NONLINEAR STATE ESTIMATION USING PARTICLE FILTERS ON THE STIEFEL MANIFOLD

Claudio J. Bordin Jr.

Universidade Federal do ABC
Santo André, SP - 09210-580 - Brazil
claudio.bordin@ufabc.edu.br

Marcelo G. S. Bruno

Instituto Tecnológico de Aeronáutica
São José dos Campos, SP - 12228-900 - Brazil
bruno@ita.br

ABSTRACT

Many problems in statistical signal processing involve tracking the state of a dynamic system that evolves on a Stiefel manifold. To this aim, we introduce in this paper a novel particle filter algorithm that approximates the optimal importance function on the Stiefel manifold and is capable of handling nonlinear observation functions. To sample from the required importance function, we develop adaptations of previous MCMC algorithms. We verify via numerical simulations that, in a scenario with a strongly nonlinear observation model, the new proposed method outperforms existing algorithms that use the prior importance function at the cost, however, of increased computational complexity.

Index Terms— Stiefel manifold, Particle filters, MCMC.

1. INTRODUCTION

Most state estimation algorithms found in the literature assume that the unknown state vector lies on a linear Euclidean space. However, many engineering applications, such as attitude estimation in navigation systems [1], image processing [2], robotics [3] and digital communications [4], [5], lead to models with dynamic systems whose states are constrained to the Stiefel manifold $\mathcal{V}_{k,l}$, i.e. the space of the real $k \times l$ orthonormal matrices, of which the unit (hyper)sphere [5] and the special orthogonal group $\mathrm{SO}(n)$ [6] are special cases, see also [7] for an extended list of references to practical signal processing problems where the Stiefel manifold constraint applies.

A particle filter [8] algorithm for state estimation on the Stiefel manifold was introduced in [9] using a data-independent (prior) importance function and assuming a linear observation model. In this paper, we extend the work in [9] in two ways. First, we consider a scenario in which the observations are nonlinear functions of the state contaminated by additive Gaussian noise. Second, rather than using the prior importance function, we propose a novel matrix Fisher-Bingham [10] parametric approximation to the optimal (data-driven) importance function, which is the equivalent, on $\mathcal{V}_{k,l}$, to the Gaussian approximation to the optimal importance function employed in [11–15] for particle filters specified on $\mathbb{R}^L$ with arbitrary nonlinear observation models. This paper also generalizes to an arbitrary Stiefel manifold the results in [5], which considered only the special case in which states are constrained to the unit hypersphere.

The paper is divided into 5 sections. Sec. 1 is this introduction. In Sec. 2 we describe the problem setup and briefly review how it has been tackled by [9]. In Sec. 3, we detail the new proposed algorithm: first, we derive the closed-form expression of the parameters of the matrix Fisher-Bingham approximation to the optimal importance function; next, we propose a Gibbs sampler [16] to draw from the proposed approximation and discuss a numerical method to compute its normalization constant. In Sec. 4, we present computer experiments with simulated data to assess the performance of the proposed method and compare it to the performance of the algorithm in [9]. Finally, we present our conclusions and discuss possible future work in Sec. 5.

This work was supported by Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), process 2018/26191-0.

2. PROBLEM SETUP

Let $\mathbf{S}_n$ denote the state of a system at the time instant n, supposed to evolve on the Stiefel manifold $\mathcal{V}_{k,l}$, defined as the set $\{\mathbf{V} \in \mathbb{R}^{k\times l} : \mathbf{V}^T\mathbf{V} = \mathbf{I}_l\}, k > l$, where $\mathbf{I}_l$ is the $l \times l$ identity matrix. We assume that

$$\mathbf{S}_n|\mathbf{S}_{n-1} \sim \mathrm{vMF}(\mathbf{S}_n|\kappa\mathbf{S}_{n-1}), \tag{1}$$

i.e., conditioned on $\mathbf{S}_{n-1}$, $\mathbf{S}_n$ has a matrix Von Mises-Fisher distribution (vMF) [17, p. 31], whose probability density function (p.d.f.) is defined as

$$\mathrm{vMF}(\mathbf{S}_n|\kappa\mathbf{S}_{n-1}) = \frac{\mathrm{etr}\left(\kappa\mathbf{S}_{n-1}^T\mathbf{S}_n\right)}{{}_0F_1\left(\frac{k}{2}, \frac{\kappa^2}{4}\mathbf{S}_{n-1}^T\mathbf{S}_{n-1}\right)}, \tag{2}$$

where etr stands for the exponential of the trace of a square matrix, $\kappa \in \mathbb{R}^+$ is a fixed hyperparameter, and ${}_0F_1$ is the hypergeometric function with matrix argument [17].

We assume that $\{\mathbf{S}_n\}$ gives rise to the observation sequence $\{\mathbf{Y}_n\}$, $\mathbf{Y}_n \in \mathbb{R}^{k\times l}$, such that

$$\mathbf{Y}_n|\mathbf{S}_n \sim \mathbf{N}_{k,l}(\mathbf{Y}_n|\mathbf{G}(\mathbf{S}_n), \mathbf{\Omega}, \mathbf{\Gamma}), \tag{3}$$

where $\mathbf{G} : \mathbb{R}^{k\times l} \to \mathbb{R}^{k\times l}$ is a possibly nonlinear function, and $\mathbf{N}_{k,l}$ is a matrix normal distribution on $\mathbb{R}^{k\times l}$, defined as [9] [18]

$$\mathbf{N}_{k,l}(\mathbf{Y}_n|\mathbf{G}(\mathbf{S}_n), \mathbf{\Omega}, \mathbf{\Gamma}) = \frac{\mathrm{etr}\left[-\frac{1}{2}\mathbf{\Omega}^{-1}(\mathbf{Y}_n - \mathbf{G}(\mathbf{S}_n))^T\mathbf{\Gamma}^{-1}(\mathbf{Y}_n - \mathbf{G}(\mathbf{S}_n))\right]}{(2\pi)^{kl/2}|\mathbf{\Omega}|^{k/2}|\mathbf{\Gamma}|^{l/2}}, \tag{4}$$

where $\mathbf{\Omega} \in \mathbb{R}^{l\times l}$ and $\mathbf{\Gamma} \in \mathbb{R}^{k\times k}$ are symmetric positive-definite matrices.

We intend to develop a particle filter [8] to approximate the states' posterior probabilities as

$$\mathbf{Pr}(\{\mathbf{S}_0, \cdots, \mathbf{S}_n\} \in \Delta|\mathbf{Y}_0, \cdots, \mathbf{Y}_n) \approx \sum_{q=1}^{Q} w_n^{(q)} \delta_{\mathbf{S}_0^{(q)}, \cdots, \mathbf{S}_n^{(q)}}(\Delta), \tag{5}$$

where Δ is a subset of $\mathcal{V}_{k,l}^{n+1}$, the Cartesian product on $n+1$ copies of $\mathcal{V}_{k,l}$, $\delta_x(X)$ is a Dirac measure, defined as 1 if $x \in X$ and 0 otherwise, $\mathbf{S}_j^{(q)}$, $0 \leq j \leq n$ are the so-called *particles*, sequentially sampled from an arbitrary p.d.f. called *importance function* as $\mathbf{S}_n^{(q)} \sim \pi\left(\mathbf{S}_n|\mathbf{S}_{n-1}^{(q)}, \mathbf{Y}_n\right)$, $Q \gg 1$ is the number of particles, and $w_n^{(q)}$ are the particle weights, which are recursively evaluated as [8]

$$w_n^{(q)} \propto w_{n-1}^{(q)} \frac{p(\mathbf{Y}_n|\mathbf{S}_n^{(q)})p(\mathbf{S}_n^{(q)}|\mathbf{S}_{n-1}^{(q)})}{\pi\left(\mathbf{S}_n^{(q)}|\mathbf{S}_{n-1}^{(q)}, \mathbf{Y}_n\right)}, \tag{6}$$

where $p(\cdot)$ denotes the p.d.f. of the indicated variates and the symbol $\propto$ denotes "proportional to" with the implied proportionality constant such that $\sum_{q=1}^{Q} w_n^{(q)} = 1$.

Reference [9] introduced a particle filtering algorithm for the signal model given by (1)-(3) restricted to $\mathbf{G}(\mathbf{S}_n) = \mathbf{S}_n$ and using the prior importance function, i.e., sampling the particles as

$$\mathbf{S}_n^{(q)} \sim p\left(\mathbf{S}_n|\mathbf{S}_{n-1}^{(q)}, \mathbf{Y}_n\right) = \text{vMF}(\mathbf{S}_n|\kappa\mathbf{S}_{n-1}^{(q)}). \tag{7}$$

The restriction on $\mathbf{G}$ can be trivially lifted, leading to the weight update equation

$$w_n^{(q)} \propto w_{n-1}^{(q)} \mathbf{N}_{k,l}(\mathbf{Y}_n|\mathbf{G}(\mathbf{S}_n^{(q)}), \mathbf{\Omega}, \mathbf{\Gamma}). \tag{8}$$

To generate samples from the vMF p.d.f., Ref. [9] employed the rejection method described in [17].

3. PROPOSED METHOD

Instead of sampling particles from the data-blind prior importance function, we propose to use the so-called *optimal* importance function [8], i.e., sample $\mathbf{S}_n^{(q)}$ according to

$$\begin{aligned}\mathbf{S}_n^{(q)} &\sim p(\mathbf{S}_n|\mathbf{Y}_n, \mathbf{S}_{n-1}^{(q)}) \\ &= \frac{p(\mathbf{Y}_n|\mathbf{S}_n)p(\mathbf{S}_n|\mathbf{S}_{n-1}^{(q)})}{\int_{\mathcal{V}_{k,l}} p(\mathbf{Y}_n|\mathbf{S}_n)p(\mathbf{S}_n|\mathbf{S}_{n-1}^{(q)})d\mathcal{V}_{k,l}(\mathbf{S}_n)},\end{aligned} \tag{9}$$

where $d\mathcal{V}_{k,l}(\mathbf{S}_n)$ represents the volume element on $\mathcal{V}_{k,l}$. The weights are then updated as

$$w_n^{(q)} \propto w_{n-1}^{(q)} \frac{p(\mathbf{Y}_n|\mathbf{S}_n^{(q)})p(\mathbf{S}_n^{(q)}|\mathbf{S}_{n-1}^{(q)})}{p(\mathbf{S}_n^{(q)}|\mathbf{Y}_n, \mathbf{S}_{n-1}^{(q)})}. \tag{10}$$

Unfortunately, the integral in the denominator of (9) can only be analytically evaluated if $\mathbf{G}(\mathbf{S}_n)$[1] is a linear function. Thus, we propose to approximate (9) by linearizing $\mathbf{G}(\mathbf{S}_n)$ around $\mathbf{S}_{n-1}$, i.e.,

$$\mathbf{g}(\mathbf{s}_n) \approx \mathbf{g}(\mathbf{s}_{n-1}) + \mathbf{J}(\mathbf{s}_{n-1})\left[\mathbf{s}_n - \mathbf{s}_{n-1}\right], \tag{11}$$

where $\mathbf{s}_n \triangleq \text{vec}(\mathbf{S}_n)$, $\mathbf{g}(\mathbf{s}_n) \triangleq \text{vec}(\mathbf{G}(\mathbf{S}_n))$, the Jacobian matrix $[\mathbf{J}(\mathbf{s}_{n-1})]_{kl} \triangleq \left.\frac{\partial[\mathbf{g}(\mathbf{s})]_k}{\partial[\mathbf{s}]_l}\right|_{\mathbf{s}=\mathbf{s}_{n-1}}$ and $[\cdot]_{a(b)}$ denotes the elements of a vector (matrix).

It can be verified [18] that (3) implies that

$$P(\mathbf{Y}_n|\mathbf{S}_n) = p(\mathbf{y}_n|\mathbf{s}_n) = \mathcal{N}_{kl}(\mathbf{y}_n|\mathbf{g}(\mathbf{s}_n), \mathbf{\Sigma}), \tag{12}$$

[1]For readability, we do not explicitly indicate in the sequel the dependence of the quantities with respect to the particle index (q) unless strictly necessary.

where $\mathcal{N}_{kl}$ stands for a $kl-$variate vector Gaussian p.d.f., $\mathbf{\Sigma} = \mathbf{\Gamma} \otimes \mathbf{\Omega}$, and $\mathbf{y}_n \triangleq \text{vec}(\mathbf{Y}_n)$. Substituting (11) and (12) into (9) and exploring the fact that $\text{tr}(U^T V) = \text{vec}(U)^T\text{vec}(V)$, where tr denotes the trace operator, it follows that

$$\begin{aligned}p(\mathbf{Y}_n|\mathbf{S}_n)p(\mathbf{S}_n|\mathbf{S}_{n-1}) \approx c_{MF}^{-1}(\kappa\mathbf{S}_{n-1})c_N^{-1}(\mathbf{\Sigma})\ \exp\Big\{-\frac{1}{2} \\ \left[\tilde{\mathbf{y}}_n - \mathbf{J}(\mathbf{s}_{n-1})\mathbf{s}_n\right]^T \mathbf{\Sigma}^{-1}\left[\tilde{\mathbf{y}}_n - \mathbf{J}(\mathbf{s}_{n-1})\mathbf{s}_n\right] + \kappa\mathbf{s}_{n-1}^T\mathbf{s}_n\Big\},\end{aligned} \tag{13}$$

where $\tilde{\mathbf{y}}_n \triangleq \mathbf{y}_n - \mathbf{g}(\mathbf{s}_{n-1}) + \mathbf{J}(\mathbf{s}_{n-1})\mathbf{s}_{n-1}$, $c_N(\mathbf{\Sigma}) = (2\pi)^{\frac{kl}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}$ $= (2\pi)^{\frac{kl}{2}}|\mathbf{\Omega}|^{\frac{k}{2}}|\mathbf{\Gamma}|^{\frac{l}{2}}$, and $c_{MF}(\kappa\mathbf{S}_{n-1})$ is the normalization constant for the Von Mises-Fisher p.d.f, defined in the denominator of the right-hand side (r.h.s.) of (2). Rearranging terms, the r.h.s. of (13) can be rewritten as

$$\begin{aligned}&c_{MF}^{-1}(\kappa\mathbf{S}_{n-1})c_N^{-1}(\mathbf{\Sigma})\exp\Big\{-\frac{1}{2}\tilde{\mathbf{y}}_n^T\mathbf{\Sigma}^{-1}\tilde{\mathbf{y}}_n + \Big[\tilde{\mathbf{y}}_n^T\mathbf{\Sigma}^{-1}\mathbf{J}(\mathbf{s}_{n-1}) + \\ &\qquad \kappa\mathbf{s}_{n-1}^T\Big]\mathbf{s}_n + \mathbf{s}_n^T\left[-\frac{1}{2}\mathbf{J}(\mathbf{s}_{n-1})^T\mathbf{\Sigma}^{-1}\mathbf{J}(\mathbf{s}_{n-1})\right]\mathbf{s}_n\Big\} \qquad (14) \\ &= c_{MF}^{-1}(\kappa\mathbf{S}_{n-1})c_N^{-1}(\mathbf{\Sigma})\exp\left\{-\frac{1}{2}\tilde{\mathbf{y}}_n^T\mathbf{\Sigma}^{-1}\tilde{\mathbf{y}}_n\right\}\cdot \\ &\quad \exp\Big\{\text{tr}\left(\left[\text{vec}^{-1}\Big(\mathbf{J}(\mathbf{s}_{n-1})^T\mathbf{\Sigma}^{-1}\tilde{\mathbf{y}}_n\Big) + \kappa\mathbf{S}_{n-1}\right]^T\mathbf{S}_n\right) + \\ &\quad \mathbf{s}_n^T\left[-\frac{1}{2}\mathbf{J}(\mathbf{s}_{n-1})^T\mathbf{\Sigma}^{-1}\mathbf{J}(\mathbf{s}_{n-1})\right]\mathbf{s}_n\Big\}, \qquad (15)\end{aligned}$$

where vec^{-1} denotes the inverse of the vectorization operator.

The expression (15) can be recast as

$$c_{MF}^{-1}(\kappa\mathbf{S}_{n-1})c_N^{-1}(\mathbf{\Sigma})\exp\left\{-\frac{1}{2}\tilde{\mathbf{y}}_n^T\mathbf{\Sigma}^{-1}\tilde{\mathbf{y}}_n\right\}\text{FB}(\mathbf{S}_n|\mathbf{A}_n, \mathbf{B}_n), \tag{16}$$

where FB stands for the matrix Fisher-Bingham p.d.f. on $\mathcal{V}_{n,k}$, defined as [10]

$$\text{FB}(\mathbf{S}_n|\mathbf{A}_n, \mathbf{B}_n) = \frac{\exp\left\{\text{tr}(\mathbf{A}_n^T\mathbf{S}_n) + \text{vec}(\mathbf{S}_n)^T\mathbf{B}_n\text{vec}(\mathbf{S}_n)\right\}}{c_{FB}(\mathbf{A}_n, \mathbf{B}_n)}, \tag{17}$$

where $c_{FB}(\mathbf{A}_n, \mathbf{B}_n)$ is the matrix Fisher-Bingham p.d.f. normalization constant and

$$\mathbf{A}_n \triangleq \text{vec}^{-1}(\tilde{\mathbf{y}}_n^T\mathbf{\Sigma}^{-1}\mathbf{J}(\mathbf{s}_{n-1})) + \kappa\mathbf{S}_{n-1}, \tag{18}$$

$$\mathbf{B}_n \triangleq -\frac{1}{2}\mathbf{J}(\mathbf{s}_{n-1})^T\mathbf{\Sigma}^{-1}\mathbf{J}(\mathbf{s}_{n-1}). \tag{19}$$

Plugging (16) into (9) and noting that $\tilde{\mathbf{y}}_n$, $\mathbf{A}_n$, and $\mathbf{B}_n$ do not depend on $\mathbf{S}_n$, the integral in the denominator of (9) can be evaluated by simply dropping the Fisher-Bingham density in (16). Therefore, as a result of the approximation (11), we can write

$$p(\mathbf{S}_n|\mathbf{Y}_n, \mathbf{S}_{n-1}^{(q)}) \approx \text{FB}(\mathbf{S}_n|\mathbf{A}_n^{(q)}, \mathbf{B}_n^{(q)}), \tag{20}$$

and propagate the weights, in turn, as

$$w_n^{(q)} \propto w_{n-1}^{(q)} \frac{\text{vMF}(\mathbf{S}_n^{(q)}|\kappa\mathbf{S}_{n-1}^{(q)})\ \mathbf{N}_{k,l}(\mathbf{Y}_n|\mathbf{G}(\mathbf{S}_n^{(q)}), \mathbf{\Omega}, \mathbf{\Gamma})}{\text{FB}(\mathbf{S}_n^{(q)}|\mathbf{A}_n^{(q)}, \mathbf{B}_n^{(q)})}. \tag{21}$$

To evaluate (20) and (21), we developed methods to generate samples from the matrix Fisher-Bingham p.d.f. and compute its normalization constant, which are discussed respectively in Secs. 3.1 and 3.2.

Remark It is worth noting that when $l = 1$, $\mathcal{V}_{k,l}$ reduces to the unit $k-1$ sphere $\mathcal{S}^{k-1}$. In that case, if the observation $\mathbf{y}_n$ is defined as a scalar function of $\mathbf{s}_n$ and $\mathbf{\Sigma} = \sigma^2$, $\mathbf{A}_n$ and $\mathbf{B}_n$ (Eqs. 18 and 19) equal the parameters $\mathbf{a}_n$ and $\mathbf{B}_n$ for a vector Fisher-Bingham p.d.f. derived in [5].

3.1. Sampling from a matrix Fisher-Bingham p.d.f.

As pointed out in [19], the most convenient methods to generate samples from a matrix Fisher-Bingham p.d.f. are based on MCMC [16]. Although [20] describes a general method to generate samples for p.d.f's on the Stiefel manifold, it cannot be easily extended to draw samples from (17). Therefore, we adapted the algorithm introduced in [21, Sec. 3.3], originally developed for the matrix Bingham-Von Mises-Fisher distribution[2], as follows: under the restriction that $\mathbf{B}_n$ is a block-diagonal matrix (which, as shown in Sec. 4, can be satisfied depending on $\mathbf{G}$ and $\mathbf{\Sigma}$), we can write

$$\mathrm{FB}(\mathbf{S}_n|\mathbf{A}_n,\mathbf{B}_n) \propto \prod_{m=1}^{l} \exp\Big(\mathbf{A}_n[,m]^T\mathbf{S}_n[,m] + \mathbf{S}_n[,m]^T\mathbf{B}_n(m)\mathbf{S}_n[,m]\Big), \quad (22)$$

where $[,m]$ stands for the $m-$th column of a matrix and $\mathbf{B}(m) \in \mathbb{R}^{k\times k}$ denotes the $m-$th block of the diagonal of $\mathbf{B}_n$.

As the columns of $\mathbf{S}_n$ are orthogonal with probability 1, we can write $\mathbf{S}_n = [\mathbf{N}\mathbf{z}\ \ \mathbf{S}_n[,-1]]$, where $\mathbf{S}_n[,-1]$ is the matrix formed by removing the first column of $\mathbf{S}_n$, $\mathbf{N} \in \mathbb{R}^{k\times(k-l+1)}$ is an orthonormal basis for the null space of $\mathbf{S}_n[,-1]$, and $\mathbf{z}$ is a $(k-l+1)$ unit-norm column vector. The conditional p.d.f. of $\mathbf{z}$ is then given by [21]

$$p(\mathbf{z}|\mathbf{S}_n[,-1]) \propto \exp\Big(\mathbf{A}_n[,1]^T\mathbf{N}\mathbf{z} + \mathbf{z}^T\mathbf{N}^T\mathbf{B}_n(1)\mathbf{N}\mathbf{z}\Big) \triangleq \exp(\tilde{\mathbf{a}}\mathbf{z} + \mathbf{z}^T\tilde{\mathbf{B}}\mathbf{z}), \quad (23)$$

which is a Fisher-Bingham density on the unit sphere [22].

As shown in [21], a Markov chain in $\mathbf{S}_n$ with stationary p.d.f. $\mathrm{FB}(\mathbf{S}_n|\mathbf{A}_n,\mathbf{B}_n)$ can be obtained via the Gibbs sampler [16] defined as follows:

Given $\mathbf{S}_n^{<j>} = \mathbf{S}$, the $j-$th element of the chain, compute steps 1 to 4 for each $m \in \{1,\ldots,l\}$ in a random order:

1. compute $\mathbf{N}$, an orthonormal basis for the null space of $\mathbf{S}[,-m]$;

2. compute $\tilde{\mathbf{a}} = \mathbf{A}_n[,m]^T\mathbf{N}$ and $\tilde{\mathbf{B}} = \mathbf{N}^T\mathbf{B}_n(m)\mathbf{N}$;

3. sample $\mathbf{z}$ from a Fisher-Bingham density on the unit sphere with parameters $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{B}}$ via the algorithm introduced in [5, Sec. 4.1].

4. set $\mathbf{S}[,m] = \mathbf{N}\mathbf{z}$.

Set $\mathbf{S}_n^{<j+1>} = \mathbf{S}$.

[2]The algorithm of [21, Sec. 3.3] cannot be directly applied to draw samples from (17) since, for the problem at hand, it is not possible to represent $\mathbf{B}_n$ as a Kronecker product of two matrices.

3.2. Computation of the matrix Fisher-Bingham p.d.f. normalization constant

To update the weights (Equation 21), it is necessary to compute the normalization constants[3]

$$c_{FB}(\mathbf{A}_n,\mathbf{B}_n) \triangleq \int_{\mathcal{V}_{k,l}} \exp\Big\{\mathrm{tr}(\mathbf{A}_n^T\mathbf{S}) + \mathrm{vec}(\mathbf{S})^T\mathbf{B}_n\mathrm{vec}(\mathbf{S})\Big\}\, d\mathcal{V}_{k,l}(\mathbf{S}), \quad (24)$$

$$c_{MF}(k\mathbf{S}_{n-1}) \triangleq \int_{\mathcal{V}_{k,l}} \exp\Big\{\mathrm{tr}(\kappa\mathbf{S}_{n-1}^T\mathbf{S})\Big\}\, d\mathcal{V}_{k,l}(\mathbf{S}), = c_{FB}(k\mathbf{S}_{n-1},\mathbf{0}). \quad (25)$$

Although the saddlepoint approximation method introduced in [10] could be used to evaluate (24), we verified experimentally that the first-order approximation was not accurate enough, while the second- and third-order approximations required excessive computer runtime. Therefore, following the reasoning of the Monte Carlo algorithm implemented in [10][4], we developed a quasi-Monte Carlo method [23] to approximate (24). This method generates low-discrepancy uncorrelated Gaussian random vectors which, when mapped to $\mathcal{V}_{k,l}$, result in uniformly distributed samples that are used to approximate the required integral via (26). The algorithm runs as follows:

A. Generate samples uniformly distributed on $\mathcal{V}_{k,l}$:

1. Generate a sample α_i of a $k-$dimensional low-discrepancy [23] Sobol sequence, uniformly distributed on $[0\ \ 1]^k$;

2. Compute $[\beta_i]_r = \Phi^{-1}([\alpha_i]_r)$, $1 \le r \le k$, where Φ^{-1} denotes the (normalized) Gaussian inverse cumulative distribution function;

3. Evaluate $\mathbf{x}_i = \beta_i/||\beta_i||$;

4. Take $\mathbf{x}_1$ as the first column of the matrix $\mathbf{X}$;

5. To compute the next $l-1$ columns of $\mathbf{X}$, repeat steps 1-3, apply the Gram-Schmidt method to $\mathbf{x}_i$ to obtain the component orthogonal to the previous columns and, if its norm is above an arbitrary threshold (set to 10^{-6}), normalize it (as in step 3). Otherwise, discard the result and repeat steps 1-3;

6. Return the matrix $\mathbf{X}$.

B. Compute the quasi-Monte Carlo Estimate:

Approximate (24) as

$$c_{FB}(\mathbf{A}_n,\mathbf{B}_n) \approx \mathrm{Vol}(\mathcal{V}_{k,l})\cdot \frac{1}{N_S}\sum_{i=1}^{N_S}\Big\{\mathrm{tr}(\mathbf{A}_n^T\mathbf{X}^{<i>}) + \mathrm{vec}(\mathbf{X}^{<i>})^T\mathbf{B}_n\mathrm{vec}(\mathbf{X}^{<i>})\Big\}, \quad (26)$$

where N_S is the number of samples, $\mathbf{X}^{<i>}$ is the $i-$th sample generated by the procedure A, and

$$\mathrm{Vol}(\mathcal{V}_{k,l}) = \frac{2^l\pi^{kl/2}}{\Gamma_l(k/2)} \quad (27)$$

[3]Note that (25) can also be evaluated as a matrix hypergeometric function (Equation 2), which also demands intensive numerical computations.

[4]In the reference's supplementary material, file `stiefel/logNormConstSP.m`.

is the volume of $\mathcal{V}_{k,l}$ [24], where $\Gamma_m(a) \triangleq \pi^{m(m-1)/4} \prod_{i=1}^{m} \Gamma(a-(i-1)/2)$ and Γ stands for the gamma function.

3.3. Computation of the weighted averages on the Stiefel manifold

To compute an estimate $\hat{\mathbf{S}}_n$ given the particle approximation to the posterior probabilities of the states $\left\{w_n^{(q)}, \mathbf{S}_n^{(q)}\right\}_{q=1}^{Q}$, one could use the heuristic procedure described in [9], which consisted of computing the average $\bar{\mathbf{S}}_n \triangleq \sum_{q=1}^{Q} w_n^{(q)} \mathbf{S}_n^{(q)}$ and projecting $\bar{\mathbf{S}}_n$ onto $\mathcal{V}_{k,l}$ using the polar decomposition [17]. Ideally, however, one would estimate the state as a *Karcher mean* [25], defined as the value of $\hat{\mathbf{S}}_n$ that minimizes the weighted mean square *geodesic distance* to the particle set (see, e.g., [5, Sec. 4.2]). This computation requires the calculation of Stiefel manifold's exponential and logarithmic maps. Although these computations can be performed via the algorithms recently introduced in [26], they revealed to be computationally too complex for our aim. As a compromise, we evaluated the weighted averages over the Stiefel manifold adapting the method described in [27] as

$$\mathbf{S}_n^{<i+1>} = \mathcal{M}_{\mathbf{S}_n^{<i>}} \left(\sum_{q=1}^{Q} w_n^{(q)} \, \mathcal{M}^{-1}_{\mathbf{S}_n^{<i>}} \left(\mathbf{S}_n^{(q)} \right) \right), \; i \geq 0 \quad (28)$$

where $\mathbf{S}_n^{<i>}$ denotes the $i-$th estimate of the weighted average, with $\mathbf{S}_n^{<0>}$ chosen as a random element of the particle set, and $\mathcal{M}$ and $\mathcal{M}^{-1}$ are the orthographic retraction and lifting maps [27], respectively, whose computation is far less expensive than that of the exponential and logarithmic maps. The algorithm of (28) was run until $\|\mathbf{S}_n^{<i+1>} - \mathbf{S}_n^{<i>}\|_F < 10^{-6}$, where $\|\cdot\|_F$ denotes the Frobenius norm. When the algorithm stops, the estimate $\hat{\mathbf{S}}_n^{<i+1>}$ is taken as $\mathbf{S}_n^{<i+1>}$.

4. NUMERICAL EXPERIMENT

To evaluate the performance of the algorithm proposed in Sec. 3, we performed a numerical simulation consisting of 150 independent trials. In each trial, we processed 100 consecutive synthetic data samples generated from the model (1)-(3). For comparison, we ran in the same setup the algorithm described in [9]. The particle filters used $Q = 300$ particles and performed systematic resampling [8] at each time step; the particles were initialized with samples from a uniform p.d.f. on the Stiefel manifold, and the initial weights calculated accordingly.

We assumed that the function $\mathbf{G}$ acts elementwise on its argument and has the same expression for all entries, i.e., $[\mathbf{g}(\mathbf{s}_{n-1})]_i = [\mathbf{g}([\mathbf{s}_{n-1}]_i)]_i \triangleq g([\mathbf{s}_{n-1}]_i)$. As a consequence, the Jacobian $\mathbf{J}(\mathbf{s}_{n-1})$ is a diagonal matrix, whose entries are given by $[\mathbf{J}(\mathbf{s}_{n-1})]_{ii} = \left.\frac{\partial g([\mathbf{s}]_i)}{\partial [\mathbf{s}]_i}\right|_{\mathbf{s}=\mathbf{s}_{n-1}}$. We also assumed that the observation noise entries are decorrelated, such that $\mathbf{\Omega} \triangleq \mathbf{I}_l$, $\mathbf{\Gamma} \triangleq \mathbf{I}_k \sigma^2$ and, consequently, $\mathbf{\Sigma} = \mathbf{I}_{kl}\sigma^2$. As a result, $\mathbf{B}_n$ is diagonal. The parameters were set to $\kappa = 150$, $\sigma^2 = 0.05$, $k = 3$ and $l = 2$.

To sample from the matrix Von Mises-Fisher p.d.f., we used the algorithm described in [21, Sec. 2.2]. To draw samples from the matrix Fisher-Bingham p.d.f., in turn, the algorithm of Sec. 3.1 ran 50 Gibbs iterations. To evaluate the matrix p.d.f's normalizations constants (Sec. 3.2), we used $N_S = 10^5$ pseudo-random samples, which were computed at initialization and reused in all function calls. The algorithm's performance was evaluated in terms of the mean geodesic distance from the true state $\mathbf{S}_n$ to the estimated state $\hat{\mathbf{S}}_n$ (Sec. 3.3), defined as [28]

$$d(\mathbf{S}_n, \hat{\mathbf{S}}_n) = \|\mathrm{Exp}_{\mathbf{S}_n}^{-1}(\hat{\mathbf{S}}_n)\|_F, \quad (29)$$

where $\mathrm{Exp}_{\mathbf{S}_n}^{-1}(\cdot)$ is the logarithmic map [26] of the Stiefel manifold evaluated at $\mathbf{S}_n$.

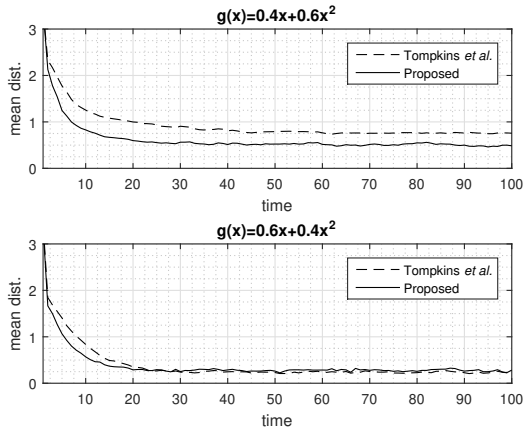

Fig. 1. Mean geodesic distance for the proposed algorithm (Sec. 3) and that of Tompkins *et al.* [9] as a function of time, for distinct nonlinear observation functions $g(x)$.

Figure 1 displays the results obtained for $g(x) = 0.4x + 0.6x^2$ (top) and $g(x) = 0.6x + 0.4x^2$ (bottom). Note that, for the former expression, the observations will often be positive, while for the latter, they will more frequently follow the state's true signal. As one may observe, for stronger nonlinearity (top), the proposed method exhibited an asymptotic error about 30% smaller than the method of [9], which uses the prior importance function, at the cost of increased computational complexity: in a similarly optimized execution setup, each realization of the proposed algorithm took 451s, while the algorithm of [9] required 5.4s. On the other hand, for the *softer* nonlinearity (bottom), the proposed method exhibits no statistically significant steady-state performance advantage.

5. CONCLUSIONS

We introduced in this paper a new particle filtering algorithm to estimate the states of a dynamic system that evolves on the Stiefel manifold. Although the optimal importance function is intractable when the observation function $\mathbf{G}$ is nonlinear, by linearizing this function around the previous state estimate, the optimal importance function could be approximated as a matrix Fisher-Bingham density. As we verified experimentally, for certain choices of $\mathbf{G}$, the proposed method is capable of outperforming a previous approach that uses the prior importance function, at the expense, however, of increased computational complexity. In particular, the advantage of using the optimal importance function in lieu of the prior was more evident in our simulations in a scenario with stronger nonlinearity in the observation model.

Most of the computational complexity of the proposed method is related to drawing samples from and computing normalization constants for the matrix Fisher-Bingham density. As future work, we intend to investigate further these issues with the goal to reduce complexity.

6. REFERENCES

[1] V. Solo, "Attitude estimation and Brownian motion on SO (3)," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 4857–4862.

[2] Y. M. Lui, "Advances in matrix manifolds for computer vision," *Image and Vision Computing*, vol. 30, no. 6-7, pp. 380–388, 2012.

[3] J. Kwon and F. C. Park, "Visual tracking via particle filtering on the affine group," *Intl. J. Robotics Research*, vol. 29, no. 2-3, pp. 198–217, 2010.

[4] C. J. Bordin and M. G. S. Bruno, "Sequential Bayesian Algorithms for Identification and Blind Equalization of Unit-Norm Channels," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2157–2161, Nov 2015.

[5] C. J. Bordin Jr., C. G. de Figueredo, and M. G. S. Bruno, "Nonlinear state estimation on unit spheres using manifold particle filtering," *Digital Signal Processing*, vol. 81, pp. 50–56, 2018.

[6] J. Boulanger, S. Said, N. Le Bihan, and J. H. Manton, "Filtering from observations on Stiefel manifolds," *Signal Processing*, vol. 122, pp. 52–64, 2016.

[7] G. Marjanovic and V. Solo, "An Engineer's Guide to Particle Filtering on Matrix Lie Groups," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 3969–3973.

[8] A. Doucet, A. Smith, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Information Science and Statistics. New York: Springer, 2010.

[9] F. Tompkins and P. J. Wolfe, "Bayesian Filtering on the Stiefel Manifold," in *Computational Advances in Multi-Sensor Adaptive Processing, 2007. CAMPSAP 2007. 2nd IEEE International Workshop on*. IEEE, 2007, pp. 261–264.

[10] A. Kume, S. P. Preston, and A. T. A. Wood, "Saddlepoint Approximations for the Normalizing Constant of Fisher–Bingham Distributions on Products of Spheres and Stiefel Manifolds," *Biometrika*, vol. 100, no. 4, pp. 971–984, 2013.

[11] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," Tech. Rep. CUED/F-INFENG/TR 359, Cambridge University Engineering Department, 1999.

[12] A. Doucet, S. Godsill, and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statist. Comput.*, vol. 10, no. 3, pp. 197–208, Jul. 2000.

[13] R. van der Merwe, A. Doucet, N. de Freitas, and E. A. Wan, "The unscented particle filter," in *Advances in neural information processing systems*, 2001, pp. 584–590.

[14] S. Saha, P. K. Mandal, Y. Boers, H. Driessen, and A. Bagchi, "Gaussian proposal density using moment matching in SMC methods," *Stats. Comput.*, vol. 19, no. 2, pp. 203–208, 2009.

[15] M. G. S. Bruno, *Sequential Monte Carlo Methods for Nonlinear Discrete-Time Filtering*, Synthesis Lectures on Signal Processing. Morgan & Claypool, 2013.

[16] C. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer Texts in Statistics. Springer New York, 2013.

[17] Y. Chikuse, *Statistics on special manifolds*, vol. 174, Springer Science & Business Media, 2012.

[18] A. K. Gupta and D. K. Nagar, *Matrix variate distributions*, Chapman and Hall/CRC, 2018.

[19] J. T. Kent, A. M. Ganeiber, and K. V. Mardia, "A new unified approach for the simulation of a wide class of directional distributions," *Journal of Computational and Graphical Statistics*, vol. 27, no. 2, pp. 291–301, 2018.

[20] S. Byrne and M. Girolami, "Geodesic Monte Carlo on embedded manifolds," *Scandinavian Journal of Statistics*, vol. 40, no. 4, pp. 825–845, 2013.

[21] P. D. Hoff, "Simulation of the matrix Bingham–von Mises–Fisher distribution, with applications to multivariate and relational data," *Journal of Computational and Graphical Statistics*, vol. 18, no. 2, pp. 438–456, 2009.

[22] A. Kume, "Saddlepoint Approximations for the Bingham and Fisher-Bingham Normalising Constants," *Biometrika*, vol. 92, no. 2, pp. 465–476, Jun. 2005.

[23] I. Radović, I. M. Sobol, and R. F. Tichy, "Quasi-monte carlo methods for numerical integration: Comparison of different low discrepancy sequences," *Monte Carlo Methods and Applications*, vol. 2, no. 1, pp. 1–14, 1996.

[24] A. Edelman, "18.325: Finite Random Matrix Theory Volumes and Integration, Handout #4," `http://web.mit.edu/18.325/www/handouts.html`, March 2005.

[25] H. Karcher, "Riemannian Center of Mass and Mollifier Smoothing," *Communications on Pure and Applied Mathematics*, vol. 30, no. 5, pp. 509–541, 1977.

[26] R. Zimmermann, "A matrix-algebraic algorithm for the Riemannian logarithm on the Stiefel manifold under the canonical metric," *SIAM Journal on Matrix Analysis and Applications*, vol. 38, no. 2, pp. 322–342, 2017.

[27] T. Kaneko, S. Fiori, and T. Tanaka, "Empirical arithmetic averaging over the compact Stiefel manifold," *IEEE Transactions on Signal Processing*, vol. 61, no. 4, pp. 883–894, 2013.

[28] R. Chakraborty and B. Vemuri, "Statistics on the (compact) Stiefel manifold: Theory and Applications," *arXiv preprint arXiv:1708.00045*, 2017.