & 0 & 0 \end{pmatrix}$$

and $C(A,B)$ is computed from

$$C(A,B) = \begin{pmatrix} c_{11} & c_{12} & 0 \\ c_{12} & c_{22} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \textit{where} \quad \begin{aligned} c_{11} &= b_{33}a_{22} - b_{32}a_{23} - b_{23}a_{32} + b_{22}a_{33} \\ c_{12} &= -b_{33}a_{21} + b_{31}a_{23} + b_{23}a_{31} - b_{21}a_{33} \\ c_{22} &= b_{33}a_{11} - b_{31}a_{13} - b_{13}a_{31} + b_{11}a_{33}. \end{aligned}$$

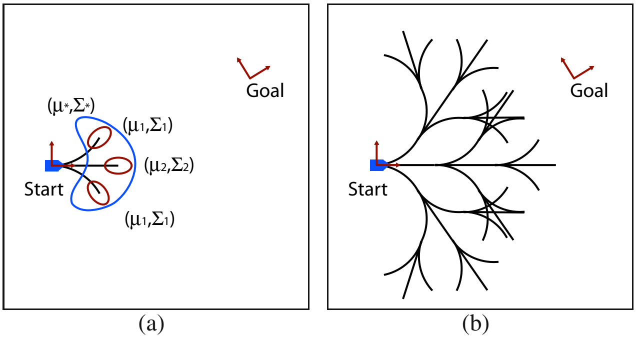


Fig. 1 (a) shows a planar robot with three noisy path options for a single step with means μ_i and covariances Σ_i. The combined distribution of all possible trajectories for a single step has mean μ_* and covariance Σ_*. (b) depicts an ideal tree of possible poses after three steps.

The proof of this theorem is similar to that in [23] except that it is for the planar case. The derivation of the covariance expression is shown in the Appendix.

4 Path-of-Probability (POP) Algorithm

We are interested in finding a path consisting of N intermediate steps that maximizes the probability of moving from a starting pose to a final pose. We assume that we have a finite set ζ of small trajectories that we can choose to use for a given step. For example, as shown in Fig. 1(a), we can choose from three different segments. Due to uncertainty, each segment has a distribution of possible poses with mean μ_i and covariance Σ_i as illustrated by the red ellipses. We can then add multiple steps to begin to cover the environment as shown in Fig. 1(b) for three such steps. It can be computationally expensive to keep track of all possible trajectories and to propagate the uncertainty along each of these trajectories, especially as the number of choices in ζ and number of steps increases. In this section, we present the Path-Of-Probability (POP) algorithm, which is a computationally efficient way of determining which option we should choose for each step. The POP algorithm has been applied previously in slightly different manners for manipulator arms [10] and needle steering [18].

Let $g_i \in \zeta$ be the mean of the selected trajectory for the ith step. Suppose that the first $(i-1)$ steps have been selected. In the POP algorithm, we want to select the step $g_i \in \zeta$ that has the highest probability of reaching the goal with the remaining $(N-i)$ steps. This can be written as

$$g_i = \arg\max_{g \in \zeta} f((g_1 \cdots g_{i-1} \circ g)^{-1} \circ g_{goal}; \mu_{N-i* \cdots *N}, \Sigma_{1* \cdots *N}), \tag{22}$$

where ζ is the set of possible moves. If we have a way of exactly measuring ($\Sigma_{act} = 0$) the actual pose g_{act} before choosing a step, then the formulation is

$$g_i = \arg\max_{g \in \zeta} f((g_{act} \circ g)^{-1} \circ g_{goal}; \mu_{N-i*\cdots*N}, \Sigma_{N-i*\cdots*N}). \quad (23)$$

Another variant of this algorithm could be to fuse uncertainty in measurements with the uncertainty in propagation. Here we restrict our attention to the problems in (22) and (23). The key in the POP algorithm is to calculate the convolved means and covariances of the remaining steps quickly.

In this paper, we assume that we do not have any prior knowledge of the type of path we would like to use for the remaining steps. Therefore, we combine all the individual distributions for a single step into a new distribution with mean μ_* and covariance Σ_* as shown with the blue contour in Fig. 1(a). We can then use the propagation formulas in (20) and (21) recursively to propagate the distribution for the remaining steps. In a sense, this allows us to calculate the probability of arriving at the goal pose given all possible remaining trajectories.

We can sample data points from each individual distribution then use the following formulas to estimate the mean and covariance for a single step

$$\mu_* = \mu_* \circ \exp\left(\frac{1}{Q}\sum_{j=1}^{Q} \log(\mu_*^{-1} \circ g_j)\right) \quad (24)$$

$$\Sigma_* = \frac{1}{Q}\sum_{j=1}^{Q} \log^\vee(\mu_*^{-1} \circ g_j)[\log^\vee(\mu_*^{-1} \circ g_j)]^T \quad (25)$$

where Q is the total number of data points from all trajectories in a single step. Depending on the number of samples, this can become computationally expensive, especially since the mean equation is recursive. However, if the individual distributions are highly focused around the individual means, we observed that we can obtain approximately the same results if we use

$$\mu_* = \mu_* \exp\left(\frac{1}{Z}\sum_{j=1}^{Z} \log(\mu_*^{-1} \circ \mu_j)\right) \quad \text{and} \quad (26)$$

$$\Sigma_* = \frac{1}{Z}\sum_{j=1}^{Z} \log^\vee(\mu_*^{-1} \circ \mu_j)[\log^\vee(\mu_*^{-1} \circ \mu_j)]^T \quad (27)$$

where Z is the total number of possible trajectories in one step. With the latter method, we do not need to sample from the individual distributions, we only need the means $\{\mu_i\}$. If the set ζ is known beforehand, the mean μ_* and Σ_* can be calculated numerically before beginning the POP algorithm.

The previous versions of the POP algorithm in [10] and [18] assume that the goal is fixed through out time and that the number of steps to reach the goal is known. We can remove these assumptions by using a two-part POP algorithm. In the first part, for step i with a maximum number of steps N, we estimate the best number of remaining steps M to reach the target with

$$M = \underset{m \in 1 \ldots N-i}{\arg\max}\, f((g_{act})^{-1} \circ g_{goal}((i+m)T); \mu_{1*\cdots*m}, \Sigma_{1*\cdots*m}) \tag{28}$$

where we use m convolutions of (μ_*, Σ_*) and T is the time of a single step. Note that the goal pose is evaluated at different discrete times in this part. After selecting the best number of steps to take, we choose a trajectory for a single step with

$$g_i = \underset{g \in \zeta}{\arg\max}\, f((g_{act} \circ g)^{-1} \circ g_{goal}((i+M)T); \mu_{i+2*\cdots*(M+i)}, \Sigma_{i*\cdots*(i+M)}). \tag{29}$$

By breaking the problem into a two-part algorithm, we only have to evaluate $(N - i + Z)$ pdfs for each step i. We repeat this two-step process for each step until the current pose is within some tolerance of the goal pose. In the next section, we demonstrate this POP algorithm on a simple numerical example.

5 Example: Stochastic Kinematic Disc

In this section, we are interested in numerically testing a couple key aspects of this paper with the simple example of a disc that can roll but not slip in the plane. First, we verify that a Gaussian in exponential coordinates is a good fit. Second, we verify that the second-order propagation formulas can be used to calculate the mean and covariance of the convolution of two distributions. Finally, we demonstrate the POP algorithms outlined above.

The governing stochastic differential equation for this example is given by

$$(g^{-1}\dot{g})^\vee dt = \mathbf{h}\,dt + H\,d\mathbf{w} = \begin{pmatrix} v \\ 0 \\ \omega \end{pmatrix} dt + \begin{pmatrix} \sqrt{D_v} & 0 \\ 0 & 0 \\ 0 & \sqrt{D_\omega} \end{pmatrix} \begin{pmatrix} dw_v \\ dw_\omega \end{pmatrix}. \tag{30}$$

where $(g^{-1}\dot{g})^\vee$ are body fixed velocities [7], v and ω are translational and angular velocities, respectively, D_v and D_ω are noise coefficients and $d\mathbf{w} = [dw_v, dw_\omega]^T$ are unit-strength Wiener processes.

For small diffusion, the mean and covariance defined in (10) and (11) can be approximated with [18]

$$\mu(t) = \exp\left(\int_0^t \hat{\mathbf{h}}\,d\tau\right) \quad \text{and} \quad \Sigma(t) = \int_0^t Ad^{-1}(\mu(\tau))HH^T Ad^{-T}(\mu(\tau))d\tau, \tag{31}$$

which are essentially the deterministic path from integrating $\mathbf{h}dt$ and covariance propagation with only the A and B terms in (21) with infinitesimal steps. For an ideally straight trajectory ($\omega = 0$), the mean and covariance at time t is given by

$$\mu(t) = \begin{pmatrix} 1 & 0 & vt \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma(t) = \begin{pmatrix} D_v t & 0 & 0 \\ 0 & \frac{1}{3} D_\omega v^2 t^3 & \frac{1}{2} D_\omega v t^2 \\ 0 & \frac{1}{2} D_\omega v t^2 & D_\omega t \end{pmatrix}. \tag{32}$$

For a trajectory with constant angular and translational velocity, the disc ideally moves along a circular path. The mean and covariance of this case are

$$\mu(t) = \begin{pmatrix} \cos(\omega t) & -\sin(\omega t) & \frac{v}{\omega}\sin(\omega t) \\ \sin(\omega t) & \cos(\omega t) & \frac{v}{\omega}(1-\cos(\omega t)) \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma(t) = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{pmatrix}, \tag{33}$$

where

$$\begin{aligned} \sigma_{11} &= \tfrac{1}{4\omega^3}\left[2t\omega(3D_\omega v^2 + D_v\omega^2) - 8D_\omega v^2 \sin(\omega t) + (D_w v^2 + D_v \omega^2)\sin(2\omega t)\right], \\ \sigma_{12} &= \tfrac{1}{\omega^3}\left[D_\omega v^2 - D_v\omega^2 - (D_\omega v^2 + D_v\omega^2)\cos(\omega t)\right]\sin^2(\tfrac{\omega t}{2}), \\ \sigma_{13} &= \tfrac{1}{\omega^2} D_\omega v(\omega t - \sin(\omega t)), \quad \sigma_{22} = \tfrac{1}{4\omega^3}(D_\omega v^2 + D_v\omega^2)(2\omega t - \sin(2\omega t)), \\ \sigma_{23} &= \tfrac{1}{\omega^2} D_\omega v(1-\cos(\omega t)), \quad \text{and} \quad \sigma_{33} = D_\omega t. \end{aligned}$$

To verify that the integral version of the means and covariances are accurate, we numerically integrated the stochastic differential equation in (30) using a modified version of the Euler-Maruyama method [11] with a time step of $dt = 0.001$ for $T = 1$ for 25000 samples with noise levels of $D_v = 0.001$ and $D_\omega = 0.1$. We can calculate an estimate of the actual mean and covariance from (24) and (25), respectively, where Q is the number of sample data points.

For the ideal straight path with $v = 1$ and $\omega = 0$, we obtain from sample data

$$\mu_{\text{data}} = \begin{pmatrix} 1.000 & -0.001 & 1.000 \\ 0.001 & 1.000 & -0.000 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma_{\text{data}} = \begin{pmatrix} 0.001 & -0.000 & -0.000 \\ -0.000 & 0.034 & 0.050 \\ -0.000 & 0.050 & 0.100 \end{pmatrix};$$

from the integral propagation formula,

$$\mu_{\text{prop}} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma_{\text{prop}} = \begin{pmatrix} 0.001 & 0 & 0 \\ 0 & 0.033 & 0.050 \\ 0 & 0.050 & 0.100 \end{pmatrix}.$$

For an ideal circular path with $v = 1$ and $\omega = \frac{\pi}{2}$ we have from data

$$\mu_{\text{data}} = \begin{pmatrix} 0.000 & -1.000 & 0.637 \\ 1.000 & 0.000 & 0.637 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma_{\text{data}} = \begin{pmatrix} 0.010 & 0.012 & 0.023 \\ 0.012 & 0.021 & 0.041 \\ 0.023 & 0.041 & 0.101 \end{pmatrix}.$$

The integral propagation formulas provide

$$\mu_{\text{prop}} = \begin{pmatrix} 0 & -1 & 0.637 \\ 1 & 0 & 0.637 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma_{\text{prop}} = \begin{pmatrix} 0.010 & 0.013 & 0.231 \\ 0.013 & 0.021 & 0.041 \\ 0.023 & 0.041 & 0.100 \end{pmatrix}.$$

Given the sample data from each example, we can calculate normalized histogram contours and compare them to the contours generated from the Gaussian in (12) marginalized over the heading as shown in Fig. 2. Note that if we used the standard

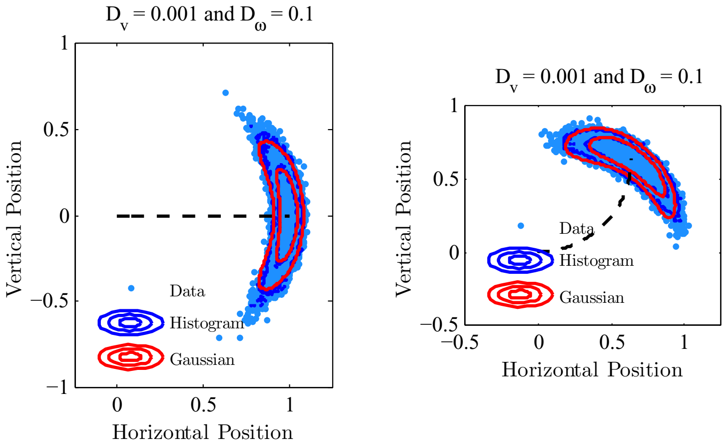


Fig. 2 25,000 sample points after integrating the stochastic differential equation of a noisy rolling disc for a straight path (left) and a circular path (right). Gaussian pdf contours marginalized over the heading from a propagated mean and covariance are compared to the contours from a histogram of the sample points.

formula for a Gaussian in $\mathbb{R}^n$ we would have elliptical contours, which does not fit the data as well.

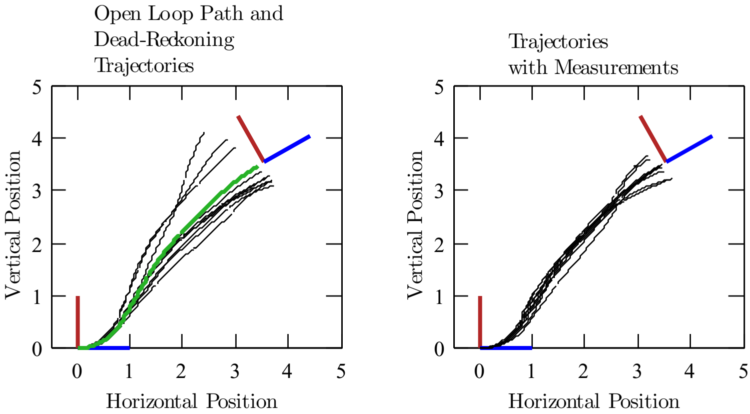
Now let's verify that the second-order propagation formulas are accurate for convolving two distributions. Here we assume the robot follows the straight example followed by the arc example with the same parameters as above. The mean and covariance from sample data following these paths were calculated to be

$$\mu_{\text{data}} = \begin{pmatrix} -0.003 & -1.000 & 1.636 \\ 1.000 & -0.003 & 0.636 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma_{\text{data}} = \begin{pmatrix} 0.145 & 0.083 & 0.137 \\ 0.083 & 0.065 & 0.104 \\ 0.137 & 0.104 & 0.201 \end{pmatrix}.$$

By using the propagation formulas in (20) and (21) with the means and covariances from (32) and (33) we obtain

$$\mu_{\text{prop}} = \begin{pmatrix} 0.000 & -1.000 & 1.638 \\ 1.000 & 0.000 & 0.637 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Sigma_{\text{prop}} = \begin{pmatrix} 0.146 & 0.083 & 0.137 \\ 0.083 & 0.065 & 0.104 \\ 0.137 & 0.104 & 0.200 \end{pmatrix}.$$

Finally, we want to demonstrate the POP algorithm. Given a set ζ of ideally straight and circular trajectories, we can approximate the mean and covariance of each individual trajectory in closed form using the expressions in (32) and (33). Before applying the POP algorithm, we numerically calculate the mean μ_* and covariance Σ_* for a single step. In the remaining examples, we use $D_v = 0.0001$ and $D_\omega = 0.01$ as well as a constant translational velocity $v = 1$ and a range of angular velocities $\omega \in [-\pi/3 : \pi/12 : \pi/3]$ for a step time of $T = 1$. When planning a path open loop



(a) (b)

Fig. 3 (a) shows the open-loop optimal trajectory of the POP algorithm in green. Dead-reckoning trajectories are shown in black executing this sequence of steps. (b) shows ten paths executing the POP algorithm with absolute measurements.

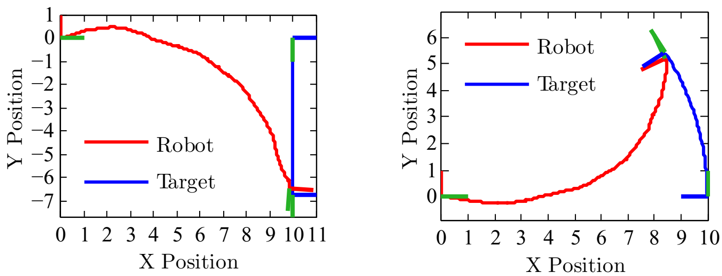


Fig. 4 The modified POP algorithm with absolute measurements implemented for two examples of a moving target and a variable number of steps to intercept

(no measurements) for a fixed goal, we use the POP algorithm with (22) to obtain the desired trajectories for each step. Fig. 3(a) shows an ideal path with five intermediate steps for an example goal pose with several dead-reckoning trajectories that try to follow this trajectory. Note that due to noise in the velocities the trajectories move away from the goal. When we have access to exact full pose information at each step, we select different step segments according to (23). Fig. 3(b) shows several actual trajectories using measurements at the end of each step.

The modified POP algorithm from (28) and (29) with measurements was tested with moving targets and a variable number of steps as shown in the two examples of Fig. 4 for approximately 25 steps with the same parameters as above. For both examples, the velocity of the moving target was chosen to be 50 percent of translational velocity of the robot to enable the robot to intercept it. This example demonstrates the propagation formulas as well as the modified POP algorithm.

6 Conclusion

In this paper, we focus on motion planning with uncertainty for planar systems. We represent all distributions with probability density functions. By using Lie group theory, we are able to derive closed-form expressions to efficiently propagate the mean and covariance of these distributions. We discuss the Path-of-Probability algorithm with these tools and extend this algorithm to include moving targets and a variable number of steps. This algorithm and propagation formulas were tested on a disc that can roll but not slip in the plane. In future work, we plan to introduce obstacles in the planning and to incorporate noisy measurements of both the robot and the target. We are also interested in applying these techniques to more complicated planar systems. Overall, these propagation formulas and the POP algorithm allow for quick and efficient motion planning of uncertain planar systems.

7 Appendix

7.1 Covariance Formula Proof

The proof of the expression for the mean to second order was shown in [23]. The proof of the covariance expression is slightly different than in [23] since we are working with $SE(2)$ instead of $SE(3)$. The covariance of the convolution $(f_1 * f_2)(g)$ about the mean μ_{1*2} is given by

$$\Sigma_{f_1 * f_2} = \int_G \log^\vee(\mu_{1*2}^{-1} \circ g) \left[\log^\vee(\mu_{1*2}^{-1} \circ g)\right]^T (f_1 * f_2)(g) dg. \tag{34}$$

With a change of coordinates $g \to \mu_1 \circ \mu_2 \circ g$, this can be rewritten as

$$\Sigma_{f_1 * f_2} = \int_G \log^\vee(\mu^{-1}_{\rho_1^{\mu_2} * \rho_2} \circ g) \left[\log^\vee(\mu^{-1}_{\rho_1^{\mu_2} * \rho_2} \circ g)\right]^T (\rho_1^{\mu_2} * \rho_2)(g) dg = \Sigma_{\rho_1^{\mu_2} * \rho_2}. \tag{35}$$

Define the covariance C about the identity to be

$$C = \int_G \log^\vee(g) \left[\log^\vee(g)\right]^T f(g) dg. \tag{36}$$

If a pdf $f(g)$ has mean μ and covariance Σ, then expanding (36) with the Baker-Campbell-Hausdorff formula (BCH) (see next subsection)

$$C = \Sigma + \log^\vee(\mu) \left[\log^\vee(\mu)\right]^T + \frac{1}{2} \left(\Sigma ad^T(\log(\mu)) + ad(\log(\mu))\Sigma\right). \tag{37}$$

Since $\mu_{\rho_1^{\mu_2} * \rho_2}$ is second order, this means to second order

$$C_{\rho_1^{\mu_2} * \rho_2} = \Sigma_{\rho_1^{\mu_2} * \rho_2}. \tag{38}$$

The covariance C about the identity for $f(g) = (\rho_1^{\mu_2} * \rho_2)(g)$ is given by

$$C_{\rho_1^{\mu_2} * \rho_2} = \int_G \int_G \log^\vee(g) \left[\log^\vee(g)\right]^T \rho_1^{\mu_2}(h)\rho_2(h^{-1} \circ g) dh\, dg$$

Applying a change of coordinates $g \to h^{-1} \circ g$,

$$C_{\rho_1^{\mu_2} * \rho_2} = \int_G \int_G \log^\vee(h \circ g) \left[\log^\vee(h \circ g)\right]^T \rho_1^{\mu_2}(h)\rho_2(g) dh\, dg.$$

Let $X = \log(h)$ and $Y = \log(g)$. In the expansion of the log terms with the BCH, any terms in linear in X or Y will integrate to zero, which leaves the even terms as

$$\begin{aligned} &\left\{(Z(X,Y))^\vee \left[(Z(X,Y))^\vee\right]^T\right\}_{\text{even}} \\ =&\mathbf{x}\mathbf{x}^T + \mathbf{y}\mathbf{y}^T + \tfrac{1}{4} ad(X)\mathbf{y}\mathbf{y}^T ad^T(X) \\ &+ \tfrac{1}{12} ad(X)ad(X)\mathbf{y}\mathbf{y}^T + \tfrac{1}{12}\mathbf{y}\mathbf{y}^T ad^T(X)ad^T(X) \\ &+ \tfrac{1}{12} ad(Y)ad(Y)\mathbf{x}\mathbf{x}^T + \tfrac{1}{12}\mathbf{x}\mathbf{x}^T ad^T(Y)ad^T(Y) + \dots \end{aligned} \tag{39}$$

Integrating the first two terms, we obtain the matrix A from

$$A = \int_G \mathbf{x}\mathbf{x}^T \rho_1^{\mu_2}(h) dh = \Sigma_{\rho_1^{\mu_2}} = Ad(\mu_2^{-1})\Sigma_{\rho_1} Ad^T(\mu_2^{-1}) \tag{40}$$

and the matrix B from

$$B = \int_G \mathbf{y}\mathbf{y}^T \rho_2(g) dg = \Sigma_{\rho_2}. \tag{41}$$

Integration of the third term over g gives us

$$\int_G ad(X)\mathbf{y}\mathbf{y}^T ad^T(X)\rho_2(g) dg = ad(X) B ad^T(X)$$

and after integration over h we obtain

$$C(A,B) = \int_G ad(X) B ad^T(X) \rho_1^{\mu_2}(h) dh.$$

Additionally, we have

$$\int_G ad(X)ad(X)\rho_1^{\mu_2}(h) dh \int_G \mathbf{y}\mathbf{y}^T \rho_2(g) dg = A'' B,$$

where the $''$ operator rearranges the covariance matrix A. The remaining terms in the log expansion can be found similar to (7.1) by switching the order of A and B and using transposes.

7.2 Baker-Campbell-Hausdorff Formula

The *Baker-Campbell-Hausdorff formula* (BCH) [7] is a useful expression that will be used extensively in this paper to relate between the matrix exponential and the Lie bracket. The BCH is given by

$$\begin{aligned} Z(X,Y) &= \log(e^X e^Y) \\ &= X+Y+\frac{1}{2}[X,Y]+\frac{1}{12}([X,[X,Y]]+[Y,[Y,X]])+\frac{1}{24}([X,[Y,[Y,X]]])+\dots \end{aligned}$$

If the $\vee$ operator is applied to this formula, we obtain

$$\begin{aligned} \mathbf{z} =& \mathbf{x}+\mathbf{y}+\frac{1}{2}ad(X)\mathbf{y} \\ &+\frac{1}{12}(ad(X)ad(X)\mathbf{y}+ad(Y)ad(Y)\mathbf{x})+\frac{1}{24}ad(X)ad(Y)ad(Y)\mathbf{x}+\dots \end{aligned}$$

Acknowledgments. This work was funded in part by NSF grant IIS-0915542 “RI: Small: Robotic Inspection, Diagnosis, and Repair” and Andrew Long’s NDSEG fellowship.

References

[1] Alterovitz, R., Siméon, T., Goldberg, K.: The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty. In: Robotics: Science and Systems, pp. 246–253 (2007)
[2] Alterovitz, R., Branicky, M., Goldberg, K.: Motion planning under uncertainty for image-guided medical needle steering. Intl J. of Robotics Res. 27(11-12), 1361–1374 (2008)
[3] Barraquand, J., Ferbach, P.: Motion planning with uncertainty: The information space approach. In: Proceedings of the IEEE Intl Conf. on Robotics and Automation, vol. 2, pp. 1341–1348 (1995)
[4] Bertsekas, D.: Dynamic Programming and Optimal Control, 3rd edn., vol. II. Athena Scientific, Nashua (2011)
[5] Bry, A., Roy, N.: Rapidly-exploring random belief trees for motion planning under uncertainty. In: Proceedings of the IEEE Intl Conf. on Robotics and Automation, pp. 723–730 (2011)
[6] Censi, A., Calisi, D., De Luca, A., Oriolo, G.: A Bayesian framework for optimal motion planning with uncertainty. In: Proceedings of the IEEE Intl Conf. on Robotics and Automation, pp. 1798–1805 (2008)
[7] Chirikjian, G.: Stochastic Models, Information Theory, and Lie Groups, vol. 2. Birkhäuser (2012)
[8] Chirikjian, G., Kyatkin, A.: Engineering Applications of Noncommutative Harmonic Analysis. CRC Press, Boca Raton (2001)
[9] Choset, H., Lynch, K., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L., Thrun, S.: Principles of robot motion: theory, algorithms, and implementation. MIT Press (2005)

[10] Ebert-Uphoff, I., Chirikjian, G.: Inverse kinematics of discretely actuated hyper-redundant manipulators using workspace densities. In: Proceedings of the IEEE Intl Conf. on Robotics and Automation, vol. 1, pp. 139–145 (1996)
[11] Higham, D.: An algorithmic introduction to numerical simulation of stochastic differential equations. SIAM Review, 525–546 (2001)
[12] Kalman, R.: A new approach to linear filtering and prediction problems. J. of Basic Eng. 82, 35–45 (1960)
[13] Lambert, A., Gruyer, D.: Safe path planning in an uncertain-configuration space. In: Proceedings of the IEEE Intl Conf. on Robotics and Automation, vol. 3, pp. 4185–4190 (2003)
[14] LaValle, S.: Planning algorithms. Cambridge University Press (2006)
[15] Lazanas, A., Latombe, J.: Landmark-based robot navigation. Algorithmica 13(5), 472–501 (1995)
[16] Mason, R., Burdick, J.: Trajectory planning using reachable-state density functions. In: Proceedings of the IEEE Intl Conf. on Robotics and Automation, vol. 1, pp. 273–280 (2002)
[17] Murray, R., Li, Z., Sastry, S.: A mathematical introduction to robotic manipulation. CRC (1994)
[18] Park, W., Wang, Y., Chirikjian, G.: The path-of-probability algorithm for steering and feedback control of flexible needles. Intl J. of Robotics Res. 29(7), 813–830 (2010)
[19] Pepy, R., Lambert, A.: Safe path planning in an uncertain-configuration space using rrt. In: Proceedings of the IEEE/RSJ Intl Conf. on Intelligent Robots and Systems, pp. 5376–5381 (2006)
[20] Smith, R., Self, M., Cheeseman, P.: Estimating uncertain spatial relationships in robotics. In: Proceedings of the Second Annual Conf. on Uncertainty in Artificial Intelligence, pp. 435–461 (1986)
[21] Thrun, S., Burgard, W., Fox, D.: Probabilistic robotics. MIT Press (2006)
[22] Van Den Berg, J., Abbeel, P., Goldberg, K.: LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information. Intl J. of Robotics Res. 30(7), 895 (2011)
[23] Wang, Y., Chirikjian, G.S.: Nonparametric second-order theory of error propagation on motion groups. Intl J. of Robotics Res. 27(11-12), 1258–1273 (2008)