Convex Relaxations for Pose Graph Optimization With Outliers

Luca Carlone, *Member, IEEE*, and Giuseppe C. Calafiore, *Fellow, IEEE*

***Abstract*— *Pose graph optimization* consists in the estimation of a set of poses from pairwise measurements and provides a formalization for many problems arising in mobile robotics and geometric computer vision. In this letter, we consider two-dimensional pose estimation problems in which a subset of the measurements is spurious. Our first contribution is to develop robust estimators that can cope with heavy-tailed measurement noise, hence increasing robustness to the presence of outliers. Since the resulting estimators require solving nonconvex optimization problems, we further develop convex relaxations that approximately solve those problems via semidefinite programming. We then provide conditions for testing the exactness of the proposed relaxations. Contrary to existing approaches, our convex relaxations do not rely on the availability of an initial guess for the unknown poses, hence they are more suitable for setups in which such guess is not available (e.g., multirobot localization, recovery after localization failure). We tested the proposed techniques in extensive simulations, and we show that some of the proposed relaxations are indeed *tight* (i.e., they solve the original nonconvex problem exactly) and ensure accurate estimation in the face of a large number of outliers.**

***Index Terms*—SLAM, localization, sensor fusion, optimization and optimal control, multi-robot systems.**

I. Introduction

POSE Graph Optimization (PGO) consists in the estimation of a set of poses (i.e., rotations and translations) from pairwise relative pose measurements. This model often arises from a *maximum likelihood* (ML) approach to geometric estimation problems in robotics and computer vision. For instance, in the context of robot *Simultaneous Localization and Mapping* (SLAM), PGO is used to estimate the trajectory of the robot (as a discrete set of poses), which in turn allows the reconstruction of a map of the environment, see, e.g., [3]. Another example is multi robot localization, in which multiple robots estimate their poses from pairwise measurements, see [1]. Similarly, a variant of PGO arises in *Structure from Motion* (SfM) in computer vision, as a workhorse for estimating the poses (up to scale) of the cameras observing a 3D scene.

Manuscript received September 8, 2017; accepted January 3, 2018. Date of publication January 15, 2018; date of current version February 8, 2018. This letter was recommended for publication by Associate Editor U. Frese and Editor C. Stachniss upon evaluation of the reviewers' comments. This work was supported by the Army Research Laboratory's DCIST Program. *(Corresponding author: Luca Carlone.)*

L. Carlone is with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: lcarlone@mit.edu).

G. C. Calafiore is with the Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino, Torino 10129, Italy, and also with the Institute of Electronics, Computer and Telecommunication Engineering (IEIIT-CNR), Torino 10129, Italy (e-mail: giuseppe.calafiore@polito.it).

Digital Object Identifier 10.1109/LRA.2018.2793352

PGO leads to a hard non-convex optimization problem, whose (global) solution is the ML estimate for the unknown poses. While a standard approach to solve PGO was to use iterative nonlinear optimization methods, such as the Gauss-Newton method [18], [20] or the gradient method [14], [25], to obtain locally optimal solutions, a very recent set of works shows how to compute *globally* optimal solutions to PGO via convex relaxations [5], [8], [9], [28]. These works demonstrate that in the noise regimes encountered in practical applications, the (nonconvex) PGO problem exhibits *zero duality gap*, which implies that it can be solved exactly via convex relaxations.

An outstanding problem in PGO is how to make the pose estimation robust to the presence of spurious measurements. Standard PGO assum es a nominal distribution for the measurement noise (e.g., Gaussian noise on translation measurements), and it produces largely incorrect estimates in presence of *outliers*, i.e., measurements that move away from this nominal distribution. This issue limits robust operation in practical applications in which the presence of outliers is unavoidable. Outliers may be due to sensor failures, but they are more commonly caused by incorrect *data association*, see [3].

Related Work: We partition the existing literature into *outlier mitigation* and *outlier rejection* techniques. The former techniques estimate the poses while trying to reduce the influence of the outliers. The latter explicitly include binary decision variables to establish whether a given measurement is an outlier or not. Traditionally, outlier mitigation in SLAM and SfM relied on the use of robust M-estimators, see [17]. For instance, the use of the Huber loss is a fairly standard choice in SfM, see [16]. Along this line, Agarwal *et al.*, in [26], dynamically adjust the measurement covariances to reduce the influence of measurements with large errors. Olson and Agarwal, in [24], use a max-mixture distribution to accommodate multiple hypotheses on the noise distribution of a measurement. Casafranca *et al.*, in [10], propose to minimize the ℓ_1-norm of the residual errors and design an iterative scheme to locally solve the corresponding optimization. Lee *et al.*, in [22], use expectation maximization. Pfingsthorn and Birk [27] model ambiguous measurements using hyperedges and multimodal mixture of Gaussian constraints that augment the pose graph.

Outlier rejection techniques aim at explicitly identifying the spurious measurements. A popular technique is RANSAC, see [11], in which subsets of the measurements are sampled in order to identify an outlier-free set. Sünderhauf and Protzel, in [30], [31], propose to augment the PGO problem with latent binary variables that are responsible for deactivating outliers. Similar ideas appear in the context of robust estimation, e.g., in the *Penalized Trimmed Squares* estimator of Zioutas and Avramidis, [34]. Latif *et al.* in [21] and Graham *et al.* in [12] look for "internally consistent" constraints, which are in mutual

agreement. Carlone *et al.* in [7] use ℓ_1-relaxation to find a large set of mutually-consistent measurements.

A main drawback of the techniques mentioned above is that they rely on the availability of an initial estimate of the poses. This is problematic for two reasons. First, the performance of these techniques heavily depends on the initial guess and they perform poorly when the initial guess is noisy, as it happens in practice. Second, in many applications, the initial guess is simply not available (e.g., in multi robot localization). Recent work in computer vision attempts to overcome the need for an initial guess. In particular, Wang and Singer in [33] provide a convex relaxation for robust rotation estimation. In this letter, we extend [33] to work on poses (rather than rotations), and consider a broader set of robust cost functions.

Contribution: Our first contribution is to propose robust PGO formulations that can cope with heavy-tailed measurement noise. We consider cost functions with unsquared ℓ_2 norm, ℓ_1 norm, and Huber loss, and, when possible, we provide a probabilistic justification in terms of ML estimation.

Since the resulting optimization problems are nonconvex, our second contribution is to provide a systematic way to derive convex relaxations for those problems, taking the form of semidefinite programs (SDP). The key advantage of our convex relaxations is that they do not require an initial guess, which makes them suitable for problem instances where such estimate is not available or unreliable.

As a third contribution, we provide conditions under which the proposed relaxations are exact, as well as general bounds on the suboptimality of the relaxed solution. Similarly to related works such as [5], [28], these conditions involve the rank of a matrix appearing in the semidefinite relaxation.

Finally, we test the performance of the proposed relaxations in extensive Monte Carlo simulations. The experimental results show that a subset of the convex relaxations discussed in this letter are indeed tight, and work extremely well in simulated problem instances, showing insensitivity to a large amount of outliers, and outperforming related techniques. The paper is tailored to 2D PGO problems. However, the approach and the theoretical guarantees can be extended to the 3D setup. Extra visualizations are given in the supplemental material [4].

Notation: We use lower and upper case bold letters to denote vectors ($\boldsymbol{v}$) and matrices ($\boldsymbol{M}$), respectively. Non-bold face letters are used for scalars (j) and function names ($f(\cdot)$). The identity matrix of size n is denoted with $\mathbf{I}_n$. An $m \times n$ zero matrix is denoted by $\mathbf{0}_{m\times n}$. The symbols $\|\cdot\|_1$, $\|\cdot\|_2$, and $\|\cdot\|_F$ denote the ℓ_1, ℓ_2, and Frobenious norm, respectively. Given a random variable x, we use the notation $\mathcal{D}(x;\beta_1,\beta_2,\ldots)$ to denote the probability density of x with parameters $\beta_1,\beta_2,\ldots$, and we also write $x \sim \mathcal{D}(\beta_1,\beta_2,\ldots)$.

II. Pose Graph Optimization

PGO estimates n poses from m relative pose measurements. We consider a planar setup, in which each to-be-estimated pose $\boldsymbol{x}_i \doteq (\boldsymbol{R}_i, \boldsymbol{t}_i), i = 1,\ldots,n$, comprises a *translation* vector $\boldsymbol{t}_i \in \mathbb{R}^2$ and a rotation matrix $\boldsymbol{R}_i \in \mathrm{SO}(2)$. For a pair of poses (i,j), a relative pose measurement $\bar{\boldsymbol{x}}_{ij} \doteq (\bar{\boldsymbol{R}}_{ij}, \bar{\boldsymbol{t}}_{ij})$, with $\bar{\boldsymbol{t}}_{ij} \in \mathbb{R}^2$ and $\bar{\boldsymbol{R}}_{ij} \in \mathrm{SO}(2)$, describes a noisy measurement of the relative pose between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Each measurement is assumed to be sampled from the following *generative model*:

$$\bar{\boldsymbol{t}}_{ij} = \boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j - \boldsymbol{t}_i) + \boldsymbol{t}_{ij}^{\epsilon}, \qquad \bar{\boldsymbol{R}}_{ij} = \boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j\boldsymbol{R}_{ij}^{\epsilon} \tag{1}$$

where $\boldsymbol{t}_{ij}^{\epsilon} \in \mathbb{R}^2$ and $\boldsymbol{R}_{ij}^{\epsilon} \in \mathrm{SO}(2)$ represent translation and rotation measurement noise, respectively.

The problem can be modeled through graph formalism: each to-be-estimated pose is associated to a vertex (or node) in the vertex set $\mathcal{V}$ of a directed graph, while each measurement is associated to an edge in the edge set $\mathcal{E}$ of the graph. We denote by $n \doteq |\mathcal{V}|$ the number of nodes and by $m \doteq |\mathcal{E}|$ the number of edges. The resulting graph, namely $\mathcal{G}(\mathcal{V},\mathcal{E})$, is usually referred to as a *pose graph*.

A. Standard PGO

Given the generative model (1), standard approaches to PGO formulate the pose estimation problem in terms of ML estimation: the best pose estimate is the one maximizing the likelihood of the available measurements $\bar{\boldsymbol{x}}_{ij}, \forall (i,j) \in \mathcal{E}$. Since the expression of the measurement likelihood depends on the assumed distribution of the measurement noise, an ubiquitous assumption in this endeavour is that the translation noise $\boldsymbol{t}_{ij}^{\epsilon}$ is distributed according to a zero-mean Normal distribution with given covariance $\boldsymbol{\Sigma}_{ij}^{t}$, i.e., $\boldsymbol{t}_{ij}^{\epsilon} \sim \mathrm{Normal}(\mathbf{0}, \boldsymbol{\Sigma}_{ij}^{t})$. For the rotation noise, it has been recently proposed to use the Von Mises distribution as noise model, as this leads to simpler estimators and it is easier to analyze, see [5]. In particular, it is assumed that $\boldsymbol{R}_{ij}^{\epsilon}$ in (1) represents a rotation of a random angle θ^{ϵ}, distributed according to the Von Mises distribution:

$$\mathrm{VonMises}(\theta^{\epsilon};\mu,\kappa) = c(\kappa)\exp(\kappa\cos(\theta^{\epsilon}-\mu)) \tag{2}$$

with mean μ, concentration parameter κ, and where $c(\kappa)$ is a normalization constant that is irrelevant for the purpose of ML estimation. These choices of the noise model lead to the following standard PGO formulation:

Proposition 1 (Standard PGO Formulation): Given the generative model (1) and assuming that the translation noise is $\boldsymbol{t}_{ij}^{\epsilon} \sim \mathrm{Normal}(\mathbf{0}, \frac{1}{\omega_{ij}^{t}}\mathbf{I}_2)$ and that the rotation noise is $\boldsymbol{R}_{ij}^{\epsilon} \sim \mathrm{VonMises}(\mathbf{0}, \omega_{ij}^{R})$, the maximum likelihood estimate of the poses is the solution of the following minimization:

$$\min_{\substack{\boldsymbol{t}_i\in\mathbb{R}^2 \\ \boldsymbol{R}_i\in\mathrm{SO}(2)}} \sum_{(i,j)\in\mathcal{E}} \omega_{ij}^{t}\|\boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j-\boldsymbol{t}_i)-\bar{\boldsymbol{t}}_{ij}\|_2^2 + \frac{\omega_{ij}^{R}}{2}\left\|\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}\right\|_F^2. \tag{3}$$

A proof of this statement is given in [5, Proposition 1]. The estimation in (3) involves a nonconvex optimization problem, due to the nonconvexity of the set $\mathrm{SO}(2)$. Surprisingly, recent results [5], [28] show that one can still compute a globally optimal solution to (3), whenever the measurement noise is reasonable, using convex relaxations.

III. Robust PGO Formulations

The ML estimator in Proposition 1 is known to perform arbitrarily bad in the presence of even a single outlier. This is due to the quadratic nature of the cost, for which measurements with large residual errors dominate the other terms. This in turn depends on the fact that the estimator assumes a light-tailed noise distribution, i.e., the Normal distribution. Therefore, a natural way to gain robustness is to change the noise assumptions to take into account the presence of measurements with larger errors, and use noise distributions with "heavier-than-normal tails," see, e.g., [19]. In the rest of this section we consider three alternative noise models and their induced cost functions. Since the corresponding optimization problems remain nonconvex,

we discuss how to relax these formulation to convex programs in Sections IV and V.

A. Least Unsquared Deviation: ℓ_2-Norm Cost

We next introduce two distributions: the *multivariate exponential power distribution*, which we later use to model the translation noise, and the *directional Laplace distribution*, which we use to model noise in $\mathrm{SO}(2)$.

Definition 2 (Multivariate Exponential Power Distribution): For a random variable $\boldsymbol{x} \in \mathbb{R}^d$, the *multivariate exponential power distribution* with parameters $\boldsymbol{\mu} \in \mathbb{R}^n$, $\boldsymbol{\Sigma} \succ 0$, and $\beta > 0$ is defined as:

$$\mathrm{ExpPow}(\boldsymbol{x}; \mu, \Sigma, \beta) = c \exp\left(-\tfrac{1}{2}[(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})]^{\beta} \right) \tag{4}$$

where c is a normalization constant, independent of $\boldsymbol{x}$.

Definition 3 (Directional Laplace Distribution): For a random variable $\theta \in (-\pi, +\pi]$, the Directional Laplace Distribution with mean $\mu \in (-\pi, +\pi]$ and scale parameter $\kappa > 0$ is defined as:

$$\mathrm{DLaplace}(\theta; \mu, \kappa) = c \exp\left(-\kappa \left| \sin\left(\frac{\theta-\mu}{2} \right) \right| \right) \tag{5}$$

where c is a normalization constant, independent of θ.

Definition 3 is a slight variation of the definition given in [23], in that it considers $(-\pi, +\pi]$ as the angular domain, rather than $[0, +\pi)$. Using Definitions 2–3, we can now introduce our first robust estimator.

Proposition 4 (Least Unsquared Deviation Estimator): Given the generative model (1) and assuming that the translation noise is $\boldsymbol{t}_{ij}^{\epsilon} \sim \mathrm{ExpPow}(\boldsymbol{0}, \frac{1}{(w_{ij}^t)^2}\mathbf{I}_2, \frac{1}{2})$ and the rotation noise is $\boldsymbol{R}_{ij}^{\epsilon} \sim \mathrm{DLaplace}(\boldsymbol{0}, w_{ij}^R)$, then the ML estimate of the poses is the solution of the following minimization problem:

$$\min_{\substack{\boldsymbol{t}_i \in \mathbb{R}^2 \\ \boldsymbol{R}_i \in \mathrm{SO}(2)}} \sum_{(i,j)\in\mathcal{E}} w_{ij}^t \|\boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j - \boldsymbol{t}_i) - \bar{\boldsymbol{t}}_{ij}\|_2 + \frac{w_{ij}^R}{\sqrt{2}} \|\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}\|_F . \tag{6}$$

The derivation of the ML estimator of Proposition 4 is straightforward, and proceeds by inspection, starting from the noise distributions (4) and (5), and using the relation

$$\|\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}\|_F = 2\sqrt{2}|\sin(\bar{\theta}_{ij} - \theta_j + \theta_i)|, \tag{7}$$

where $\bar{\theta}_{ij}$, θ_j, and θ_i are the rotation angles corresponding to $\bar{\boldsymbol{R}}_{ij}$, $\boldsymbol{R}_j$, and $\boldsymbol{R}_i$, respectively.

The estimator (6) is similar to the standard PGO estimator (3), except for the fact that the ℓ_2-norm and the Frobenius norm are unsquared. The intuition behind the use of (6) is that, since the cost is no longer quadratic, it does not favor measurements with large residuals.

B. Least Absolute Deviation: ℓ_1-Norm Cost

In this section we consider a formulation in which the cost includes the ℓ_1-norm rather than the Euclidean norm.

Definition 5 (Least Absolute Deviation Estimator): The Least Absolute Deviation Estimator of the poses in the pose graph is the solution of the following minimization problem:

$$\min_{\substack{\boldsymbol{t}_i \in \mathbb{R}^2 \\ \boldsymbol{R}_i \in \mathrm{SO}(2)}} \sum_{(i,j)\in\mathcal{E}} w_{ij}^t \|\boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j - \boldsymbol{t}_i) - \bar{\boldsymbol{t}}_{ij}\|_1 + \frac{w_{ij}^R}{2} \|\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}\|_1 . \tag{8}$$

The formulation in Definition 5 adopts an ℓ_1-norm penalty, which is known to be less sensitive to outliers, see, e.g., [29, p. 10], hence we expect this formulation to perform better in presence of spurious measurements. However, contrarily to Proposition 4, we currently only have a partial probabilistic justification for the cost (8). More precisely, while it is known that the first term in the cost (8) follows from the assumption that the translation noise is distributed according to a Laplace distribution, see [29, p. 10], the choice of the second term ($\|\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}\|_1$) is currently arbitrary, and only justified by the symmetry w.r.t. the first term.

C. Huber Loss

In this section we consider a popular M-estimator, based on the Huber loss function. While this is a commonly used robust estimator in SLAM and SfM, our novel contribution is to provide a convex relaxation, which is discussed in Section IV.

Definition 6 (Huber Estimator): The Huber Estimator of the poses in the pose graph is the minimizer of the following optimization problem:

$$\min_{\substack{\boldsymbol{t}_i \in \mathbb{R}^2 \\ \boldsymbol{R}_i \in \mathrm{SO}(2)}} \sum_{(i,j)\in\mathcal{E}} \mathrm{h}(w_{ij}^t \|\boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j - \boldsymbol{t}_i) - \bar{\boldsymbol{t}}_{ij}\|_2) + \mathrm{h}(w_{ij}^R \|\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}\|_F) \tag{9}$$

where $\mathrm{h}(\cdot)$ is the Huber loss function, defined as:

$$\mathrm{h}(x) = \begin{cases} |x|^2 & |x| \leq 1 \\ 2|x| - 1 & \text{otherwise.} \end{cases} \tag{10}$$

The Huber loss in (10) is a quadratic function when the argument belongs to the interval $[-1, +1]$, while it is linear otherwise. Ideally, the inliers should fall in the quadratic region, where the Huber estimator behaves as the least squares estimator of Proposition 1; on the other hand, the outliers should ideally fall in the linear region, in which the Huber loss behaves as the least unsquared deviation estimator of Proposition 4. We note that while an alternative definition of the Huber loss, e.g., [16, p. 619], includes an extra parameter δ that defines the size of the quadratic region ($[-\delta, +\delta]$), we are implicitly using the terms w_{ij}^t and w_{ij}^R to define the region in which the cost is quadratic. For instance, in the first term of (9), the Huber loss becomes quadratic when:

$$w_{ij}^t \|\boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j - \boldsymbol{t}_i) - \bar{\boldsymbol{t}}_{ij}\|_2 \leq 1 \Leftrightarrow \|\boldsymbol{R}_i^{\mathsf{T}}(\boldsymbol{t}_j - \boldsymbol{t}_i) - \bar{\boldsymbol{t}}_{ij}\|_2 \leq \frac{1}{w_{ij}^t} \tag{11}$$

i.e., the terms w_{ij}^t and w_{ij}^R define the boundaries between the quadratic and the linear behavior.

IV. Convex Relaxations and Rounding Procedure

In this section we discuss a systematic approach for deriving convex relaxations for the problems (6), (8), (9). We refer to the approaches presented in this section as *1-stage techniques* since they require the solution of a single convex program. In Section V, instead, we propose an alternative way to solve

problems (6), (8), (9), by decoupling rotation and translation estimation. We refer to the latter as *2-stage techniques*.

A. Convex Relaxations of Robust PGO Formulations

We consider each formulation in Section III.

1) Relaxation of the ℓ_2-Norm Formulation: Problem (6) is nonconvex and therefore hard to solve globally. The source of nonconvexity is the constraint $\boldsymbol{R}_i \in \mathrm{SO}(2)$, while the cost can be made convex (in fact, quadratic) by rearranging the rotations $\boldsymbol{R}_i$ and leveraging the invariance to rotation of the ℓ_2 norm. While in our previous work [5] we used a quadratic reformulation of the cost, in the following we propose a more direct relaxation approach.

The first step towards our convex relaxation is to reparametrize Problem (6). We rearrange the unknown poses $(\boldsymbol{R}_i, \boldsymbol{t}_i)$, with $i = 1, \ldots, n$, into a single matrix $\boldsymbol{Z}$:

$$\boldsymbol{Z} \doteq [\boldsymbol{R}_1 \ \ldots \ \boldsymbol{R}_n \ \boldsymbol{t}_1 \ \ldots \ \boldsymbol{t}_n] \in \mathbb{R}^{2\times 3n} \tag{12}$$

Moreover we define the following square matrix:

$$\boldsymbol{X} \doteq \boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z} = \left[\begin{array}{ccc|ccc} \boldsymbol{R}_1^{\mathsf{T}}\boldsymbol{R}_1 & \ldots & \boldsymbol{R}_1^{\mathsf{T}}\boldsymbol{R}_n & \boldsymbol{R}_1^{\mathsf{T}}\boldsymbol{t}_1 & \ldots & \boldsymbol{R}_1^{\mathsf{T}}\boldsymbol{t}_n \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{R}_n^{\mathsf{T}}\boldsymbol{R}_1 & \ldots & \boldsymbol{R}_n^{\mathsf{T}}\boldsymbol{R}_n & \boldsymbol{R}_n^{\mathsf{T}}\boldsymbol{t}_1 & \ldots & \boldsymbol{R}_n^{\mathsf{T}}\boldsymbol{t}_n \\ \hline \boldsymbol{t}_1^{\mathsf{T}}\boldsymbol{R}_1 & \ldots & \boldsymbol{t}_1^{\mathsf{T}}\boldsymbol{R}_n & \boldsymbol{t}_1^{\mathsf{T}}\boldsymbol{t}_1 & \ldots & \boldsymbol{t}_1^{\mathsf{T}}\boldsymbol{t}_n \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{t}_n^{\mathsf{T}}\boldsymbol{R}_1 & \ldots & \boldsymbol{t}_n^{\mathsf{T}}\boldsymbol{R}_n & \boldsymbol{t}_n^{\mathsf{T}}\boldsymbol{t}_1 & \ldots & \boldsymbol{t}_n^{\mathsf{T}}\boldsymbol{t}_n \end{array}\right] \doteq \left[\begin{array}{c|c} \boldsymbol{X}^{RR} & \boldsymbol{X}^{Rt} \\ \hline \boldsymbol{X}^{tR} & \boldsymbol{X}^{tt} \end{array}\right] \in \mathbb{R}^{3n\times 3n} \tag{13}$$

where we distinguished 4 block matrices within $\boldsymbol{X}$, with $\boldsymbol{X}^{RR} \in \mathbb{R}^{2n\times 2n}$, $\boldsymbol{X}^{tt} \in \mathbb{R}^{n\times n}$, and $\boldsymbol{X}^{Rt} = (\boldsymbol{X}^{tR})^{\mathsf{T}} \in \mathbb{R}^{2n\times n}$. Problem (6) can be written as function of $\boldsymbol{Z}$ and $\boldsymbol{X}$:

$$\begin{aligned} \min_{\boldsymbol{Z},\boldsymbol{X}} \ & \sum_{(i,j)\in\mathcal{E}} w_{ij}^t \|[\boldsymbol{X}]_{ij}^{Rt} - [\boldsymbol{X}]_{ii}^{Rt} - \bar{\boldsymbol{t}}_{ij}\|_2 + \frac{w_{ij}^R}{\sqrt{2}} \|[\boldsymbol{X}]_{ij}^{RR} - \bar{\boldsymbol{R}}_{ij}\|_F \\ \text{s.t.} \quad & \boldsymbol{X} = \boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z}, \ [\boldsymbol{X}]_{ii}^{RR} = \mathbf{I}_2, \ \det[\boldsymbol{Z}]_i = +1, \\ & i = 1, \ldots, n, \end{aligned} \tag{14}$$

where we used the notation $[\boldsymbol{X}]_{ij}^{RR}$ to identify the 2×2 block entry of $\boldsymbol{X}^{RR}$ at row i and column j, and $[\boldsymbol{X}]_{ij}^{Rt}$ to identify the 2×1 block entry of $\boldsymbol{X}^{Rt}$, according to the partition shown in (13). The optimization problem imposes the constraint that $\boldsymbol{X}$ can be computed from $\boldsymbol{Z}$, since $\boldsymbol{X} = \boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z}$, while the constraints $[\boldsymbol{X}]_{ii}^{RR} = \boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_i = \mathbf{I}_2$ and $\det[\boldsymbol{Z}]_i = +1$ enforce that the estimated rotation matrices are orthogonal.[1] In the following, we drop the determinant constraint for simplicity, and perform estimation over the *Orthogonal group* rather than the Special Orthogonal group $\mathrm{SO}(2)$. Empirical evidence from this and previous works, such as [9], shows that dropping the determinant constraint does not impact the relaxation.

Now we note that the constraint $\boldsymbol{X} = \boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z}$ is equivalent to (i) $\boldsymbol{X} \succeq 0$, and (ii) $\mathrm{rank}\,(\boldsymbol{X}) = 2$, hence we can rewrite (14) in the sole variable $\boldsymbol{X}$ as follows:

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \sum_{(i,j)\in\mathcal{E}} w_{ij}^t \|[\boldsymbol{X}]_{ij}^{Rt} - [\boldsymbol{X}]_{ii}^{Rt} - \bar{\boldsymbol{t}}_{ij}\|_2 + \frac{w_{ij}^R}{\sqrt{2}} \|[\boldsymbol{X}]_{ij}^{RR} - \bar{\boldsymbol{R}}_{ij}\|_F \\ \text{s.t.} \quad & \boldsymbol{X} \succeq 0, \ \mathrm{rank}\,(\boldsymbol{X}) = 2, \ [\boldsymbol{X}]_{ii}^{RR} = \mathbf{I}_2, i = 1, \ldots, n \end{aligned} \tag{15}$$

Problem (15) has a convex objective and the only nonconvex constraint is the rank constraint on $\boldsymbol{X}$. Therefore, to obtain a convex problem we drop the rank constraint and we obtain the following convex relaxation.

Proposition 7 (Convex Relaxation With ℓ_2-Norm): The following semidefinite program is a convex relaxation of the nonconvex problem (6):

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \sum_{(i,j)\in\mathcal{E}} w_{ij}^t \|[\boldsymbol{X}]_{ij}^{Rt} - [\boldsymbol{X}]_{ii}^{Rt} - \bar{\boldsymbol{t}}_{ij}\|_2 + \frac{w_{ij}^R}{\sqrt{2}} \|[\boldsymbol{X}]_{ij}^{RR} - \bar{\boldsymbol{R}}_{ij}\|_F \\ \text{s.t.} \quad & \boldsymbol{X} \succeq 0, \quad [\boldsymbol{X}]_{ii}^{RR} = \mathbf{I}_2, \quad i = 1, \ldots, n \end{aligned} \tag{16}$$

The fact that (16) is a convex relaxation (6) trivially follows from the first part of this section: Problem (16) is convex and its feasible set contains the one of Problem (15), which is simply a reformulation of Problem (6).

2) Relaxation of the ℓ_1-Norm Formulation: The convex relaxation of Problem (8) can be derived in full analogy with the one presented in the previous section, by introducing the matrix $\boldsymbol{X}$ and noting that the terms $\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j$, $\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{t}_i$, and $\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{t}_j$ can be written using $\boldsymbol{X}$. Therefore, we obtain the following.

Proposition 8 (Convex Relaxation With ℓ_1-Norm): The following semidefinite program is a convex relaxation of the nonconvex problem (8):

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \sum_{(i,j)\in\mathcal{E}} w_{ij}^t \|[\boldsymbol{X}]_{ij}^{Rt} - [\boldsymbol{X}]_{ii}^{Rt} - \bar{\boldsymbol{t}}_{ij}\|_1 + \frac{w_{ij}^R}{2} \|[\boldsymbol{X}]_{ij}^{RR} - \bar{\boldsymbol{R}}_{ij}\|_1 \\ \text{s.t.} \quad & \boldsymbol{X} \succeq 0, \quad [\boldsymbol{X}]_{ii}^{RR} = \mathbf{I}_2, \quad i = 1, \ldots, n. \end{aligned} \tag{17}$$

3) Relaxation of Huber Formulation: This paragraph provides a relaxation of the robust formulation (9), using the same derivation of the previous sections.

Proposition 9 (Convex Relaxation With Huber Loss): The following semidefinite program is a convex relaxation of the nonconvex problem (9):

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \sum_{(i,j)\in\mathcal{E}} \mathrm{h}(w_{ij}^t \|[\boldsymbol{X}]_{ij}^{Rt} - [\boldsymbol{X}]_{ii}^{Rt} - \bar{\boldsymbol{t}}_{ij}\|_2) + \mathrm{h}(w_{ij}^R \|[\boldsymbol{X}]_{ij}^{RR} - \bar{\boldsymbol{R}}_{ij}\|_F) \\ \text{s.t.} \quad & \boldsymbol{X} \succeq 0, \quad [\boldsymbol{X}]_{ii}^{RR} = \mathbf{I}_2, \quad i = 1, \ldots, n. \end{aligned} \tag{18}$$

Again, the proof of the statement follows from the derivation we provided for the ℓ_2-norm case. While the convexity of the costs in (16) and (17) trivially follows from the convexity of the ℓ_1 and ℓ_2 norms, the convexity of the cost in (18) is less straightforward. To ascertain convexity of the cost in (18) we note that the first term in the cost has the form "$\mathrm{h}(\|\boldsymbol{x}\|_2)$", and is the composition of (i) a convex function ($\|\boldsymbol{x}\|_2$) and (ii) a function ($\mathrm{h}(\cdot)$) which is convex and nondecreasing when restricted to nonnegative arguments, see [16, p. 617]. Properties (i) and (ii) guarantee that the resulting function $\mathrm{h}(\|\boldsymbol{x}\|_2)$ is convex, see [2, p. 84]. An analogous argument holds for the second summand in (18).

[1] By definition $\mathrm{SO}(2) \doteq \{\boldsymbol{R} \in \mathbb{R}^{2\times 2} : \boldsymbol{R}^{\mathsf{T}}\boldsymbol{R} = \boldsymbol{R}\boldsymbol{R}^{\mathsf{T}} = \mathbf{I}_2, \det(\boldsymbol{R}) = +1\}$.

B. Rounding the Relaxed Solutions

The solution of each of the convex relaxations (16), (17), and (18) is a matrix, that we call $\hat{\boldsymbol{X}}$. However, our goal is to estimate a set of poses. In this section we show how to retrieve the poses $(\boldsymbol{R}_i, \boldsymbol{t}_i), i = 1, \dots, n$, from $\hat{\boldsymbol{X}}$.

Rotation rounding: The computation of the rotation estimates from $\hat{\boldsymbol{X}}$ proceeds along the same lines of [33] and [5]. Given the matrix $\hat{\boldsymbol{X}}$ we first compute a rank-2 approximation of $\hat{\boldsymbol{X}}$ via singular value decomposition. The resulting matrix $\hat{\boldsymbol{Z}} \in \mathbb{R}^{2\times 3n}$ is such that $\hat{\boldsymbol{Z}}^{\mathsf{T}}\hat{\boldsymbol{Z}} \approx \hat{\boldsymbol{X}}$ (the previous is an actual equality when $\hat{\boldsymbol{X}}$ has rank 2). Then, since the $\hat{\boldsymbol{Z}}$ has the structure described in (12), we know that the n first 2×2 blocks of $\hat{\boldsymbol{Z}}$ contain our rotation estimates. Therefore we project each block to $\mathrm{SO}(2)$, as prescribed in [15, Section 5], and obtain our rotation estimates $\hat{\boldsymbol{R}}_i, i = 1, \dots, n$.

Translation rounding: The translation computation leverages the rotation estimates $\hat{\boldsymbol{R}}_i, i = 1, \dots, n$, computed in the previous section. Let us call $\hat{\boldsymbol{R}} \doteq [\hat{\boldsymbol{R}}_1 \ \dots \ \hat{\boldsymbol{R}}_n]$ the matrix stacking all these estimates. Then, by inspection of the matrix $\boldsymbol{X}$ in (13) we realize that $\boldsymbol{R}\boldsymbol{X}^{Rt} = n[\boldsymbol{t}_1 \ \dots \ \boldsymbol{t}_n]$. Based on this observation, we build our translation estimate as:

$$\hat{\boldsymbol{t}} \doteq [\hat{\boldsymbol{t}}_1 \ \dots \ \hat{\boldsymbol{t}}_n] = \frac{1}{n}\hat{\boldsymbol{R}}\hat{\boldsymbol{X}}^{Rt}. \tag{19}$$

V. Decoupling Rotations and Translations: 2-Stage Convex Relaxations

In this section we discuss a slightly different approach for computing approximate solutions for the problems (6), (8), (9). These three problems share the fact that the corresponding cost functions have two different terms: the first involving rotation and translations and the second involving only rotations. Based on the empirical observation that the contribution of the first term to the rotation estimate is often negligible (see [6]), in this section we propose to first compute a rotation estimate by minimizing the second summand, and then compute the translation estimate by minimizing the first summand with given rotations. While the decoupling of rotation and translation estimation might sound counterintuitive, Section VII shows that this *2-stage approach* is indeed the most effective, leading to accurate estimate in presence of many outliers.

Stage 1. Rotation estimation: The rotation subproblem in (6), (8), (9), assumes the following general form:

$$\min_{\boldsymbol{R}_i \in \mathrm{SO}(2)} \sum_{(i,j)\in\mathcal{E}} f_R(\boldsymbol{R}_i^{\mathsf{T}}\boldsymbol{R}_j - \bar{\boldsymbol{R}}_{ij}) \tag{20}$$

where, depending on the formulation ((6), (8), (9)), the function $f_R(\cdot)$ denotes the ℓ_2-norm, the ℓ_1-norm, or the Huber loss, properly weighted by w_{ij}^R.

We now propose a convex relaxation which proceeds along the lines of Section IV. First we define two matrices, $\boldsymbol{R} \doteq [\boldsymbol{R}_1 \ \dots \ \boldsymbol{R}_n] \in \mathbb{R}^{2\times 2n}$, and

$$\boldsymbol{X}^{RR} \doteq \boldsymbol{R}^{\mathsf{T}}\boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_1^{\mathsf{T}}\boldsymbol{R}_1 & \dots & \boldsymbol{R}_1^{\mathsf{T}}\boldsymbol{R}_n \\ \vdots & \ddots & \vdots \\ \boldsymbol{R}_n^{\mathsf{T}}\boldsymbol{R}_1 & \dots & \boldsymbol{R}_n^{\mathsf{T}}\boldsymbol{R}_n \end{bmatrix} \in \mathbb{R}^{2n\times 2n} \tag{21}$$

Then, we repeat the same derivation of Section IV (but applied to the smaller matrix $\boldsymbol{X}^{RR}$) and get the following convex relaxation of the rotation subproblem (20):

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \sum_{(i,j)\in\mathcal{E}} f_R([\boldsymbol{X}]_{ij}^{RR} - \bar{\boldsymbol{R}}_{ij}) \\ \text{s.t.} \ & \boldsymbol{X}^{RR} \succeq 0, \quad [\boldsymbol{X}]_{ii}^{RR} = \mathbf{I}_2, \quad i = 1, \dots, n. \end{aligned} \tag{22}$$

Calling $\hat{\boldsymbol{X}}^{RR}$ the solution of the convex program (22), we can recover the rounded rotation estimates, $\hat{\boldsymbol{R}}_i$, $i = 1, \dots, n$, from $\hat{\boldsymbol{X}}^{RR}$ as discussed in Section IV-B. The relaxation (22) is similar to the one used by Wang and Singer in [33], except for the fact that we accommodate different robust cost functions.

Stage 2. Translation estimation: The translation subproblem corresponds to the first summand in (6), (8), (9), and can be written in the following general form:

$$\min_{\boldsymbol{t}_i \in \mathbb{R}^2} \sum_{(i,j)\in\mathcal{E}} f_t\left(\hat{\boldsymbol{R}}_i^{\mathsf{T}}\boldsymbol{t}_j - \hat{\boldsymbol{R}}_i^{\mathsf{T}}\boldsymbol{t}_i - \bar{\boldsymbol{t}}_{ij}\right) \tag{23}$$

where, depending on the formulation ((6), (8), (9)), the function $f_t(\cdot)$ denotes the ℓ_2-norm, the ℓ_1-norm, or the Huber loss, properly weighted by w_{ij}^t. In (23), we already substituted the rotation estimates computed in Stage 1, hence the translations are the only unknowns.

Problem (23) is already a convex program, since the problem is unconstrained and $f_t(\cdot)$ is a convex function in all the considered formulations. Therefore, Stage 2 simply consists of solving (23) with an off-the-shelf convex solver to get the translation estimates $\hat{\boldsymbol{t}} \doteq [\hat{\boldsymbol{t}}_1, \dots, \hat{\boldsymbol{t}}_n]$.

VI. A Posteriori Performance Guarantees

This section provides *a posteriori* checks that can ascertain the quality of the relaxed solution after solving the convex relaxation. We give the following result, applied to (6), while the very same statement holds for Problems (8), (9).

Proposition 10 (Tightness in robust PGO formulations): Let $f(\boldsymbol{R}, \boldsymbol{t})$ be the cost function of the robust PGO formulation (6), with $\boldsymbol{R} \doteq [\boldsymbol{R}_1, \dots, \boldsymbol{R}_n]$ and $\boldsymbol{t} \doteq [\boldsymbol{t}_1, \dots, \boldsymbol{t}_n]$, and call $f^\star$ the corresponding optimal cost. With slight abuse of notation, call $f(\boldsymbol{X})$ the cost function of the convex relaxation (16), and denote with $\hat{\boldsymbol{X}}$ the optimal solution of (16) and with $(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$ the corresponding rounded estimate. Then, it holds:

$$f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}}) - f^\star \leq f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}}) - f(\hat{\boldsymbol{X}}) \tag{24}$$

i.e., we can compute a bound on the suboptimality gap $f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}}) - f^\star$ by using the optimal cost of the relaxation $f(\hat{\boldsymbol{X}})$. Moreover, if the relaxed solution $\hat{\boldsymbol{X}}$ has rank 2 and its rank-2 decomposition $\hat{\boldsymbol{Z}}$ is such that the first n 2×2 blocks of $\hat{\boldsymbol{Z}}$ are in $\mathrm{SO}(2)$, then $f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}}) = f^\star$ and the rounded solution $(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$ is optimal for the original problem (6).

Proof: Since (16) is a relaxation of Problem (6), it follows that $f(\hat{\boldsymbol{X}}) \leq f^\star$, which implies the inequality (24). Moreover, if $\hat{\boldsymbol{X}}$ has rank 2, then $f(\hat{\boldsymbol{X}}) = f(\hat{\boldsymbol{Z}}^{\mathsf{T}}\hat{\boldsymbol{Z}})$, where $\hat{\boldsymbol{Z}}$ is the rank-2 decomposition of $\hat{\boldsymbol{X}}$. If the n first 2×2 blocks of $\hat{\boldsymbol{Z}}$ are already in $\mathrm{SO}(2)$, then the rounding does not alter $\hat{\boldsymbol{Z}}$, i.e., $\hat{\boldsymbol{Z}} = [\hat{\boldsymbol{R}}\ \hat{\boldsymbol{t}}]$. Therefore, it holds that (i) $f(\hat{\boldsymbol{X}}) = f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}}) \leq f^\star$. Since $(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$ is feasible for (6), by optimality of $f^\star$ it follows that (ii) $f^\star \leq f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$. Combining the inequalities (i) and (ii), it follows that $f(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}}) = f^\star$, proving optimality of $(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$. ■

Proposition 10 provides computational tools to quantify the suboptimality of the rounded solution $(\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$. Moreover, it gives an a posteriori condition under which the relaxation is tight. Tightness is attained under two conditions. The first condition is that the rank of $\hat{\boldsymbol{X}}$ is 2: this guarantees that we did not lose anything when relaxing the rank constraint in (15). The second condition is that $\hat{\boldsymbol{Z}}$ (the rank 2 decomposition of $\hat{\boldsymbol{X}}$) contains rotation matrices: this guarantees that we did not lose anything by dropping the determinant constraint in (15). As mentioned in Section IV, indeed empirical evidence suggests that the determinant constraint does not impact the results, hence we are mainly interested in the rank of $\hat{\boldsymbol{X}}$.

An analogous result applies to the rotation estimation of the 2-stage approaches, as discussed below (the proof is identical to the one of Proposition 10).

Proposition 11 (Tightness in 2-stage formulations): Let $f_R(\boldsymbol{R})$ be the cost function of the rotation subproblem in (20), with $\boldsymbol{R} \doteq [\boldsymbol{R}_1, \ldots, \boldsymbol{R}_n]$, and call $f_R^\star$ the corresponding optimal cost. With slight abuse of notation, call $f_R(\boldsymbol{X}^{RR})$ the cost function of the convex relaxation (22), and denote with $\hat{\boldsymbol{X}}^{RR}$ the optimal solution of (22) and with $\hat{\boldsymbol{R}}$ the corresponding rounded estimate. Then, it holds that

$$f_R(\hat{\boldsymbol{R}}) - f_R^\star \leq f_R(\hat{\boldsymbol{R}}) - f_R\left(\hat{\boldsymbol{X}}^{RR}\right) \tag{25}$$

i.e., we can compute a bound on the suboptimality gap $f_R(\hat{\boldsymbol{R}}) - f_R^\star$ by using the optimal cost of the relaxation $f_R(\hat{\boldsymbol{X}}^{RR})$. Moreover, if the relaxed solution $\hat{\boldsymbol{X}}^{RR}$ has rank 2 and its rank-2 decomposition $\hat{\boldsymbol{Z}}$ is such that the first n 2×2 blocks of $\hat{\boldsymbol{Z}}$ are in $\mathrm{SO}(2)$, then $f_R(\hat{\boldsymbol{R}}) = f_R^\star$ and the rounded solution $\hat{\boldsymbol{R}}$ is optimal for the rotation subproblem in (20).

Note that we only discuss the tightness in the relaxation of the rotation estimation problem (Stage 1), while translation estimation (Stage 2) is already convex.

VII. Numerical Evaluation

This section presents extensive numerical simulations and provides empirical evidence that (i) the *1-stage* convex relaxations of Section IV are *not* tight, while the relaxations of the rotation subproblems in Section V are indeed tight in the tested scenarios; (ii) the *2-stage* approaches are robust with respect to increasing probability of outlying measurements and ensure excellent performance in practice; (iii) the *2-stage* approaches allows a reliable detection of the outliers; (iv) the *2-stage* approaches outperform state-of-the-art techniques based on local optimization, such as *Dynamic Covariance Scaling* [26].

A. Summary of the Proposed Approaches

In the experimental evaluation we compared the six proposed approaches for solving the PGO with outliers:

1) ℓ_1 *on poses:* solution of Problem (17);
2) ℓ_1 *2-stage:* solution of Problems (22) and (23) with $f_R(\cdot) = \|\cdot\|_1$ and $f_t(\cdot) = \|\cdot\|_1$;
3) ℓ_2 *on poses:* solution of Problem (16);
4) ℓ_2 *2-stage:* solution of Problems (22) and (23) with $f_R(\cdot) = \|\cdot\|_F$ and $f_t(\cdot) = \|\cdot\|_2$;
5) *Huber on poses:* solution of Problem (18);
6) *Huber 2-stage:* solution of Problems (22) and (23) with $f_R(\cdot) = \mathrm{h}(\|\cdot\|_F)$ and $f_t(\cdot) = \mathrm{h}(\|\cdot\|_2)$.

All problems above are convex and can be solved globally by off-the-shelf SDP solvers. We use `cvx` [13] as convex solver. After solving the convex relaxation, we apply the *rounding procedure*, in order to obtain a feasible solution of the original problems. We compute statistics over 30 runs.

B. Monte Carlo Analysis and Simulation Setup

We performed a Monte Carlo analysis on synthetic datasets consisting of randomly generated pose graphs. Ground truth positions of the n nodes are drawn from a uniform distribution in a square; ground truth orientations are drawn from a uniform distribution over $(-\pi, +\pi]$. Connections among nodes are generated according to (i) the *Erdős-Rényi* random graph model, where each edge is included in the graph with probability p independently from every other edge (in our tests we set $p = 0.5$); or (ii) the *Geometric* random graph model, where all nodes closer than a specified distance (set to $\frac{\Delta}{4}$ in our tests, where Δ is the size of the environment) are connected by an edge. Illustrative examples of random graphs are provided in the supplemental material [4]. Once the graph is created, the relative measurements $(\check{\boldsymbol{t}}_{ij}, \check{\boldsymbol{R}}_{ij})$ associated to each edge $(i,j) \in \mathcal{E}$ are generated as:

$$\begin{aligned} \check{\boldsymbol{t}}_{ij} &= (1 - \delta_{ij})\bar{\boldsymbol{t}}_{ij} + \delta_{ij}\tilde{\boldsymbol{t}}_{ij}, \\ \check{\boldsymbol{R}}_{ij} &= (1 - \delta_{ij})\bar{\boldsymbol{R}}_{ij} + \delta_{ij}\tilde{\boldsymbol{R}}_{ij} \end{aligned} \tag{26}$$

where δ_{ij} are i.i.d. *Bernoulli* random variables with parameter p^{out}, and are such that with probability p^{out}, $\delta_{ij} = 1$ and the measurement $(\check{\boldsymbol{t}}_{ij}, \check{\boldsymbol{R}}_{ij})$ is assigned an outlier measurement $(\tilde{\boldsymbol{R}}_{ij}, \tilde{\boldsymbol{t}}_{ij})$, while if $\delta_{ij} = 0$ the measurement $(\check{\boldsymbol{t}}_{ij}, \check{\boldsymbol{R}}_{ij})$ is assigned an inlier (but noisy) measurement $(\bar{\boldsymbol{R}}_{ij}, \bar{\boldsymbol{t}}_{ij})$. Inliers and outliers models are as follows: the inliers $(\bar{\boldsymbol{t}}_{ij}, \bar{\boldsymbol{R}}_{ij})$ are noisy measurements generated according to model (1) with $\boldsymbol{t}_{ij}^\epsilon \sim \text{Normal}(\boldsymbol{0}_2, \sigma_T^2)$ and $\boldsymbol{R}_{ij}^\epsilon = \boldsymbol{R}(\epsilon_{ij}^R)$, $\epsilon_{ij}^R \sim \text{Normal}(0, \sigma_R^2)$, where $\boldsymbol{R}(\epsilon_{ij}^R)$ is the planar rotation matrix of angle ϵ_{ij}^R. For our tests we set $\sigma_R = 0.01$ and $\sigma_T = 0.1$, which are good proxies for the noise levels found in practice. The outliers $(\tilde{\boldsymbol{t}}_{ij}, \tilde{\boldsymbol{R}}_{ij})$ are completely wrong measurements, and are obtained from the model (1) by adding large uniformly distributed noise: $\boldsymbol{t}_{ij}^\epsilon \sim \text{Uniform}(-\frac{\Delta}{4}, \frac{\Delta}{4})$ (where Δ is the size of the environment), and $\boldsymbol{R}_{ij}^\epsilon = \boldsymbol{R}(\epsilon_{ij}^R)$, $\epsilon_{ij}^R \sim \text{Uniform}(-\pi, +\pi)$.

Since the general purpose solver in `cvx` does not scale to large instances, we focus on relatively small graphs, with $n = 20$ and $n = 50$.

C. Tightness of Convex Relaxations

In this section we evaluate the tightness of the convex relaxations, i.e., their capability of producing accurate solutions for the original (nonconvex) problem. In order to check if the convex relaxations in *1-stage* approaches are tight, we check the rank of the solution matrices $\hat{\boldsymbol{X}}$ of problems (16), (17) and (18) (see Proposition 10). In all instances we tested, the rank of matrix $\hat{\boldsymbol{X}}$ is very high, even in absence of outliers ($p^{\text{out}} = 0$). The solution matrices of problem (17) and (18) have rank equal to n, while the rank of the solution matrix $\hat{\boldsymbol{X}}$ of (16) is around 10 and 25 for instances having respectively 20 and 50 nodes. This empirical evidence shows that the SDP relaxations (16), (17) and (18) are not tight in presence of noise and outlying measurements.

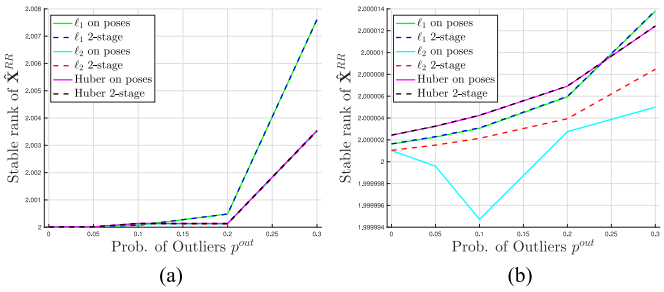

Fig. 1. Stable rank of the matrix $\hat{\boldsymbol{X}}^{RR}$ computed by the different approaches and for increasing probability of outliers. (a) $n = 20$. (b) $n = 50$.

On the other hand, in *2-stage approaches* the relaxations of the rotation subproblem (i.e., problem (22)) are practically always tight. We computed the *stable rank* of the matrix $\hat{\boldsymbol{X}}^{RR}$, i.e., the squared ratio between the Frobenius norm and the spectral norm. The stable rank is a real number rather than an integer, and provides a more detailed picture of the rank of the matrix, without requiring to commit to a given numerical tolerance for the rank computation. Fig. 1(a) shows that the stable rank of $\hat{\boldsymbol{X}}^{RR}$ is very close to 2 for all considered values of p^{out} and for $n = 20$. For the 2-stage approaches, this confirms the tightness of our convex relaxations. Interestingly, also if the 1-stage approaches, even if the matrix $\hat{\boldsymbol{X}}$ has large rank, the submatrix $\hat{\boldsymbol{X}}^{RR}$ has rank close to 2. This suggests that the presence of the translations in the 1-stage formulations breaks the tightness. These results are further confirmed by Fig. 1(b), which shows results for graphs with $n = 50$ poses.

D. Robust PGO

In this section we evaluate, for increasing levels of outliers p^{out}, the quality of the pose estimates produced by the proposed approaches. The estimation quality is quantified by the mean rotation and mean translation error, computed with respect to the ground truth. Fig. 2 shows the mean errors, averaged on 30 runs. We note that there is a huge divide between the two set of approaches. On the one hand, the 1-stage approaches are very sensitive to the presence of outliers, since mean translation and rotation errors quickly increase with p^{out}. On the other hand, the estimates provided by the 2-stage approaches have small errors and are remarkably insensitive to the increase of the probability of outliers (y-axis is in log scale). In particular, *Huber 2-stage* ensures top performance, followed by ℓ_2 *2-stage* and ℓ_1 *2-stage*.

E. Sensor Failure Detection via Convex Relaxations

The proposed techniques can be also used for outlier *identification*, which is an important tool for sensor failure detection.

A simple outlier detection technique would first solve one of the proposed convex relaxations to obtain an estimate of the poses $(\boldsymbol{R}, \boldsymbol{t})$, with $\boldsymbol{R} = [\boldsymbol{R}_1, \ldots, \boldsymbol{R}_n]$ and $\boldsymbol{t} = [\boldsymbol{t}_1, \ldots, \boldsymbol{t}_n]$. Then, for each measurement $(\check{\boldsymbol{R}}_{ij}, \check{\boldsymbol{t}}_{ij})$, it would use such estimate to compute the residual error for that measurement, and classify as outliers measurements with large residuals. Due to space reasons, we report precision/recall curves for outlier detection in [4]. The results confirm that 2-stage techniques lead to high precision/recall and are effective in identifying spurious measurements.

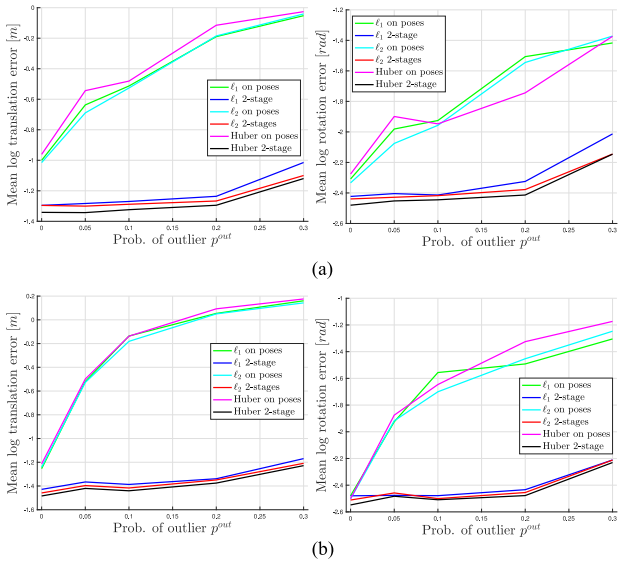

Fig. 2. Estimation errors for the six approaches proposed in this letter. Left column: average translation error. Right column: average rotation error. (a) *Erdős-Rényi* random graphs and (b) *Geometric* random graphs. Similar results for graphs with $n = 50$ nodes are given in the supplemental material [4].

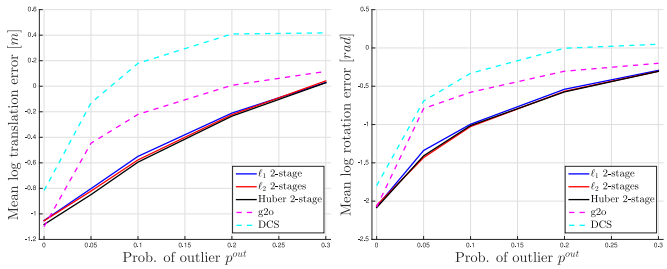

Fig. 3. Estimation errors for the proposed 2-stage approaches, g2o, and DCS. (a) Average translation error; (b) average rotation error. Similar results for graphs with $n = 50$ nodes are given in the supplemental material [4].

F. Comparison Against the State of the Art

In this section we show that the proposed techniques outperform the *Dynamic Covariance Scaling* approach [26].

We consider a *Grid* graph, i.e., a Manhattan world model similar to the one used in related work [5], [32]; we simulate a complete grid (an example is given in [4]), including an odometric path connecting all nodes. Each node in the Grid has at most 4 neighbors (corresponding to adjacent nodes in the grid) hence it is less connected than the graphs in Section VII-B, and closer to pose graphs found in practice. In this section, we focus on the 2-stage approaches, since we already observed that they ensure the best performance. We benchmark the proposed 2-stage techniques against g2o [20], a non-robust PGO solver, and *Dynamic Covariance Scaling* [26] (DCS), a state-of-the-art robust PGO approach. Both approaches need an initial guess: in our tests we compute the initial guess from the odometric edges, following common practice. We use default parameters for both g2o and DCS.

Fig. 3 shows the mean translation and rotation errors, averaged over 30 runs, for increasing amount of outliers. The proposed 2-stage approaches dominate g2o and DCS in all test instances. g2o is not a robust solver, hence this result is expected.

On the other hand, we observe that also DCS has degraded performance: in our tests the initial guess to DCS is computed from the odometry, and some of the odometric edges may be spurious; in those cases, DCS starts from an incorrect initialization and its robust cost has the undesirable effect of "disabling" the correct edges which are not consistent with the initial guess.[2] This further stresses the importance of designing global techniques for robust PGO that do not rely on an initial guess. On the downside, we observe that while the proposed techniques outperform DCS, their performance is significantly worse than the one observed in Section VII-D. This suggests that the performance of the proposed convex relaxations degrades for graphs with low connectivity.

VIII. Conclusion

We proposed robust approaches for the solution of PGO with outliers. We considered three robust cost functions and provided a systematic way of computing a convex relaxation of the resulting (nonconvex) optimization problems. Numerical experiments suggest that a subset of the proposed techniques ensures accurate estimation in the face of many outliers, particularly when tested on well-connected graphs. This also enables outlier identification and failure detection.

This work opens several avenues for future research, including the analysis of 3D problem instances, a deeper theoretical investigation of the performance of the proposed convex relaxations, and the design of ad-hoc SDP solvers: while the general purpose SDP solver used in our current implementation scales poorly in the problem size (it requires tens of seconds to solve an instance with 50 poses), our ultimate goal is to solve large PGO instances ($>$1000 poses) efficiently and robustly.

Acknowledgment

The authors would like to thank Carlo Tommolillo for his help with the numerical evaluation.

References

[1] R. Aragues, L. Carlone, G. Calafiore, and C. Sagues, "Multi-agent localization from noisy relative pose measurements," in *Proc. 2011 IEEE Int. Conf. Robot. Autom.*, 2011, pp. 364–369.

[2] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[3] C. Cadena *et al.*, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.

[4] L. Carlone and G. Calafiore, "Convex relaxations for pose graph optimization with outliers," 2017. [Online]. Available: https://arxiv.org/abs/1801.02112

[5] L. Carlone, G. Calafiore, C. Tommolillo, and F. Dellaert, "Planar pose graph optimization: Duality, optimal solutions, and verification," *IEEE Trans. Robot.*, vol. 32, no. 3, pp. 545–565, Jun. 2016.

[6] L. Carlone and A. Censi, "From angular manifolds to the integer lattice: Guaranteed orientation estimation with application to pose graph optimization," *IEEE Trans. Robot.*, vol. 30, no. 2, pp. 475–492, Apr. 2014.

[7] L. Carlone, A. Censi, and F. Dellaert, "Selecting good measurements via ℓ_1 relaxation: A convex approach for robust estimation over graphs," in *Proc. 2014 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 2667–2674.

[8] L. Carlone and F. Dellaert, "Duality-based verification techniques for 2D SLAM," in *Proc. 2015 IEEE Int. Conf. Robot. Autom.*, 2015, pp. 4589–4596.

[9] L. Carlone, D. Rosen, G. Calafiore, J. J. Leonard, and F. Dellaert, "Lagrangian duality in 3D SLAM: Verification techniques and optimal solutions," in *Proc. 2015 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 125–132.

[10] J. J. Casafranca, L. M. Paz, and P. Piniés, "A back-end l1 norm based solution for factor graph SLAM," in *Proc. 2013 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 17–23.

[11] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[12] M. C. Graham, J. P. How, and D. E. Gustafson, "Robust incremental slam with consistency-checking," in *Proc. 2015 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2015, pp. 117–124.

[13] M. Grant and S. Boyd, "CVX: MATLAB software for disciplined convex programming, version 2.1." Mar. 2014. [Online]. Available: http://cvxr.com/cvx

[14] G. Grisetti, C. Stachniss, and W. Burgard, "Non-linear constraint network optimization for efficient map learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 428–439, Sep. 2009.

[15] R. Hartley, J. Trumpf, Y. Dai, and H. Li, "Rotation averaging," *Int. J. Comput. Vis.*, vol. 103, no. 3, pp. 267–305, 2013.

[16] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press 2003.

[17] P. Huber, *Robust Statistics*. New York, NY, USA: Wiley, 1981.

[18] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *Int. J. Robot. Res.*, vol. 31, pp. 217–236, Feb. 2012.

[19] S. Kotz, T. J. Kozubowski, and K. Podgórski, "Asymmetric multivariate laplace distribution" in *The Laplace Distribution and Generalizations: A Revisit With Applications to Communications, Economics, Engineering, and Finance*. Boston, MA, USA: Birkhäuser, 2001, pp. 239–272.

[20] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *Proc. 2011 IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 2011, pp. 3607–3613.

[21] Y. Latif, C. D. C. Lerma, and J. Neira, "Robust loop closing over time," in *Proc. Robot., Sci. Syst.*, 2012.

[22] G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Robust pose-graph loop-closures with expectation-maximization," in *Proc. 2013 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 556–563.

[23] N. Mitianoudis, "A directional Laplacian density for underdetermined audio source separation," in *Proc. Int. Conf. Artif. Neural Netw.*, 2010, vol. 6352, pp. 450–459.

[24] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," in *Proc. Robot., Sci. Syst.*, Sydney, NSW, Australia, Jul. 2012.

[25] E. Olson, J. Leonard, and S. Teller, "Fast iterative alignment of pose graphs with poor initial estimates," in *Proc. 2006 IEEE Int. Conf. Robot. Autom.*, May 2006, pp. 2262–2269.

[26] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proc. 2013 IEEE Int. Conf. Robot. Autom.*, 2013, pp. 62–69.

[27] M. Pfingsthorn and A. Birk, "Generalized graph SLAM: Solving local and global ambiguities through multimodal and hyperedge constraints," *Int. J. Robot. Res.*, vol. 35, no. 6, pp. 601–630, 2016.

[28] 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," in *Proc. Int. Workshop Algorithmic Found. Robot.*, 2016.

[29] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. New York, NY, USA: Wiley, 1981.

[30] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose graph SLAM," in *Proc. 2012 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 1879–1884.

[31] N. Sunderhauf and P. Protzel, "Towards a robust back-end for pose graph SLAM," in *Proc. 2012 IEEE Int. Conf. Robot. Autom.*, 2012, pp. 1254–1261.

[32] N. Sünderhauf and P. Protzel, "Switchable constraints vs. max-mixture models vs. RRR–A comparison of three approaches to robust pose graph SLAM," in *Proc. 2013 IEEE Int. Conf. Robot. Autom.*, 2013, pp. 5198–5203.

[33] L. Wang and A. Singer, "Exact and stable recovery of rotations for robust synchronization," *Inf. Inference, J. IMA*, vol. 2, no. 2, pp. 145–193, 2013.

[34] G. Zioutas and A. Avramidis, "Deleting outliers in robust regression with mixed integer programming," *Acta Math. Appl. Sinica*, vol. 21, no. 2, pp. 323–334, 2005.

[2] DCS would instead work well when bootstrapped with a good initial guess. Unfortunately, a good guess is non-trivial to compute in presence of outliers.