Diffusion Particle Filtering on the Special Orthogonal Group Using Lie Algebra Statistics

Claudio J. Bordin Jr. Ⓞ, Member, IEEE, Caio G. de Figueiredo Ⓞ, and Marcelo G. S. Bruno Ⓞ, Senior Member, IEEE

Abstract—In this letter, we introduce new distributed diffusion algorithms to track a sequence of hidden random matrices that evolve on the special orthogonal group. The algorithms are based on the Adapt-then-Combine and the Random Exchange methods, and diffuse Gaussian approximations of posterior densities computed in the Lie algebra of the special orthogonal group. Simulation results show that, in scenarios with nonlinear observation functions, the proposed algorithms perform closely to the centralized particle filter estimator and can outperform competing Extended Kalman Filters.

Index Terms—Statistics on manifolds, matrix lie groups, diffusion method, particle filters.

I. INTRODUCTION

A MULTITUDE of applications in engineering, for example, in signal [1], [2], [3], [4], [5], [6], [7] and image processing [8], robotics [9], [10], [11], and attitude estimation and control [12], [13], [14], [15], require the estimation of variables [16] with built-in nonlinear constraints, which leads them to be naturally modeled as points on a manifold. Although methods developed to process unconstrained variables can be used in such scenarios, achieving optimal performance requires approaches that properly take into account the underlying geometry. Additionally, in many such applications, it is advantageous to distribute the computational burden of estimation methods among separate remote nodes that cooperate with each other via message passing over a partially connected communication network. Notwithstanding the sustained interest in distributed filtering [17], [18], [19], [20], most of the proposed algorithms found in the literature only apply to the case in which both the observations and the hidden variables lie in Euclidean spaces. On the other hand, previous works on estimation of manifold-constrained variables [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] do not consider distributed processing architectures.

Particle filtering [21], [22], [23], [24], [25] is now a well-established technique to track the states of systems in nonlinear/non-Gaussian setups. In previous works [26], [27], we introduced diffusion particle filter (PF) algorithms for fully distributed estimation of hidden variables that were constrained to the unit hypersphere and the Stiefel manifold, respectively. The algorithms in [26], [27] employed both the Random Exchange (RndEx) [28] and the Adapt-then-Combine (ATC) [17], [29] diffusion techniques [30], [31] and used methods from directional statistics to represent probability density functions (p.d.f.’s) that are transmitted between nodes over the network. In [32], instead, we built Gaussian parametric approximations in tangent spaces to the Stiefel manifold to represent such p.d.f.’s. A different approach to tangent space parametrization for the Stiefel manifold is also found in [4].

In this letter, we consider a state tracking problem in which the hidden state evolves as a random walk [11] on the special orthogonal group, denoted $SO(n)$, i.e., the set of the real orthogonal matrices of dimension $n \times n$ with determinant equal to one. We follow a different approach to represent the p.d.f.’s that are transmitted over the network, using Gaussian approximations computed in the Lie algebra of $SO(n)$. The state estimates at each network node, on the other hand, are determined via the averaging algorithm described in [33] that approximates the Karcher mean of the particle set, which, in turn, generalizes to non-Euclidean spaces the notion of minimum mean-square error estimates.

The signal model in this letter can be employed, for example, to track the attitude of a rigid body [14], [15]. The proposed algorithms can deal with arbitrary nonlinearities in the observation function, which could be used to model limitations or defects of the sensing components, encoders, transmitters or receivers. Experimental results show that, by explicitly taking into account that the hidden state matrices evolve on $SO(n)$, the proposed distributed PFs can achieve significant performance gains compared to previous methods.

The remainder of the letter is organized as follows: in Section II-A we present a brief summary of the properties of $SO(n)$. The signal model is then described in Section II-B. We present the proposed ATC and RndEx algorithms in Sections III and IV, respectively. Results of numerical simulations are reported in Section V. Finally, our conclusions are left to Section VI.

Notation: We use regular weight lowercase or uppercase fonts to denote deterministic scalars (e.g., $k, \lambda, Q$), lowercase boldface italic fonts to denote deterministic vectors ($\mathbf{x}$) and uppercase boldface italic fonts to denote deterministic matrices ($\mathbf{X}$). Random vectors and matrices are denoted by lowercase ($\mathbf{x}$) and uppercase ($\mathbf{X}$) boldface sans-serif fonts, respectively. The sentence “Sample $\mathbf{S}^{(q)} \sim p(S)$" means the same as “Draw a real matrix sample $\mathbf{S}^{(q)}$ of a random matrix $\mathbf{S}$ with p.d.f. $p(S)$”. Similarly “Sample $\mathbf{s}^{(q)} \sim p(s)$" means that a vector sample $\mathbf{s}^{(q)}$ of $\mathbf{s}$ is drawn according to $p(s)$.
II. Signal Model

A. Geometry of \( SO(n) \)

The Special Orthogonal Group is a matrix Lie group \([34]\) closed with respect to matrix multiplications and, accordingly, we will denote it in the remainder of this letter as \( \mathcal{G} \).

The matrix group \( \mathcal{G} \) is also a smooth manifold \([35]\). The tangent space to \( \mathcal{G} \) at a point \( M \in \mathcal{G} \), denoted \( T_M \mathcal{G} \), is the space of \( n \times n \) real matrices \( X \) such that \( M^T X \) is skew-symmetric, where \((\cdot)^T\) denotes the transpose of a matrix. Furthermore, it is possible to map a point \( S \in \mathcal{G} \) into a point \( X \) in the tangent space \( T_M \mathcal{G} \) and vice-versa using respectively the logarithmic and exponential maps, which, for an arbitrary tangency point \( M \in \mathcal{G} \), are given by \([33]\)

\[
\begin{align*}
\text{Log}_M(S) &= M \log(M^T S), \\
\text{Exp}_M(X) &= M \exp(M^T X),
\end{align*}
\]

where \(\log(\cdot)\) and \(\exp(\cdot)\) are the matrix logarithm and matrix exponential functions \([33]\), respectively.

In particular, making \( M = I \), the identity matrix \( I \), the tangent space \( T_I \mathcal{G} \) is by definition the Lie algebra \( \mathfrak{g} \) of the matrix Lie group \( \mathcal{G} \), which, in this case, is the set of all skew-symmetric matrices of dimension \( n \times n \).

If \( X = \text{Log}_M(S) \), then the geodesic connecting \( M \) and \( S \) on \( \mathcal{G} \) is the curve \( \gamma : \mathbb{R} \rightarrow \mathcal{G} \), such that \( \gamma(t) = \text{Exp}_M(Xt) \) \([36]\). Moreover, the geodesic distance \( d_{\mathcal{G}}(M, S) \) between \( M \) and \( S \) can be then shown to be \( \| X \|_F \) \([36]\), where \( \| \cdot \|_F \) is the usual matrix Frobenius norm.

It is also possible to move from a point \( X \in T_M \mathcal{G} \) to another point in \( T_M \mathcal{G} \), with \( M_1, M_2 \in \mathcal{G} \), using the transport operator \([34]\) \( T : T_M \mathcal{G} \rightarrow T_M \mathcal{G} \) such that \( T(X) = M_2 M_1^{-1} X \). In particular, if \( M_2 = I \), the transport of \( X \in T_M \mathcal{G} \) to the Lie algebra \( \mathfrak{g} \) is simply the skew-symmetric matrix \( M_1^T X \).

Finally, since an \( n \times n \) skew-symmetric matrix has only \( d = n(n-1)/2 \) free entries, we can define a bijective mapping \( \Phi(\cdot) \) that associates each skew-symmetric matrix to a vector in \( \mathbb{R}^d \), collecting its free entries in an arbitrarily chosen order. The inverse map is denoted by \( \Phi^{-1}(\cdot) \).

B. Problem Formulation

Let \( \{S_k\} \) be a sequence of \( n \times n \) random matrices that evolve on the Special Orthogonal Group \( \mathcal{G} \) according to the random walk model \([6],[11]\)

\[
S_k = S_{k-1} \exp(\Phi^{-1}(\nu_k)),
\]

where \( k \in \mathbb{N} \) is the time index, \( \{\nu_k\} \) is a sequence of independent, identically distributed Gaussian random vectors taking values in \( \mathbb{R}^d \) with zero mean and covariance matrix \( \lambda^2 I \), and \( \lambda \in \mathbb{R} \) is a hyperparameter.

Multiple nodes on a sensor network record at each time instant \( k \) the observations \( \{Y_{k,r}\} \), taking values in \( \mathbb{R}^{n \times n} \), such that

\[
p(Y_{k,r} ; S_k) = \mathcal{N}_{\mathbb{R}^{n \times n}}(Y_{k,r}, \mathcal{H}_{k,r}(S_k), \Omega_r, \Gamma_r),
\]

where the index \( r \in \{1, \ldots, R\} \) denotes the \( r \)-th node in the network, \( \mathcal{H}_{k,r} : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n} \) is a (possibly nonlinear) function, and \( \mathcal{N}_{\mathbb{R}^{n \times n}} \) is the matrix normal p.d.f. in \( \mathbb{R}^{n \times n} \) \([37]\), whose expression is

\[
\text{etr} \left[ -\frac{1}{2} \Omega_r^{-1}(Y_{k,r} - \mathcal{H}_{k,r}(S_k))^T \Gamma_r^{-1}(Y_{k,r} - \mathcal{H}_{k,r}(S_k)) \right] \times \left[ (2\pi)^{n^2/2} |\Omega_r|^{n/2} |\Gamma_r|^{n/2} \right]^{-1},
\]

where \( \Omega_r \in \mathbb{R}^{n \times n} \) and \( \Gamma_r \in \mathbb{R}^{n \times n} \) are symmetric, positive-definite matrices, and \( \text{etr}(\cdot) \) and \( | \cdot | \) denote the exponential of the trace and the determinant of a matrix, respectively.

Estimation Objectives Given a realization \( \{Y_{j,r}\}, 1 \leq j \leq k, 1 \leq r \leq R \), of the observations \( \{Y_{j,r}\} \), we want to recursively estimate the value assumed by the matrix state \( S_k \) in a distributed, cooperative fashion.

III. ATC Diffusion Particle Filter on the Lie Group (ATC-PF)

Assume that, at instant \( k - 1 \), node \( r \) has a posterior p.d.f. \( p(Y_{k-1,1:r} ; S_{k-1}) \) conditioned on all network measurements that have been assimilated by node \( r \) up to instant \( k - 1 \). The ATC diffusion filtering method \([38]\) is divided into two steps:

**Adapt Step** We update \( p(Y_{k-1,1:r} ; S_{k-1}) \) to \( \tilde{p}_{k|k-1,r}(S_k) \) as the posterior p.d.f. obtained by assimilating the measurements \( \{Y_{k,u}\} \) available at instant \( k \) at all nodes \( u \in \mathcal{N}(r) \), the neighborhood of node \( r \), defined as the set of all nodes connected to \( r \) including itself. Assuming that the observation vectors \( \{Y_{k,u}\} \) are conditionally independent from node to node given the state, we make

\[
\tilde{p}_{k|k-1,r}(S_k) \propto \prod_{u \in \mathcal{N}(r)} p(Y_{k,u} ; S_k) \int_{\mathcal{G}} p(S_k ; S_{k-1}) \left| p_{k|k-1,r}(S_{k-1}) \right| d\mathcal{G}.
\]

If \( p_{k-1|k-1,r}(S_{k-1}) \) is represented by a weighted particle set \( \{w_{k-1|k-1,r}^{(q)}, S_{k-1}^{(q)}\}, q \in \{1, \ldots, Q\} \), we can build a new weighted representation \( \{\bar{w}_{k,r}^{(q)}, \bar{S}_{k,r}^{(q)}\} \) for \( \tilde{p}_{k|k-1,r}(S_k) \) using a marginal particle filter \([39]\) as follows:

1) Sample \([40]\)

\[
\bar{S}_{k,r}^{(q)} \sim \sum_{q'=1}^{Q} w_{k-1|k-1,r}^{(q')} p(S_k | S_{k-1}^{(q')}),
\]

2) Evaluate the weights

\[
\bar{w}_{k,r}^{(q)} \propto \prod_{u \in \mathcal{N}(r)} p(Y_{k,u}, \bar{S}_{k,r}^{(q)}),
\]

with proportionality constant such that \( \sum_{q=1}^{Q} \bar{w}_{k,r}^{(q)} = 1 \).

The optimal local state estimate \( \bar{S}_{k|k,r} \) prior to sensor fusion at node \( r \) is the intrinsic mean \( \bar{S}_{k|k,r} \) in \( \mathcal{G} \) that minimizes \( \mathbb{E} [d_G^2(S_k, \bar{S}_{k|k,r})] \) \([36]\), where the previous expectation is taken with respect to \( \tilde{p}_{k|k-1,r}(S_k) \). The particle filter approximates the intrinsic mean by the Karcher mean \([33],[41]\) of the weighted particle

---

1For simplicity of notation, we omit in this section the conditioning on the measurements in the notation \( p_{k-1|k-1,r}(S_{k-1}) \), assuming it to be implicit.
set \( \{ \bar{w}_{k,r}^q, \bar{S}_{k,r}^q \} \) on the group \( \mathcal{G} \), i.e.,

\[
\bar{S}_{k|k,r} = \arg \min_{\bar{S} \in \mathcal{G}} \sum_{q=1}^Q \bar{w}_{k,r}^q d_G^2(\bar{S}, \bar{S}_{k|k,r}^q),
\]

and transmits it to nodes \( u \in \{ N(r) \setminus \{r\} \} \).

**Combine Step** We fuse the local state estimates \( \bar{S}_{k|k,u} \), \( u \in N(r) \) to obtain a new merged estimate at node \( r \). We propose to do it in a principled way that conforms to the geometry of the matrix group \( \mathcal{G} \).

Upon receiving the local state estimates \( \bar{S}_{k|k,u} \) from each node \( u \in \{ N(r) \setminus \{r\} \} \), node \( r \) first computes the Karcher mean

\[
\bar{S}_{k|k,r} = \arg \min_{\bar{S} \in \mathcal{G}} \sum_{u \in N(r)} a_{r,u} d_G^2(\bar{S}, \bar{S}_{k|k,u}),
\]

where \( \{ a_{r,u} \} \) is a set of positive real weights such that \( \sum_{u} a_{r,u} = 1 \) for all \( r \).

Next, node \( r \) executes the following steps:

1. Compute \( X_{k|k,r} = \Log_{\bar{S}_{k|k,r}}(\bar{S}_{k|k,r}^q) \in T_{\bar{S}_{k|k,r}} \mathcal{G} \), with \( \Log(\cdot) \) defined as in (1), evaluate the skew-symmetric matrix \( Z_{k,r} = \bar{S}_{k|k,r}^T X_{k|k,r} \), and determine \( \bar{z}_{k,r} = \Phi(Z_{k,r}^q) \in \mathbb{R}^d \).
2. Fit a \( d \)-variate Gaussian approximation \( \pi_{k|k,r} \) to the weighted particle set \( \{ \bar{w}_{k,r}^q, \bar{z}_{k,r}^q \} \), \( q \in \{1, \ldots, Q\} \), computing the corresponding sample mean \( m_{k|k,r} \) and sample covariance matrix \( P_{k|k,r} \).
3. Transmit \( m_{k|k,r} \) and \( P_{k|k,r} \) to nodes \( u \in \{ N(r) \setminus \{r\} \} \).
4. Receive \( m_{k|k,u} \) and \( P_{k|k,u} \) from nodes \( u \in \{ N(r) \setminus \{r\} \} \).
5. Combine the local p.d.f.’s \( \pi_{k|k,u} \), \( u \in N(r) \), into a fused p.d.f. \( \bar{\pi}_{k|k,r} \) using the Geometric Average (GA) fusion rule [29], [42]

\[
\bar{\pi}_{k|k,r}(z_k) \propto \prod_{u \in N(r)} \left[ \pi_{k|k,u}(z_k) \right]^{a_{r,u}},
\]

using the same network weights \( a_{r,u} \) as in (10).

6. Resample \( \tilde{X}_{k|k}^q \sim \bar{\pi}_{k|k,r}(z_k) \in \mathbb{R}^d \), evaluate \( \tilde{X}_{k|k}^q = S_{k|k,r} \tilde{Z}_{k,r}^q \) \( \Delta \tilde{S}_{k|k,r}^q \tilde{Z}_{k,r}^q \in T_{\tilde{S}_{k|k,r}} \mathcal{G} \), then recover \( S_{k|k,r}^q = \Exp_{\tilde{S}_{k|k,r}}(\tilde{X}_{k|k}^q) \in \mathcal{G} \), where \( \Exp(\cdot) \) is defined as in (2).

7. Reset the particle weights \( w_{k,r}^q = 1/Q, q \in \{1, \ldots, Q\} \).
8. Compute the fused state estimate \( \bar{S}_{k|k,r} \) at node \( r \) at instant \( k \) as the Karcher mean of the particle set \( \{ w_{k,r}^q, \bar{S}_{k|k,r}^q \} \), which represents the final posterior \( p_k|k|_r \) that is propagated to the next time step.

The fusion rule in (11) finds the p.d.f. \( \bar{\pi} \) that minimizes \( \sum_u a_{r,u} D_{KL}(\bar{\pi}||\pi_{k|k,u}) \), where \( D_{KL} \) is the Kullback-Leibler divergence. An extensive theoretical analysis of that and other alternative p.d.f. fusion rules is found in [43] and [44].

**Implementation Details** Given (1), the points \( Z_{k|k,u}^q \) in the Lie algebra \( \mathcal{L} \) are obtained directly at each node \( u \) as

\[
Z_{k,u}^q = S_{k|k,u} T S_{k|k,u} \logm(S_{k|k,u}, S_{k,u}^q),
\]

Similarly, from (2), the points \( S_{k|k,r}^q \) on the manifold \( \mathcal{G} \) at each node \( r \) are computed as

\[
S_{k|k,r}^q = \Exp_{\bar{S}_{k|k,r}}(S_{k|k,r} T \bar{S}_{k|k,r} \bar{Z}_{k,r}^q) = S_{k|k,r} \Exp_{\bar{S}_{k|k,r}}(\bar{Z}_{k,r}^q).
\]

On the other hand, since all underlying p.d.f.’s \( \pi_{k|k,u} \) defined in \( \mathbb{R}^d \) are Gaussian, then the merged p.d.f. \( \bar{\pi}_{k|k,r} \) on the left-hand side of (11) is also Gaussian with fused mean and covariance matrices given by [17]

\[
P_{k|k,r}^{-1} = \sum_{u \in N(r)} a_{r,u} [P_{k|k,u}]^{-1}
\]

Finally, to obtain the Karcher mean of a weighted particle set \( \{w_{k,r}^q, S_{k|k,r}^q\} \), we used the iterative steepest descent algorithm in [33]. For numerical stability, the \( \expm(\cdot) \) and \( \logm(\cdot) \) functions were implemented using the Rodrigues’ Formula (see [45]).

**IV. RNdEx DIFFUSION PARTICLE FILTER ON THE LIE GROUP (RNdEx-PF)**

Assume that, at instant \( k - 1 \), node \( l \) has a posterior p.d.f. \( p(S_{k-1|l|} Y_{0:k-1,l}) \) conditioned on all measurements \( Y_{0:k-1,l} \) that have been assimilated by node \( l \) up to instant \( k - 1 \). As explained in [28], [38], in the RNdEx diffusion method, node \( l \) first exchanges a representation of its posterior p.d.f. with another random node \( r \) in the network using a pre-specified random exchange protocol. Node \( r \), upon receiving that representation, updates the received posterior p.d.f. assimilating the measurements \( Y_{k,r} \) that are available at the nodes \( u \in N(r) \).

Specifically, let \( \{ w_{k-1,l}^q, S_{k-1|l|}^q \} \) be a weighted particle set that represents \( p(S_{k-1|l|} Y_{0:k-1,l}) \) and let \( S_{k-1|k-1|} \) be its corresponding Karcher mean. The RNdEx-PF executes then the following steps:

1. Compute \( X_{k-1|l}^q = \Phi(S_{k-1|k-1,l} T \logm(S_{k-1|k-1,l}, S_{k-1|l}^q)) \) in \( \mathbb{R}^d \).
2. Adjust a Gaussian p.d.f. \( \bar{\pi}_{k-1|k-1,l} \) to the particle set \( \{ w_{k-1,l}^q, X_{k-1|l}^q \} \) with \( q \in \{1, \ldots, Q\} \) using sample moment matching, and transmit the parameters to a random node \( r \) according to the RNdEx protocol.
3. Upon receiving the parameters, node \( r \), similarly to a density-assisted PF [46], samples \( \tilde{X}_{k-1|l}^q \sim \bar{\pi}_{k-1|k-1,l}(x_{k-1|l}) \), then recovers

\[
S_{k-1|l}^q = \Exp_{\tilde{S}_{k-1|k-1,l}}(\tilde{S}_{k-1|k-1,l} \Phi^{-1}(\tilde{X}_{k-1|l}^q)) \in \mathcal{G},
\]

and resets the particle weights \( \tilde{w}_{k-1,l}^q = 1/Q, \forall q \in \{1, \ldots, Q\} \).

Next, in the data assimilation step, we employ a PF to build an updated weighted particle representation of \( p(S_k|Y_{k,r}, Y_{0:k-1,r}) \triangleq p(S_k|Y_{0:k,r}) \) at node \( r \) at instant \( k \) as follows:

1. Sample \( S_{k|k}^q \sim p(S_k|X_{k|k}^q) \) according to the dynamic model in (3).
2) Compute the importance weights

\[ w_{k,r}^{(q)} = \prod_{u \in N(r)} p(Y_{k,u}|S_{k,r}^{(q)}). \]

The state estimate \( \hat{S}_{k,k,r} \) is then the Karcher mean of the particle set \( \{w_{k,r}^{(q)}, S_{k,r}^{(q)}\} \).

V. SIMULATION RESULTS

To evaluate the performance of the proposed algorithms we ran Monte Carlo simulations consisting of 1,000 independent runs. We used a network with five nodes: nodes 1 to 4 are on the vertices of a square and node 5 is at its center and is connected to all other nodes, as in [26], [27]. The noise covariance matrices were set to \( \Omega_r = I \) and \( \Gamma_r = \beta \cdot 10^{-\alpha_r/10} \), with \( \alpha_r \) equal to 3, 6, 10, 13 and 20 dB for \( r = 1, \ldots, 5 \), respectively. The weights \( \{a_{r,u}\} \) in (10) were obtained using the Metropolis rule (see [17]). We assumed that \( n = 3, Q = 200 \), and \( k = 0.15 \).

The observation function \( h_{k,r}(S_k) \) in (4) was defined as \( h_{k,r}(S_k)_{i,j} = h((S_k)_{i,j}) \), where \( (.)_{i,j} \) denotes the matrix element with the given indexes and \( h(\cdot) : \mathbb{R} \to \mathbb{R} \) a scalar function. We employed in the simulations two formulations for \( h(\cdot) \), namely, A) \( h(x) = x^3 - 1/2 \), and B) \( h(x) = \text{sat}(x; 1/2) \), where

\[ \text{sat}(x; \beta) = \begin{cases} x, & |x| < \beta, \\ \beta \cdot x/|x|, & |x| \geq \beta. \end{cases} \]

For comparison, we ran in the same setup the following competing methods:

i) A centralized\(^2\) EKF (Euclid.-EKF), based on a vectorized version of the model in (3) and (4); the state evolves according to an adjusted Gaussian random walk in the Euclidean space \( \mathbb{R}^{n^2 \times 1} \), and its estimates were projected onto \( \mathcal{G} \) using their polar factors;

ii) A centralized manifold-constrained EKF (MC-EKF), based on [12] and [13];

iii) Particle filters (Isol-PF) that operate in isolation at each node;

iv) A centralized particle filter (Joint-PF);

v) A modified likelihood consensus distributed PF (LC-PF), which follows the algorithm in [47], but is adapted to the geometry of the Special Orthogonal Group for a fairer comparison.

Methods iii)-v) sample particles from the state transition p.d.f. in \( \mathcal{G} \) defined by (3), and compute estimates as the Karcher means of their local particle sets as in (9).

Fig. 1 displays the simulations results. As shown in the Figure, the ATC-PF has steady-state performance similar to that of the centralized particle filter for both formulations of \( h(x) \). On the other hand, the RndEx-PF performs worse than the ATC-PF for the polynomial nonlinearity, but better than the LC-PF. In both cases, the proposed methods outperform the particle filters operating in isolation and the EKFs by a large margin.

For the saturation-type nonlinearity (Fig. 1(b)), all distributed PFs perform similarly to the centralized PF. The proposed methods have, however, smaller internode communication costs than the LC-PF, as it can be seen by replacing the simulation parameters \( n = d = 3, k_c = 8 \) [47], and \( k_c \leq (|N(r)| - 1) \) in the expressions on the right column of Table I.

\(^2\)We use the term \textit{centralized} to characterize an algorithm that runs in a single node and has access to all observations in the network.

VI. CONCLUSION

In this letter, we introduced new distributed diffusion PF algorithms to track a sequence of hidden random matrices that evolve on the Special Orthogonal Group. The algorithms employ a Gaussian parametric approximation of the weighted particle set defined in an isomorphism of the Lie algebra of the matrix group. Simulation results show that, in setups with different nonlinear observation functions, the performance of the proposed algorithms surpasses or equals that of an alternative distributed PF at a lower communication cost, and can outperform a manifold-constrained centralized EKF by a large margin.
