Abstract

We address general filtering problems on the Euclidean group $SE(3)$. We first generalize, to stochastic nonlinear systems evolving on $SE(3)$, the particle filter of Liu and West [17] for simultaneous estimation of the state and covariance. The filter is constructed in a coordinate-invariant way, and explicitly takes into account the geometry of $SE(3)$ and $P(n)$, the space of symmetric positive definite matrices. Some basic results for bilinear systems on $SE(3)$ with linear and quadratic measurements are also derived. Three examples—GPS attitude estimation, needle tip location, and vision-based robot end-effector pose estimation—are presented to illustrate the framework.
1 Introduction

Algorithms for the simultaneous localization and mapping (SLAM) problem, probabilistic motion planning (see, e.g., the recent monograph [34] and the references cited therein), and vision-based tracking [3], are some of the more prominent successes that the probabilistic approach to robotics has recently produced. While probabilistic approaches to object recognition and tracking continue to make the most explicit contact with geometry, e.g., [30], [31], [32], probabilistic ideas are appearing in increasingly diverse settings in which geometry plays a role, e.g., the problem of needle steering [24], [39] and mobile robot path planning [42], [43], mechanism workspace generation [36], and investigating the kinematic effects of backlash on parallel mechanisms [38].

A common theme found throughout many of these recent works is that the state equations, which are stochastic, evolve on the Euclidean group $SE(3)$. Motivated by these works, and with a view toward extending this stochastic framework to a larger class of problems in robotics, this paper addresses general filtering problems on the Euclidean group.

Our first set of contributions is a generalization, to stochastic systems evolving on $SE(3)$, of the particle filter of Liu and West [17] for simultaneously estimating the state and model parameters. Here the geometry of $SE(3)$ and $P(n)$, the space of $n \times n$ symmetric positive definite matrices, plays a key role, and our results are relevant to generalizations of other particle filtering algorithms. With a view toward approximating general nonlinear stochastic systems on $SE(3)$ with bilinear state equations and linear or quadratic measurements, we also derive some basic properties of bilinear systems on $SE(3)$ with linear-quadratic measurements, a case arguably corresponding to the most basic nontrivial system on $SE(3)$.

The second set of contributions involves three detailed examples that illustrate how particle filtering problems on $SE(3)$ naturally arise in quite distinct settings: (i) attitude estimation using GPS measurements; (ii) tip position and covariance estimation during needle steering; (iii) vision-based robot end-effector pose estimation. Simulation results are provided for the first two examples, while experimental results involving a small, low-cost humanoid robot are presented for the third example.

The practical advantages of particle filtering as an alternative to, e.g., the extended Kalman filter (EKF), have by now been well-documented in the literature (see, e.g., [1], [25]). Aside from a few excep-
tions mentioned below, however, none of the previous particle filtering works address systems that evolve on curved spaces like $SE(3)$. In principle one could, after choosing a suitable set of local coordinates, apply any number of vector space filtering algorithms; under assumptions of small noise such methods may work locally. However, globally and under large noise assumptions, their performance is quite likely to vary widely. The usual problems associated with lack of coordinate invariance (e.g., both the noise distribution and filter performance depend on the choice of local coordinates and the identity element, and switching between different coordinate charts must be managed) will moreover persist, making such approaches at best cumbersome and unreliable.

The first paper to explicitly address Monte Carlo filtering on Lie groups is [6], who generalize the sequential importance sampling particle filter to state equations of the form

$$dX = e^V X \, dt, \quad dV = dW,$$

where $X$ is an element of a matrix Lie group, $V$ an element of its corresponding Lie algebra, and $W$ a diffusion process defined on the Lie algebra. The associated measurement equation $y = h(X) + \eta$ includes an additive measurement noise term $\eta$. Case studies involving $SO(3)$ and $SE(3)$ are also presented. Earlier groundbreaking studies by Chirikjian et al. on Brownian motions on $SO(3)$ and $SE(3)$ in a robotics context, and stochastic differential equation models of the kinematics of mobile robots and surgical needles, as well as exact solutions to the associated Fokker-Planck equations, convolution formulas on $SE(3)$, and their applications to motion planning and accuracy analysis, have also been presented in [5], [24], [37], [42], [43]. Geometric difusions on $SO(3)$ also appear in the recent work of Srivastava [30], [31], [32] on object recognition and tracking.

Among the previous works, only [6] and [30] explicitly address particle filtering on $SE(3)$. In this paper we shall consider combined state-covariance estimation on systems whose state equations are more general than either [6] or [30], with object tracking and needle steering reduced to specific instances of these. We also construct the filter in a coordinate-invariant way, to avoid the pitfalls associated with local coordinate-based approaches. This involves consideration of a number of geometric issues on $SE(3)$ and $P(n)$ that we address, e.g., distance metrics, sample means and covariances, discretization and integration of differential equations, and coordinate-invariant constructions of probability distributions on these spaces.
The paper is organized as follows. Section 2 develops the geometric framework for the general filtering problem on $SE(3)$. The generalization to $SE(3)$ of the combined state-parameter particle filtering algorithm of [17] is described in Section 3. Sections 4, 5, and 6 respectively illustrate the methodology for GPS-based attitude estimation, needle steering position and covariance estimation, and vision-based robot end-effector pose estimation. Finally, in keeping with the spirit of this special issue, we conclude in Section 7 by suggesting several open problems, of both a fundamental and applied nature, that we feel are suitable for collaboration among geometers, roboticists, and control theorists.

2 General Framework

2.1 Nonlinear Filtering on Matrix Lie Groups

As is well-known, for the general vector space nonlinear filtering problem

$$dx = f(x, t) \, dt + G(x, t) \, dw$$

$$dy = h(x, t) \, dt + d\eta,$$

the conditional density $\rho(x(t)|Y_t)$, where $Y_t = \{y(s)|t_0 \leq s \leq t\}$, cannot, in general, be characterized by a finite parameter set (the well-known exception is the Kalman filter, whose conditional density is Gaussian, so that the filtering algorithm needs only to propagate the conditional mean and conditional variance). In most nonlinear problems of practical interest where this fortuitous situation does not exist, one is therefore forced to consider suboptimal filters.

With advances in both Monte Carlo simulation techniques and computing power, particle methods, which generate a set of samples that approximate the conditional density $\rho(x(t)|Y_t)$, have become an attractive means of obtaining approximate solutions. Many different versions of particle filters exist (see Doucet et al. [8] for a survey), and particle methods for online estimation continue to be a highly active area of research. This paper aims to extend particle filtering methods developed on general vector spaces to stochastic systems evolving on $SE(3)$, and by way of extension general matrix Lie groups.

We first consider the following general setting. Let $G$ be an $m$-dimensional matrix Lie group and $\mathfrak{g}$ its corresponding matrix Lie algebra, with basis elements $E_1, \ldots, E_m \in \mathfrak{g}$. The state equations and
measurements are assumed to be in left-invariant form (the development for right-invariant systems is analogous, and is not repeated here):

\[ dX = X \cdot A(X) \, dt + X \sum_{i=1}^{m} b_i(X) E_i \, dw_i \]  \hspace{1cm} (4)

\[ dy = c(X) \, dt + d\eta, \]  \hspace{1cm} (5)

where \( X \in G \) is the state, \( y \in \mathbb{R}^p \) is the measurement vector, the maps \( A : G \to g, c : G \to \mathbb{R}^p \), and \( b_i : G \to \mathbb{R} \) are assumed \( C^2 \), and \( dw_i \in \mathbb{R}, d\eta \in \mathbb{R}^p \) denote independent Wiener processes. The objective is to estimate the current state \( X(t) \) from the measurements \( y \) up to the current time \( t \).

2.2 The Bilinear Case

We now consider the case when both \( A(X) \) and the \( b_i(X) \) in (4) are constant. The resulting state equation then becomes linear in the state, and also linear in the driving term, but not jointly linear; such systems are referred to as bilinear systems in the literature, and we shall adopt this same terminology for the case of constant \( A(X) \) and \( b_i(X) \). Our interest in the bilinear case can be traced to several reasons: (i) it is the simplest nontrivial example of a stochastic system on \( G \); (ii) models for many real-world problems are bilinear (for example, our case studies on attitude estimation and needle steering); (iii) in the case of systems on vector spaces, arbitrarily good bilinear approximations to deterministic nonlinear systems can be obtained (see, e.g., [26] and the references cited therein); presumably such approximations can in principle also be constructed for nonlinear systems evolving on matrix Lie groups.

The Conditional Density Equations

In [24] the Fokker-Planck equations are derived for the probability density \( \rho(X,t) \) corresponding to the bilinear case: the Fokker-Planck operator, denoted \( \mathcal{L} \), is shown to be

\[ \mathcal{L} = -\sum_{i=1}^{m} a_i E_i^R + \frac{1}{2} \sum_{j=1}^{m} \sum_{k=1}^{m} b_j b_k E_j^R E_k^R \]  \hspace{1cm} (6)

where \( E_i^R \) is defined as the Lie derivative

\[ E_i^R \rho(X,t) = \frac{d}{dt} \rho(X \circ \exp(tE_i), t)|_{t=0} \]  \hspace{1cm} (7)

with respect to the basis element \( E_i \) of \( g \). [24] also shows how solutions to the Fokker-Planck equation \( \frac{\partial \rho}{\partial t} = \mathcal{L} \rho \) can be obtained efficiently using methods from noncommutative harmonic analysis.
Taking the measurement equations (5) into account, it is possible to write down the corresponding conditional density equations in unnormalized form. Denoting the unnormalized conditional density by \( \sigma(X, t) \), from the Duncan-Mortensen-Zakai equation \([9],[19],[41]\), \( \sigma(X, t) \) satisfies

\[
\frac{d\sigma(X, t)}{dt} = \left( \mathcal{L} - \frac{1}{2} \sum_{i=1}^{p} h_i(X)^2 \right) \sigma(X, t) + \sum_{i=1}^{p} h_i(X) \sigma(X, t) dy_i(t),
\]

with \( \sigma(X, 0) = \sigma_0(t) \), where \( \sigma_0 \) is the probability density of the initial point \( X_0 \). Solutions to (8), particularly in recursive form, are in general quite difficult to obtain. The idea of using estimation Lie algebras to construct finite-dimensional nonlinear filters was first proposed in [4], where it is shown that as long as the estimation algebra is finite dimensional, a finite dimensional recursive filter can be obtained. Details of this methodology and various extensions have been worked out by Yau in [40]; further connections with our problem are briefly mentioned in the last section.

The Itô Rule for Linear and Quadratic Measurements

Without loss of generality we can rewrite the bilinear state equations (4) as

\[
\frac{dX}{dt} = X \cdot A \, dt + X \, dW
\]

where \( dW \in g \) is a Wiener process on the Lie algebra with arbitrary covariance \( S \in \mathbb{R}^{m \times m} \) rather than \( I \).

In this section we consider a quadratic function on \( G \) of the form \( h(X) = \frac{1}{2} \text{tr}(QXNX^\top) \) for some given symmetric \( Q, N \in \mathbb{R}^{n \times n} \), and derive the stochastic dynamics for \( h \). Choose the inner product on \( g \) given by \( \langle \Omega_1, \Omega_2 \rangle = \text{tr}(\Omega_1^\top \Omega_2) \). Given \( X_0 \in G \), we parametrize a neighborhood of \( X_0 \) by \( X_0 e^\Omega \), with \( \Omega \in g \):

\[
X = X_0(I + \Omega + \frac{\Omega^2}{2!} + \ldots).
\]

\( h(X) \) is then expanded to second order in \( \Omega \), leading to

\[
h(X) = h(X_0) + \langle X_0^\top QX_0N, \Omega \rangle + \frac{1}{2} \langle X_0^\top QX_0\Omega N, \Omega \rangle + \frac{1}{2} \langle X_0^\top QX_0N, \Omega^2 \rangle + o(\Omega^2),
\]

where we make use of the matrix trace identity \( \text{tr}(ABC) = \text{tr}(CAB) = \text{tr}(BCA) \). To derive the stochastic dynamics for \( h \), we replace \( \Omega \) by \( X^{-1} \cdot \frac{dX}{dt} = A \, dt + dW \) and \( X_0 \) by \( X \), ignore the higher-order \( o(\Omega^2) \), and apply the standard Itô rules, \( i.e., dt \cdot dt = 0, dw_i \cdot dt = dt \cdot dw_i = 0 \), and \( dw_i \cdot dw_j = \rho_{ij} \, dt \), where \( dw_i, i = 1, \ldots, m \) denote the independent entries of \( dW \in g \), and \( \rho_{ij} \) denotes the correlation between \( \omega_i \) and \( \omega_j \). Based on the above, the dynamics for \( h \) can now be explicitly derived via calculation:
**Proposition 1**  For the bilinear stochastic equation \( dX = XA \ dt + X \cdot dW \), where \( X \in G \), \( A, dW \in g \) with \( A \) constant, and \( dW \) is a diffusion process on \( g \) whose independent entries are denoted by \( dw_i \). Given the quadratic function
\[
h(X) = \frac{1}{2} \text{tr} (QXX^\top),
\]
where \( Q, N \in \mathbb{R}^{n \times n} \) are assumed symmetric, the dynamics for \( h \) are given by
\[
dh = \langle X^\top QX, A \ dt + dW + \frac{1}{2} dW \cdot dW \rangle + \frac{1}{2} \langle X^\top QX \cdot dW \cdot N, dW \rangle,
\]
(11)
where the standard Itô rules \( dt \cdot dt = dw_i \cdot dt = dt \cdot dw_i = 0, dw_i \cdot dw_j = \rho_{ij} \ dt \), with \( \rho_{ij} = \text{correlation between } w_i \text{ and } w_j \), are applied.

The corresponding results for the case when \( h \) is linear in \( X \) can also be derived similarly.

**Proposition 2** Under the same conditions as those given in Proposition 1, but with \( h(X) = \text{tr}(MX) \), \( M \) symmetric, the corresponding dynamics for \( h \) are given by
\[
dh = \frac{1}{2} \langle X^\top M, A \ dt + dW + \frac{1}{2} dW \cdot dW \rangle.
\]
(12)

### 3 Particle Filtering on the Euclidean Group

As in [6], to avoid the many technicalities associated with stochastic calculus on manifolds, we formulate our particle filtering algorithms in a discrete-time setting. The discretization of the state equation (4) must be performed such that, at each iteration, \( X(t) \) always remain on \( G \). Beginning with the work of Crouch and Grossman [7], considerable literature exists on the subject (see the recent works [20], [13]).

The primary motivation in these works is to generalize Runge-Kutta and other numerical integration methods developed for ODEs on \( \mathbb{R}^n \) to general Lie groups. The simplest first-order discretization, and one that is for the most part sufficient for our purposes, is the exponential Euler discretization given by
\[
X_{i+1} = X_i \exp \left( A(X_i) \Delta t + \sum_{j=1}^{m} b_j(X) E_j \sqrt{\Delta t} \epsilon_{i+1,j} \right),
\]
(13)
where each \( \epsilon_i = (\epsilon_{i,1}, \ldots, \epsilon_{i,m}) \) is an \( m \)-dimensional zero-mean Gaussian with specified covariance matrix \( S \). Higher-order discretization methods, and other aspects of the numerical integration of stochastic differential equations, are discussed in [15].
Within this discrete-time setting we now consider the measurement equation

\[ y = h(X) + \eta, \]  

where \( \eta \) is independent Gaussian white noise. Given a set of measurements \( \{y_0, \ldots, y_k\} \equiv y_{1:k} \), the objective is to estimate the marginal density \( \rho(X_k|y_{1:k}) \). Assuming \( \rho(X_{k-1}|y_{1:k-1}) \) is available, and under a first-order Markovian assumption, the predicted density \( \rho(X_k|y_{1:k-1}) \) can be obtained as

\[ \rho(X_k|y_{1:k-1}) = \int \rho(X_k|X_{k-1})\rho(X_{k-1}|y_{1:k-1}) \, d\mu(X_{k-1}), \]  

(15)

where \( d\mu(X) \) denotes the Haar measure on \( G \) (in the case of \( SE(3) \) one can employ the bi-invariant volume form, see, e.g., [5]). If the measurement \( y_k \) at time \( k \) is available, the filtering density \( \rho(X_k|y_{1:k}) \) is obtained by updating the predicted density \( \rho(X_k|y_{1:k-1}) \) via Bayes Rule:

\[ \rho(X_k|y_{1:k}) = \frac{\rho(y_k|X_k)\rho(X_k|y_{1:k-1})}{\int \rho(y_k|X_k)\rho(X_k|y_{1:k-1}) \, d\mu(X_k)}, \]  

(16)

where the denominator is simply a constant normalization factor. We can therefore sequentially estimate \( \rho(X_k|y_{1:k}) \) by repeating the prediction (15) and update (16) steps upon sequential arrival of \( y_k \).

Particle filtering algorithms perform the above sequential estimation via Monte Carlo simulation. In particular, any geometrically well-defined particle filtering algorithm on a group \( G \) must resolve, in addition to the discretization and propagation of the state equations discussed earlier, the following two issues in a coordinate-invariant way: (i) constructing distributions on \( G \); (ii) formulas for the sample mean and covariance based on the the metric structure of \( G \). The construction of coordinate-invariant probability distributions on \( SE(3) \) is detailed in Wang and Chirikjian [38]; the reader is referred to their paper for further details. In what follows we focus on the second issue of geometrically well-defined sample means and covariances.
3.1 Sample Means and Covariances

The Special Euclidean Group

We first consider the special Euclidean group $G = SE(3)$ and its Lie algebra $g = se(3)$; recall that these admit the matrix representations

$$
\begin{bmatrix}
R & p \\
0 & 1
\end{bmatrix} \in SE(3), \quad \begin{bmatrix}
[\omega] & v \\
0 & 0
\end{bmatrix} \in se(3), \quad [\omega] = \begin{bmatrix}
0 & -\omega_3 & \omega_2 \\
\omega_3 & 0 & -\omega_1 \\
-\omega_2 & \omega_1 & 0
\end{bmatrix}, \quad (17)
$$

where $R \in SO(3)$, $p, \omega, v \in \mathbb{R}^3$. There is extensive literature on distance metrics on $SE(3)$ that we do not recount here; we instead cite the main result of Moakher [18], who shows that the mean rotation (in the Euclidean sense) of $N$ rotations $\{R_1, \ldots, R_N\}$, defined as the $R \in SO(3)$ that minimizes

$$
\arg \min_{R \in SO(3)} \sum_{i=1}^{N} \| R_i - R \|_F^2, \quad (18)
$$

where $\| \cdot \|_F$ denotes the Frobenius norm, is given by the orthogonal projection of $\bar{R} = \frac{1}{N} \sum_{i=1}^{N} R_i$. This can be explicitly evaluated as

$$
R = \begin{cases} 
V U^\top, & \text{if } \det(\bar{R}^\top) > 0, \\
V H U^\top, & \text{otherwise},
\end{cases} \quad (19)
$$

where $U$ and $V$ are obtained from the singular value decomposition of $\bar{R}^\top$, i.e., $\bar{R}^\top = U \Sigma V^\top$, and $H = \text{diag}[1, 1, -1]$. The sample mean on $SE(3)$ can be obtained by augmenting the previous sample mean formula for $SO(3)$ with the traditional algebraic sample mean for the translational component in $SE(3)$. Alternatively, the mean rotation in the Riemannian sense is defined as the $R$ that minimizes

$$
\arg \min_{R \in SO(3)} \sum_{i=1}^{N} \| \log(R_i^\top R) \|_F^2. \quad (20)
$$

Closed-form solutions to the above are generally not available, however.

We remark that any distance metric on $SE(3)$ fundamentally involves the combination of rotational and translational units, and there is no natural way to do this. In this sense any distance metric on $SE(3)$ will be ad hoc to some extent; this will be captured by the choice of length scale for physical space. The reference [22] contains further discussion of these issues, as well as suggested heuristics for choosing a length scale appropriate for the task at hand.
Given a set of elements \( \{X_1, \ldots, X_n\} \) of some Lie group \( G \), whose mean is denoted \( \bar{X} \), the associated sample covariance matrix is defined geometrically as follows. First, denote the minimal geodesic from the mean \( \bar{X} \) to \( X_i \) by \( \gamma_i(t) \), such that \( \gamma_i(0) = \bar{X} \) and \( \gamma_i(1) = X_i \). Then clearly \( \dot{\gamma}_i(0) \in g \). Representing \( \dot{\gamma}_i(0) \) in vector form as \( v_i \), the sample covariance \( \Sigma \) is then defined in the usual way as

\[
\Sigma = \frac{1}{n+1} \sum_{i=1}^{n} v_i v_i^\top.
\]

Assuming the chosen Riemannian metric on \( G \) is left (right) invariant, it can be seen that the resulting \( \Sigma \) as constructed above will also be left (right) invariant. The sample covariance on \( SE(3) \) can be constructed using our previous formulas for calculating means and minimal geodesics on \( SE(3) \). [38] provides a more detailed discussion of covariances on \( SE(3) \).

The Symmetric Positive-Definite Matrices

Recall that covariance matrices are characterized by being symmetric and positive-definite; we denote the space of such matrices by \( P(n) \), and note that it is not a group because it is not closed under matrix multiplication. A Riemannian structure on \( P(n) \) whose tangent space can be identified as \( S(n) \), the space of symmetric \( n \times n \) real matrices, can be constructed by the Riemannian metric given by

\[
(X,Y)_P = \text{tr}(P^{-1}XP^{-1}Y).
\]

Then the length of a curve \( P(t) \in P(n) \), \( a \leq t \leq b \), is given by

\[
L(P) = \int_a^b \sqrt{\text{tr}((P^{-1}(t)\dot{P}(t))^2)} \, dt.
\]

This notion of length is invariant not only under reparametrizations of \( [a,b] \), but also under congruent transformations of the form \( GPG^\top \), where \( G \) is any fixed element in \( GL(n) \). Using the fact that \( P(n) \) is a complete space (i.e., the geodesics are well-defined for all \( t \)), the minimal geodesic \( \gamma(t) : [0,1] \rightarrow [A,B] \) connecting two points \( A, B \in P(n) \) is given by

\[
\gamma(t) = A^{1/2}(A^{-1/2}BA^{-1/2})^tA^{1/2} = \Gamma_{A^{1/2}}(\Gamma_{A^{-1/2}}(B))^t
\]

where the square roots are taken to be symmetric positive-definite. Defining the distance between \( A \) and \( B \) in the usual way by the length of the above minimal geodesic, we have

\[
d(A,B) = \left( \sum_{i=1}^{n} \log^2 \lambda_i \right)^{1/2},
\]
where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of the matrix $AB^{-1}$. Since $AB^{-1}$ is similar to $A^{-1/2}(AB^{-1})A^{1/2}$, which in turn is symmetric positive-definite, the eigenvalues of $AB^{-1}$ are all positive, and $\log \lambda_i$ is well defined for each $i$. Note also that $d(A, \gamma(t)) = td(A, B)$. The references [10], [16], and [27] provide further details on the metric structure of $P(n)$.

With the above metric structure on $P(n)$, we now discuss sample means and covariances on $P(n)$. A widely used formula for the sample mean of $N$ symmetric positive-definite matrices $\{P_1, \ldots, P_N\}$ is the arithmetic mean $\frac{1}{N} \sum_{i=1}^{N} P_i$. While the arithmetic mean clearly lies in $P(n)$, it has a number of undesirable properties (e.g., the arithmetic mean of matrices with equal determinant can have a larger determinant). We therefore focus on the intrinsic mean, defined as

$$\arg \min_{\bar{P} \in P(n)} \sum_{i=1}^{N} d(\bar{P}, P_i)^2. \quad (25)$$

For the case of two points the intrinsic mean is simply the midpoint of the minimal geodesic. For arbitrary $N$ the above intrinsic mean is unique on $P(n)$. [10] provides a simple steepest descent algorithm for numerically obtaining the intrinsic mean, using the gradient of (25) given by $\sum_{i=1}^{N} \log(\bar{P} P_i^{-1})$.

Definition of the sample covariance on $P(n)$, as well as its numerical computation, are discussed in [16] and [10]. Given $N$ elements $P_1, \ldots, P_N$, and an intrinsic mean $\bar{P}$, the covariance matrix relative to $\bar{P}$ is defined by

$$\Sigma_{\bar{P}} = \frac{1}{N} \sum_{i=1}^{N} V_i V_i^\top, \quad (26)$$

where $V_i$ is the tangent vector at $\bar{P}$ such that the geodesic $\gamma(t)$ goes through $\gamma(0) = \bar{P}$ and $\gamma(1) = P_i$ with $\dot{\gamma}(0) = V_i$. Based on these mean and covariance formulas, a generalized normal distribution on $P(n)$ is constructed by taking the curvature into account; see [16] for details.

We remark in closing that in addition to covariances, there exist other examples from robotics in which the geometry of $P(n)$ plays a central role, e.g., inertia and stiffness matrices, either in joint or task space, are all symmetric positive-definite.

### 3.2 Particle Filtering with Unknown Covariance

We now generalize the combined state-parameter estimation particle filter of Liu and West [17] to $SE(3)$, where the parameter space is restricted to be $P(n)$ (i.e., the noise covariance $S$ is assumed unknown a
priori). We work with the discretized state equations of (13) and the measurement equation \( y = h(X) + \eta \), with \( \eta \) Gaussian white noise. The filtering problem on \( SE(3) \) that we consider is the estimation of the state \( X(t) \) and \( S \), the covariance of the \( dw_i \)'s, given the time series of measurements, \( y(t_0), y(t_1), \ldots, y(t_n) \). Clearly the covariance \( S \) of the \( m \)-dimensional Wiener process \( dw(t) \) strongly affects the pose estimates, and in general it is difficult to a priori select an appropriate value of \( S \). Instead, one can simultaneously estimate the pose and the relevant covariance parameters, by embedding the covariance parameters into the state. This is a fairly common technique in system identification, and the mathematical and engineering justification for this technique in the context of particle filtering is covered in, e.g., [14], [17], [12], [33].

Collecting the unknown parameters of the covariance \( S \) into a vector \( \theta \), the state \( X_t \) can be augmented with \( \theta_t \) as \( \{X_t, \theta_t\} \). To improve filter performance one usually adds an artificial dynamics to the static parameter \( \theta_t \), e.g., \( \theta^{(i)}_t \sim N(\theta^{(i)}_{t-1}, V_{t-1}) \) where the covariance \( V_{t-1} \) is decreasing with time. While the mean \( \bar{\theta}_{t-1} \) of the particles associated with \( \theta_{t-1} \) remains the same at both \( t \) and \( t-1 \), the covariance \( C_{t-1} \) now increases to \( C_{t-1} + V_{t-1} \).

Liu and West [17] explicitly point out this undesirable “over-dispersion” or “loss of information” problem, and propose kernel smoothing with shrinkage as a remedy, with normal kernels of the form \( N(a\theta^{(i)}_{t-1} + (1-a)\bar{\theta}_{t-1}, h^2C_{t-1}) \) used for each parameter particle \( \theta^{(i)}_t \). In this case the mean is still preserved, while the covariance \( C_t \) now becomes \( (a^2 + h^2)C_{t-1} \); by setting \( a = \sqrt{1-h^2} \), the covariance \( C_t \) can be matched to \( C_{t-1} \), and the “over-dispersion” induced by the artificial noise corrected. A discount factor \( \delta \) with typical values between 0.95 and 0.99 is used as the control factor, with \( a = (3\delta - 1)/2\delta \) [17].

Since in our case the unknown parameter \( \theta_t \) evolves on \( P(n) \), kernel smoothing must be performed to always ensure that \( \theta_t \) lies on \( P(n) \). Means and covariances on \( P(n) \) can be calculated from (25) and (26).

The kernel mean, \( a\theta^{(i)}_{t-1} + (1-a)\bar{\theta}_{t-1} \), can be understood as the corresponding point on the geodesic \( \gamma(t) \) connecting \( \theta^{(i)}_{t-1} \) and \( \bar{\theta}_{t-1} \) with \( \gamma(0) = \theta^{(i)}_{t-1} \) and \( \gamma(1) = \bar{\theta}_{t-1} \); that is, \( a\theta^{(i)}_{t-1} + (1-a)\bar{\theta}_{t-1} = \gamma(1-a) \), which can be calculated from (23). Gaussian sampling with the kernel mean \( \mu \) and covariance \( \Sigma_\mu \) (performed in Step 2(c)-ii of the algorithm below) can be realized following the method described in [16], i.e., perform the Cholesky decomposition \( \Sigma_\mu = CC^T \), generate a zero-mean, unit-variance Gaussian random vector \( z \in \mathbb{R}^{n(n+1)/2} \) and form \( V = Cz \), and compute \( P = \mu^{1/2} \exp(\mu^{-1/2}[V]\mu^{-1/2}) \mu^{1/2} \), where \([V] \) denotes the
original vector $V$ reshaped as an element of $S(n)$.

With the above geometric constructions, we present the particle filtering algorithm for simultaneous covariance and state estimation, employing kernel smoothing with shrinkage (in what follows we continue to use the vector space kernel mean notation for the geometric kernel mean of (23)).

**Algorithm**

1. **Initialization:** $t = 0$
   
   - Set number of particles $N$.
   - Set $\delta$ between $0.95 \sim 0.99$.
   - For $i = 1, \ldots, N$, draw the states $X_0^{(i)}$ and the covariance parameters $\theta_0^{(i)}$ for $S$ from the priors $\rho(X_0)$ and $\rho(\theta_0)$, respectively.

2. **Importance Sampling:**
   
   (a) Set $t = t + 1$.
   
   (b) Compute the intrinsic mean $\bar{\theta}_{t-1}$ from (25) and covariance $\Sigma_{\bar{\theta}_{t-1}}$ of $\{\theta_{t-1}^{(i)}, i = 1, \cdots, N\}$ from (26).
   
   (c) For $i = 1, \ldots, N$, draw $X_t^{(i)} \sim \rho(X_t|X_{t-1}^{(i)}, \theta_{t-1}^{(i)])$, i.e.,
      
      i. Determine the kernel mean $a\bar{\theta}_{t-1}^{(i)} + (1-a)\bar{\theta}_{t-1}$ from (23).
      
      ii. Draw $\theta_{t}^{(i)}$ from $N(a\bar{\theta}_{t-1}^{(i)} + (1-a)\bar{\theta}_{t-1}, h^2\Sigma_{\bar{\theta}_{t-1}})$ where $a = (3\delta - 1)/2\delta$ and $h^2 = 1 - a^2$.
      
      iii. Generate the Gaussian $\epsilon_i$ from $N(0, \sigma_{\epsilon_i}^2 \Delta t)$, and propagate $X_{t-1}^{(i)}$ to $X_{t}^{(i)}$ via
          
          $$X_{t}^{(i)} = X_{t-1}^{(i)} \exp \left( A(X_t, t) \Delta t + \sum_{j=1}^{N} b_j(X) E_j \sqrt{\Delta t} \epsilon_{t,j} \right).$$

   (d) For $i = 1, \ldots, N$, weight each draw by
      
      $$w_t^{(i)} \propto \rho(y_t|X_t^{(i)}).$$

   (e) For $i = 1, \ldots, N$, normalize the importance weights:
      
      $$\tilde{w}_t^{(i)} = w_t^{(i)} \left[ \sum_{j=1}^{N} w_t^{(j)} \right]^{-1}$$
3. Resampling

(a) Resample from \( \{X_i^{(1)}, \ldots, X_i^{(N)}\} \) and \( \{\theta_i^{(1)}, \ldots, \theta_i^{(N)}\} \) with probability proportional to \( \tilde{w}_i^{(i)} \) to produce a random sample \( \{X_i^{(1)}, \ldots, X_i^{(N)}\} \) and \( \{\theta_i^{(1)}, \ldots, \theta_i^{(N)}\} \).

(b) For \( i = 1, \ldots, N \), set \( w_i^{(i)} = \tilde{w}_i^{(i)} = \frac{1}{N} \).

4. Go to Importance Sampling Step

Various methods exist for drawing \( X_0^{(i)} \) and \( \theta_0^{(i)} \) from the priors \( \rho(X_0) \) and \( \rho(\theta_0) \) in the initialization. For our purposes, given some initial \( X_0 \), we set \( X_0^{(i)} = X_0 \exp(N) \) for some randomly generated \( N \in se(3) \) (e.g., zero-mean Gaussian). For \( \rho(\theta_0) \), [12] suggests a uniform independent distribution for each element. Since in our case the parameter space is \( P(n) \), we use the following simple but \textit{ad hoc} method: sample a symmetric matrix \( S \) from a uniform distribution, obtain its singular value decomposition \( S = U\Sigma V^\top \), and replace \( V \) with \( U \) to make the sampled symmetric matrix positive definite.

For practical implementation purposes, in our case studies we assume \( S \) is block-diagonal, i.e., \( S = \text{Diag}[S_1, S_2] \), where \( S_1, S_2 \in P(3) \) are covariances for the orientation and position, respectively. The output of the algorithm is a set of samples that can be used to approximate the posterior distribution of \( X(t) \) on the Lie group:

\[
\pi(X_t|y_t) \approx \hat{\pi}(X_t|y_t) = \frac{1}{N} \sum_{i=1}^{N} \delta(X_t^{-1}X_t^{(i)}),
\]

where \( \delta(X) = 1 \) for \( X = I \) and 0 otherwise. The optimal estimate \( \hat{X}(t) \) is then obtained using the sample mean formula on \( SE(3) \) given earlier.

4 Application to GPS Attitude Estimation

As our first case study, we consider the problem of attitude estimation using global positioning system (GPS) data as measurements. Following the physical setup as described in [23], the state equations can be written in the form

\[
dR = RA_t \, dt + R \, dW,
\]

where \( R \in SO(3), \) \( dW \in so(3), \) and \( A_t \in so(3) \) possibly time-varying. The noise accounts for errors due to various factors such as parameter uncertainty, unmodelled dynamics, etc. The measurement equations
can be conveniently expressed in matrix form as

\[ Y = S^\top RB + V, \tag{32} \]

where the measurements are arranged as elements of the matrix \( Y \in \mathbb{R}^{n \times m} \), \( B = (b_1, b_2, \ldots, b_m) \in \mathbb{R}^{3 \times m} \) and \( S = (s_1, s_2, \ldots, s_n) \in \mathbb{R}^{3 \times n} \) are given matrices obtained from the measurement data, and the elements of \( V \in \mathbb{R}^{n \times m} \) represent standard independent identically distributed Gaussian noise. Note that the measurement equations are linear in \( R \); it is straightforward to rearrange them in the form

\[ y_{ij} = \text{tr}(M_{ij}R) + v_{ij}, \tag{33} \]

\( i = 1, \ldots, n, \ j = 1, \ldots, m \), with \( M_{ij} \) appropriately defined matrices. We thus have a set of bilinear state equations on \( SO(3) \), with measurements that are linear in the state \( R \).

We perform simulation studies of the particle filter assuming that \( S \) is known a priori (only state estimation is performed). Arbitrary constant values are chosen for \( A \) and \( M \), and the measurement data generated accordingly with the given noise. Figure 1 shows the errors (in degrees), in which the variance of the measurement noise is taken to be 0.001 \( m \). Figure 2 shows the corresponding errors when the measurement noise variance is increased to 0.01 \( m \). The rows in the graph correspond to estimation errors in the roll, pitch, and yaw angles, respectively. Table 1 shows the error values averaged over 100 simulations.

<table>
<thead>
<tr>
<th>Noise variance</th>
<th>Roll</th>
<th>Pitch</th>
<th>Yaw</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001 m</td>
<td>1.3623</td>
<td>1.9958</td>
<td>1.671</td>
</tr>
<tr>
<td>0.01 m</td>
<td>4.0091</td>
<td>5.4993</td>
<td>4.1445</td>
</tr>
</tbody>
</table>

Table 1: Average of estimation errors over 100 simulations.

5 Application to Needle Tip Estimation

In [39] a kinematic model of needle steering is developed that captures the nonholonomy introduced by the bevel tip. Based on this work, in [24] a probabilistic needle motion planning algorithm is presented, in which the ensemble of reachable states of the needle tip are obtained via Monte Carlo simulation,
and the inverse kinematics solved taking into account the resulting distribution. The performance of the method depends to a considerable extent on the accuracy of the noise model used in the simulation. In this case study we consider the problem of estimating both the needle tip’s Cartesian position and the covariance of the noise model via our particle filtering algorithm.

Because of the bevel tip, straight insertion of the needle without any rotation will result in the needle tip tracing a screw-like trajectory. The needle's high torsional stiffness means that any twisting of the needle at the insertion point is transmitted to the tip motion. The needle is further deflected by inhomogeneous properties of the tissue [21]. In addition to the time-varying translational and rotation insertion speeds of the needle at the entry point, the primary factors determining the needle motion are properties of the ambient material and the geometry of the bevel tip.

To capture the uncertainties in the tip motion, we adopt the kinematic model proposed in [24], but with a general six-dimensional Wiener noise term; the state equation is of the form $dX = XA dt + X dW$,
where $dW \in se(3)$ is standard Wiener noise with a covariance $S$, and $A = (\omega, v) \in se(3)$ is of the form

$$\omega = (\kappa u_1, 0, u_2)$$

$$v = (0, 0, u_1)$$

Here $\kappa$ is the curvature that describes the amount of bending, and $u_1$ and $u_2$ are respectively the translational and rotational insertion speeds along the needle at the point of insertion. The term $\kappa u_1$ reflects the fact that the needle will bend with curvature $\kappa$, depending on the bevel angle and material properties of the tissue.

Currently, electromagnetic tracking is usually used to track the needle tip’s Cartesian position in an intra-operative and minimally-invasive way [29], [2], [28]. Methods for measuring the orientation of the bevel tip have yet to be developed, and because of a lack of observability it is not possible to determine the needle tip orientation from position measurements alone. Errors in electromagnetic tracking are also inevitable. Denoting the rotation and translation components of the state $X$ by $R \in SO(3)$ and $p \in \mathbb{R}^3$, respectively, the measurement equations for the needle tip position $y \in \mathbb{R}^3$ with respect to the fixed
Figure 3: Needle tip position estimation errors (solid line) and the measurement errors (dotted line) with the measurement noise covariances $Q = \text{Diag}(1, 1, 1)$ (top) and $Q = \text{Diag}(5, 5, 5)$ (bottom). Blue, green, and red lines represent (1,4), (2,4) and (3,4) elements of the frame.

The electromagnetic frame are assumed to be of the form

$$y = p + n$$  \hspace{1cm} (34)

where $n$ represents three-dimensional Gaussian noise with covariance $Q$.

For our simulation experiments, the actual data is generated via simulation of the state equations with Gaussian noise of covariance $S = \text{Diag}(S_1, S_2)$, where $S_1 = \text{Diag}(0.005, 0.005, 0.005)$ and $S_2 = \text{Diag}(0.5, 0.5, 0.5)$. The particle filter is run with Gaussian measurement noise of covariance $Q = \text{Diag}(1, 1, 1)$. For the particle filter we assume the $S_1$ to be estimated is diagonal, whereas no assumptions are made about $S_2$ (other than that it is symmetric positive-definite).

The results of the particle filtering estimation are shown in the top of Figure 3. It can be seen that the needle tip position is estimated reasonably accurately by our filter. The norm of the position estimation errors is 5.8845, while that of the measurement errors is 11.6944. The estimated covariances $\hat{S}_1$ and $\hat{S}_2$ are:

$$\hat{S}_1 = \begin{bmatrix} 0.002784 & 0 & 0 \\ 0 & 0.005202 & 0 \\ 0 & 0 & 0.003501 \end{bmatrix}$$  \hspace{1cm} (35)
\[ \hat{S}_2 = \begin{bmatrix} 0.5223 & -0.04909 & 0.2183 \\ -0.04909 & 0.4603 & -0.08547 \\ 0.2183 & -0.08547 & 0.4776 \end{bmatrix} \]  

The same simulation experiments are performed, but this time with increased Gaussian measurement noise of covariance \( Q = \text{Diag}(5, 5, 5) \); the results are shown in the bottom of Figure 3. Due to the larger measurement noise, the estimation accuracy is noticeably diminished compared to the prior simulation. However, even in this case, note that the position estimation errors are smaller than the measurement errors. The norm of the position estimation errors is 12.4053, while that of the measurement errors is 24.2633. The estimated covariances \( \hat{S}_1 \) and \( \hat{S}_2 \) are:

\[ \hat{S}_1 = \begin{bmatrix} 0.007575 & 0 & 0 \\ 0 & 0.008393 & 0 \\ 0 & 0 & 0.005072 \end{bmatrix} \]  

\[ \hat{S}_2 = \begin{bmatrix} 0.5585 & 0.03821 & 0.2778 \\ 0.03821 & 0.3431 & 0.007572 \\ 0.2778 & 0.007572 & 0.8104 \end{bmatrix} \]  

In the above two simulation experiments, position estimation is possible to some degree even in the case of unknown covariance. The covariance estimation results are not as accurate, although it is evident that the diagonal elements of \( \hat{S}_2 \) are clearly dominant. Better filtering performance can be expected in the event that a technique for measuring both the position and orientation of the needle tip becomes available.

6 Application to Vision-Based End-Effector Pose Estimation

For our third and most detailed case study, we consider the problem of estimating the end-effector pose of a robot using vision sensors. For robots in typical laboratory or industrial settings, one usually has available accurate kinematic models of the robot (often obtained via a kinematic calibration procedure involving the use of laser, ball-bar, or other position measurement sensors of reasonably high accuracy), together with accurate joint encoders, so that it is enough to evaluate the forward kinematics for reliably
determining the end-effector pose.

Things change considerably for low-cost personal robots operating in unstructured home settings. The kinematic models provided by the manufacturer are often inaccurate and unreliable, and high precision sensors are usually not available for performing kinematic calibration. Further compounding matters is that the robots themselves are subject to greater manufacturing tolerances (and thus greater errors), resulting in a greater degree of parametric errors, backlash, friction, joint and link elastic deformations, and tracking control errors. Torque-level control is also typically not feasible, making attempts to construct accurate dynamic models and associated observers moot.

The vision-based pose estimation algorithm presented here is intended for a low-cost, six degree-of-freedom serial chain robot. We also assume a low-cost noisy vision sensor (such as a webcam) capable of measuring a pre-specified set of feature points on the end-effector, and a nominal kinematic model with potentially large parameter errors. Because of the relatively noisy vision sensor, accurate end-effector pose estimation based on vision measurements alone is inadequate (not least of all because the classical depth ambiguity problem cannot be easily resolved in our setting); it is essential to make use of the vision measurements in conjunction with the available kinematic model.

To make explicit the connection between the kinematic equations and the Lie algebra $se(3)$, we choose to express the nominal forward kinematics of the serial chain in product of exponentials form, i.e.,

$$ X = e^{A_1 q_1} e^{A_2 q_2} \cdots e^{A_6 q_6} M $$

where $X \in SE(3)$ represents the location of the end-effector frame relative to the fixed frame, $q_1, \cdots, q_6$ denote the joint variables, $A_1, \cdots, A_6 \in se(3)$ denote the kinematic parameters, and $M \in SE(3)$ denotes the end-effector location when the robot is in its home position. In our application we assume the kinematic parameters $A_i \in se(3)$ and $M \in SE(3)$ are not known exactly, contributing to the positioning errors arising from other effects such as friction, backlash, joint and link elasticity, and tracking control errors.

For the vision sensor, we assume that a single perspective projection camera is available, and that the image plane coordinates can be measured for a fixed number $n$ of feature points on the end-effector frame. Let $Y = (y_1, \cdots, y_n)$ denote the measurements from the vision sensor, where each $y_i$ represents the 2-D homogeneous coordinates of feature point $i$ in the image plane and can be represented as a function of
\( y_i = K \begin{bmatrix} R & -Re \end{bmatrix} X x_i \). 

(40)

Here \( R \in SO(3) \) is the camera rotation matrix, \( \hat{c} \in \mathbb{R}^3 \) is the camera center in the fixed frame, \( x_i \) represents the 3-D homogeneous coordinates of each feature point attached to the end-effector with respect to the end-effector frame, and \( K \in \mathbb{R}^{3 \times 3} \) is the camera calibration matrix of the form

\[
K = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix},
\]

with \( f \) the focal length and \((p_x, p_y)\) the principal point on the image plane. The measurement equation can be expressed in the form \( Y = h(X) \), where \( h(\cdot) \) represents the aforementioned perspective projection mapping onto the 2-D image plane.

We choose to express the kinematic state equations in right-invariant form, i.e.,

\[
dX = V(u_1, \cdots, u_6) \cdot X \cdot dt + dW \cdot X
\]

(42)

\[
V(u_1, \cdots, u_6) = A_1 u_1 + Ad_{\hat{c}_1 q_1} (A_2) u_2 + \cdots + Ad_{\hat{c}_1 q_1 \cdots c_5 q_5} (A_6) u_6
\]

(43)

\[
Y = h(X) + \eta.
\]

(44)

where \( Ad \) denotes the adjoint mapping on \( SE(3) \) (i.e., \( Ad_X(V) = XVX^{-1} \) for \( X \in SE(3) \) and \( V \in se(3) \)), and \( u_i = \dot{q}_i \), \( i = 1, \cdots, 6 \) the joint velocity inputs. The joint velocities are taken to be the control inputs. Equations (42) and (44) are discretized as

\[
X_t = exp(V(u_1, \cdots, u_6) \Delta t + dW) \cdot X_{t-1}
\]

(45)

\[
Y_t = h(X_t) + \eta.
\]

(46)

Experiments are performed with the Cycloid 3 shown in Figure 4(a), a small, inexpensive 19-DOF humanoid robot developed for educational and entertainment purposes. For the experiment we fix all joints except for the waist joint, and the three joints in the right arm, so that we can effectively regard the Cycloid as a 4-DOF open chain. For the vision sensor we use the 640 \( \times \) 480 resolution 3Com HomeConnect PC Digital WebCam shown in Figure 4(b).

To evaluate the experimental performance of our pose estimation algorithm, we use the Krypton GCS300 real-time stereo camera and sensor probe system (see Figure 4(b)) to measure the actual 3-D
Figure 4: (a) The Cycloid 3 currently being used in experimental studies. (b) The “3Com HomeConnect PC Digital WebCam”, “Krypton GCS300” stereo camera, and its probe.

Pose of the end-effector frame with respect to the fixed frame. The probe, now regarded as the end-effector frame, is attached to the right hand of the Cycloid, and the camera frame is taken to be the fixed frame. The four infrared LED’s on the probe are used as feature points for the end-effector frame.

We use the direct linear algorithm of [11] to obtain the camera matrix of the webcam, from a sufficiently large set of point correspondences between the measured 3-D points of the four feature points and their 2-D corresponding points on the image plane. Nominal values for the Cycloid’s highly imprecise kinematic parameters (which are not provided by the manufacturer) are obtained via direct measurement, and are shown in Table 2.

The Cycloid is manipulated for 20 seconds according to a set of pre-specified joint control inputs, while the webcam captures the end-effector feature points every 0.5 seconds. The variance of the additive

<table>
<thead>
<tr>
<th>Joint</th>
<th>$\omega$</th>
<th>$v$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joint 1</td>
<td>(0.9651, -0.1075, -0.2389)</td>
<td>(-199.3, -1834.4, 20.2)</td>
</tr>
<tr>
<td>Joint 2</td>
<td>(-0.1700, -0.9507, -0.2593)</td>
<td>(-1790.9, 311.7, 31.7)</td>
</tr>
<tr>
<td>Joint 3</td>
<td>(0.1993, -0.2909, 0.9358)</td>
<td>(-646.7, -344.4, 30.7)</td>
</tr>
<tr>
<td>Joint 4</td>
<td>(-0.1700, -0.9507, -0.2593)</td>
<td>(-1793.6, 331.5, -39.1)</td>
</tr>
</tbody>
</table>

Table 2: Kinematic parameters for each joint of the Cycloid 3.
Gaussian noise $\eta$ is heuristically set to 4. Trajectories of the feature points captured by the webcam, i.e., $Y_{1:40}$, are shown in Figure 5. We set $\delta$ to 0.99 and initialize the covariance matrices. For $S_1$, a symmetric matrix is uniformly sampled between 0 and 0.00001 for the diagonal elements, and between -0.000005 and 0.000005 for the off-diagonal elements. For $S_2$, a uniform distribution between 0 and 1 is used for the diagonal elements, and between -0.5 and 0.5 for the off-diagonal elements. Both $S_1$ and $S_2$ are then forced to be symmetric positive definite matrices using the method indicated earlier.

Figure 6 shows the end-effector pose trajectory $X_{1:40}$ as measured by the Krypton GCS300. The end-effector pose trajectory calculated from the nominal forward kinematics equation (39), $\tilde{X}_{1:40}$, is also shown together as the dotted line, clearly showing large errors from the actual end-effector pose.

5000 particles are used in the filtering algorithm and initialized from $\tilde{X}_0$. The estimated end-effector pose trajectory $\hat{X}_{1:40}$ is shown in Figure 7, while Figure 8 shows the estimation errors. The errors between the measured end-effector frame and that obtained from the nominal forward kinematics frame are shown together as a dotted line for comparison. The norms of the errors for $\hat{X}_{1:40}$ are 0.2495 for the orientation component and 9.6908 for the position, while those for $\tilde{X}_{1:40}$ are 0.5454 and 88.1265, respectively. The orientation component of the errors are clearly less severe compared to position errors. Both orientation and position estimation accuracy are improved, with the position components showing particularly large improvement. Finally, Figure 9 shows the elements of the estimated covariances.
Figure 6: The measured end-effector frame (solid line) and ideal end-effector frame (dotted line). Top: Blue, green, and red lines represent the three independent elements of the logarithm of the rotation matrix. Bottom: Blue, green, and red lines represent the (1,4), (2,4), and (3,4) elements of the frame.

Figure 7: The measured end-effector frame (solid line) and estimated end-effector frame (dotted line). Top: Blue, green, and red lines represent the three independent elements of the logarithm of the rotation matrix. Bottom: Blue, green, and red lines represent the (1,4), (2,4), and (3,4) elements of the frame.
Figure 8: The estimation errors (solid line) and the errors between the measured end-effector frame and the ideal end-effector frame (dotted line). Top: Blue, green, and red lines represent the errors for the three independent elements of the logarithm of the rotation matrix. Bottom: Blue, green, and red lines represent the errors for the (1,4), (2,4), and (3,4) elements of the frame.

Figure 9: The estimated covariance parameters.
7 Conclusions

This paper has investigated particle filtering on the Euclidean group $SE(3)$. Assuming the covariance of the state space noise model is \textit{a priori} unknown, we generalize the particle filter of Liu and West [17] for simultaneously estimating state and model parameters to $SE(3) \times P(n)$, where $P(n)$ denotes the space of $n \times n$ symmetric positive-definite matrices. A second set of contributions are three case studies—involving GPS attitude estimation, needle insertion, and vision-based end-effector pose estimation—that illustrate the potential applications of the filter to diverse problems in robotics.

We discuss both the theoretical underpinnings of the filter and the implementation and performance issues. The geometry of $SE(3)$ and $P(n)$ in particular need to be accounted for in any generalization of vector space particle filtering algorithms to these manifolds. In this regard we discuss the metric structure, formulas for the sample mean and covariance, the construction of distributions, discretization and propagation of the state equations, and other geometric issues involving the two manifolds.

The framework and examples presented in the paper represent just one aspect of the larger problem of Monte Carlo estimation on Lie groups, all presented in a robotics context, and as such several open research problems can be posed in a more general setting:

- **Extending the estimation algebra approach to stochastic systems on Lie groups:** extending the recent work of [40] to matrix Lie groups, it may be possible to construct finite-dimensional filters for certain classes of stochastic systems on Lie groups. The bilinear case on basic groups such as $SO(3)$ would be the simplest and most obvious candidate for investigation.

- **Determining the minimum variance linear filter for bilinear systems on Lie groups:** The minimum variance linear filter for bilinear systems on vector spaces has been derived by Vallot [35], and determining a corresponding result for bilinear systems on matrix Lie groups would be of interest. In principle one can recast the Lie group bilinear system into its corresponding vector bilinear form, by simply arranging the elements of the matrix $X$ as a vector. However, the optimal vector space linear filter will typically produce estimates that deviate from the Lie group. A geometric construction of the minimum variance linear filter based on fundamental principles, and taking into account the geometric structure of the underlying space, would be of interest.
• **Generalization of other particle filtering algorithms**: There exist many types of particle filtering algorithms, and a geometric generalization of these algorithms to the Euclidean group would be of interest.

• **Improving the computational efficiency of the algorithms**: The algorithms in their current form are computationally quite intensive, and online estimation is possible only under various simplifying assumptions. Ways to improve the computational efficiency are clearly desirable.

• **Extension to other Lie groups and other applications**: There exist many other matrix Lie groups that arise in various engineering applications, and it would be of interest to derive geometric particle filtering algorithms for other Lie groups. For example, the group $SL(3)$ of volume preserving linear transformations plays a prominent role in vision and medical imaging, as does the affine group. The case studies presented in this paper are also just a small sample of the possible applications of geometric particle filtering. Both the needle insertion and vision-based pose estimation algorithms have room for improvement, and object tracking can be approached by, *e.g.*, taking the dynamic equations into account, and even attempting to identify the inertial and mass properties (which present another opportunity for estimation on $P(n)$).

These are but a small sample of potential research problems in this domain that would benefit from a collaboration among geometers, control theorists, and roboticists.

**Acknowledgements**

Financial support for this research was provided in part by the Institute of Advanced Machinery and Design, the College of Engineering Research Foundation, all at Seoul National University, and by the Center for Intelligent Robotics at KIST. The authors wish to acknowledge the contributions of Wooram Park, who brought the needle estimation case study to our attention.
References


