Robust estimation of rotations from relative measurements by maximum likelihood

Nicolas Boumal, Amit Singer and P.-A. Absil

***Abstract*— We estimate unknown rotation matrices R_i from a set of measurements of relative rotations $R_iR_j^T$. Measurements are strongly affected by noise such that a small fraction of them are well concentrated around the true relative rotations while the majority of measurements are outliers bearing little or no information. We propose a maximum likelihood estimator (MLE) that explicitly acknowledges this noise model, yielding a robust estimation algorithm. The MLE is computed via Riemannian trust-region optimization using the Manopt toolbox. Comparisons of the MLE with Cramér-Rao bounds suggest the estimator is asymptotically efficient.**

I. Introduction

Synchronization of rotations is the problem of estimating rotation matrices $R_1, \ldots, R_N \in \mathrm{SO}(n)$ from noisy measurements of relative rotations $R_iR_j^\top$, where $\mathrm{SO}(n)$ is the special orthogonal group:

$$\mathrm{SO}(n) = \{R \in \mathbb{R}^{n\times n} : R^\top R = I_n, \det(R) = +1\}.$$

Some of the rotations may be known in advance or not. This problem appears naturally in a number of applications, such as the calibration of a network of cameras [1], sensor network localization [2], the molecule problem [3], picture pose estimation [4] and cryo-EM imaging [5], for example.

In some of these applications, a large subset of the measurements may be of poor quality. These applications call for robust synchronization algorithms, capable of withstanding outliers. Hartley et al. [4] propose to estimate the rotations by minimizing an L1 norm of the disagreement between the model and the measurements, the Weiszfeld algorithm. The resulting algorithm is simple, fast and is shown to produce good results, but comes with little theoretical guarantees because of the nonconvexity and the nonsmoothness of the optimization problem they solve. Wang et al. [6] propose a convex relaxation of the synchronization problem, the LUD algorithm. LUD achieves exact recovery when a given portion of the measurements are exact, the other measurements being uniformly random. When the former measurements are slightly noisy rather than perfect, LUD is stable.

N. Boumal and P.-A. Absil are with the Department of Mathematical Engineering, ICTEAM Institute, Université catholique de Louvain, Belgium. NB is an FNRS research fellow. nicolasboumal@gmail.com, pa.absil@uclouvain.be

A. Singer is with the Program in Applied and Computational Mathematics, Princeton University, NJ, USA. amits@math.princeton.edu

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office. It was financially supported by the Belgian FRFC (Fonds de la Recherche Fondamentale Collective). A. Singer acknowledges support by AFSOR award FA9550-12-1-0317 and by the Simons Foundation award LTR DTD 06-05-2012.

Both the Weiszfeld algorithm and LUD address the synchronization problem by proposing a certain cost function at the onset. In contrast, following previous work [7], we will address synchronization by first assuming a specific noise model on the measurements. This statistical approach to the problem has a number of advantages: (i) the underlying assumptions about the noise are clear and could be adapted to individual applications; (ii) Cramér-Rao bounds can be derived that provide a meaningful target to compare algorithms against [7]; and (iii) the definition of maximum likelihood estimator naturally suggests an algorithm.

In the following, we first propose a noise model and define the associated maximum likelihood estimator (Section II), then we describe an algorithm to compute this estimator (Section III) and we study its performance against the Cramér-Rao bounds (Section IV).

Notation: bold letters always denote a tuple of matrices: $\mathbf{R} = (R_1, \ldots, R_N)$; the product acts component-wise: $\mathbf{R\Omega} = (R_1\Omega_1, \ldots, R_N\Omega_N)$. Given two nodes i and j in a graph, $i \sim j$ indicates they are neighbors.

II. A noise model for robust synchronization

In synchronization, the target quantities (the parameters) are the rotation matrices

$$R_1, \ldots, R_N \in \mathrm{SO}(n).$$

In order to estimate these rotations, we are given measurements H_{ij}. Each H_{ij} is a noisy measurement of the relative rotation $R_iR_j^\top$. The available measurements define an undirected *measurement graph* or *synchronization graph*

$$\mathcal{G} = (V, \mathcal{E}), \tag{1}$$

with $V = \{1, \ldots, N\}$ and $\{i, j\} \in \mathcal{E}$ if a measurement H_{ij} is available. By symmetry, $H_{ij} = H_{ji}^\top$. The measurements are modeled as follows:

$$H_{ij} = Z_{ij}R_iR_j^\top, \tag{2}$$

where $Z_{ij} \in \mathrm{SO}(n)$ is a random variable. In order to model a measurement that is concentrated around the true relative rotation, one can give Z_{ij} a probability density function (pdf) that is concentrated around the identity matrix I_n.

A popular Gaussian-like distribution on $\mathrm{SO}(n)$ is the *Langevin distribution*, which has the following pdf:

$$\ell_\kappa \colon \mathrm{SO}(n) \to \mathbb{R}^+, \quad \ell_\kappa(Z) = \frac{1}{c_n(\kappa)}\exp\big(\kappa\,\mathrm{trace}(Z)\big), \tag{3}$$

where $\kappa \geq 0$ is the *concentration parameter*. The pdf $\ell_\kappa(Z)$ attains it maximum at $Z = I_n$. The larger κ is, the

more ℓ_κ is concentrated around the identity. As an extreme case, ℓ_0 is constant over $\mathrm{SO}(n)$, i.e., it corresponds to a uniform distribution. If Z_{ij} is uniformly distributed, then so is H_{ij} and the measurement contains no information. On the other end, ℓ_∞ is the point-mass function at the identity. If Z_{ij} is deterministically equal to I_n, then H_{ij} is a noiseless measurement. For $0 < \kappa < \infty$, the measurement H_{ij} is isotropically distributed around its mean $R_i R_j^\top$. The normalization constant $c_n(\kappa)$ ensures that ℓ_κ has unit integral over $\mathrm{SO}(n)$ with respect to the Haar measure. Explicit formulas are worked out in [7] in terms of the modified Bessel functions of the first kind, I_ν. For $n = 2, 3$ we have:

$$c_2(\kappa) = I_0(2\kappa), \quad c_3(\kappa) = \exp(\kappa)(I_0(2\kappa) - I_1(2\kappa)).$$

In order to model the fact that only a fraction $0 \leq p \leq 1$ of the measurements are of decent quality while the remaining measurements contain little or no information, we propose to consider the following pdf for the noise rotations Z_{ij}:

$$f\colon \mathrm{SO}(n) \to \mathbb{R}^+, \quad f(Z) = p\,\ell_\kappa(Z) + (1-p)\,\ell_{\kappa'}(Z). \quad (4)$$

This mixture of Langevin's indeed captures the presence of a fraction $1-p$ of outliers if we let κ' be small compared to κ. With probability p, a measurement H_{ij} is distributed around $R_i R_j^\top$ with high concentration κ, and with probability $1-p$, it is distributed around the same mean with low concentration κ'. In the sequel, we will often consider $\kappa' = 0$.

We make the following assumption on the noise rotations:

Assumption 1: The Z_{ij}'s pertaining to different measurements are independent, identically distributed, with probability density function f (4).

The assumption that the Z_{ij}'s are identically distributed is merely to allow for a shorter exposition. All of what follows goes through if one assumes specific values of p, κ and κ' for each measurement individually (indeed, the Matlab code used in Section IV allows for such freedom). Independence, on the other hand, is a central assumption in the present work and cannot be relaxed easily. For convenience we further assume connectivity:

Assumption 2: The measurement graph (1) is connected.

If the graph is not connected, all of what follows may be applied to each connected component separately. With a little more care, this assumption can be relaxed too.

Under Assumption 1, the *log-likelihood function* L for synchronization of rotations is as follows:

$$L\colon \mathrm{SO}(n)^N \to \mathbb{R}, \quad L(\hat{\mathbf{R}}) = \sum_{i\sim j} \log f(H_{ij}\hat{R}_j\hat{R}_i^\top). \quad (5)$$

The summation is over the edges of the measurement graph.

Depending on the application, some of the rotations may be known in advance. They are called *anchors*. If no anchor is provided, synchronization can only be performed up to a global rotation. It is then acceptable, for the purpose of obtaining an estimator, to fix an arbitrary rotation to, say, the identity matrix. Let $A \subset \{1, \ldots, N\}$ denote the set of indices of anchors. The parameter space, that is, the space of acceptable values for an estimator, is

$$\mathcal{P}_A = \{\hat{\mathbf{R}} \in \mathrm{SO}(n)^N : \forall i \in A, \hat{R}_i = R_i\}. \quad (6)$$

The *maximum likelihood estimator* (MLE) $\hat{\mathbf{R}}_{\mathrm{MLE}}$ is the parameter that maximizes the log-likelihood function L:

$$\hat{\mathbf{R}}_{\mathrm{MLE}} = \operatorname*{argmax}_{\hat{\mathbf{R}}\in\mathcal{P}_A} L(\hat{\mathbf{R}}). \quad (7)$$

That it, $\hat{\mathbf{R}}_{\mathrm{MLE}}$ is the assignment of rotations $R_1, \ldots, R_N$ that best explains the observations H_{ij} under the assumed noise model and in the absence of prior information on the rotations. Since L is a smooth function defined over the smooth and compact manifold $\mathcal{P}_A$, a global maximizer exists.

Remark 1 (least-squares case): Assuming $p = 1$ (no outliers), the pdf of the Z_{ij}'s reduces to a simple Langevin prior: $f(Z) = \ell_\kappa(Z)$. The log-likelihood function then reads:

$$L(\hat{\mathbf{R}}) = \sum_{i\sim j} \kappa \operatorname{trace}(H_{ij}\hat{R}_j\hat{R}_i^\top) + \text{constant}. \quad (8)$$

Maximizing L is then equivalent to minimizing $\sum_{i\sim j} \kappa \left\|H_{ij}\hat{R}_j - \hat{R}_i\right\|_{\mathrm{F}}^2$. Hence, synchronization algorithms based on the minimization of this least-squares criterion over $\mathcal{P}_A$ are maximum likelihood estimators under a Langevin prior. This notably explains why such algorithms cannot be robust against outliers.

III. Computing the maximum likelihood estimator

We now propose an algorithm to compute $\hat{\mathbf{R}}_{\mathrm{MLE}}$. Because the optimization problem (7) is nonconvex, we only guarantee the computation of a local maximizer, so that our "MLE" is really only a proxy for the true $\hat{\mathbf{R}}_{\mathrm{MLE}}$. Nevertheless, Section IV shows that the algorithm performs well in practice, as compared to Cramér-Rao bounds.

The parameter space $\mathcal{P}_A$ (6) is a Riemannian submanifold of $(\mathbb{R}^{n\times n})^N$. The log-likelihood function

$$L_A = L|_{\mathcal{P}_A}, \quad (9)$$

that is, the restriction of L (5) to $\mathcal{P}_A$, is a smooth objective function defined over that manifold. Maximizing L_A over $\mathcal{P}_A$ is thus an instance of a smooth optimization problem on a manifold [8].

In this section, we start by describing a procedure to obtain an initial guess (a first iterate). It is based on a relaxation of the synchronization problem under a Langevin prior. This relaxation (as proposed in [9]) reduces the problem to the computation of a few dominant eigenvectors of a generalized eigenvalue problem. These vectors are further processed to obtain a feasible point on the parameter space $\mathcal{P}_A$. We then go on to establish the gradient and the Hessian of the cost function L_A to be maximized. A second-order Riemannian trust-region method, GenRTR [10], is then used within the Manopt framework [11] to improve on the initial guess, exploiting the gradient and Hessian information.

Notice that the parameters of the noise model (κ, κ' and p) are assumed known. In practice, these would have to be estimated from the data, for example through cross-validation. We leave this to further work.

A. An initial guess based on a spectral relaxation

The spectral relaxations of the synchronization problem developed in [9] for $\mathrm{SO}(2)$ then [5] for $\mathrm{SO}(3)$ and finally in [12] for the general case are suitable to produce cheap yet good solutions. Here, we show how [12, Algorithm 16] can be used to produce $\hat{\mathbf{R}}^{(0)}$ in $\mathcal{P}_A$ from a structured eigenvalue problem. Algorithm 1 summarizes the procedure.

Let $D \in \mathbb{R}^{N\times N}$ be a diagonal matrix such that $D_{ii} = \sum_{i\sim j} \kappa$, i.e., κ times the degree of node i. Following notations in [12], define $D_1 = D \otimes I_n$ (Kronecker product). Let $W_1 \in \mathbb{R}^{nN\times nN}$ be a symmetric matrix composed of $n \times n$ blocks such that the (i, j)-block $(W_1)_{ij}$ is κH_{ij} if nodes i and j are connected, and zero otherwise.

Let $X \in \mathbb{R}^{nN\times n}$ be composed of N stacked $n \times n$ blocks $X_1, \ldots, X_N$. Consider the following quadratic expressions:

$$X^\top D_1 X = \sum_{i=1}^{N} D_{ii}\, X_i^\top X_i, \tag{10}$$

$$X^\top W_1 X = \sum_{i\sim j} \kappa\, X_i^\top H_{ij} X_j + \kappa\, X_j^\top H_{ji} X_i. \tag{11}$$

Maximizing $\mathrm{trace}(X^\top W_1 X)$ subject to $X_i \in \mathrm{SO}(n)$ is equivalent to computing the maximum likelihood estimator for synchronization under a Langevin prior (see Remark 1, eq. (8)). This is difficult because of the nonconvexity of the constraints. Now observe that, under these same constraints, $X^\top D_1 X = \mathrm{trace}(D)\, I_n$. If we relax and simply impose the latter, i.e., that the columns of X be D_1-orthogonal, then maximizing $\mathrm{trace}(X^\top W_1 X)$ becomes easy: it is a generalized eigenvector problem with pencil (W_1, D_1).

This observation underpins [12, Algorithm 16]. Compute the n dominant D_1-orthonormal eigenvectors of W_1, i.e., compute $X \in \mathbb{R}^{nN\times n}$ as the solution of (notice that the scaling of X is irrelevant as long as it is fixed):

$$\max_X \mathrm{trace}(X^\top W_1 X) \text{ such that } X^\top D_1 X = I_n. \tag{12}$$

The global optimum of this problem can be computed efficiently, for example using `eigs` in Matlab.

In a noiseless scenario, the blocks X_i in the obtained solution will be orthogonal matrices (up to scaling). Because of noise in the measurements, this is, in general, not the case and one needs to project the X_i's to construct a feasible solution for the original problem. The proposed rounding procedure is to project each block to $\mathrm{SO}(n)$ as $R_i^{(a)} = \Pi_{\mathrm{SO}(n)}(X_i)$, where $\Pi_{\mathrm{SO}(n)} \colon \mathbb{R}^{n\times n} \to \mathrm{SO}(n)$ assigns to $R_i^{(a)}$ the rotation matrix that is closest to X_i in the sense of the Frobenius norm in $\mathbb{R}^{n\times n}$. This may be computed via the SVD decomposition $S = U\Sigma V^\top$, $s = \det(UV^\top)$ [13]:

$$\Pi_{\mathrm{SO}(n)}(S) = U\, \mathrm{diag}(1, \ldots, 1, s)\, V^\top, \tag{13}$$

where Σ is diagonal with decreasing entries and s is either 1 or -1 since U and V are orthogonal. As long as the smallest singular value of S has multiplicity one, this is uniquely defined [13, Prop. 3.3].

The solution X of the eigenvalue problem (12) is defined up to an orthogonal transformation. This means that even in the noiseless case where the individual blocks X_i would be orthogonal (up to scaling), they could turn out not to be rotation matrices, having negative determinant. To resolve this ambiguity, we also compute the projections of XJ, with $J = \mathrm{diag}(1, \ldots, 1, -1)$. Compute $R_i^{(b)} = \Pi_{\mathrm{SO}(n)}(X_i J)$. Finally, keep either $\mathbf{R}^{(a)}$ or $\mathbf{R}^{(b)}$ depending on which is more likely (eq. (5)). That is, set $\tilde{\mathbf{R}} = \mathbf{R}^{(a)}$ if $L(\mathbf{R}^{(a)}) \geq L(\mathbf{R}^{(b)})$, and $\tilde{\mathbf{R}} = \mathbf{R}^{(b)}$ otherwise.

This procedure yields an initial guess of rotations $\tilde{\mathbf{R}}$ that does not, in general, comply with the anchor constraints. We thus further globally align $\tilde{\mathbf{R}}$ with the anchors by computing [13]:

$$Q = \min_{Q\in\mathrm{SO}(n)} \sum_{i\in A} \|R_i - \tilde{R}_i Q\|_{\mathrm{F}}^2 = \Pi_{\mathrm{SO}(n)}\Big(\sum_{i\in A} \tilde{R}_i^\top R_i\Big).$$

The initial guess for the optimization step is $\hat{\mathbf{R}}^{(0)}$, where $\hat{R}_i^{(0)}$ is set to R_i if node i is anchored and to $\tilde{R}_i Q$ otherwise.

Algorithm 1 EIG: Compute the initial guess $\hat{\mathbf{R}}^{(0)}$

1: Form the sparse matrices D_1 (10) and W_1 (11) ;
2: Compute $X \in \mathbb{R}^{nN\times n}$, the dominant eigenvectors of the pencil (W_1, D_1) [Matlab: $[X, \sim]$ = `eigs`(W_1, D_1, n)] ;
3: **for all** $i \in 1 \ldots N$ **do**
4: $\quad R_i^{(a)} = \Pi_{\mathrm{SO}(n)}(X_i)$ and $R_i^{(b)} = \Pi_{\mathrm{SO}(n)}(X_i J)$;
5: **end for**
6: $\tilde{\mathbf{R}} = \begin{cases} \mathbf{R}^{(a)} & \text{if } L(\mathbf{R}^{(a)}) \geq L(\mathbf{R}^{(b)}), \\ \mathbf{R}^{(b)} & \text{otherwise} ; \end{cases}$
7: Anchor alignment: $Q = \Pi_{\mathrm{SO}(n)}\Big(\sum_{i\in A} \tilde{R}_i^\top R_i\Big)$;
8: **for all** $i \in 1 \ldots N$ **do**
9: $\quad \hat{R}_i^{(0)} = \begin{cases} R_i & \text{if } i \in A, \\ \tilde{R}_i Q & \text{otherwise} ; \end{cases}$
10: **end for**

B. Gradient of the log-likelihood L_A

The function L_A (9) is defined, and must be maximized, on $\mathcal{P}_A$ (6). The latter is a Riemannian submanifold of $(\mathbb{R}^{n\times n})^N$, that is, it is a smooth surface in the Euclidean space $(\mathbb{R}^{n\times n})^N$ endowed with the usual inner product

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{i=1}^{N} \mathrm{trace}(X_i^\top Y_i). \tag{14}$$

The gradient of a function defined on such a space is a tangent vector to the surface which we now define.

Let $\bar{L}$ be the function defined on $(\mathbb{R}^{n\times n})^N$ by the same analytic formula as L (5), such that L_A is merely the restriction of $\bar{L}$ to $\mathcal{P}_A$:

$$\bar{L} \colon (\mathbb{R}^{n\times n})^N \to \mathbb{R}, \quad \bar{L}(\hat{\mathbf{R}}) = \sum_{i\sim j} \log f(\hat{R}_i^\top H_{ij} \hat{R}_j). \tag{15}$$

(We permuted the matrices in the argument to f, which is fine since f only depends on the trace of its input.) The gradient of $\bar{L}$ can be computed in the usual way. Because $\mathcal{P}_A$ is a Riemannian submanifold of $(\mathbb{R}^{n\times n})^N$, the gradient

of L_A at a point $\hat{\mathbf{R}} \in \mathcal{P}_A$ is related to the gradient of $\bar{L}$ by this simple equation [8, eq. (3.37)]:

$$\operatorname{grad} L_A(\hat{\mathbf{R}}) = \mathrm{P}_{\hat{\mathbf{R}}}\big(\operatorname{grad} \bar{L}(\hat{\mathbf{R}})\big), \tag{16}$$

where $\mathrm{P}_{\hat{\mathbf{R}}}$ is the orthogonal projector (w.r.t. the metric (14)) from the ambient space $(\mathbb{R}^{n\times n})^N$ to the tangent space to the manifold (the surface) $\mathcal{P}_A$ at $\hat{\mathbf{R}}$. Explicitly, the i^{th} component of the gradient of L_A, that is, the gradient of L_A w.r.t. the i^{th} rotation $\hat{R}_i$, is given by:

$$\operatorname{grad}_i L_A(\hat{\mathbf{R}}) = \begin{cases} \hat{R}_i \operatorname{skew}\Big(\hat{R}_i^\top \operatorname{grad}_i \bar{L}(\hat{\mathbf{R}})\Big) & \text{if } i \notin A, \\ 0 & \text{if } i \in A, \end{cases} \tag{17}$$

where $\operatorname{skew}(M) = (M - M^\top)/2$. Gradient components pertaining to anchored rotations are forced to zero by the projector since these rotations cannot move. The other components are projected to a form $\hat{R}_i\Omega_i$ where Ω_i is skew-symmetric, since vectors of this form are the tangent vectors to $\mathrm{SO}(n)$ at $\hat{R}_i$. Indeed, differentiating the constraint $\hat{R}_i^\top \hat{R}_i = I_n$ yields the condition $\hat{R}_i^\top \dot{R} + \dot{R}^\top \hat{R}_i = 0$ for a vector $\dot{R}$ to be a tangent vector.

By definition, $\operatorname{grad}_i \bar{L}(\hat{\mathbf{R}})$ is the unique matrix in $\mathbb{R}^{n\times n}$ satisfying, for all X in $\mathbb{R}^{n\times n}$,

$$\operatorname{trace}\Big(X^\top \operatorname{grad}_i \bar{L}(\hat{\mathbf{R}})\Big) = \mathrm{D}_i \bar{L}(\hat{\mathbf{R}})[X], \tag{18}$$

where the r.h.s. is the directional derivative of $\bar{L}$ at $\hat{\mathbf{R}}$ w.r.t. the i^{th} rotation $\hat{R}_i$ along the direction X. In order to compute the gradient of $\bar{L}$, we thus compute its directional derivatives and proceed by identification in (18). Let us define

$$\hat{Z}_{ij} \triangleq \hat{R}_i^\top H_{ij} \hat{R}_j. \tag{19}$$

By the chain rule,

$$\mathrm{D}_i \bar{L}(\hat{\mathbf{R}})[X] = \sum_{i\sim j} \frac{1}{f(\hat{Z}_{ij})} \mathrm{D}f(\hat{Z}_{ij})[X^\top H_{ij} \hat{R}_j]. \tag{20}$$

The summation is over the nodes j that are neighbors of node i. The differential of f (4) is obtained as follows:

$$\mathrm{D}f(Z)[Y] = p\,\mathrm{D}\ell_\kappa(Z)[Y] + (1-p)\,\mathrm{D}\ell_{\kappa'}(Z)[Y], \tag{21}$$

$$\mathrm{D}\ell_\kappa(Z)[Y] = \kappa \ell_\kappa(Z) \operatorname{trace}(Y). \tag{22}$$

Combining (20)–(22), we further obtain:

$$\mathrm{D}_i \bar{L}(\hat{\mathbf{R}})[X] = \sum_{i\sim j} g(\hat{Z}_{ij}) \operatorname{trace}(X^\top \hat{R}_i \hat{Z}_{ij}),$$

$$g(\hat{Z}_{ij}) = \frac{p\kappa\ell_\kappa(\hat{Z}_{ij}) + (1-p)\kappa'\ell_{\kappa'}(\hat{Z}_{ij})}{f(\hat{Z}_{ij})}. \tag{23}$$

By identification with (18) and in combination with (17), this establishes the gradient of L_A:

$$\operatorname{grad}_i L_A(\hat{\mathbf{R}}) = \begin{cases} \hat{R}_i \sum_{i\sim j} g(\hat{Z}_{ij}) \operatorname{skew}\Big(\hat{Z}_{ij}\Big) & \text{if } i \notin A, \\ 0 & \text{if } i \in A. \end{cases} \tag{24}$$

Notice that the i^{th} component of the gradient can be computed based solely on the information pertaining to node i and its neighbors. This hints toward gradient-based decentralized synchronization algorithms (which we do not discuss).

C. Hessian of the log-likelihood L_A

Second-order optimization algorithms on Riemannian manifolds require the computation of the Riemannian Hessian of the objective function. For the particular case of Riemannian submanifolds such as $\mathcal{P}_A$, the Hessian admits a simple formulation in terms of the differential of the gradient in the ambient space.

For unanchored nodes ($i \notin A$), let us introduce the functions $G_i \colon (\mathbb{R}^{n\times n})^N \to \mathbb{R}^{n\times n}$ (see (23) for g):

$$G_i(\hat{\mathbf{R}}) = \hat{R}_i \sum_{i\sim j} g(\hat{Z}_{ij}) \operatorname{skew}\Big(\hat{Z}_{ij}\Big).$$

From (24), we know that the restriction of G_i to $\mathcal{P}_A$ yields the i^{th} gradient component of L_A. Then, the i^{th} component of the Hessian of L_A at $\hat{\mathbf{R}}$ applied to the tangent vector $\hat{\mathbf{R}}\boldsymbol{\Omega}$ is given by [8, eq. (5.15)] ($\boldsymbol{\Omega}$ is a tuple of skew-symmetric matrices):

$$\operatorname{Hess}_i L_A(\hat{\mathbf{R}})[\hat{\mathbf{R}}\boldsymbol{\Omega}] = \hat{R}_i \operatorname{skew}\Big(\hat{R}_i^\top \mathrm{D}G_i(\hat{\mathbf{R}})[\hat{\mathbf{R}}\boldsymbol{\Omega}]\Big).$$

That is, it is sufficient to differentiate the gradient vector field in the ambient space and then to (orthogonally) project the resulting vector field to the tangent spaces of $\mathcal{P}_A$. By the chain rule and the product rule:

$$\hat{R}_i^\top \mathrm{D}G_i(\hat{\mathbf{R}})[\hat{\mathbf{R}}\boldsymbol{\Omega}] = \sum_{i\sim j} \mathrm{D}g(\hat{Z}_{ij})[\hat{\Omega}_{ij}] \operatorname{skew}\Big(\hat{Z}_{ij}\Big)$$

$$+ \Omega_i \sum_{i\sim j} g(\hat{Z}_{ij}) \operatorname{skew}\Big(\hat{Z}_{ij}\Big) + \sum_{i\sim j} g(\hat{Z}_{ij}) \operatorname{skew}\Big(\hat{\Omega}_{ij}\Big),$$

where $\hat{\Omega}_{ij}$ is the directional derivative of $\hat{Z}_{ij}$ when $\hat{R}_i$ and $\hat{R}_j$ are moved (infinitesimally) along $\hat{R}_i\Omega_i$ and $\hat{R}_j\Omega_j$:

$$\hat{\Omega}_{ij} = \Omega_i^\top \hat{R}_i^\top H_{ij} \hat{R}_j + \hat{R}_i^\top H_{ij} \hat{R}_j \Omega_j = \hat{Z}_{ij}\Omega_j - \Omega_i \hat{Z}_{ij}.$$

This is not, in general, a skew-symmetric matrix. Some algebra yields the following formula for $\mathrm{D}g(\hat{Z}_{ij})[\hat{\Omega}_{ij}]$:

$$\left(\frac{p\kappa^2\ell_\kappa(\hat{Z}_{ij}) + (1-p)\kappa'^2\ell_{\kappa'}(\hat{Z}_{ij})}{f(\hat{Z}_{ij})} - g^2(\hat{Z}_{ij})\right) \operatorname{trace}(\hat{\Omega}_{ij}).$$

Combining equations in this subsection yields an explicit expression for the tangent vector $\operatorname{Hess} L_A(\hat{\mathbf{R}})[\hat{\mathbf{R}}\boldsymbol{\Omega}]$ for non-anchored nodes. For anchored nodes, $\operatorname{Hess}_i L_A(\hat{\mathbf{R}})$ vanishes.

D. Maximizing the likelihood

We use the GenRTR algorithm [10], which implements a second-order Riemannian trust-region method, to maximize the likelihood over the manifold $\mathcal{P}_A$. The Manopt toolbox [11] makes it straightforward to specialize GenRTR to the specific manifold $\mathcal{P}_A$ and the specific cost function L_A at hand. GenRTR converges globally (that is, from any initial guess) toward critical points (typically local optimizers) with quadratic local convergence.

The optimization algorithm is stopped once the norm of the gradient drops below $10^{-6}/|\mathcal{E}|$, where $|\mathcal{E}|$ is the number of measurements. The maximum trust-region radius is $\bar{\Delta} = \pi\sqrt{n(N - |A|)}$, which scales likes the diameter of the compact manifold $\mathcal{P}_A$; the initial radius is $\Delta_0 = \bar{\Delta}/8$. We allow up to 100 Hessian evaluations to solve each inner problem. The other parameters are set to their default value.

IV. Experiments

We now perform a few experiments on synthetic data to showcase properties of the proposed maximum likelihood estimator. Our main goal in this section is to study the performance of the MLE compared to known theoretical limits (Cramér-Rao bounds [7]) under the assumed noise model. Hence, in all tests, the measurements are generated following the noise model proposed in Section II and the specific parameters (κ, κ' and p) are known to the algorithm. We will see that, under these favorable conditions, the MLE appears to reach the Cramér-Rao bound in many cases.

A. Performance criterion and Cramér-Rao bounds

For a given estimator $\hat{\mathbf{R}}$ of the true rotations $\mathbf{R}$ with anchors indexed by A, assuming there is at least one anchor, the performance criterion we choose is the mean squared error (MSE) based on the geodesic distance on $\mathcal{P}_A$:

$$\mathrm{MSE}(\mathbf{R}, \hat{\mathbf{R}}) = \frac{1}{N - |A|} \sum_{i \notin A} \|\log(R_i^\top \hat{R}_i)\|_{\mathrm{F}}^2.$$

For rotations in the plane or in space ($n = 2$ or 3), $\|\log(R_i^\top \hat{R}_i)\|_{\mathrm{F}}/\sqrt{2}$ is the angle in radians of the rotation $R_i^\top \hat{R}_i$. For small errors, $\|\log(R_i^\top \hat{R}_i)\|_{\mathrm{F}} \approx \|R_i - \hat{R}_i\|_{\mathrm{F}}$.

In the absence of anchors, this performance criterion is unsuitable because the sought rotations can only be recovered up to a global rotation—see [7] for a discussion.

Because measurements are noisy, there is no hope to reduce the mean squared error to zero all the time. From [7], it is known that the expected MSE for synchronization is lower-bounded by some number which heavily relies on the topology of the measurement graph. The relevant features of the topology of the graph are captured by the spectrum of the *graph Laplacian*.

Following [7, § 5], the *information weight* of a measurement $ZR_iR_j^\top$ is defined by

$$w = \mathbb{E}\left\{ \|\mathrm{grad} \log f(Z)\|^2 \right\}, \tag{25}$$

where the expectation is taken w.r.t. Z distributed with pdf f. In the extreme case, if Z is uniformly distributed over $\mathrm{SO}(n)$, then f is constant and w is zero, i.e., the measurement contains no information. The more f is concentrated (that is, the less uncertainty there is), the larger the gradient of $\log f$ and thus the larger w is. The weight w can be computed by numerical integration using Weyl's integration formula [7].

Let us weigh each edge of the measurement graph with w. The Laplacian of the resulting graph is the symmetric, positive semidefinite matrix $\mathcal{L} \in \mathbb{R}^{N \times N}$ defined by:

$$\mathcal{L}_{ij} = \begin{cases} wd_i & \text{if } i = j, \\ -w & \text{if } i \sim j, \\ 0 & \text{otherwise,} \end{cases}$$

where d_i is the degree of node i. Further define the masked Laplacian $\mathcal{L}_A$ which is obtained by forcing to zero the rows and columns of $\mathcal{L}$ that correspond to anchored rotations:

$$(\mathcal{L}_A)_{ij} = \begin{cases} \mathcal{L}_{ij} & \text{if } i, j \notin A, \\ 0 & \text{otherwise.} \end{cases}$$

Then, the Cramér-Rao bounds (CRB) on the expected MSE of any unbiased estimator $\hat{\mathbf{R}}$ for the anchored synchronization problem is lower-bounded as follows [7, § 6.1]:

$$\mathbb{E}\left\{ \mathrm{MSE}(\mathbf{R}, \hat{\mathbf{R}}) \right\} \geq \frac{(n(n-1)/2)^2}{N - |A|} \mathrm{trace}(\mathcal{L}_A^\dagger) + \text{curvature terms},$$

where † denotes Moore-Penrose pseudoinversion. This bound is valid in a large signal-to-noise ratio (SNR) regime.

For complete graphs with one anchor, at large SNR, the CRB for rotations in $\mathrm{SO}(3)$ reduces to:

$$\mathbb{E}\left\{ \mathrm{MSE}(\mathbf{R}, \hat{\mathbf{R}}) \right\} \geq \frac{18}{wN} \left(1 - \frac{1}{wN} \right).$$

The larger the SNR (that is, the larger κ, κ' and p), the larger w (25) is and the lower the CRB.

Because the parameter space $\mathcal{P}_A$ is compact, even estimators that disregard measurements completely and return a random estimator would have finite MSE. Any reasonable estimator should perform at least as well as a random estimator. Hence, an upperbound on the MSE for rotations in $\mathrm{SO}(3)$ is given by [7, § 7]:

$$\mathbb{E}\left\{ \mathrm{MSE}(\mathbf{R}, \hat{\mathbf{R}}) \right\} \leq \frac{2\pi^2}{3} + 4.$$

B. Synthetic experiments

Figures 1, 2 and 3 show the expected mean squared error reached by the maximum likelihood estimator for varying noise parameters, in comparison with the Cramér-Rao bound and the expected MSE reached by the initial guess alone, as well as the LUD algorithm [6] (which does not have perfect knowledge of the noise model).

All tests are performed for synchronization of rotations in $\mathbb{R}^3$ ($n = 3$) with one anchor ($A = \{1\}$) on complete measurement graphs with $N = 400$ nodes, i.i.d. noise, $\kappa' = 0$. Setting $\kappa' = 0$ means measurements are either complete outliers (w.p. $1 - p$) or concentrated around the true relative rotation they measure with concentration κ (w.p. p).

The performance plots display an estimate of the expected MSE of estimators by averaging the MSE's obtained over a number of realizations of the noise. As a means to interpret the experiments, we point out that Langevin measurements with concentration $\kappa = 0.1, 1, 5$ and 10 are, on average, off by $123°, 81°, 30°$ and $21°$, resp. Likewise, it is useful to understand how good or bad an MSE level is. For $n = 3$, assuming the error is spread over all the rotations equally so that each rotation is off by an angle θ, then θ and the MSE are related by $\theta = \frac{180}{\pi}\sqrt{\frac{\mathrm{MSE}}{2}}$ (in degrees). MSE's of $10^{-2}, 10^{-1}$ and 10^0 correspond to average errors of, respectively, $4°, 13°$ and $40°$ on each rotation.

Figures 1–2 show that the MLE rapidly reaches the CRB and solves the synchronization problem as well as possible even at unfavorable SNR's. Figure 3 shows that for extremely low SNR's, the MLE may not reach the CRB. In future work, we will exploit the fact that most rotations are actually well estimated by the MLE, the error being mostly concentrated on a few unlucky (and detectable) rotations.

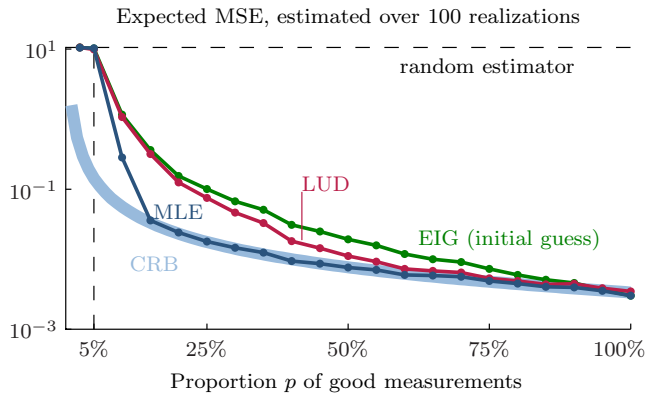

Fig. 1. Synchronization of a complete graph of $N = 400$ rotations in SO(3) with a variable proportion p of good measurements (concentration $\kappa = 5$). The remaining measurements are complete outliers ($\kappa' = 0$). As predicted in [9], for $p < 1/\sqrt{N} = 5\%$, even if the good measurements were perfect, they would bear insufficient information and both estimators perform as badly as a random estimator. For $p > 1/\sqrt{N}$, the MLE rapidly reaches the Cramér-Rao bound and appears to be efficient. The initial guess, based on the eigenvector method, is much improved by the MLE. LUD is the method proposed in [6]: it has no knowledge of the noise model. The initial guess is computed in 4 to 6 seconds. The MLE needs 2 to 20 additional seconds for p larger than 15%. For smaller p (corresponding to harder problems), the MLE may need 4 to 6 minutes to converge.

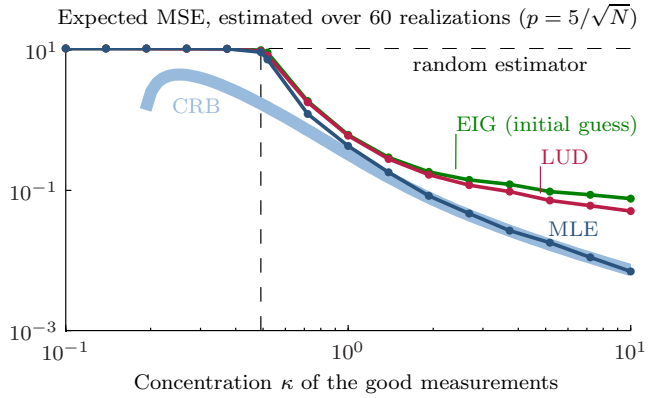

Fig. 2. Synchronization of a complete graph of $N = 400$ rotations in SO(3). A (comfortable) $p = 25\%$ of the measurements have variable concentration κ. The remaining 75% are complete outliers ($\kappa' = 0$). The proposed maximum likelihood estimator seems to rapidly reach the Cramér-Rao bound as the SNR increases. The vertical dashed line indicates the theoretically predicted point at which the eigenvector method starts performing better than a random estimator [9].

V. Conclusions

We proposed an outlier-aware noise model for synchronization of rotations. The maximum likelihood estimator for this model is the solution of a smooth optimization problem on a manifold. To compute the MLE, we first obtain an initial guess by means of a known relaxation which reduces synchronization to an eigenvector problem. Then, this initial guess is used as a starting point for an advanced second-order Riemannian trust-region optimization scheme. While this algorithm does not guarantee global optimality, comparing the variance of the obtained estimator with Cramér-Rao bounds suggests that it is indeed asymptotically efficient.

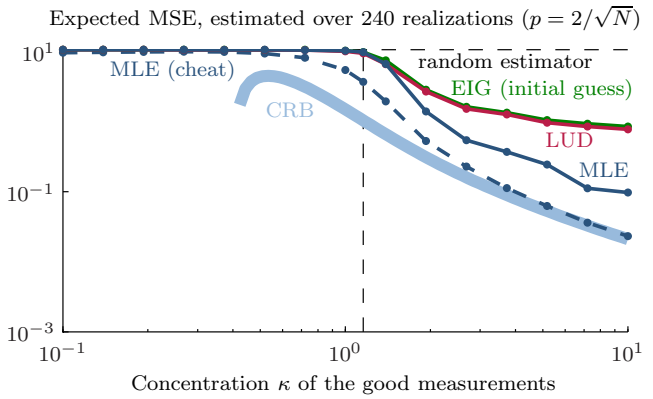

Fig. 3. Same experiment as Fig. 2, with a challenging $p = 10\%$. The remaining 90% of the measurements bear no information. In this extreme scenario, the computed MLE does not seem to reach the Cramér-Rao bound. This is in part due to nonglobal optimization of the likelihood function. Indeed, the *MLE (cheat)* dashed curve shows the expected mean square error reached when using the true rotations as initial guess for the optimization stage. For $\kappa > 1$ (experimentally), the such reached critical point is more likely than the one reached with the legitimate MLE algorithm, and indeed performs better, indicating that the performance could be improved using global optimization techniques.

References

[1] R. Tron and R. Vidal, "Distributed image-based 3D localization of camera sensor networks," in *Decision and Control, held jointly with the 28th Chinese Control Conference. Proceedings of the 48th IEEE Conference on*. IEEE, 2009, pp. 901–908.

[2] M. Cucuringu, Y. Lipman, and A. Singer, "Sensor network localization by eigenvector synchronization over the Euclidean group," *ACM Transactions on Sensor Networks*, vol. 8, no. 3, pp. 19:1–19:42, 2012.

[3] M. Cucuringu, A. Singer, and D. Cowburn, "Eigenvector synchronization, graph rigidity and the molecule problem," *Information and Inference: A Journal of the IMA*, vol. 1, no. 1, pp. 21–67, 2012.

[4] R. Hartley, K. Aftab, and J. Trumpf, "L1 rotation averaging using the Weiszfeld algorithm," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3041–3048.

[5] A. Singer and Y. Shkolnisky, "Three-Dimensional Structure Determination from Common Lines in Cryo-EM by Eigenvectors and Semidefinite Programming," *SIAM Journal on Imaging Sciences*, vol. 4, no. 2, pp. 543–572, 2011. [Online]. Available: http://link.aip.org/link/?SII/4/543/1

[6] L. Wang and A. Singer, "Exact and stable recovery of rotations for robust synchronization," to appear in Information and Inference: A Journal of the IMA, 2013.

[7] N. Boumal, A. Singer, P.-A. Absil, and V. Blondel, "Cramér-Rao bounds for synchronization of rotations," to appear in Information and Inference: A Journal of the IMA, 2013.

[8] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008.

[9] A. Singer, "Angular synchronization by eigenvectors and semidefinite programming," *Applied and Computational Harmonic Analysis*, vol. 30, no. 1, pp. 20–36, 2011.

[10] P.-A. Absil, C. G. Baker, and K. A. Gallivan, "Trust-region methods on Riemannian manifolds," *Found. Comput. Math.*, vol. 7, no. 3, pp. 303–330, July 2007.

[11] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, "Manopt: a matlab toolbox for optimization on manifolds," *arXiv preprint arXiv:1308.5200 [cs.MS]*, 2013.

[12] A. Bandeira, A. Singer, and D. Spielman, "A Cheeger Inequality for the Graph Connection Laplacian," *Arxiv preprint arXiv:1204.3873*, 2012.

[13] A. Sarlette and R. Sepulchre, "Consensus optimization on manifolds," *SIAM J. Control and Optimization*, vol. 48, no. 1, pp. 56–76, 2009.