Second-Order Theory of Error Propagation on Motion Groups

Yunfeng Wang[1] and Gregory S. Chirikjian[2]

[1] Department of Mechanical Engineering, The College of New Jersey, Ewing, NJ 08628
jwang@tcnj.edu
[2] Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD 21218
gregc@jhu.edu

Summary. Error propagation on the Euclidean motion group arises in a number of areas such as and in dead reckoning errors in mobile robot navigation and joint errors that accumulate from the base to the distal end of manipulators. We address error propagation in rigid-body poses in a coordinate-free way. In this paper we show how errors propagated by convolution on the Euclidean motion group, $SE(3)$, can be approximated to second order using the theory of Lie algebras and Lie groups. We then show how errors that are small (but not so small that linearization is valid) can be propagated by a recursive formula derived here. This formula takes into account errors to second-order, whereas prior efforts only considered the first-order case [8, 9].

Keywords: Recursive error propagation, Euclidean group, spatial uncertainty.

1 Introduction

In this section we review the literature on error propagation, and review the terminology and notation used throughout the paper.

1.1 Literature Review

Murray, Li and Sastry [3], and Selig [4] presented Lie-group-theoretic notation and terminology to the robotics community, which has now become standard vocabulary. Chirikjian and Kyatkin [1] showed that many problems in robot kinematics and motion planning can be formulated as the convolution of functions on the Euclidean group. The representation and estimation of spatial uncertainty has also received attention in the robotics and vision literature. Two classic works in this area are due to Smith and Cheeseman [6] and Su and Lee [7]. Recent work on error propagation by Smith, Drummond and Roussopoulos [5] describes the concatenation of random variables on groups and applies this formalism to mobile robot navigation. In all three of these works, errors are assumed small enough that covariances can be propagated by the formula [8, 9]

$$\Sigma_{1*2} = Ad(g_2^{-1})\Sigma_1 Ad^T(g_2^{-1}) + \Sigma_2, \tag{1}$$

S. Akella et al. (Eds.): Algorithmic Foundation of Robotics VII, STAR 47, pp. 155–168, 2008.
springerlink.com

where Ad is the adjoint operator for $SE(3)$. This equation essentially says that given two 'noisy' frames of reference $g_1, g_2 \in SE(3)$, each of which is a Gaussian random variable with 6×6 covariance matrices[1] Σ_1 and Σ_2, respectively, the covariance of $g_1 \circ g_2$ will be Σ_{1*2}. This approximation is very good when errors are very small. We extend this linearized approximation to the quadratic terms in the expansion of the matrix exponential parametrization of $SE(3)$. Results for $SO(3)$ are generated in the process.

1.2 Review of Rigid-Body Motions

The Euclidean motion group, $SE(3)$, is the semi direct product of $I\!R^3$ with the special orthogonal group, $SO(3)$. We denote elements of $SE(3)$ as $g = (\mathbf{a}, A) \in SE(3)$ where $A \in SO(3)$ and $\mathbf{a} \in I\!R^3$. For any $g = (\mathbf{a}, A)$ and $h(\mathbf{r}, R) \in SE(3)$, the group law is written as $g \circ h = (\mathbf{a} + A\mathbf{r}, AR)$, and $g^{-1} = (-A^T\mathbf{a}, A^T)$. Alternately, one may represent any element of $SE(3)$ as a 4×4 homogeneous transformation matrix of the form

$$H(g) = \begin{pmatrix} A & \mathbf{a} \\ \mathbf{0}^T & 1 \end{pmatrix},$$

in which case the group law is matrix multiplication.

For small translational (rotational) displacements from the identity along (about) the i^{th} coordinate axis, the homogeneous transforms representing infinitesimal motions look like

$$H_i(\epsilon) \stackrel{\triangle}{=} \exp(\epsilon \tilde{E}_i) \approx I_{4\times 4} + \epsilon \tilde{E}_i$$

where

$$\tilde{E}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \quad \tilde{E}_2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \quad \tilde{E}_3 = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix};$$

$$\tilde{E}_4 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \quad \tilde{E}_5 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \quad \tilde{E}_6 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

These are related to the basis elements $\{E_i\}$ for $so(3)$ as

$$\tilde{E}_i = \begin{pmatrix} E_i & \mathbf{0} \\ \mathbf{0}^T & 0 \end{pmatrix}$$

when $i = 1, 2, 3$.

[1] Exactly what is meant by a covariance for a Lie group is quantified later in the paper.

Large motions are also obtained by exponentiating these matrices. For example,

$$\exp(t\tilde{E}_3) = \begin{pmatrix} \cos t & -\sin t & 0 & 0 \\ \sin t & \cos t & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \exp(t\tilde{E}_6) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

More generally, it can be shown that every element of a matrix Lie group G can be described with the exponential parametrization

$$g = g(x_1, x_2, ..., x_N) = \exp\left(\sum_{i=1}^{N} x_i \tilde{E}_i\right). \tag{2}$$

This kind of relationship is common in the study of Lie groups and algebras.

One defines the 'vee' operator, $\vee$, such that

$$\left(\sum_{i=1}^{N} x_i \tilde{E}_i\right)^{\vee} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{pmatrix}$$

The vector, $\mathbf{x} \in I\!R^N$, can be obtained from $g \in G$ from the formula

$$\mathbf{x} = (\log g)^{\vee}. \tag{3}$$

When integrating a function over a group in a neighborhood of the identity, a weight $w(\mathbf{x})$ is defined as

$$\int_G f(g)dg = \int_{I\!R^N} f(g(\mathbf{x}))w(\mathbf{x})d\mathbf{x}.$$

It may be shown that due to the nature of the exponential parameterization, $w(\mathbf{x}) = 1 + O(\|\mathbf{x}\|^2)$ near the identity, and so the approximation $w(\mathbf{x}) = 1$ can be used in the first order theory. However, in the current presentation we retain $w(\mathbf{x})$ for higher order errors.

We calculate

$$w(\mathbf{x}) = \det\left[\left(g^{-1}\frac{\partial g}{\partial x_1}\right)^{\vee}, \cdots, \left(g^{-1}\frac{\partial g}{\partial x_N}\right)^{\vee}\right]. \tag{4}$$

If the approximation $g = I + X + X^2/2 + X^3/6$ is used, then to second order we can write

$$w(\mathbf{x}) = 1 - \frac{1}{2}\mathbf{x}^T K \mathbf{x}$$

for some matrix K that depends on the structure of the group. K is computed for $SO(3)$ and $SE(3)$ in the Appendix.

2 Nonparametric Second-Order Theory

Let $g_1, g_2 \in SE(3)$ be two precise reference frames. Then $g_1 \circ g_2$ is the frame resulting from stacking one relative to the other. Now suppose that each has some uncertainty. Let $\{h_i\}$ and $\{k_j\}$ be two sets of frames of reference that are distributed around the identity. Let the first have N_1 elements, and the second have N_2. What will the covariance of the set of $N_1 \cdot N_2$ frames $\{(g_1 \circ g_2)^{-1} \circ g_1 \circ h_i \circ g_2 \circ k_j\}$ (which are assumed to be distributed around the identity) look like ?

A pdf, ρ, on a Lie group G is said to have mean at the identity if the function

$$C(g) = \int_G \|[\log(g^{-1} \circ h)]^\vee\|^2 \rho(h) dh$$

is minimized at $g = e$. For this kind of pdf, the covariance is defined as

$$\Sigma = \int_G \log(g)^\vee [\log(g)^\vee]^T \rho(g) dg. \tag{5}$$

A similar expression can be defined for discrete cloud of frames, which is equivalent to replacing $\rho(g)$ with a weighted sum of Dirac delta functions.

Let $\rho_i(g)$ be a unimodal pdf with mean at the identity and which has a preponderance of its mass concentrated in a unit ball around the identity (where distance from the identity is measured as $\|(\log g)^\vee\|$). Then $\rho_i(g_i^{-1} \circ g)$ will be a distribution with the same shape centered at g_i. In general, the convolution of two pdfs is defined as

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

and in particular if we make the change of variables $k = g_1^{-1} \circ h$, then

$$\rho_1(g_1^{-1} \circ g) * \rho_2(g_2^{-1} \circ g) = \int_G \rho_1(k) \rho_2(g_2^{-1} \circ k^{-1} \circ g_1^{-1} \circ g) dk.$$

Making the change of variables $g = g_1 \circ g_2 \circ q$, where q is a relatively small displacement measured from the identity, the above can be written as

$$\rho_{1*2}(g_1 \circ g_2 \circ q) = \int_G \rho_1(k) \rho_2(g_2^{-1} \circ k^{-1} \circ g_2 \circ q) dk. \tag{6}$$

The essence of this paper is the efficient approximation of covariances associated with (6) when those of ρ_1 and ρ_2 are known. This problem reduces to the efficient approximation of

$$\Sigma_{1*2} = \int_G \int_G \log(q)^\vee [\log(q)^\vee]^T \rho_1(k) \rho_2(g_2^{-1} \circ k^{-1} \circ g_2 \circ q) dk dq. \tag{7}$$

In many practical situations, discrete data are sampled from ρ_1 and ρ_2 rather than having complete knowledge of the distributions themselves. Therefore, sampled covariances can be computed by making the following substitutions:

$$\rho_1(g) = \sum_{i=1}^{N_1} \alpha_i \Delta(h_i^{-1} \circ g) \tag{8}$$

and

$$\rho_2(g) = \sum_{j=1}^{N_2} \beta_j \Delta(k_j^{-1} \circ g) \tag{9}$$

where

$$\sum_{i=1}^{N_1} \alpha_i = \sum_{j=1}^{N_2} \beta_j = 1.$$

Here $\Delta(g)$ is the Dirac delta function for the group G, which has the properties

$$\int_G f(g)\Delta(h^{-1} \circ g)dg = f(h) \quad \text{and} \quad \Delta(h^{-1} \circ g) = \Delta(g^{-1} \circ h).$$

Using these properties, if we substitute (8) into (5), the result is

$$\Sigma_1 = \int_G \log(g)^\vee [\log(g)^\vee]^T \sum_{i=1}^{N_1} \alpha_i \Delta(h_i^{-1} \circ g)dg = \sum_{i=1}^{N_1} \alpha_i \log(h_i)^\vee [\log(h_i)^\vee]^T \tag{10}$$

Substitution of the sampled ρ_1 into (7) yields

$$\Sigma_{1*2} = \sum_{i=1}^{N_1} \alpha_i \int_G \log(q)^\vee [\log(q)^\vee]^T \rho_2(g_2^{-1} \circ h_i^{-1} \circ g_2 \circ q)dq. \tag{11}$$

Similarly, substitution of the sampled ρ_2 into the above equation kills the integral and substitutes values of q for which $g_2^{-1} \circ h_i^{-1} \circ g_2 \circ q = k_j$. This yields

$$\Sigma_{1*2} = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \alpha_i \beta_j \log(g_2^{-1} \circ h_i \circ g_2 \circ k_j)^\vee [\log(g_2^{-1} \circ h_i \circ g_2 \circ k_j)^\vee]^T. \tag{12}$$

While this equation is exact, it has the drawback of requiring $O(N_1 \cdot N_2)$ arithmetic operations. In the first-order theory of error propagation, we made the approximation

$$\log(k^{-1} \circ q) = X - Y,$$

or equivalently

$$[\log(k^{-1} \circ q)]^\vee = \mathbf{x} - \mathbf{y},$$

where $k = \exp Y$ and $q = \exp X$ are elements of the Lie group $SE(3)$. This decouples the summations and makes the computation $O(N_1 + N_2)$. However, the first-order theory breaks down for large errors. Therefore, we explore here a second-order theory that has the benefits of greater accuracy, while retaining good computational performance.

In the second-order theory of error propagation on Lie groups (and $SE(3)$ in particular), we now make the approximation

$$\log(k^{-1} \circ q) = X - Y + \frac{1}{2}[X, Y],$$

or equivalently

$$[\log(k^{-1} \circ q)]^{\vee} = \mathbf{x} - \mathbf{y} + \frac{1}{2} ad(\mathbf{x})\mathbf{y}. \tag{13}$$

Interestingly, terms such as X^2 and Y^2 do not appear in the approximation (13).

Here $[\cdot, \cdot]$ denotes the Lie bracket, and

$$[X, Y] = \sum_{i,j,k} C_{ij}^{k} x_i y_j \tilde{E}_k,$$

which means that the k^{th} component of $[X, Y]^{\vee}$ will be of the form $\sum_{i,j} C_{ij}^{k} x_i y_j$ which is a weighted product of elements from $\mathbf{x}$ and $\mathbf{y}$. We therefore write

$$[X, Y]^{\vee} = \mathbf{x} \wedge \mathbf{y} = ad(X)\mathbf{y}.$$

$ad(X)$ should not be confused with $Ad(g)$, which is defined by $Ad(g)\mathbf{x} = (gXg^{-1})^{\vee}$. The relationship between these two is $Ad(\exp X) = \exp(ad(X))$. See the Appendix for a more complete review.

In addition to the approximation in (13) we use two additional properties of the log function:

$$[\log(k^{-1})]^{\vee} = -[\log(k)]^{\vee} \tag{14}$$

and

$$[\log(g \circ h \circ g^{-1})]^{\vee} = Ad(g)[\log(h)]^{\vee}. \tag{15}$$

Using (13), (14) and (15), then to second order,

$$\log(g_2^{-1} \circ h_i \circ g_2 \circ k_j)^{\vee} = Ad(g_2^{-1})\mathbf{y}_i + B_i \mathbf{z}_j$$

where $\mathbf{y}_i = (\log h_i)^{\vee}$, $\mathbf{z}_j = (\log k_j)^{\vee}$, and $B_i = B(Ad(g_2^{-1})\mathbf{y}_i)$ where

$$B(\mathbf{x}) = I + \frac{1}{2} ad(\mathbf{x}).$$

Note also that $B(\mathbf{y})\mathbf{y} = \mathbf{y}$ because $[Y, Y] = 0$, and therefore $[B(\mathbf{y})]^{-1}\mathbf{y} = \mathbf{y}$ as well. Substitution into the formula (12) for Σ_{1*2} then yields

$$\Sigma_{1*2} = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \alpha_i \beta_j (B_i \mathbf{z}_j + A\mathbf{y}_i)(B_i \mathbf{z}_j + A\mathbf{y}_i)^T$$

where $A = Ad(g_2^{-1})$.

Assuming that the sampled distributions are centered around the identity (so that cross terms sum to zero), allows the summations over i and j to decouple. The result is written as

$$\Sigma_{1*2} = A \Sigma_1 A^T + \sum_{i=1}^{N_1} \alpha_i B_i \Sigma_2 B_i^T \tag{16}$$

In practice, $\alpha_i = 1/N_1$ and $\beta_j = 1/N_2$.

Note, in the first order theory we approximated $B_i = I$, and the above reduced to

$$\Sigma_{1*2} = Ad_{g_2^{-1}} \Sigma_1 Ad^T_{g_2^{-1}} + \Sigma_2.$$

3 Numerical Examples

Evaluating the robustness of the first-order (1) and the second-order (16) covariance propagation formula over a wide range of kinematic errors is essential to understand effectiveness of these formulas. In this section, we test these two covariance propagation formulas with concrete numerical examples.

Consider a spatial serial manipulator, PUMA 560. The link-frame assignments of PUMA 560 for D-H parameters is the same as those given [2]. Table 1 lists the D-H parameters of PUMA 560, where $a_2 = 431.8$ mm, $a_3 = 20.32$ mm, $d_3 = 124.46$ mm, and $d_4 = 431.8$ mm. The solution of forward kinematics is the homogeneous transformations of the relative displacements from one D-H frame to another multiplied sequentially.

In order to test these covariance propagation formulas, we first need to create some kinematic errors. Since joint angles are the only variables of the PUMA 560, we assume that errors exist only in these joint angles. We generated errors by deviating each joint angle from its ideal value with uniform random absolute errors of $\pm\epsilon$. Therefore, each joint angle was sampled at three values: $\theta_i - \epsilon$, θ_i, $\theta_i + \epsilon$. This generates $n = 3^6$ different frames of references $\{g^i_{ee}\}$ that are clustered around desired g_{ee}. Here g_{ee} denotes the position and orientation of the distal end of the manipulator relative to the base in the form of homogeneous transformation matrix.

Three different methods for computing the same error covariances for the whole manipulator are computed. The first is to apply brute force enumeration, which gives the actual covariance of the whole manipulator:

$$\Sigma = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T \tag{17}$$

where $\mathbf{x}_i = [\log(g^{-1} \circ g_i)]^\vee$, and the formula (17) is used to all the 3^6 different frames of references $\{g^i_{ee}\}$. The second method is to apply the first-order propagation formula (1). The third is to apply the second-order propagation formula (16). For the covariance propagation methods, we only need to find the covariance of each individual link. Then the covariance of the whole manipulator can be recursively calculated using the corresponding propagation formula. In our case, all the individual links have the same covariance since we assumed the same kinematic errors at each joint angle.

In order to quantify the robustness of the two covariance approximation methods, we define a measure of deviation of results between the first/second order formula and the actual covariance using the Hilbert-Schmidt (Frobenius) norm as

Table 1. DH PARAMETERS of PUMA 560

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90^o	0	0	θ_2
3	0	a_2	d_3	θ_3
4	-90^o	a_3	d_4	θ_4
5	90^o	0	0	θ_5
6	-90^o	0	0	θ_6

$$deviation = \frac{\|\Sigma_{prop} - \Sigma_{actual}\|}{\|\Sigma_{actual}\|}, \tag{18}$$

where Σ_{prop} is the covariance of the whole manipulator calculated using either the first-order (1) or the second-order (16) propagation formula, Σ_{actual} is the actual covariance of the whole manipulator calculated using (17), and $\|\cdot\|$ denotes the Hilbert-Schmidt (Frobenius) norm.

With all the above information, we now can conduct the specific computation and analysis. Our numerical simulations have showed that different configurations of the manipulator will not influence the end-effector covariances too much. Here the ideal joint angles from θ_1 to θ_6 were taken as $[0, \pi/2, -\pi/2, 0, 0, \pi/2]$. The joint angle errors ϵ were taken from 0.1 rad to 0.6 rad. The covariances of the whole manipulator corresponding to these kinematic errors were then calculated through the three aforementioned methods. The results of the first-order and second-order propagation formula were graphed in Fig. 1 in terms of deviation defined by Eq. (18). It was shown that the second-order propagation formula makes significant improvements in terms of accuracy than that of the first-order formula. The second-order propagation theory is much more robust than the first-order formula over a wide range of kinematic errors. These two methods both work well for small errors, and deviate from the actual value more and more as the errors become large. However, the deviation of the first-order formula grows rapidly and breaks down while the second-order propagation method still retains a reasonable value.

To give the readers a sense of what these covariances look like, we listed the values of the covariance of the whole manipulator for the joint angle error $\epsilon = 0.3$ rad below.

The ideal pose of the end effector can be found easily via forward kinematics to be

$$g_{ee} = \begin{pmatrix} 0.0000 & -1.0000 & 0 & 0.0203 \\ -1.0000 & -0.0000 & 0 & 0.1245 \\ 0 & 0 & -1.0000 & -0.8636 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix}.$$

The actual covariance of the whole manipulator calculated using equations (17) is

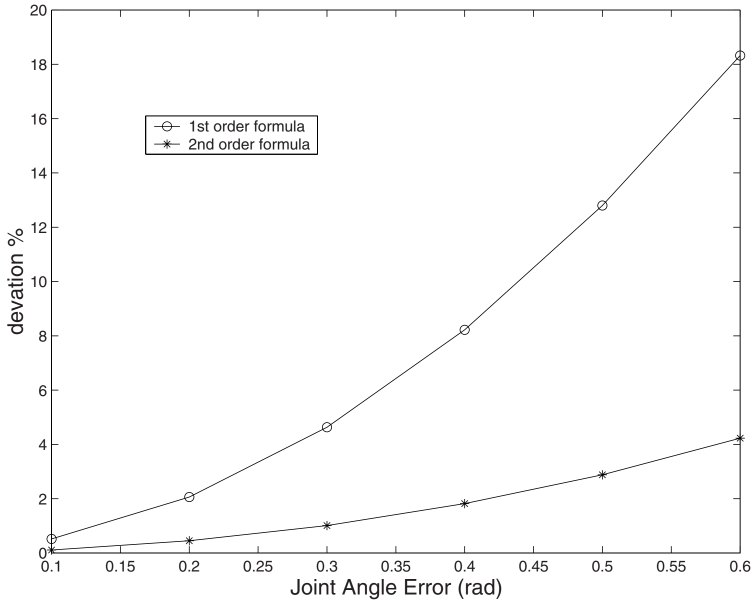

Fig. 1. The Deviation of the First and Second-order Propagation Methods

$$\Sigma_{actual} = \begin{pmatrix} 0.1748 & 0.0000 & 0.0000 & 0.0000 & -0.0755 & -0.0024 \\ 0.0000 & 0.0078 & 0.0000 & 0.0034 & -0.0000 & 0.0003 \\ 0.0000 & 0.0000 & 0.1747 & 0.0012 & -0.0072 & -0.0000 \\ 0.0000 & 0.0034 & 0.0012 & 0.0025 & -0.0001 & 0.0001 \\ -0.0755 & -0.0000 & -0.0072 & -0.0001 & 0.0546 & 0.0015 \\ -0.0024 & 0.0003 & -0.0000 & 0.0001 & 0.0015 & 0.0011 \end{pmatrix},$$

the covariance using the first-order propagation formula (1) is

$$\Sigma_{prop,\,1st} = \begin{pmatrix} 0.1800 & 0.0000 & 0 & 0.0000 & -0.0777 & -0.0024 \\ 0.0000 & 0.0000 & 0 & 0.0000 & -0.0000 & -0.0000 \\ 0 & 0 & 0.1800 & 0.0012 & -0.0075 & 0 \\ 0.0000 & 0.0000 & 0.0012 & 0.0000 & -0.0002 & -0.0000 \\ -0.0777 & -0.0000 & -0.0075 & -0.0002 & 0.0569 & 0.0016 \\ -0.0024 & -0.0000 & 0 & -0.0000 & 0.0016 & 0.0000 \end{pmatrix},$$

and the covariance using the second-order propagation formula (16) is

$$\Sigma_{prop,\,2nd} = \begin{pmatrix} 0.1765 & 0.0000 & 0.0000 & 0.0000 & -0.0762 & -0.0024 \\ 0.0000 & 0.0079 & 0.0000 & 0.0034 & -0.0000 & 0.0003 \\ 0.0000 & 0.0000 & 0.1765 & 0.0012 & -0.0072 & 0.0000 \\ 0.0000 & 0.0034 & 0.0012 & 0.0025 & -0.0001 & 0.0001 \\ -0.0762 & -0.0000 & -0.0072 & -0.0001 & 0.0551 & 0.0015 \\ -0.0024 & 0.0003 & 0.0000 & 0.0001 & 0.0015 & 0.0011 \end{pmatrix},$$

where the covariance of one link is

$$\Sigma_{one-link} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0600 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

4 Conclusions

In this paper, first-order kinematic error propagation formulas are modified to include second-order effects. This extends the usefulness of these formulas to errors that are not necessarily small. In fact, in the example to which the methodology is applied, errors in orientation can be as large as a radian or more and the second-order formula appears to capture the error well. The second-order propagation formula makes significant improvements in terms of accuracy than that of the first-order formula. The second-order propagation theory is much more robust than the first-order formula over a wide range of kinematic errors.

References

1. Chirikjian, G.S., Kyatkin, A.B.: Engineering Applications of Noncommutative Harmonic Analysis. CRC Press, Boca Raton (2001)
2. Craig, J.J.: Introduction to Robotics Mechanics and Control, 3rd edn. Prentice Hall, Englewood Cliffs (2005)
3. Murray, R.M., Li, Z., Sastry, S.S.: A Mathematical Introduction to Robotic Manipulation. CRC Press, Ann Arbor MI (1994)
4. Selig, J.M.: Geometrical Methods in Robotics. Springer, New York (1996)
5. Smith, P., Drummond, T., Roussopoulos, K.: Computing MAP trajectories by representing, propagating and combining PDFs over groups. In: Proceedings of the 9th IEEE International Conference on Computer Vision, vol. II, pp. 1275–1282, Nice (2003)
6. Smith, R.C., Cheeseman, P.: On the Representation and Estimation of Spatial Uncertainty. The International Journal of Robotics Research 5(4), 56–68 (1986)
7. Su, S., Lee, C.S.G.: Manipulation and Propagation of Uncertainty and Verification of Applicability of Actions in assembly Tasks. IEEE Transactions on Systems, Man, and Cybernetics 22(6), 1376–1389 (1992)
8. Wang, Y., Chirikjian, G.S.: Error Propagation on the Euclidean Group with Applications to Manipulator Kinematics. IEEE Transactions on Robotics (in press)
9. Wang, Y., Chirikjian, G.S.: Error Propagation in Hybrid Serial-Parallel Manipulators. In: IEEE International Conference on Robotics and Automation, Orlando Florida, pp. 1848–1853 (May 2006)

A Appendix

A.1 Background

Given $g \in SE(3)$ of the form,

$$g = \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}$$

and $X \in se(3)$ of the form

$$X = \begin{pmatrix} \Omega & \mathbf{v} \\ \mathbf{0}^T & 0 \end{pmatrix},$$

if

$$\mathbf{x} = (X)^\vee = \begin{pmatrix} \boldsymbol{\omega} \\ \mathbf{v} \end{pmatrix},$$

then $Ad(g)$ is defined by the expression

$$(gXg^{-1})^\vee = Ad(g)\mathbf{x}$$

and explicitly

$$Ad(g) = \begin{pmatrix} R & 0 \\ TR & R \end{pmatrix}. \tag{19}$$

The matrix T is skew-symmetric, and $\text{vect}(T) = \mathbf{t}$.

Similarly, $ad(X)$ (which can also be written as $ad(\mathbf{x})$), is defined by

$$[X, Y]^\vee = ad(X)\mathbf{y},$$

where $[X, Y] = XY - YX$ is the Lie bracket. Explicitly,

$$ad(X) = \begin{pmatrix} \Omega & 0 \\ V & \Omega \end{pmatrix} \tag{20}$$

where the matrix V is skew-symmetric, and $\text{vect}(V) = \mathbf{v}$.

A.2 Second Order Approximation of Volume Weighting Function for $SO(3)$ and $SE(3)$

Let g be an element of the matrix Lie group G, and X be an arbitrary element of the associated Lie algebra, $\mathcal{G}$. Let $g = \exp X$. Then we can truncate the Taylor series expansion for g and g^{-1} as:

$$g = I + X + X^2/2 + X^3/6 + O(X^4) \quad \text{and} \quad g^{-1} = I - X + X^2/2 - X^3/6 + O(X^4).$$

If $X = \sum_i x_i E_i$, this means that

$$\frac{\partial g}{\partial x_i} = E_i + \frac{1}{2}(E_i X + X E_i) + \frac{1}{6}(E_i X^2 + X E_i X + X^2 E_i) + O(X^3).$$

Therefore,

$$g^{-1}\frac{\partial g}{\partial x_i} = E_i + \frac{1}{2}[E_i, X] - \frac{1}{3}XE_iX + \frac{1}{6}(E_iX^2 + X^2E_i) + O(X^3).$$

Taking the $\vee$ of both sides yields the columns of the Jacobian matrix, the determinant of which provides the desired weighting function. Note that $(E_i)^\vee = \mathbf{e}_i$ and $([E_i, X])^\vee = -([X, E_i])^\vee = -ad(X)\mathbf{e}_i$, and so we can write the i^{th} column as:

$$\left(g^{-1}\frac{\partial g}{\partial x_i}\right)^\vee = \mathbf{e}_i - \frac{1}{2}ad(X)\mathbf{e}_i - \frac{1}{3}(XE_iX)^\vee + \frac{1}{6}(E_iX^2 + X^2E_i)^\vee. \quad (21)$$

If we define,

$$J_1(\mathbf{x}) = [(XE_1X)^\vee, (XE_2X)^\vee, (XE_3X)^\vee]$$

and

$$J_2(\mathbf{x}) = [(E_1X^2 + X^2E_1)^\vee, (E_2X^2 + X^2E_2)^\vee, (E_3X^2 + X^2E_3)^\vee],$$

then to second order,

$$J(\mathbf{x}) = I - \frac{1}{2}ad(X) - \frac{1}{3}J_1(\mathbf{x}) + \frac{1}{6}J_2(\mathbf{x}). \quad (22)$$

Details for $SO(3)$

In the case of $SO(3)$,

$$X = \begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix}$$

and $ad(X) = X$. Direct calculation shows that the matrix $J_1^{so(3)}$ can be written as

$$J_1^{so(3)}(\mathbf{x}) = -\begin{pmatrix} x_1^2 & x_1x_2 & x_1x_3 \\ x_1x_2 & x_2^2 & x_2x_3 \\ x_1x_3 & x_2x_3 & x_3^2 \end{pmatrix}$$

Similarly, for $J_2^{so(3)}$ one finds

$$J_2^{so(3)}(\mathbf{x}) = -\begin{pmatrix} 2x_1^2 + x_2^2 + x_3^2 & x_1x_2 & x_1x_3 \\ x_1x_2 & x_1^2 + 2x_2^2 + x_3^2 & x_2x_3 \\ x_1x_3 & x_2x_3 & x_1^2 + x_2^2 + 2x_3^2 \end{pmatrix}$$

Now, to second order, the full Jacobian is

$$J^{so(3)}(\mathbf{x}) = I - \frac{1}{2}X - \frac{1}{3}J_1^{so(3)}(\mathbf{x}) + \frac{1}{6}J_2^{so(3)}(\mathbf{x}) = I - \frac{1}{2}X + \frac{1}{6}X^2,$$

where of course

$$X^2 = \begin{pmatrix} -x_2^2 - x_3^2 & x_1x_2 & x_1x_3 \\ x_1x_2 & -x_1^2 - x_3^2 & x_2x_3 \\ x_1x_3 & x_2x_3 & -x_1^2 - x_2^2 \end{pmatrix}$$

The second-order approximation of the determinant of $J^{so(3)}(\mathbf{x})$ is then

$$w^{so(3)}(\mathbf{x}) = \det J^{so(3)}(\mathbf{x}) \approx \det(I - \frac{1}{2}X) \cdot \det(I + \frac{1}{6}X^2) \approx (1 + \frac{1}{4}\|\mathbf{x}\|^2)(1 - \frac{1}{3}\|\mathbf{x}\|^2).$$

The reason why this is justified is that all terms in the cofactor expansion of the det depend on X^2 will be of higher than second order, except those on the diagonal. This is due to the fact that second-order terms here will multiply the diagonal entries of the identity matrix yielding second-order terms.

Finally, this means that to second order,

$$w^{so(3)}(\mathbf{x}) = 1 - \frac{1}{2}\mathbf{x}^T K \mathbf{x} \quad \text{where} \quad K = \frac{1}{6}I \tag{23}$$

Details for $SE(3)$

It is convenient to write an arbitrary element of $se(3)$ as

$$X = \begin{pmatrix} \Omega & \mathbf{v} \\ \mathbf{0}^T & 0 \end{pmatrix}$$

where Ω is an arbitrary element of $so(3)$ and $\mathbf{v}$ is an arbitrary element of $\mathbb{R}^3$.

In this case,

$$ad(X) = \begin{pmatrix} \Omega & 0 \\ V & \Omega \end{pmatrix}$$

where $(V)^\vee = \mathbf{v}$.

Referring back to (21), we can compute each term directly to find:

$$(XE_iX)^\vee = \begin{pmatrix} (\Omega E_i \Omega)^\vee \\ \Omega E_i \mathbf{v} \end{pmatrix}$$

for rotational components $(i = 1, 2, 3)$ and

$$(XE_iX)^\vee = \mathbf{0}$$

for transitional components $(i = 4, 5, 6)$. This means that

$$J_1^{se(3)}(\boldsymbol{\omega}, \mathbf{v}) = \begin{pmatrix} J_1^{so(3)}(\boldsymbol{\omega}) & 0 \\ \boldsymbol{\omega} \wedge (I \wedge \mathbf{v}) & 0 \end{pmatrix}.$$

Similarly,

$$(E_iX^2 + X^2E_i)^\vee = \begin{pmatrix} (E_i\Omega^2 + \Omega^2 E_i)^\vee \\ E_i \Omega \mathbf{v} \end{pmatrix}.$$

for rotational components $(i = 1, 2, 3)$ and

$$(E_i X^2 + X^2 E_i)^\vee = \begin{pmatrix} \mathbf{0} \\ \Omega^2 \mathbf{e}_i \end{pmatrix}.$$

for transitional components $(i = 4, 5, 6)$ and

$$J_2^{se(3)}(\boldsymbol{\omega}, \mathbf{v}) = \begin{pmatrix} J_2^{so(3)}(\boldsymbol{\omega}) & 0 \\ (\Omega \mathbf{v})^\wedge & \Omega^2 \end{pmatrix}.$$

Substituting into (22) and taking the determinant,

$$\det J^{se(3)}(\boldsymbol{\omega}, \mathbf{v}) = |\det J^{so(3)}(\boldsymbol{\omega})|^2$$

This means that to second order,

$$w^{se(3)}(\mathbf{x}) = 1 - \frac{1}{2}\mathbf{x}^T K \mathbf{x} \quad \text{where} \quad K = \begin{pmatrix} \frac{1}{3}I & 0 \\ 0 & 0 \end{pmatrix}. \tag{24}$$