A Unified Approach for Pose Graph Optimization

Gabriel Antunes Moreira
gmoreira@isr.tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa, Portugal

September 2020

Abstract

Pose Graph Optimization (PGO) is an important problem in Computer Vision, particularly in motion estimation, whose objective consists of finding the rigid transformations that achieve the best global alignment of visual data on a common reference frame. The majority of PGO approaches rely on iterative techniques which refine an initial estimate until convergence is achieved. On the other hand, recent works have identified a global constraint which has cast this problem into the matrix completion domain. The success which both of these formulations have had in computing accurate solutions efficiently has been overshadowed by large-scale industrial applications such as autonomous flight, self-driving cars and smart-cities, where it is necessary to fuse numerous images covering large areas but where each one of them has few pairwise observations. We developed a methodology that unifies these alternative formulations, which until now, coexisted unable to share the advantages of each other, resulting in a closed-form solution in the basin of attraction of the global optimum. Our formulation allows for high scalability, low computational cost and high precision, simultaneously.

1. Introduction

The registration of 3D point sets obtained by LiDARs and RGB-D cameras is one of the core problems in Robotics and Computer Vision, with applications ranging from dense scene reconstruction to localization. If we have two points sets, Iterative Closest Point (ICP) [4][18] is a common procedure to solve the registration problem. This class of algorithms requires an initialization and iterates between estimating point correspondences and computing the best transformation between them.

Consider now that we have a set of point clouds corresponding to different views of the same scene. An example is represented on the left side of Fig. 1. The different point sets may be obtained via an array of 3D scanners. Alternatively, they may represent the visual data acquired by single observer as it moves through space. Both situations have seen a rise in popularity recently with applications such as smart-cities and autonomous transportation systems e.g. self-driving cars and drones, where the registered 3D data may be used to perform object detection, tracking, and mapping. For known point correspondences, the rigid transformations that allow for an optimal registration of the point sets can be solved for in closed-form via Generalized Procrustes Analysis [9]. For unknown correspondences, optimization strategies analogous to ICP have been proposed [20]. Notwithstanding, such optimization schemes are computationally inefficient even for moderately sized registration tasks. To circumvent this issue, Pose Graph Optimization (PGO) is usually the method of choice in 3D-SLAM and dense scene reconstruction.

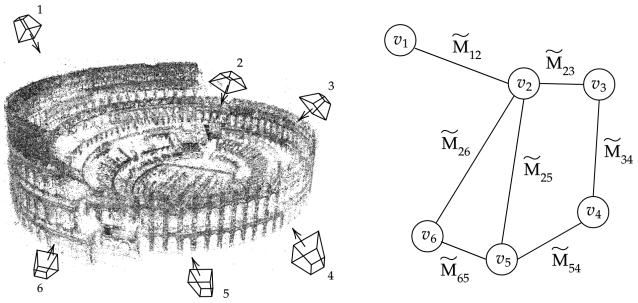

Figure 1: Pose graph example. Colosseo dataset.

PGO or motion averaging consists of estimating a set of rigid transformations, given a subset of pairwise measurements of their ratios. The latter is usually computed via ICP or 2D image feature matching between overlapping frames. By associating the rigid transformation measurement $\widetilde{M}_{ij}$, from point cloud i to point cloud j, to an edge $(i, j) \in E$, we obtain a simple graph $\mathcal{G} = (V, E)$ which, if connected, can be used to derive a cost function that is minimized by the global transformations M_i that best fit the measured

data. An example of a possible cost function is the one from Eq. (1), which makes no assumptions about the measured transformations.

$$\underset{M_i \in \mathrm{SE(3)}}{\arg\min} \sum_{(i,j)\in E} ||\widetilde{M}_{ij} - M_i M_j^{-1}||_F^2 \tag{1}$$

A pose graph example is represented on the right side of Fig. 1. In real world applications, guaranteeing globally optimal solutions to PGO is paramount. However, this optimization task is a high-dimensional and non-concave problem. Second-order methods bootstrapped with robust initializations can converge to the sought-after optimum. Notwithstanding, they do not scale well. On the other hand, even if certain relaxations of the original problem allow for more efficient implementations, their solutions are often far from the those of the original problem.

We address the problem of PGO in the context of point cloud registration. The desiderata for our solution are: scalability, efficiency and accuracy. While existing methods, namely those proposed in [7] [2] [15] [17] satisfy a subset of these requirements, they usually incur an efficiency accuracy trade-off. This paper builds on the ML function put forward by Carlone et al. [6], the spectral decomposition relaxation derived by Fusiello et al. [3] and the optimality verification techniques in SO(3) put forward by Eriksson et al. [11]. We close the gap between these formulations and derive a quasi-optimal solution for PGO in SO(3) and SE(3) that can be computed faster than all other state-of-the-art approaches. More specifically, we make a two-fold contribution:

- We demonstrate that in applications where a high Signal-to-Noise Ratio (SNR) is a valid assumption, e.g. in RGB-D registration, the problem of finding the optimal rigid transformations can be split into two manageable subproblems which deal separately with rotations and translations.

- We propose two solutions corresponding to the subproblems mentioned. We solve rotation averaging by means of the Krylov-Schur [19] algorithm for spectral decomposition and Cholesky factorization. Solving for translations is accomplished by using the same Cholesky solver, retaining the preconditioning used in the Krylov-Schur method. Our solution is shown empirically to be in the basin of attraction of the global optimum. Additionally, a simple outlier detection method is presented which succeeds in improving the closed-form solution when the noise model assumption is not verified.

A C++ implementation of our algorithm is available at https://github.com/gabmoreira/pipe.

2. Related work

The literature on PGO can be segmented into three different clusters: Maximum Likelihood Estimation (MLE) by means of nonlinear iterative solvers; suboptimal relaxations which do not guarantee local optimality but may produce accurate solutions that can be used to bootstrap other methods; optimality verification and globally optimal strategies.

MLE is arguably the most popular approach to PGO. The optimization problem it gives rise to, which is both non-convex and high-dimensional, is usually tackled via state-of-the-art Gauss-Newton and Levenberg-Marquandt methods. Noteworthy examples of such solvers are g2o by Kummerle et al. [14] and GTSAM by Dellaert [10]. To ensure global convergence however, these techniques rely on initializations in the basin of attraction of the global optimum. This problem has been addressed by Carlone et al. [6], who studied different rotation initialization techniques.

In SO(3), Tron et al. [21] established a link between graph consensus algorithms and the Riemannian gradient descent on the SO(3) manifold. In spite of its good convergence properties, this technique is arguably slower than most approaches. Martinec et al. [15] addressed the same problem by means of a chordal relaxation, whereby the solution to a least-squares problem is projected to SO(3). Despite its inherent suboptimality, it scales well and can be used to initialize iterative solvers. Other remarkable works in rotation averaging include the seminal paper by Hartley et al. [13], a Quasi-Newton method set forth by Chatterjee et al. [8], and more recently, a deep learning approach by Purkait et al. [16].

In contrast to the formulations mentioned hitherto, Arrigoni et al. [2] have cast PGO into the Low-Rank and Sparse (LRS) decomposition domain. The relaxation underlying this methodology allows it to work both in SE(3) and SO(3), but lends itself to invariably suboptimal solutions. Foregoing the sparse term, introduced to capture outliers, this approach degenerates to the well studied problem of low-rank matrix completion [1][12]. In another paper, the same authors derived a different relaxation which admits a closed-form solution that can be computed efficiently via spectral decomposition [3].

In recent literature, global optimality has been the focus of extensive research. Carlone et al. [6] derived the Lagrangian dual problem for PGO in SE(3), which can be used to validate an estimate as the global optimum and, in certain cases, allows for the direct retrieval of the optimal solution. A similar strategy was adopted to solve rotation averaging in SO(3) by Eriksson et al. [11], who also put forward several results pertaining to optimality verification. A faster and certifiably correct approach in SE(3) via semidefinite relaxation was proposed by Rosen et al. [17].

While it is clear that the research in this field seems to converge towards globally optimal solutions, these ap-

proaches are still too computationally expensive to be used in real-time applications. Conversely, efficient suboptimal relaxations are often not robust enough to be applicable in industrial settings. As we will demonstrate, when considering Computer Vision applications, namely RGB-D camera registration, our quasi-optimal solution can be computed at a fraction of the cost of state-of-the-art algorithms.

3. Proposed approach

Let $\mathcal{G} = (V, E)$ be a simple and connected graph with $|V| = n$ the number of poses and $\{\widetilde{R}_{ij}, \widetilde{t}_{ij}\}$ for $(i, j) \in E$ the rigid transformation measurement from pose i to pose j. To render algebraic manipulation more tractable we will henceforth make use of the following block-matrix notation. Let $t \in \mathbb{R}^{3n}$ and $R \in \mathrm{SO(3)}^n \subset \mathbb{R}^{3n\times 3}$ such that

$$t = \begin{bmatrix} t_1^\top & \dots & t_n^\top \end{bmatrix}^\top$$

$$R = \begin{bmatrix} R_1^\top & \dots & R_n^\top \end{bmatrix}^\top, \quad R_i \in \mathrm{SO(3)}$$

with $\{R_i, t_i\}_{i=1,\dots,n}$ the rigid transformation corresponding to the i-th pose. Let $\theta = \{R, t\}$ and $y = \{\widetilde{R}_{ij}, \widetilde{t}_{ij}\}_{(i,j)\in E}$ be the set of parameters and observations, respectively. Assuming an isotropic Gaussian generative noise model with variance σ_t^2 for the translation measurements, an isotropic Langevin noise model with concentration parameter $1/\sigma_R^2$ for the rotations [5] and inter-independence amongst the random variables involved, the log-likelihood function is given by Eq. (2).

$$\log L(\theta|y) = -\frac{1}{2\sigma_t^2} \sum_{(i,j)\in E} ||\widetilde{t}_{ij} - t_i + R_i R_j^\top t_j||^2 + \frac{1}{\sigma_R^2} \sum_{(i,j)\in E} \mathrm{tr}\, \widetilde{R}_{ij} R_j R_i^\top \quad (2)$$

PGO can be formulated as seeking $\theta^* = \{R^*, t^*\}$ that maximizes $\log L(\theta|y)$, i.e.

$$\underset{\theta\in \mathrm{SO(3)}^n \times \mathbb{R}^{3n}}{\arg\max} \log L(\theta|y) \quad (3)$$

The optimization task from Eq. (3) is high-dimensional, constrained and non-concave. Notwithstanding, provided there is an estimate of R close to R^*, solving for translations is a trivial convex least-squares problem. We can thus raise the question regarding the validity of separating the optimization problem into two manageable subproblems: $\mathcal{R}$ which is known in the literature as multiple rotation averaging and $\mathcal{T}$ which computes the set of optimal translations for a set of rotation estimates $\widehat{R}$.

$$\mathcal{R} : \underset{R\in \mathrm{SO(3)}^n}{\arg\max}\, \mathrm{tr}\, \widetilde{R}_{ij} R_j R_i^\top$$

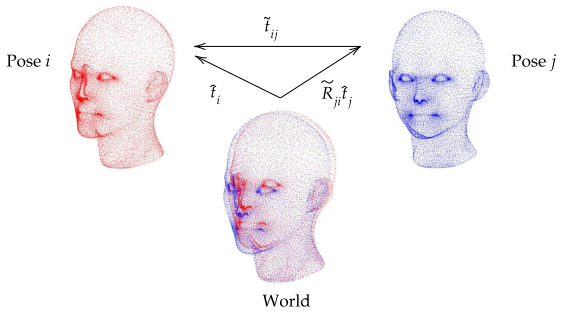

Figure 2: Representation of the high SNR hypothesis. The point clouds were purposefully registered with an angular error of 4 deg. The translations $\hat{t}_i$, $\widetilde{t}_{ij}$ and $\widetilde{R}_{ji}\hat{t}_j$ are in the coordinate system i.

$$\mathcal{T} : \underset{t\in\mathbb{R}^{3n}}{\arg\min} \sum_{(i,j)\in E} ||\widetilde{t}_{ij} - t_i + \widehat{R}_i \widehat{R}_j^\top t_j||^2$$

As it turns out, provided there is a high Signal-to-Noise Ratio (SNR), this approximation, which we formalize in Hypothesis 1, is valid.

Hypothesis 1 (High SNR) *Let t_i^* be the optimal translation corresponding to pose i. We define high SNR as*

$$\forall\, (i,j) \in E \;:\; \widetilde{R}_{ji} t_j^* \overset{H1}{\approx} t_i^* - \widetilde{t}_{ij}$$

As a sanity check, note that this approximation becomes an equality when the measurements are equal to the ground truth.

In Computer Vision applications, this hypothesis is usually verified since the relative transformations are obtained either via Iterative Closest Point (ICP) algorithms, 2D image feature matching or a combination of both. As an example, we show in Fig. 2, that a pairwise registration with an angular error of 4 degrees, which could otherwise be considered a small figure, can be deemed inaccurate. In the subsequent sections, we will derive a quasi-optimal solution to problem $\mathcal{R}$, which we then use to solve $\mathcal{T}$.

3.1. Multiple rotation averaging

In order to find an approximate solution to problem $\mathcal{R}$, we will show that the closed-form eigenspace solution derived by Fusiello et al. [3], which is optimal for $\sigma_R = 0$, is actually a good approximation of the global optimum under the high SNR hypothesis we put forward. Furthermore, we show that the respective eigenvalues, which are implicitly computed in the process, provide an insight to the optimality of the solution. Finally, we put forward a simple fixed-point iteration, which refines this solution, should it be deemed suboptimal.

We will stack the rotation measurements $\widetilde{R}_{ij}$ in a block matrix $\widetilde{R} \in \mathbb{R}^{3n\times 3n}$, defined as

$$\widetilde{R} = \begin{cases} \widetilde{R}_{ij} & \text{if } (i,j) \in E \\ I_3 & \text{if } i = j \\ 0_3 & \text{otherwise} \end{cases}$$

where $I_3 \in \mathbb{R}^{3\times 3}$ and $0_3 \in \mathbb{R}^{3\times 3}$ denote the identity matrix and null matrix, respectively. By doing so, we can restate the problem of multiple rotation averaging $\mathcal{R}$ as seeking R^* such that

$$R^* = \underset{R\in \mathrm{SO(3)}^n}{\arg\min} - \text{tr } R^\top \widetilde{R} R$$

which will be hereafter referred to as the primal problem. By relaxing the group constraint $R \in \mathrm{SO(3)}^n$ for $R \in \mathrm{O(3)}^n$ we can write the Lagrangian [11] as

$$L(R,\Lambda) = -\text{tr } R^\top \widetilde{R} R - \text{tr } \Lambda(I - RR^\top)$$

where the symmetric diagonal block matrix Λ acts as the Lagrange multiplier. The dual problem consists of solving the semidefinite program

$$\underset{\Lambda - \widetilde{R} \succeq 0}{\arg\max} -\text{tr } \Lambda$$

In the trivial case, with $\sigma_R = 0$, the expanded graph degree matrix defined as $\Lambda^* = D\otimes I_3 + I_{3n}$ is the optimal Lagrange multiplier. In fact, the symmetric matrix

$$\Lambda^* - \widetilde{R} = (\mathcal{L} \otimes \mathbb{1}_3) \circ \widetilde{R}$$

where $\mathcal{L}$ denotes the graph Laplacian and $\mathbb{1}_3 \in \mathbb{R}^{3\times 3}$ a matrix with ones, has a null eigenvalue with geometric and algebraic multiplicities 3. Furthermore, the respective eigenspace $E_0 \in \mathbb{R}^{3n\times 3}$ intersects $\mathrm{SO(3)}^n$. For any estimate $\widehat{R} \in E_0$, the dual problem is worth, in this case,

$$-\text{tr } \Lambda^* = -3\sum_{i=1}^{n} 1 + \deg(v_i) = -3\big(n + 2|E|\big)$$

and the primal

$$-\text{tr } \widehat{R}^\top \widetilde{R} \widehat{R} = -\text{tr } \widehat{R}^\top (D \otimes I_3 + I_{3n}) \widehat{R} = -3\big(n + 2|E|\big)$$

where we used the following equality, derived from the eigenvector equation

$$(D \otimes I_3 + I_{3n}) \widehat{R} = \widetilde{R} \widehat{R}$$

The duality gap being zero implies that this is the global optimum. To make the transition to the realistic case where $\sigma_R > 0$, we put forward the following Conjecture, verified empirically.

Conjecture 1 *The expected value of the smallest eigenvalue of $(\mathcal{L} \otimes \mathbb{1}_3) \circ \widetilde{R}$, hereafter denoted by $\overline{\lambda_1}$, converges monotonically to a positive scalar c, for increasing σ_R. Mathematically this translates to:*

$$\lim_{\sigma_R \to \pi} \overline{\lambda_1}((\mathcal{L} \otimes \mathbb{1}_3) \circ \widetilde{R}) = c$$

Let $\widehat{\Lambda} = D$. According to Conjecture 1, we expect $\widehat{\Lambda} - \widetilde{R}$ to become positive definite for increasing σ_R. Nevertheless, within the validity of Hypothesis 1, we can expect $\lambda_1 \approx 0$, especially for sparse, poorly connected graphs (with a small Fiedler value), which is usually the case in 3D SLAM. Under this assumption, we have

$$(\widehat{\Lambda} - \widetilde{R})\widehat{R} \approx 0$$

which makes $\widehat{R}$ a quasi-stationary point. In this approximation, the dual problem will still yield the same value, but we can expect the duality gap to increase with σ_R, i.e. the degree matrix becomes a bad approximation of the Lagrange multiplier for noisier measurements.

Our solution, named EigenRA, consists thus of computing the three eigenvectors of $(\mathcal{L} \otimes \mathbb{1}_3) \circ \widetilde{R}$ corresponding to the smallest eigenvalues, and then projecting them to $\mathrm{SO(3)}^n$ via Singular Value Decomposition (SVD). This estimate can be accepted as the global optimum, provided the smallest eigenvalues lie below a positive threshold.

For large and sparse matrices, the spectral decomposition can be efficiently computed using the Krylov-Schur algorithm by Stewart [19]. This method computes an orthonormal basis $\{v_1, \ldots, v_s\}$ for the Krylov subspace of

$$\big((\mathcal{L} \otimes \mathbb{1}_3) \circ \widetilde{R} - \sigma I\big)^{-1}$$

where σ denotes the shift operator, in our case close to zero. The Krylov iterations, as defined in Eq. (4), can take advantage of the symmetry of the shifted matrix by solving for v_{k+1} via Cholesky $LDL^\top$ factorization.

$$\big((\mathcal{L} \otimes \mathbb{1}_3) \circ \widetilde{R} - \sigma I\big)^{-1} v_k = v_{k+1} \tag{4}$$

To handle cases where our approximation may be flawed, albeit rare in the applications we are considering, we show that the fixed-points of the map

$$T : \mathrm{SO(3)}^n \longrightarrow \mathrm{SO(3)}^n$$
$$T = (\widetilde{R} R)_{\downarrow \mathrm{SO(3)}^n}$$

coincide with the stationary points of $-\text{tr } R^\top \widetilde{R} R$ on $\mathrm{SO(3)}^n$. Consequently, starting from our baseline solution, the following fixed-point iterations allow us to converge to the global optimum.

$$\widehat{R}^{k+1} = (R\widehat{R}^k)_{\downarrow \mathrm{SO(3)}^n}$$

3.2. Optimizing for translations

We will now show that for a set of rotation estimates $\widehat{R}$, the optimal translations can be computed by solving a symmetric linear system with the same pattern as the one used in the Krylov-Schur method, implemented in our rotation averaging solution. Let $b \in \mathbb{R}^{3n}$ be defined as

$$b = \begin{bmatrix} \sum_{j \in E(v_1)} \frac{1}{2}(\widetilde{t}_{1j} + \widehat{R}_1 \widehat{R}_j^\top \widetilde{R}_{j1} \widetilde{t}_{1j}) \\ \vdots \\ \sum_{j \in E(v_n)} \frac{1}{2}(\widetilde{t}_{nj} + \widehat{R}_n \widehat{R}_j^\top \widetilde{R}_{jn} \widetilde{t}_{nj}) \end{bmatrix}$$

The derivative of the log-likelihood w.r.t to t is given by

$$\frac{\partial}{\partial t} \log L(\theta|y) = \big((\mathcal{L} \otimes \mathbb{1}_3) \circ R R^\top\big) t + b$$

It follows then that the solution to the convex problem $\mathcal{T}$ can be obtained by solving Eq. (5).

$$\big((\mathcal{L} \otimes \mathbb{1}_3) \circ R R^\top\big) t + b = 0 \tag{5}$$

Since $(\mathcal{L} \otimes \mathbb{1}_3) \circ \widehat{R}\widehat{R}^\top$ is symmetric, we can use Cholesky $LDL^\top$ factorization to solve for t. The advantage of doing so lies in the fact that the rotation eigenvectors computed via Krylov-Schur make use of the same solver. Attending to the fact that the matrices being factorized share the same structure of zero and non-zero entries, the preconditioning performed by the solver while solving for rotations is valid afterwards, when optimizing for translations. This results in a considerable gain in performance. Our algorithm for PGO in SE(3) will be hereafter referred to as EigenMA.

3.3. Outlier detection

Since the computation of pairwise transformations via ICP or feature matching is prone to generate outliers, we devised an iterative method, EigenRAO, to identify model-incoherent rotation measurements as follows.

A weight matrix $W \in \mathbb{R}^{n \times n}$ is initialized with ones. Iteratively: the block matrix $\widetilde{R}$ with the measurements, weighted by W, i.e. $(W \otimes \mathbb{1}_3) \circ \widetilde{R}$ is used to compute the eigenvector solution $\widehat{R}$, let λ_1 be the associated smallest eigenvalue; an error matrix P is computed containing the errors of all the edges of the graph; rotation measurements with an error above a certain threshold η are replaced in $\widetilde{R}$ by their estimate $\widehat{R}_i \widehat{R}_j^\top$ (instead of removing the respective edge from the graph which would reduce its connectivity); the weight matrix is updated as $W \leftarrow W \circ \rho(P)$ (where ρ is a loss function); the sum of the entries of the i-th line of W, with the exception of the diagonal entry, is set to $\deg(v_i)$ and its diagonal is set to one; finally, W is replaced by its symmetric component. The steps are repeated until $|\lambda_1|$ falls below a predefined threshold.

We found, empirically, that the choice of the loss function ρ played an important role in the rate of convergence and ground-truth errors. We chose the loss function from Eq. (6), which can be fine-tuned via the parameters a and b.

$$\rho(P; a, b) = a + b \exp\left(\pi - \frac{P}{2}\right) \tag{6}$$

The reasoning behind this algorithm comes from the knowledge that in the case without outliers λ_1 should be close to 0. By iteratively updating the weights, the transformations which remain unexplained by the eigenvector (large error) over many iterations incur increasingly larger penalties (smaller weights). Consequently, outlier transformations will have an increasingly smaller impact on the spectrum and eigenspaces and thus λ_1 decreases in absolute value. Generally, from a certain weight matrix W onward, λ_1 will cease to change and the algorithm stops.

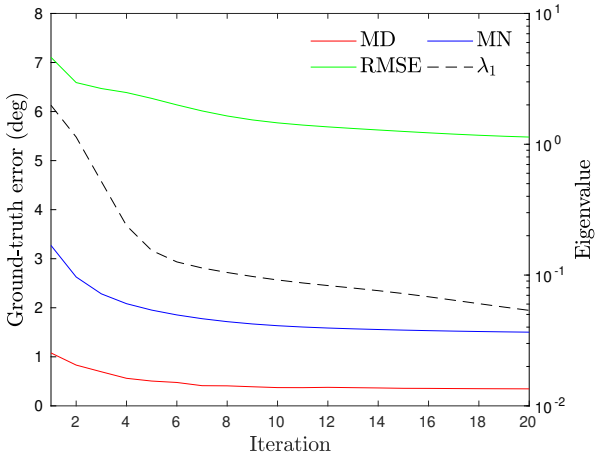

Figure 3: Convergence of the ground-truth RMSE, mean (MN) and median (MD) errors and of the smallest eigenvalue of the block matrix used in EigenRAO for the dataset *Ellis Island*.

4. Evaluation and experiments

In this section we present the results of several motion averaging simulations and benchmarks in order to assess the performance of our algorithms and establish meaningful comparisons against the state-of-the-art. All the tests were conducted using an Intel Core i7-4700HQ CPU with a maximum clock frequency of 3.4GHz and 16GB of RAM. Our code was implemented both in C++ and MATLAB and is available online.

4.1. Rotation averaging

In order to assess the quality of a set of rotation estimates $\widehat{R}$ and quantify its error w.r.t to the global optimum or ground-truth, we will use three metrics often featured in the literature. These are the mean (MN), the median (MD)

and the root-mean-square error (RMSE), now defined. Let $d_{\text{geo}}(\widehat{R}_i, R_i^{\text{GT}})$ denote the geodesic distance in SO(3) [13], between the i-th ground-truth rotation R_i^{GT} and the respective estimate $\widehat{R}_i$, then

$$\text{MD} = \text{median}\{d_{\text{geo}}(\widehat{R}_i, R_i^{\text{GT}})\}_i$$

$$\text{MN} = \frac{1}{n}\sum_{i=1}^{n} d_{\text{geo}}(\widehat{R}_i, R_i^{\text{GT}})$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} d_{\text{geo}}(\widehat{R}_i, R_i^{\text{GT}})^2}$$

4.1.1 On the quasi-optimality of our solution

We begin the analysis of the proposed rotation averaging algorithm EigenRA, by an empirical study on the quasi-optimality of our solution. Our goal is to show that, assuming a small noise standard deviation σ_R, i.e. under Hypothesis 1, our solution is in the basin of attraction of the global optimum and can be accepted as optimal with a negligeable angular error.

Consider three random connected graphs $\mathcal{G}_i = (V, E_i)$ with $|V| = 5750$ and a variable set of edges. The graphs have the following algebraic connectivities $F_1 = 0.005$, $F_2 = 0.015$, $F_3 = 0.1$. To each edge of the graph we assigned a ground-truth rotation corrupted by isotropic Langevin noise, with σ_R varying from 0 to 4 deg. We used EigenRA to compute a set of rotation estimates and the fixed-point iterations defined in Section 3.1 to obtain the global optimum. The stopping criterion was defined as the Riemannian gradient on SO(3)^n having a Frobenius norm smaller than 10^{-7}. This stationary point was then used to compute the Lagrange multiplier estimate $\widehat{\Lambda}$. Finally, we consider an estimate to be the global optimum if the duality gap is smaller than 10^{-10} and $\widehat{\Lambda} - \widetilde{R} \succeq 0$.

The suboptimality of our solution, as measured by the RMSE in degrees, w.r.t the global optimum is plotted in Fig. 4. Regardless of the graph, and for all σ_R we were able to converge towards a globally optimal solution from our estimate. This is in accordance with our claim that our solution lies in the optimal basin of attraction. More importantly however, is the fact that the angular RMSE between our estimate $\widehat{R}$ and the global optimum R^* is considerably small. Moreover, this suboptimality error appears to decrease as the algebraic connectivity of the graph increases. This is in conformity with the core principal of PGO being the redistribution of error amongst the nodes of graph.

4.1.2 Benchmarks

To test our solution with real data, we used 12 real rotation averaging datasets assembled by Wilson et al. [22].

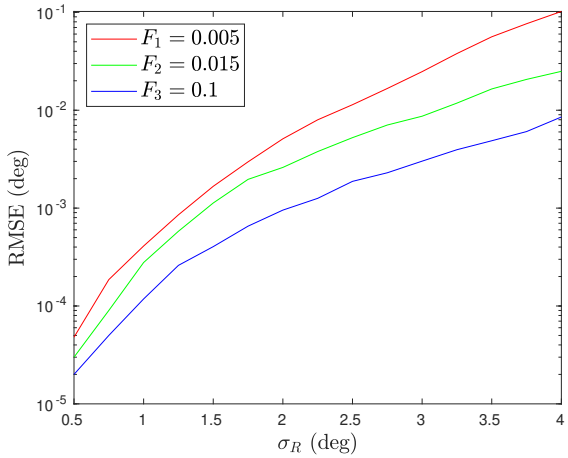

Figure 4: RMSE in degrees, between our estimate $\widehat{R}$ and the global optimum R^* for three different graph configurations with fixed $n = 5750$, different algebraic connectivities F and variable σ_R. Each result was averaged over 10 simulations.

These datasets contain not only noisy observations of the relative rotations but also ground-truth information obtained through Bundle Adjustment.

Due to the large percentage of outlier transformations in the datasets considered, EigenRA is expected to perform poorly. Consequently, we resort to EigenRAO. Our iterative method was benchmarked against a novel deep learning approach proposed by Purkait et al., NeuRoRa [16], which combines two neural networks that suppress outliers and estimate the rotations. A second benchmark was the algorithm devised by Chatterjee et al. [8], based on a Quasi-Newton optimization scheme using a $l^{\frac{1}{2}}$-norm kernel function. For these algorithms, the results we present are those claimed by the authors in the respective papers. Finally, we also compare our results to IRLS EIG-SE(3) by Fusiello et al. [3], for which the author's MATLAB implementation can be found online.

The ground-truth error was computed for three metrics defined in Section 4.1. The results are presented in Table 1. Our solution outperforms that of Chatterjee et al. [8] for nearly all metrics and datasets. The same cannot be said about NeuRoRa which produces results comparable or better than ours in certain datasets. However, EigenRAO fares better overall, especially in terms of the MD error. By comparing EigenRAO's errors with those of the IRLS method by Fusiello et al. [3], we conclude that the latter is surpassed in every dataset and metric. While the eigendecomposition step is the same in both algorithms, our iterative reweighting scheme guarantees the symmetry of the block matrix, uses a different loss function and replaces low-weight measure-

Dataset	Graph		EigenRAO			Chatterjee et al. [8]			Purkait et al. [16]			Fusiello et al. [3]		
	$\|V\|$	$\|E\|$	MN	MD	RMSE	MN	MD	RMSE	MN	MD	RMSE	MN	MD	RMSE
Alamo	627	97296	**2.26**	**0.63**	**6.97**	4.16	1.06	12.68	4.94	1.16	16.09	3.85	1.30	12.09
Ellis Island	247	20297	**1.43**	**0.27**	**5.37**	2.87	0.51	10.36	2.59	0.64	12.82	3.07	0.81	10.50
Yorkminster	458	27729	**2.04**	**0.98**	**5.11**	3.51	1.60	8.41	2.52	0.99	6.55	3.79	1.82	9.37
Montreal Notre Dame	474	52424	**1.10**	**0.28**	6.29	1.54	0.51	7.45	1.23	0.64	**2.67**	1.86	0.62	11.22
Vienna Cathedral	918	103550	**3.4**	**0.85**	10.23	8.29	1.28	27.84	3.91	1.54	**9.93**	8.59	1.59	28.59
Piazza del Popolo	354	24710	3.14	**0.50**	**6.37**	4.06	0.89	8.41	**3.05**	0.79	9.01	3.89	0.97	9.45
Union Square	930	25561	**4.30**	3.31	**7.68**	9.33	3.93	22.44	5.98	**2.01**	17.61	6.92	5.41	13.07
NY Library	376	20680	**1.89**	**0.79**	3.99	3.04	1.35	6.99	1.90	1.18	**2.89**	3.67	2.05	7.78
Notre Dame	553	103932	2.05	**0.55**	8.06	3.53	0.65	14.61	**1.65**	0.68	**6.37**	3.94	1.20	14.85
Roman Forum	1134	70187	3.05	2.62	**5.39**	3.15	1.59	10.21	**2.39**	**1.31**	5.52	26.05	4.56	44.04
Tower of London	508	23863	2.74	1.73	**5.68**	3.94	2.43	9.06	**2.63**	**1.46**	5.78	4.47	2.58	10.56
Madrid Metropolis	394	23784	4.69	**1.09**	11.49	6.97	1.29	17.28	**2.55**	1.13	**6.59**	9.80	4.35	18.69

Table 1: Graph characteristics and comparison between EigenRAO and other rotation averaging solutions.

ments by their estimates. We did not conduct any analysis on CPU time since EigenRAO and IRLS EIG-SE(3) are implemented in MATLAB.

4.2. Motion averaging in SE(3)

To assess the performance of EigenMA in 3D-SLAM we used six datasets compiled by Carlone et al. [6]. Three of them correspond to simulated trajectories (Sphere, Torus3D, Grid3D) and the rest were built from visual data (Garage, Cubicle, Rim). Since ground-truth information is not available for either dataset, we will rely on the likelihood function to compare the different approaches.

For each dataset we computed the global pose estimates in closed-form using EigenMA and EIG-SE(3) [3]. For the latter we used the author's original MATLAB implementation. Additionally, and due to the prominence of the chordal relaxation method [15] in the literature as an initialization for iterative solvers, we benchmarked this technique as well, using a C++ implementation. Since neither of these methods are optimal (locally or globally), we resort to Gauss-Newton (g2o) [14] initialized from our solution to obtain a local maximum of the likelihood function. A maximum of 10 iterations was set for all datasets, despite some of them converging in less than that. The log-likelihood maximum attained and the CPU time required by each method are presented in Table 2. As an example, in Fig. 5 we show the camera trajectory resulting from our optimization of Garage and Grid3D as well as the loop closures.

Since Garage, Grid3D and Cubicle are the datasets with the smallest eigenvalues (in absolute value) we posit that these pose graphs can be accurately optimized using EigenMA. The experiments confirm this, since 10 Gauss-Newton iterations did not increase the value of the log-likelihood by a significant amount, i.e. our solution is quasi-optimal.

The only datasets for which there is a considerable difference between the local maximum and our solution are

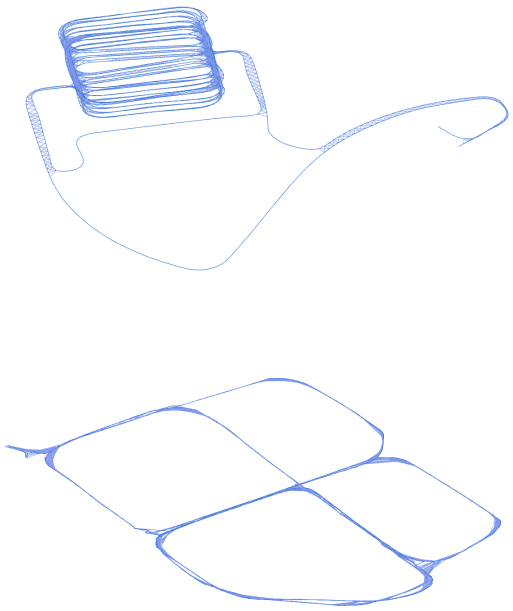

Figure 5: Camera trajectories estimated by EigenMA and loop closures. Top: Parking garage (n=1661). Bottom: Cubicle (n=5750).

Sphere and Rim. This difference is common to all three of the state-of-the-art suboptimal methods considered, ours attaining the highest objective amongst them. The Sphere case is particularly remarkable. We hypothesize that the lackluster performance of EigenMA is due to the configuration of the pose graph itself. In fact, since it simulates the successive poses of a robot travelling on a spherical surface, the relative rotations between equidistant positions are

Dataset	Graph $\lvert V\rvert$	$\lvert E\rvert$	λ_1	EigenMA $\widehat{f}_{\text{ML}}$	EigenMA $t_{\text{CPU}}(s)$	Chordal [15] $\widehat{f}_{\text{ML}}$	Chordal [15] $t_{\text{CPU}}(s)$	EIG-SE3 [3] $\widehat{f}_{\text{ML}}$	EIG-SE3 [3] $t_{\text{CPU}}(s)$	g2o [14] $\widehat{f}_{\text{ML}}$	g2o [14] $t_{\text{CPU}}(s)$
Parking garage	1661	6275	4.2e-7	**1.88e4**	**0.03**	**1.88e4**	0.28	1.41e4	1.78	**1.88e4**	0.22
Grid3D	8000	22236	8.7e-3	**6.64e4**	**0.83**	**6.64e4**	13.46	6.58e4	2.11	**6.64e4**	492.17
Cubicle	5750	16869	9.0e-6	**5.06e4**	**0.23**	**5.06e4**	1.53	3.74e4	2.12	**5.06e4**	3.65
Sphere	2200	8647	2.2e-1	1.66e4	**0.16**	1.33e4	1.04	-5.17e5	0.43	**2.47e4**	3.04
Torus3D	5000	9048	3.9e-3	**2.71e4**	**0.20**	2.70e4	1.19	2.69e4	4.04	**2.71e4**	7.01
Rim	10195	29743	1.7e-5	8.91e4	**0.46**	8.89e4	2.71	6.64e4	7.81	**8.92e4**	10.69

Table 2: Graph characteristics and comparison between EigenMA, Chordal relaxation, EIG-SE(3) and g2o (10 iterations)

nearly constant and this may hinder optimization techniques relying on eigenspaces. Furthermore, as listed in Table 2, the smallest eigenvalue for the Sphere dataset is the largest amongst the six datasets considered. Therefore, one may consider this pose graph to be outside the range of applicability of our method.

When comparing EigenMA to EIG-SE(3) one notices that, except for the Garage dataset, the latter produces poorer results. This is evident by looking at the maximum likelihood attained. The performance difference stems from the high percentage of missing data in all of the six datasets considered, combined with the fact that EIG-SE(3) performs a single eigendecomposition to obtain four eigenvectors which are then projected to $\text{SE}(3)^n$. Consequently, its translation estimates are not optimal when considering the rotation estimates produced. Moreover, the spectral decomposition in this algorithm is computed for a non-symmetric matrix, unlike our approach.

The chordal initialization is the only suboptimal technique that produces results comparable to ours. However, we attain higher objectives in Cubicle, Sphere, Torus3D and Rim and an equal objective in Grid3D. Furthermore, the computation of the eigenvalues in EigenRA, which is done *by default* within the Krylov-Schur algorithm, allows for an assessment on the consistency of the measurements. A similar procedure in the Chordal method would translate to a higher CPU time.

In terms of CPU time, our compiled C++ version of EigenMA outperforms every method currently in existence. EigenMA fares better than g2o, even if the CPU time for the latter is dependent upon the number of iterations. As an example, the optimization carried out on Grid3D using our algorithm yields approximately the same result as g2o, but it is nearly 600 times faster. Furthermore, the CPU time we have indicated for this algorithm does not take into account the initialization which dictates how well it can perform. EIG-SE(3) is also slower than EigenMA, which can be explained by the fact that it performs eigendecomposition of non-symmetric matrices. Finally, the Chordal relaxation (implemented in C++) also lags behind EigenMA by a considerable amount.

Parameter	Value
Depth cutoff (m)	2.0
SIFT specifications	Default
RANSAC max. number of iterations	7.0e+03
RANSAC RMSE threshold (m)	3.0e-06
RANSAC min. number of matches	50

Table 3: RGB-D registration pipeline parameters.

4.3. Dense 3D reconstruction

We now justify the applicability of our PGO solution to the problem of registering multiple point clouds obtained from RGB-D cameras. We built a RGB-D registration pipeline that estimates relative rigid transformations by RANSAC-filtering 2D image matches computed by SIFT. To ensure that Hypothesis 1 is verified, the RMSE threshold used in RANSAC is kept low. In the event of this technique failing to produce a rigid transformation, our pipeline resorts to ICP in order to do so. The set of transformations deemed accurate are used to build a connected graph which is then optimized using EigenMA. A MATLAB implementation is available online.

We tested our pipeline using four RGB-D datasets: Burghers, Stonewall, Sportscar and Lounge from a collection by Zhou et al. [23]. Due to the large overlap between adjacent frames in the first two datasets, we registered them with a stride of 10 images. The last two were registered in their entirety. Notwithstanding, all the objects in the four scenes were fully reconstructed. Table 3 shows the pipeline parameters used to build the pose graphs.

The reconstructed dense 3D scenes after downsampling are shown in Figs. 6 7, 8 and 9. The optimization was carried out by EigenMA which took 0.02 seconds for the smallest dataset (Stonewall with 271 poses) and 0.16 seconds for the largest (Sportscar with 6553 poses). With such CPU times, EigenMA lends itself to real-time PGO applications, namely 3D-SLAM and online reconstruction.

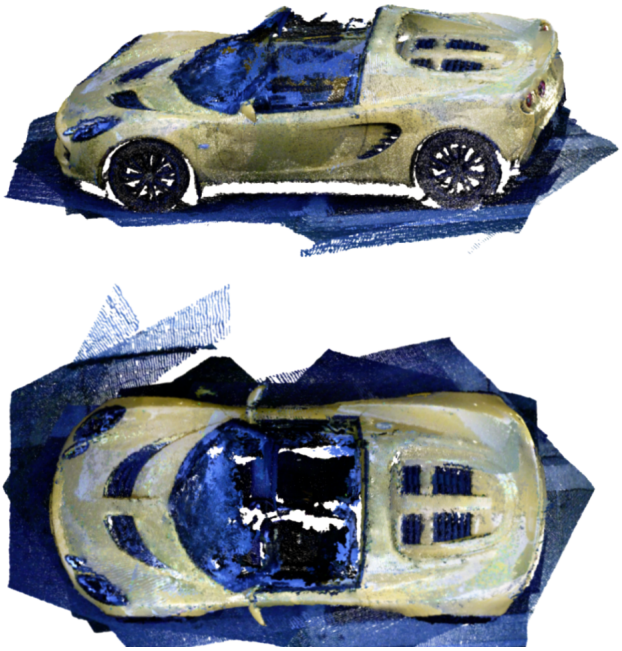

Figure 6: Reconstruction of the Sportscar dataset [23] with 6553 poses. PGO computed by EigenMA in 0.16s.

Figure 7: Reconstruction of the Stonewall dataset [23] with 271 poses. PGO computed by EigenMA in 0.02s.

Figure 8: Reconstruction of the Burghers dataset [23] with 1124 poses. PGO computed by EigenMA in 0.04s.

Figure 9: Reconstruction of the Lounge dataset [23] with 3000 poses. PGO computed by EigenMA in 0.15s.

5. Conclusions

On the one hand, MLE is capable of modeling the problem of PGO with a high degree of accuracy. Nonetheless, the optimization strategies involved are often too cumbersome and require good initializations in order to attain the global optimum. On the other hand, efficient relaxations such as EIG-SE(3), which akin to our approach, is based on spectral decomposition, often fail to achieve the same degree of accuracy as MLE. Our solution to the otherwise difficult task of PGO, when applicable, outpeforms the state-of-the-art in efficiency without compromising precision.

The empirical results we presented validate the claims we made throughout this paper and allow us to assert the following. Under Langevin noise and assuming a standard deviation no greater than 4 degrees, a quasi-optimal solution for the problem of multiple rotation averaging can be computed in closed-form using EigenRA. Our iterative method to solve multiple rotation averaging with outliers, EigenRAO, proved to be more accurate than the state-of-the-art, when tested in Bundle Adjustment datasets. For PGO in SE(3), our approach based on the Krylov-Schur method for spectral decomposition and Cholesky factorization, is faster than all the other methods. This increased performance does not compromise the accuracy of the solution. Finally, we demonstrated the applicability of our algorithm in motion estimation and dense 3D reconstruction from RGB-D data.

References

[1] Damped Newton algorithms for matrix factorization with missing data. *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, II(3):316–322, 2005. 2

[2] F. Arrigoni, B. Rossi, P. Fragneto, and A. Fusiello. Robust synchronization in SO(3) and SE(3) via low-rank and sparse matrix decomposition. *Computer Vision and Image Understanding*, 174(3):95–113, 2018. 2

[3] F. Arrigoni, B. Rossi, and A. Fusiello. Spectral Synchronization of Multiple Views in SE(3). *SIAM Journal on Imaging Sciences*, 9:1963–1990, 01 2016. 2, 3, 6, 7, 8

[4] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. 1

[5] N. Boumal, A. Singer, P. A. Absil, and V. D. Blondel. Cramér-Rao bounds for synchronization of rotations. *Information and Inference*, 3(1):1–39, 2014. 3

[6] L. Carlone, D. M. Rosen, G. Calafiore, J. J. Leonard, and F. Dellaert. Lagrangian duality in 3D SLAM: Verification techniques and optimal solutions. *IEEE International Conference on Intelligent Robots and Systems*, 2015-Decem:125–132, 2015. 2, 7

[7] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert. Initialization techniques for 3D SLAM: A survey on rotation estimation and its use in pose graph optimization. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4597–4604, 2015. 2

[8] A. Chatterjee and V. M. Govindu. Robust Relative Rotation Averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):958–972, 2018. 2, 6, 7

[9] F. Crosilla and A. Beinat. Use of generalised Procrustes analysis for the photogrammetric block adjustment by independent models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56(3):195–209, 2002. 1

[10] F. Dellaert. Factor graphs and GTSAM: A hands-on introduction. Technical report, Georgia Institute of Technology, 2012. 2

[11] A. Eriksson, C. Olsson, F. Kahl, and T. J. Chin. Rotation Averaging and Strong Duality. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 127–135, 2018. 2, 4

[12] R. F. Guerreiro and P. M. Aguiar. Factorization with missing data for 3D structure recovery. *Proceedings of 2002 IEEE Workshop on Multimedia Signal Processing, MMSP 2002*, pages 105–108, 2002. 2

[13] R. Hartley, J. Trumpf, Y. Dai, and H. Li. Rotation averaging. *International Journal of Computer Vision*, 103(3):267–305, 2013. 2, 6

[14] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. g2o: A general framework for graph optimization. In *2011 IEEE International Conference on Robotics and Automation*, pages 3607–3613, 2011. 2, 7, 8

[15] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007. 2, 7, 8

[16] P. Purkait, T.-J. Chin, and I. Reid. NeuRoRA: Neural robust rotation averaging. *ArXiv*, abs/1912.04485, 2019. 2, 6, 7

[17] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard. SE-Sync: A certifiably correct algorithm for synchronization over the special Euclidean group. *International Journal of Robotics Research*, 38(2-3):95–125, 2019. 2

[18] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. *Proceedings of International Conference on 3-D Digital Imaging and Modeling, 3DIM*, pages 145–152, 2001. 1

[19] G. W. Stewart. A Krylov–Schur algorithm for large eigenproblems. *SIAM Journal on Matrix Analysis and Applications*, 23(3):601–614, 2002. 2, 4

[20] R. Toldo, A. Beinat, and F. Crosilla. Global registration of multiple point clouds embedding the Generalized Procrustes Analysis into an ICP framework. *3Dpvt 2010*, (January):8, 2010. 1

[21] R. Tron and R. Vidal. Distributed image-based 3-D localization of camera sensor networks. *Proceedings of the IEEE Conference on Decision and Control*, pages 901–908, 2009. 2

[22] K. Wilson and N. Snavely. Robust Global Translations with 1DSfM. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 6

[23] Q. Zhou and V. Koltun. Dense Scene Reconstruction with Points of Interest. *ACM Transactions on Graphics*, 32(4), 2013. 8, 9