

MIT Open Access Articles

A convex relaxation for approximate global optimization in simultaneous localization and mapping

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Rosen, David M., Charles DuHadway, and John J. Leonard. "A Convex Relaxation for Approximate Global Optimization in Simultaneous Localization and Mapping." IEEE, 2015. 5822–5829.

As Published: <http://dx.doi.org/10.1109/ICRA.2015.7140014>

Publisher: Institute of Electrical and Electronics Engineers (IEEE)

Persistent URL: <http://hdl.handle.net/1721.1/107496>

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike



A Convex Relaxation for Approximate Global Optimization in Simultaneous Localization and Mapping

David M. Rosen*, Charles DuHadway†, and John J. Leonard*

Abstract—Modern approaches to simultaneous localization and mapping (SLAM) formulate the inference problem as a high-dimensional but sparse nonconvex M-estimation, and then apply general first- or second-order smooth optimization methods to recover a local minimizer of the objective function. The performance of any such approach depends crucially upon initializing the optimization algorithm near a good solution for the inference problem, a condition that is often difficult or impossible to guarantee in practice. To address this limitation, in this paper we present a formulation of the SLAM M-estimation with the property that, by expanding the feasible set of the estimation program, we obtain a *convex relaxation* whose solution approximates the globally optimal solution of the SLAM inference problem and can be recovered using a smooth optimization method initialized at *any* feasible point. Our formulation thus provides a means to obtain a high-quality solution to the SLAM problem *without* requiring high-quality initialization.

I. INTRODUCTION

The ability to learn a map of an initially unknown environment while simultaneously localizing within that map as it is being constructed (a procedure known as *simultaneous localization and mapping* (SLAM)) is a fundamental competency in robotics [1]. Consequently, SLAM has been the focus of a sustained research effort over the previous three decades, and there now exist a variety of mature algorithms and software libraries to solve this problem in practice (cf. [2]–[5]).

State-of-the-art approaches to SLAM typically formulate the inference problem as a high-dimensional but sparse nonconvex M-estimation [6], and then apply general first- or second-order smooth optimization methods to estimate a critical point of the objective function. This approach admits the development of straightforward and fast inference algorithms, but its computational expedience comes at the expense of robustness: specifically, the optimization methods that underpin these SLAM techniques are usually only able to guarantee convergence to a first-order critical point of the objective (i.e. a *local* minimum or saddle point), rather than the *globally* optimal solution [7]. This restriction to local rather than global solutions has several important undesirable practical ramifications.

The most serious limitation is that the solution to which any such method ultimately converges is determined by its initialization. This is particularly pernicious in the context of SLAM, in which the combination of a high-dimensional state space and significant nonlinearities in the objective function (due to e.g. the effects of rotational degrees of freedom in the

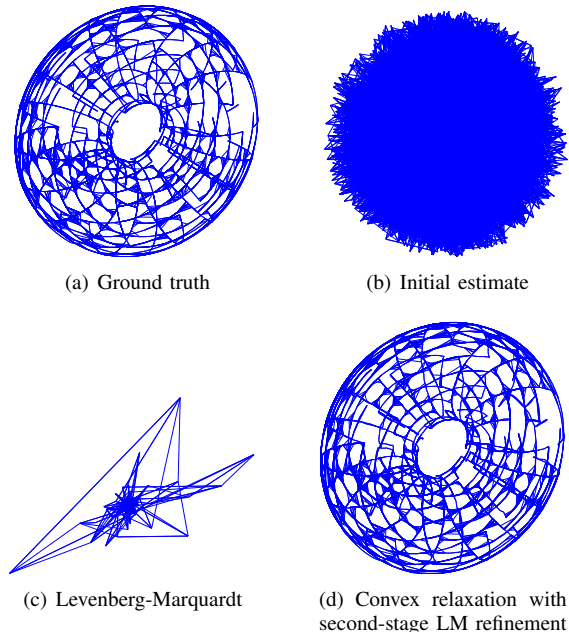


Fig. 1. The effects of poor initialization on the SLAM estimation for the torus11500 dataset (11500 6DOF poses, 22643 pose-pose measurements). (a): The ground truth map. (b): A poor (randomly sampled) initial estimate. (c): The solution obtained from Levenberg-Marquardt initialized with (b). (d): The solution obtained by first solving a convex relaxation of the original SLAM estimation problem (also initialized using (b)) to produce an approximation to the globally optimal solution, and then refining this approximation in a second-stage optimization using Levenberg-Marquardt. This approach recovers a globally consistent estimate in spite of the poor initialization.

estimated states or the use of nonlinear robust cost functions) can give rise to complex cost surfaces containing many local minima in which smooth optimization methods can become entrapped. The performance of any such SLAM technique thus depends crucially upon initializing the back-end optimization algorithm with an estimate that is close to a good solution for the inference task (Fig. 1). However, it is far from clear that this condition can always be satisfied in practice, especially in view of the fact that the entire point of solving the SLAM problem is precisely to *obtain* such a high-quality estimate.

One general strategy for attacking challenging optimization problems of this type that has proven to be very successful is *convex relaxation*. In this approach, the original problem is modified in such a way as to produce a convex approximation (for which a *globally* optimal solution can be readily obtained using *local* smooth optimization techniques initialized at *any* feasible point [8]), whose solution then approximates a solution for the original problem. While theoretical bounds on the approximation loss of such relaxations are often difficult to

*Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology, Cambridge, MA 02139, USA. Email: {dmrosen, jleonard}@mit.edu

†Google Research, Mountain View, CA 94043, USA. Email: duhadway@google.com

produce (or turn out to be quite weak), a large body of numerical experience has shown that this approach often performs remarkably well in practice when applied to “typical” problem instances (i.e., instances that are not artificially adversarial).

Motivated by this prior experience, in this paper we propose a practical approach to global optimization in SLAM via convex relaxation. Our approach formulates the SLAM problem as an instance of M-estimation with the property that, by expanding the feasible set of the estimation program, we obtain a convex relaxation whose globally optimal solution can be computed efficiently in practice; by projecting this solution onto the feasible set of the original program, we thereby obtain a good approximation of the global solution to the original SLAM M-estimation problem which is suitable for subsequent refinement with a local smooth optimization technique. Our approach thus provides a means to obtain a good estimate for the globally optimal solution of the SLAM problem *without* the need to supply a high-quality initialization.

II. THE SLAM M-ESTIMATION AND ITS CONVEX RELAXATION

A. Notation and mathematical preliminaries

Our development will make frequent use of the geometry of the special orthogonal and special Euclidean groups and their realizations as subsets of linear spaces. To that end, in this subsection we briefly establish some notational and mathematical preliminaries that will be useful in the sequel.

We let $[n] \triangleq \{1, 2, \dots, n\}$ denote the first n positive integers and $\mathbb{R}_{\geq 0}$ the nonnegative real numbers. We denote by $SO(n)$ the realization of the special orthogonal group as the set of $n \times n$ orthogonal matrices with +1 determinant:

$$SO(n) \triangleq \{R \in \mathbb{R}^{n \times n} \mid R^T R = R R^T = I, \det R = 1\}. \quad (1)$$

Similarly, we will denote by $SE(n)$ the realization of the special Euclidean group¹ as the semidirect product

$$SE(n) \triangleq \mathbb{R}^n \rtimes SO(n) \quad (2)$$

under the group operation \oplus :

$$\begin{aligned} \oplus: SE(n) \times SE(n) &\rightarrow SE(n) \\ (t_1, R_1) \oplus (t_2, R_2) &= (t_1 + R_1 t_2, R_1 R_2). \end{aligned} \quad (3)$$

$SE(n)$ also has a group action² \bullet on \mathbb{R}^n given by:

$$\begin{aligned} \bullet: SE(n) \times \mathbb{R}^n &\rightarrow \mathbb{R}^n \\ (t, R) \bullet x &= R x + t. \end{aligned} \quad (4)$$

Finally, we denote by $V(n)$ the real vector space containing the realization (1)–(3) of $SE(n)$:

$$V(n) \triangleq \mathbb{R}^n \times \mathbb{R}^{n \times n}. \quad (5)$$

¹ This Lie group models the set of robot poses in n -dimensional Euclidean space under the operation of odometric composition.

² This action transforms a point whose coordinates $x \in \mathbb{R}^n$ are specified in the global coordinate frame (the frame of the identity pose $(0, I) \in SE(n)$) to coordinates in the frame associated with the robot pose $(t, R) \in SE(n)$.

B. The SLAM M-estimation problem

In this section we provide a brief review of the pose-and-landmark SLAM inference problem; interested readers are encouraged to consult [1], [9] for a more detailed presentation.

We consider a robot attempting to learn a map of some initially unknown environment. As the robot explores, it moves through some sequence of poses $p_1, \dots, p_{n_p} \in SE(n)$ in the environment while observing some collection of landmarks $l_1, \dots, l_{n_l} \in \mathbb{R}^n$ for $n \in \{2, 3\}$. We assume that the robot is able to collect noisy observations $\bar{z}_{ij} \in SE(n)$ of pose p_j in the coordinate system of pose p_i for some subset of pairs $(i, j) \in \mathcal{P} \subset [n_p] \times [n_p]$, and noisy observations $\bar{l}_{ik} \in \mathbb{R}^n$ of the position of landmark l_k in the coordinate frame of pose p_i for some subset of pairs $(i, k) \in \mathcal{L} \subset [n_p] \times [n_l]$. The goal is then to estimate the configuration of the states

$$X \triangleq (p_1, \dots, p_{n_p}, l_1, \dots, l_{n_l}) \in \overbrace{(SE(n))^{n_p} \times (\mathbb{R}^n)^{n_l}}^{\triangleq \mathcal{F}} \quad (6)$$

(i.e. the poses p_i and landmarks l_k) given the observations

$$Z \triangleq \{\bar{z}_{ij} \mid (i, j) \in \mathcal{P}\} \cup \{\bar{l}_{ik} \mid (i, k) \in \mathcal{L}\}. \quad (7)$$

Now in principle, the poses p_i , landmarks l_k , and observations \bar{z}_{ij} and \bar{l}_{ik} should satisfy the measurement equations:

$$p_j = p_i \oplus \bar{z}_{ij} \quad (8a)$$

$$l_k = p_i \bullet \bar{l}_{ik} \quad (8b)$$

for all $(i, j) \in \mathcal{P}$ and all $(i, k) \in \mathcal{L}$; however, because the observations \bar{z}_{ij} and \bar{l}_{ik} are corrupted by sensor noise, equations (8) are generally *inconsistent* in the sense that *no* choice of states X in (6) can satisfy them all simultaneously. Thus, in practice, the SLAM inference problem is solved via *M-estimation* [6]: we define a set of cost functions

$$c_{ij}(p_i, p_j): SE(n) \times SE(n) \rightarrow \mathbb{R}_{\geq 0} \quad \forall (i, j) \in \mathcal{P} \quad (9a)$$

$$c_{ik}(p_i, l_k): SE(n) \times \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0} \quad \forall (i, k) \in \mathcal{L} \quad (9b)$$

that penalize the failure of equations (8) to hold, and then define an optimal state estimate $X^* \in \mathcal{F}$ to be one that minimizes the cumulative cost of the penalties (9):

$$\begin{aligned} X^* &= \operatorname{argmin}_{X \in \mathcal{F}} \overbrace{\sum_{(i,j) \in \mathcal{P}} c_{ij}(p_i, p_j) + \sum_{(i,k) \in \mathcal{L}} c_{ik}(p_i, l_k)}^{\triangleq f(X)} \\ &\text{subject to } \tilde{p}_1 = (0, I). \end{aligned} \quad (10)$$

In practice, the cost functions c_{ij} and c_{ik} in (9) and (10) are usually chosen as negative log-likelihoods for some assumed measurement models $p_{ij}(\bar{z}_{ij} | p_i, p_j)$ and $p_{ik}(\bar{l}_{ik} | p_i, l_k)$ (i.e. for some assumed distribution on the measurement noise affecting the observations \bar{z}_{ij} and \bar{l}_{ik}), in which case the M-estimation in (10) is actually a maximum-likelihood estimation. For example, the most common formulation of the SLAM problem assumes additive mean-zero Gaussian noise models, for which

the corresponding negative log-likelihood functions are:

$$c_{ij}^G(p_i, p_j) = \|\omega_{ij}\|_{\Sigma_{ij}^R}^2 + \|\bar{t}_{ij} - R_i^T(t_j - t_i)\|_{\Sigma_{ij}^t}^2 \quad (11a)$$

$$c_{ik}^G(p_i, l_k) = \|\bar{l}_{ik} - R_i^T(l_k - t_i)\|_{\Sigma_{ik}}^2, \quad (11b)$$

where $\|x\|_{\Sigma} \triangleq \sqrt{x^T \Sigma^{-1} x}$ is the norm corresponding to the Mahalanobis distance, $\Sigma_{ij}^R, \Sigma_{ij}^t, \Sigma_{ik} \succ 0$ are the noise covariance matrices, and $\omega_{ij} \in \mathbb{R}^3$ is the axis-angle representation for the rotational measurement residual $R_i^T R_j \bar{R}_{ij}^T \in SO(3)$ (that is, ω_{ij} satisfies $[\omega_{ij}]_{\times} = \log(R_i^T R_j \bar{R}_{ij}^T)$). However, for our purposes it will be convenient to admit more general (i.e. non-probabilistic) cost functions in the sequel.

C. Convex relaxation of the SLAM M-estimation

In general, the SLAM M-estimation (10) is a high-dimensional nonconvex nonlinear program, and thus can be quite challenging to solve using smooth numerical optimization methods [7]. To address this difficulty, in this section we describe a special class of instances of (10) that admits a straightforward *convex relaxation*, whose globally optimal solution *can* be found using these techniques.

Our approach is based upon the observation that if we model the (abstract) special Euclidean group $SE(n)$ using the realization defined in (1)–(3), then the feasible set \mathcal{F} for the SLAM M-estimation (10) embeds into the linear space

$$\mathcal{S} \triangleq (V(n))^{n_p} \times (\mathbb{R}^n)^{n_l}. \quad (12)$$

The embedding $\mathcal{F} \hookrightarrow \mathcal{S}$ provides us with a means to “convexify” \mathcal{F} by enlarging it to its convex hull within the ambient linear space \mathcal{S} . If we additionally select the objective function f in (10) such that it has a convex extension defined over the convex hull of \mathcal{F} , then the relaxation of (10) obtained by extending \mathcal{F} to $\text{conv } \mathcal{F}$ within \mathcal{S} will be a convex program.

1) *Selecting the cost functions:* We wish to determine cost functions c_{ij} and c_{ik} in (9) that have convex extensions over the convex hulls of their respective domains. Given that we are working within an ambient linear space, a natural cost function to consider is a *norm*: these are convex by definition, and the metric topology that they generate is the usual Euclidean topology (so that the “costs” they assign agree with the usual notion of “closeness” in these spaces). By virtue of (3) and (8a), we might therefore consider the pose-pose measurement error functions:

$$e_{ij}^R(p_i, p_j) = \|R_j - R_i \bar{R}_{ij}\|, \quad (13a)$$

$$e_{ij}^t(p_i, p_j) = \|t_j - (t_i + R_i \bar{t}_{ij})\|, \quad (13b)$$

for $(i, j) \in \mathcal{P}$, where $\|\cdot\|$ denotes any choice of matrix or vector norm in (13a) and (13b), respectively; these functions then serve to quantify the disagreement between the rotational and translational components of the left- and right-hand sides of (8a). Similarly, by virtue of (4) and (8b), we might consider a pose-landmark measurement error function of the form

$$e_{ik}(p_i, l_k) = \|l_k - (t_i + R_i \bar{l}_{ik})\|, \quad (14)$$

for $(i, k) \in \mathcal{L}$, where $\|\cdot\|$ is any vector norm. Finally, in order to implement a robust M-estimation, we might wish to allow

the possibility of composing the error functions in (13) and (14) with robust cost functions $\rho: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ [6]. Thus, we will consider cost functions c_{ij} and c_{ik} of the form

$$c_{ij}(p_i, p_j) = \rho_{ij}^R(\|R_j - R_i \bar{R}_{ij}\|) + \rho_{ij}^t(\|t_j - (t_i + R_i \bar{t}_{ij})\|) \quad (15a)$$

$$c_{ik}(p_i, l_k) = \rho_{ik}(\|l_k - (t_i + R_i \bar{l}_{ik})\|). \quad (15b)$$

2) *Constructing the convex relaxation:* Here we establish our main result: a prescription for constructing a convex relaxation of the SLAM M-estimation program (10).

Lemma 1. *Let \mathcal{C} be a convex set, $f: \mathcal{C} \rightarrow \mathbb{R}_{\geq 0}$ a convex function on \mathcal{C} , and $g: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ a convex and nondecreasing function on the nonnegative real numbers. Then the composition $g \circ f: \mathcal{C} \rightarrow \mathbb{R}_{\geq 0}$ is also convex.*

Proof: Let $x_1, x_2 \in \mathcal{C}$ and $\lambda \in [0, 1]$. Then

$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2) \quad (16)$$

by the convexity of f . Since g is nondecreasing and convex, then (16) implies

$$\begin{aligned} g(f(\lambda x_1 + (1 - \lambda)x_2)) &\leq g(\lambda f(x_1) + (1 - \lambda)f(x_2)) \\ &\leq \lambda g(f(x_1)) + (1 - \lambda)g(f(x_2)) \end{aligned}$$

which is the desired inequality. \square

Theorem 1 (Convex relaxation of the SLAM M-estimation). *Let c_{ij} and c_{ik} be the cost functions defined in (15) with $\rho_{ij}^R, \rho_{ij}^t, \rho_{ik}: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ convex nondecreasing functions for all $(i, j) \in \mathcal{P}$ and $(i, k) \in \mathcal{L}$, and let*

$$C(n) \triangleq \text{conv } SE(n) = \mathbb{R}^n \times \text{conv } SO(n) \quad (17a)$$

$$\tilde{\mathcal{F}} \triangleq \text{conv } \mathcal{F} = (C(n))^{n_p} \times (\mathbb{R}^n)^{n_l} \quad (17b)$$

denote the convex hulls of $SE(n)$ in $V(n)$ and \mathcal{F} in \mathcal{S} , respectively. Then the objective function $f: \mathcal{F} \rightarrow \mathbb{R}_{\geq 0}$ defined in (10) extends to a convex function $\tilde{f}: \tilde{\mathcal{F}} \rightarrow \mathbb{R}_{\geq 0}$, and the relaxation obtained from the SLAM M-estimation (10) by extending \mathcal{F} to $\tilde{\mathcal{F}}$:

$$\begin{aligned} \tilde{X}^* &= \underset{\tilde{X} \in \tilde{\mathcal{F}}}{\text{argmin}} \tilde{f}(\tilde{X}) \\ &\text{subject to } \tilde{p}_1 = (0, I) \end{aligned} \quad (18)$$

is a convex program.

Proof: The functions e_{ij}^R, e_{ij}^t , and e_{ik} defined in equations (13)–(14) are compositions of linear inner functions with convex outer functions (the norms), and are therefore convex (cf. [8, Sec. 3.2]); Lemma 1 thus guarantees that the cost functions $c_{ij} = \rho_{ij}^R \circ e_{ij}^R + \rho_{ij}^t \circ e_{ij}^t$ and $c_{ik} = \rho_{ik} \circ e_{ik}$ defined in (15) are convex, and therefore so is their sum \tilde{f} in (18) (recall (10)). We also observe that the feasible set of (18) is the intersection of the convex set $\tilde{\mathcal{F}}$ in (17b) with the affine set determined by the constraint $\tilde{p}_1 = (0, I)$, and is therefore also convex. Problem (18) is thus a convex program. \square

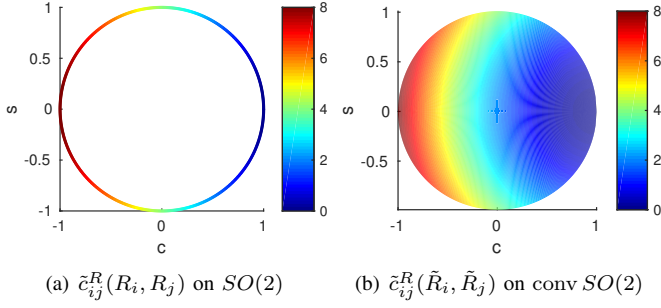


Fig. 2. Constructing the convex relaxation (18). (a): This figure plots the value of the rotational cost function $\tilde{c}_{ij}^R(R_i, R_j)$ defined in (19) as a function of $R_i \in SO(2)$ for $\alpha_{ij} = 1$ and $R_j = \bar{R}_{ij} = I_2$, using the realization $SO(2) = \{ \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \mid c^2 + s^2 = 1 \} \subset \mathbb{R}^{2 \times 2}$ described in (1). (b): The rotational cost $\tilde{c}_{ij}^R(R_i, R_j)$ shown in (a) extends to a convex function $\tilde{c}_{ij}^R(\tilde{R}_i, \tilde{R}_j)$ for $\tilde{R}_i, \tilde{R}_j \in \text{conv } SO(2)$, where the set $\text{conv } SO(2)$ is characterized explicitly by (20). The construction for $SO(3)$ using (19) and (21) is analogous.

D. The relation between the convex relaxation and the standard maximum-likelihood formulation of SLAM

The standard formulation of the SLAM problem (10) derives the cost functions (11) from an assumed Gaussian noise model, whereas our selection of the cost functions (13)–(15) was primarily motivated by a desire to obtain a convex objective for (18). It may therefore not be immediately clear how these two models relate, or whether the objective \tilde{f} in (18) constructed from (13)–(15) carries a natural interpretation in the same way that a negative log-likelihood objective does. To that end, here we briefly describe a *specific* choice of the cost functions (15) that act as *pointwise upper bounds* on the standard negative log-likelihood cost functions (11) for all $X \in \mathcal{F}$. We present the following theorem without proof, in the interest of brevity:

Theorem 2. *For the following choice of cost functions (15):*

$$\begin{aligned} \tilde{c}_{ij}(p_i, p_j) &= \underbrace{\alpha_{ij} \|R_j - R_i \bar{R}_{ij}\|_F^2}_{\tilde{c}_{ij}^R(R_i, R_j)} + \underbrace{\beta_{ij} \|t_j - (t_i + R_i \bar{t}_{ij})\|_2^2}_{\tilde{c}_{ij}^t(p_i, p_j)} \\ \tilde{c}_{ik}(p_i, l_k) &= \gamma_{ij} \|l_k - (t_i + R_i \bar{l}_{ik})\|_2^2 \end{aligned} \quad (19)$$

where $\|\cdot\|_F$ is the Frobenius norm and $\|\cdot\|_2$ is the usual Euclidean norm, it holds that $c_{ij}^G(p_i, p_j) \leq \tilde{c}_{ij}(p_i, p_j)$ and $c_{ik}^G(p_i, l_k) \leq \tilde{c}_{ik}(p_i, l_k)$ for all $p_i, p_j \in SE(3)$ and $l_k \in \mathbb{R}^3$ provided that

$$\alpha_{ij} \geq \frac{\pi^2}{4\lambda_{\min}(\Sigma_{ij}^R)}, \quad \beta_{ij} \geq \frac{1}{\lambda_{\min}(\Sigma_{ij}^t)}, \quad \gamma_{ij} \geq \frac{1}{\lambda_{\min}(\Sigma_{ik})}.$$

The practical import of Theorem 2 is that it shows how to construct a specific class of admissible cost functions (15) for the convex relaxation (18) in such a way that minimizing the convex objective \tilde{f} (as the relaxation (18) attempts to do) still “does the right thing” with respect to the usual Gaussian maximum-likelihood formulation of the SLAM problem. Fig. 2 illustrates the convex relaxation (18) obtained by using the cost functions (19) for the case of mapping in two dimensions.

III. APPROXIMATE GLOBAL OPTIMIZATION IN SLAM USING THE CONVEX RELAXATION

In this section we discuss several practical aspects of solving the convex relaxation (18) and extracting from it an approximate global solution of the original SLAM M-estimation (10).

Our point of departure (in Section III-A) is the observation that the feasible set for the program (18) has a convenient *spectrahedral* description (i.e. it can be expressed as the solution set for a linear matrix inequality [10]). We show in Section III-B how to exploit this description (in the form of log-determinant barrier functions) to enable the implementation of a fast second-order interior-point optimization method [7] to solve the relaxation (18) efficiently in practice, and discuss several of the computational advantages that this approach affords. Finally, having obtained a global solution \tilde{X}^* to the convex relaxation (18) using this interior-point technique, we describe in Section III-C a simple but effective rounding procedure to extract from \tilde{X}^* a good initial estimate \hat{X} for the global minimizer of the original SLAM M-estimation (10).

A. Spectrahedral description of $\text{conv } SO(n)$

In order to *implement* the optimization (18) in practice, it is necessary to have a computational means of testing for membership in the feasible set $\tilde{\mathcal{F}}$ (in particular, of testing for membership in $\text{conv } SO(n)$ in (17a)). Fortunately, a suitable criterion is provided by the following theorem due to Saunderson, Parrilo and Willsky [11]:

Theorem 3 (Spectrahedral description of $\text{conv } SO(n)$). *The convex hull of $SO(n)$ is a spectrahedron for all $n \in \mathbb{N}$. For the special cases $n = 2$ and $n = 3$, the corresponding spectrahedral descriptions are given explicitly by equations (20) and (21):*

$$\text{conv } SO(2) = \left\{ \tilde{R} = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \in \mathbb{R}^{2 \times 2} : \underbrace{\begin{bmatrix} 1+c & s \\ s & 1-c \end{bmatrix}}_{\triangleq A_2(\tilde{R})} \succeq 0 \right\}. \quad (20)$$

By means of Theorem 3, we can rewrite (18) as:

$$\begin{aligned} \tilde{X}^* &= \underset{\tilde{X} \in \mathcal{S}}{\text{argmin}} \tilde{f}(\tilde{X}) \\ \text{subject to } &\tilde{p}_1 = (0, I) \\ &A_n(\tilde{R}_i) \succeq 0 \quad \forall i = 2, \dots, n_p \end{aligned} \quad (22)$$

where A_n is the linear matrix operator defined in (20) or (21).

Program (22) is a constrained convex optimization problem, but in contrast to the usual (scalar) equality and inequality constraints that one most commonly encounters in mathematical programming, the constraints appearing in (22) are *positive semidefiniteness* constraints; they require that the eigenvalues of the (symmetric) matrices $A_n(\tilde{R}_i)$ be nonnegative. As there is in general no closed-form algebraic solution for the eigenvalues of a matrix in terms of its elements, it may not be immediately clear how one can effectively enforce constraints of this sort when designing a numerical optimization method to solve (22) in practice; we describe a suitable technique based upon interior-point methods in the next subsection.

$$\text{conv } SO(3) = \left\{ \tilde{R} \in \mathbb{R}^{3 \times 3} : \underbrace{\begin{bmatrix} 1 - \tilde{R}_{11} - \tilde{R}_{22} + \tilde{R}_{33} & \tilde{R}_{13} + \tilde{R}_{31} & \tilde{R}_{12} - \tilde{R}_{21} & \tilde{R}_{23} + \tilde{R}_{32} \\ \tilde{R}_{13} + \tilde{R}_{31} & 1 + \tilde{R}_{11} - \tilde{R}_{22} - \tilde{R}_{33} & \tilde{R}_{23} - \tilde{R}_{32} & \tilde{R}_{12} + \tilde{R}_{21} \\ \tilde{R}_{12} - \tilde{R}_{21} & \tilde{R}_{23} - \tilde{R}_{32} & 1 + \tilde{R}_{11} + \tilde{R}_{22} + \tilde{R}_{33} & \tilde{R}_{31} - \tilde{R}_{13} \\ \tilde{R}_{23} + \tilde{R}_{32} & \tilde{R}_{12} + \tilde{R}_{21} & \tilde{R}_{31} - \tilde{R}_{13} & 1 - \tilde{R}_{11} + \tilde{R}_{22} - \tilde{R}_{33} \end{bmatrix}}_{\triangleq A_3(\tilde{R})} \succeq 0 \right\} \quad (21)$$

B. Solving the convex relaxation

In this subsection we describe a computationally efficient optimization method for solving large-scale but sparse problems of the form (22) effectively in practice. Our approach is based upon exploiting properties of the class of interior-point methods for nonlinear programming [7] to enable the replacement of each positive semidefiniteness constraint $A_n(\tilde{R}_i) \succeq 0$ in (22) with an inequality constraint of the form $c(\tilde{R}_i) \geq 0$ for a scalar-valued function $c(\cdot)$.

1) *Interior-point methods for nonlinear programming:* Interior-point methods for nonlinear programming aim to solve constrained optimization problems of the form

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} f(x) \quad \text{subject to} \quad \begin{cases} c_i(x) = 0, & i \in \mathcal{E}, \\ c_i(x) \geq 0, & i \in \mathcal{I}, \end{cases} \quad (23)$$

(where $f, c_i: \mathbb{R}^d \rightarrow \mathbb{R}$ are real-valued, continuously-differentiable functions on \mathbb{R}^d and \mathcal{E} and \mathcal{I} are disjoint finite index sets of equality and inequality constraints, respectively) by approximating a solution x^* of (23) using the first component \hat{x} of a solution (\hat{x}, \hat{s}) to the corresponding *logarithmic barrier program*:

$$\begin{aligned} & \underset{(x,s) \in \mathbb{R}^d \times \mathbb{R}^{|\mathcal{I}|}}{\text{minimize}} f(x) - \mu \sum_{i \in \mathcal{I}} \log s_i \\ & \text{subject to} \quad \begin{cases} c_i(x) = 0, & i \in \mathcal{E}, \\ c_i(x) = s_i, & i \in \mathcal{I}, \end{cases} \end{aligned} \quad (24)$$

where $\mu > 0$ is called the *barrier parameter*. Notice that the inequality constraints in the original nonlinear program (23) have been replaced in (24) by *equalities* involving the slack variables $s \in \mathbb{R}^{|\mathcal{I}|}$. While this requires that $\hat{s}_i \geq 0$ for all $i \in \mathcal{I}$ in order for \hat{x} to be feasible in (23), observe that we needn't *explicitly* enforce this condition in (24); the fact that $\lim_{s_i \rightarrow 0^+} -\log s_i = +\infty$ together with the second set of equality constraints in (24) prevents the solutions \hat{x} from encroaching on the boundary of the feasible set of (23). In this way, the logarithmic terms in the objective in (24) act as a “barrier” that serves to keep the estimates \hat{x} in the interior of the feasible set of the original program. While the estimate \hat{x} arising from (24) will thus generally not coincide exactly with a solution x^* of (23), one can show that $\hat{x} \rightarrow x^*$ as $\mu \rightarrow 0^+$. Thus, in this approach the strategy is to solve a *sequence* of barrier problems of the form (24) for decreasing values of μ ; since each of these barrier programs is an *equality*-constrained optimization, it can in turn be solved directly using the usual method of Lagrange multipliers [7].

2) *Enforcing positive semidefiniteness constraints with interior-point methods:* Given a symmetric matrix $S \in \mathbb{R}^{n \times n}$, the condition that $S \succeq 0$ is equivalent to the condition that each of S 's (real) eigenvalues is nonnegative: $\lambda_i(S) \geq 0$ for $i = 1, \dots, n$. Now although there is in general no closed-form algebraic solution to compute the eigenvalues $\lambda_i(S)$ as a function of S 's elements, it is nevertheless still possible to enforce the nonnegativity conditions $\lambda_i(S) \geq 0$ when performing numerical optimization with an interior-point method by enforcing a clever (scalar) inequality constraint of the form $c(S) \geq 0$.

Recall that the determinant of a matrix is the product of that matrix's eigenvalues:

$$\det(S) = \prod_{i=1}^n \lambda_i(S). \quad (25)$$

Consider enforcing a constraint of the form $\det(S) \geq 0$ when applying an interior-point method; the barrier term in the corresponding logarithmic barrier program (24) is then:

$$-\mu \log \det(S) = -\mu \log \left(\prod_{i=1}^n \lambda_i(S) \right) = -\mu \sum_{i=1}^n \log \lambda_i(S). \quad (26)$$

But now observe that the right-hand side of (26) is precisely the barrier term associated with the system of inequalities:

$$\lambda_i(S) \geq 0 \quad \forall i = 1, \dots, n. \quad (27)$$

Equation (26) thus implies that a positive semidefiniteness constraint of the form $S \succeq 0$ can be replaced with a scalar constraint of the form $\det(S) \geq 0$ when solving nonlinear programs using an interior-point method, provided that the algorithm is initialized with a strictly feasible starting point $S^{(0)} \succ 0$; in that case, the fact that $\lambda_i(S^{(0)}) > 0$ for all $i = 1, \dots, n$ by construction, together with the presence of the barrier term $-\mu \log \det(S)$ in the logarithmic barrier program (24) (which prevents $\det(S^{(k)}) \rightarrow 0$, i.e., prevents any of the $\lambda_i(S^{(k)})$ from changing sign during the optimization), ensures that every iterate $S^{(k)}$ likewise satisfies $S^{(k)} \succeq 0$. (As an aside, this observation forms the basis for the class of *central path-following* interior-point methods for semidefinite programming; cf. [10, Sec. 4].)

3) *Practical implementation of the optimization:* In light of (23)–(27), we can solve (22) by applying an interior-point method to the program

$$\begin{aligned} \tilde{X}^* &= \underset{\tilde{X} \in \mathcal{S}}{\text{argmin}} \tilde{f}(\tilde{X}) \\ & \text{subject to } \tilde{p}_1 = (0, I) \\ & \det(A_n(\tilde{R}_i)) \geq 0 \quad \forall i = 2, \dots, n_p \end{aligned} \quad (28)$$

provided that the algorithm is initialized with a strictly feasible starting point (i.e. a point $\tilde{X}^{(0)}$ for which $\tilde{p}_1 = (0, I)$ and $A_n(\tilde{R}_i^{(0)}) \succ 0$ for $i = 2, \dots, n$). This is the approach that we will implement to solve (18) in practice.

4) *Computational considerations*: To close this subsection, we briefly highlight some of the attractive computational properties of the optimization approach described above.

Interior-point methods are a state-of-the-art class of accurate, high-speed techniques for solving large-scale nonlinear programs, and enjoy excellent global convergence and numerical robustness properties [7], [12]. Furthermore, the fact that the constraints in (28) involve only unary functions of the generalized orientations \tilde{R}_i means that the block-sparsity pattern of the Hessian of the Lagrangian of the barrier programs (24) coincides (after substitution to eliminate the slack variables) with that of the Hessian of $f(X)$ in (10) alone. Consequently, we expect the computational complexity of solving (28) using an interior-point method based upon the exact (i.e. second-order) Newton step (cf. [13], [14], among others) will scale gracefully to problem sizes typical of those addressed by state-of-the-art least-squares methods for SLAM [3]–[5]. Solving program (28) using interior-point methods thus provides a numerically robust and computationally efficient approach to obtain solutions \tilde{X}^* of the convex relaxation (18).

C. Rounding the solution of the relaxed program

The convex relaxation (18) is obtained from the original M-estimation (10) by *expanding* the original feasible set to its convex hull (more precisely, by expanding the feasible set $SO(n)$ for the pose orientation estimates to its convex hull $\text{conv } SO(n)$, as shown in Fig. 2). While this is desirable for the computational advantages that convex programming affords, it also means that the solution \tilde{X}^* of (18) will generally not lie in the feasible set for the original program (10); consequently, we must provide a method for transforming a solution \tilde{X}^* of (18) into a feasible point for (10), a procedure known as *rounding*.

In this case, since the objective functions for the original program (10) and its convex relaxation (18) are actually identical, we would like to design a rounding method that perturbs the solution \tilde{X}^* of (18) as little as is possible (in some sense) in order to restore feasibility in (10). Thus, one natural way to round \tilde{X}^* is by replacing each generalized orientation estimate $\tilde{R}_i^* \in \text{conv } SO(n)$ with the nearest (in some sense) valid rotation matrix; i.e., we first define a rounding procedure

$$\begin{aligned} \pi_R: \text{conv } SO(n) &\rightarrow SO(n) \\ \pi_R(\tilde{R}) &\in \underset{R \in SO(n)}{\text{argmin}} \|R - \tilde{R}\| \end{aligned} \quad (29)$$

that sends each \tilde{R}_i^* to a closest rotation matrix in some norm $\|\cdot\|$, and then define a rounding procedure $\pi: \tilde{\mathcal{F}} \rightarrow \mathcal{F}$ for \tilde{X}^* that simply fixes all translational estimates and sends each generalized orientation estimate \tilde{R}_i to one of its nearest rotation matrices $\pi_R(\tilde{R}_i)$:

$$\pi(\tilde{p}_1, \dots, \tilde{p}_{n_p}, \tilde{l}_1, \dots, \tilde{l}_{n_l}) = (\hat{p}_1, \dots, \hat{p}_{n_p}, \tilde{l}_1, \dots, \tilde{l}_{n_l}) \quad (30)$$

where

$$\hat{p}_i = (\tilde{t}_i, \pi_R(\tilde{R}_i)) \quad \forall i \in [n_p]. \quad (31)$$

Rounding procedures of the form (29) have been studied previously, and there exist straightforward and efficient methods to compute the mapping $\pi_R(\cdot)$ (based upon the singular value decomposition [15] of \tilde{R}) for the special case in which the norm $\|\cdot\|$ in (29) is the Frobenius norm [16].

IV. RELATED WORK

The formulation of the SLAM inference problem as an instance of the M-estimation (10) is originally due to Lu & Milios [17], who proposed the use of nonconvex negative log-likelihood cost functions similar to (11); this formulation remains the basis for current state-of-the-art SLAM techniques (e.g. [2]–[5]), with most modern algorithms differing principally only in how the optimization problem (10) is solved.

The most common approach is to use an approximate Newton least-squares method (as in [3]–[5]) or gradient descent (as in [2]) to estimate a local minimizer of (10); however, as these methods only guarantee convergence to *local* minima, the performance of this approach depends crucially upon initializing the back-end optimization algorithm near a good solution for the inference problem. While there has been some prior work focused specifically on improving the robustness of local search methods for optimization in SLAM with respect to poor initialization (e.g. the development of improved initialization procedures [18] or reparameterizations of the state space intended to broaden the basins of attraction of good solutions [2], [19]), ultimately these methods still depend upon identifying a good initialization for some nonconvex optimization using either random sampling or a greedy kinematic expansion along the edges of a spanning tree through the network of measurements.

Alternatively, some recent work aims to avoid the brittleness of smooth local optimization methods with respect to poor initialization through the use of convex relaxations in a spirit similar to our own. One notable example is [20], which guarantees the recovery of the *globally* optimal orientation estimates in the 2D pose-graph SLAM problem with high (user-selectable) probability; however, this approach depends upon the identification $SO(2) \cong \mathbb{R}/2\pi\mathbb{Z}$, and so appears to be limited to the 2-dimensional case. Another recent method similar to our own is [21], which formulates a specific instance of the convex relaxation (10) to solve the pointcloud registration problem; our approach can be thought of as an extension of this work from the special case of single-pose estimation based on pose-landmark observations to the more general pose-and-landmark SLAM problem.

In practice, the convex relaxation (18) can be thought of as an advanced initialization procedure that improves upon prior work by fusing information from *all* of the available measurements Z to produce \tilde{X}^* (rather than only a subset corresponding to the edges of some spanning tree in the measurement network). We thus expect our approach to be particularly advantageous versus prior techniques in cases

where the accumulated uncertainty along any single path through the network of measurements is high (e.g. high-noise scenarios or networks with deep spanning trees).

V. EXPERIMENTAL RESULTS

In this section we illustrate the performance of the approximate global optimization method of Section III on two classes of standard pose-graph SLAM benchmarks, using the performance of the standard least-squares approach as a baseline for comparison.

Our test sets for these experiments consist of 100 randomly-sampled instances of the City and sphere2500 datasets. We process each dataset twice: once using the usual least-squares SLAM formulation, and a second time using our two-stage procedure, in which we first find a minimizer \tilde{X}^* of the convex relaxation (18) (using the cost functions defined in (19)) by solving (28), compute the rounded estimate $\hat{X} = \pi(\tilde{X}^*)$, and then use \hat{X} to initialize a second-stage local refinement using the standard least-squares SLAM formulation. In both cases we initialize the first-stage optimization methods using the odometric initialization.

All experiments were run on a desktop with Intel Xeon X5660 2.80 GHz processor. The convex optimization (28) was implemented in MATLAB using the `fmincon` interior-point method (which is itself an implementation of `KNITRO`, a high-quality interior-point trust-region method for large-scale nonlinear programming [13], [14]). The least-squares minimizations in both cases were performed using the implementation of Levenberg-Marquardt available in the GTSAM library³ with the default settings, so any differences in the quality of the final estimates produced by these two approaches are due solely to the effect of refining the initialization using the convex relaxation (18).

A. Datasets

1) *City datasets*: In this experiment we evaluate the two approaches on the City problem, which simulates a robot traversing a 2D “grid world”; like the well-known Manhattan world [19], this problem is designed to be challenging for local optimization methods to solve given poor initial estimates (for example, those obtained by composing long chains of odometric measurements). Our test ensemble consists of 100 randomly-sampled problem instances, each with 3908–4285 poses and 7894–10195 pose-pose measurements. Results from this experiment are summarized in Table I; a representative instance from the test set is shown in Fig. 3.

2) *sphere2500 datasets*: We next evaluate the two methods on a high-noise version of the sphere2500 problem, a standard 3D pose-graph SLAM benchmark. Our test ensemble consists of 100 randomly-sampled problem instances, each with 2500 poses and 4949 pose-pose measurements. Results from this experiment are summarized in Table II; a representative instance from the test set is shown in Fig. 3.

³The GTSAM Library (version 2.1.0), available through <https://research.cc.gatech.edu/borg/sites/edu.borg/files/downloads/gtsam-2.1.0.tgz>

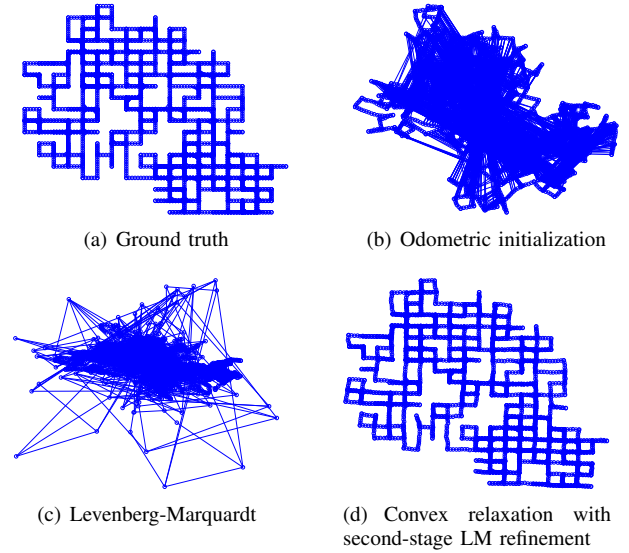


Fig. 3. A representative instance of the City datasets (4269 poses, 8861 measurements). (a): The ground truth map. (b): The odometric initialization for this example (objective function value 1.238 E8). (c): The solution obtained by Levenberg-Marquardt using the odometric initialization (6.887 E7). (d): The solution obtained using the two-stage procedure of Section III (6.864 E3); this is the same solution obtained when initializing Levenberg-Marquardt with the ground truth (a). Note that the objective function values for both (c) and (d) are within $\pm 6\%$ of the median values for the corresponding methods reported in Table I.

B. Discussion

We can see from Tables I and II that the convex relaxation approach significantly outperformed the standard least-squares approach on these examples; evidently the accumulated errors in the odometric initializations were sufficient to prevent the local search performed by Levenberg-Marquardt from recovering good solutions to the inference problem in most cases. In contrast, the two-stage convex relaxation approach consistently finds good solutions to the inference task; indeed, the fact that (18) is convex implies that the solution $\hat{X} = \pi(\tilde{X}^*)$ used to initialize the second stage refinement is *completely immune* to the effects of the accumulated error in the odometric (or any other) initialization for (18).

Of course, the enhanced performance of the two-stage approach of Section III versus the standard least-squares approach comes at the expense of additional computation (specifically, the additional computation needed to solve program (28) before applying the second-stage refinement). However, the timing results in Tables I and II show that this additional overhead is not a serious limitation to the effective use of the method (in agreement with the analysis of Section III-B3), and is well-compensated by the gain in solution quality.

VI. CONCLUSION

In this paper we proposed a practical method for approximating the globally optimal solution of the SLAM inference problem that does *not* require initialization with a high-quality estimate. Our approach is based upon a novel formulation of the SLAM M-estimation with the property that, by expanding the feasible set of the estimation program, we obtain a *convex*

	Levenberg-Marquardt			Convex relaxation with LM refinement		
	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.
Objective function value	1.839 E8	6.880 E7	2.902 E8	7.318 E3	7.247 E3	6.917 E2
Computation time (sec)	6.437	3.475	6.968	3.047 E1	3.085 E1	2.850 E1

TABLE I
SUMMARY OF RESULTS FOR CITY DATASETS

	Levenberg-Marquardt			Convex relaxation with LM refinement		
	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.
Objective function value	3.711 E7	3.451 E7	2.115 E7	7.370 E3	7.370 E3	7.853 E1
Computation time (sec)	3.527	3.316	1.176	2.197 E1	2.126 E1	1.210

TABLE II
SUMMARY OF RESULTS FOR SPHERE2500 DATASETS

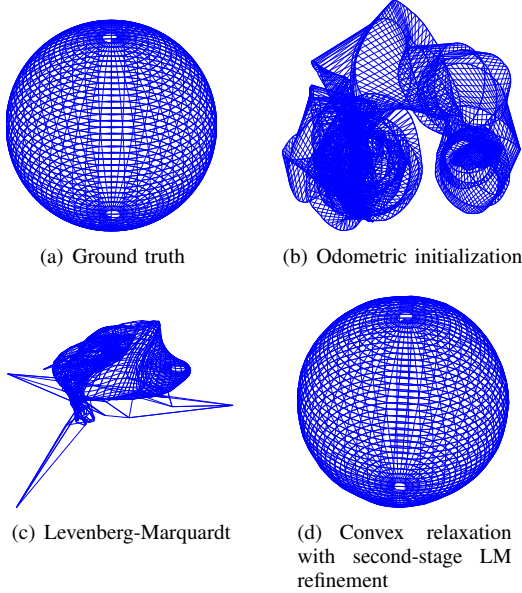


Fig. 4. A representative instance of the sphere2500 datasets. (a): The ground truth map. (b): The odometric initialization for this example (objective function value 1.318 E8). (c): The solution obtained by Levenberg-Marquardt using the odometric initialization (3.420 E7). (d): The solution obtained using the two-stage procedure of Section III (7.346 E3); this is the same solution obtained when initializing Levenberg-Marquardt with the ground truth (a). Note that the objective function values for both (c) and (d) are within $\pm 1\%$ of the median values for the corresponding methods reported in Table II.

relaxation whose solution approximates the optimal solution of the SLAM problem and can be recovered using a smooth optimization method initialized at *any* feasible point. By projecting the solution of this relaxation onto the feasible set of the original program, we obtain a good estimate of the optimal solution to the original M-estimation problem, which can itself be taken as an initialization for a second-stage local refinement using the standard least-squares approach. Our approach thus enables the efficient recovery of a high-quality map without requiring high-quality initialization.

ACKNOWLEDGEMENTS

This work was partially supported by a software engineering internship at Google, by the Office of Naval Research under grants N00014-10-1-0936, N00014-11-1-0688 and N00014-13-1-0588, and by NSF Award IIS-1318392, which we gratefully acknowledge. The authors would also like to thank Julian Mason and Kurt Konolige for numerous helpful discussions.

REFERENCES

- [1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: The MIT Press, 2008.
- [2] G. Grisetti, D. Rizzini, C. Stachniss, E. Olson, and W. Burgard, "Online constraint network optimization for efficient maximum likelihood map learning," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Pasadena, CA, May 2008, pp. 1880–1885.
- [3] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, May 2011, pp. 3607–3613.
- [4] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *Intl. J. of Robotics Research*, vol. 31, no. 2, pp. 216–235, Feb. 2012.
- [5] D. Rosen, M. Kaess, and J. Leonard, "RISE: An incremental trust-region method for robust online sparse least-squares estimation," *IEEE Trans. Robotics*, vol. 30, no. 5, pp. 1091–1108, Oct. 2014.
- [6] P. Huber, *Robust Statistics*. Hoboken, N.J.: John Wiley & Sons, 2004.
- [7] J. Nocedal and S. Wright, *Numerical Optimization*, 2nd ed. New York: Springer Science+Business Media, 2006.
- [8] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [9] F. Dellaert and M. Kaess, "Square Root SAM: Simultaneous localization and mapping via square root information smoothing," *Intl. J. of Robotics Research*, vol. 25, no. 12, pp. 1181–1203, Dec. 2006.
- [10] V. Vandenberghe and Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, Mar. 1996.
- [11] J. Saunderson, P. Parrilo, and A. Willsky, "Semidefinite descriptions of the convex hull of rotation matrices," *arXiv preprint arXiv:1403.4914*, 2014.
- [12] A. Forsgren, P. E. Gill, and M. H. Wright, "Interior methods for nonlinear optimization," *SIAM Review*, vol. 44, no. 4, pp. 525–597, 2002.
- [13] R. A. Waltz, J. L. Morales, J. Nocedal, and D. Orban, "An interior algorithm for nonlinear optimization that combines line search and trust region steps," *Math. Program.*, vol. 107, no. 3, pp. 391–408, 2006.
- [14] R. H. Byrd, J. Nocedal, and R. A. Waltz, "KNITRO: An integrated package for nonlinear optimization," in *Large-Scale Nonlinear Optimization*, G. di Pillo and M. Roma, Eds. Springer, 2006, pp. 35–59.
- [15] G. Golub and C. V. Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins University Press, 1996.
- [16] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.
- [17] F. Lu and E. Milios, "Globally consistent range scan alignment for environmental mapping," *Autonomous Robots*, vol. 4, pp. 333–349, Apr. 1997.
- [18] M. Pfingsthorn and A. Birk, "Simultaneous localization and mapping with multimodal probability distributions," *Intl. J. of Robotics Research*, vol. 32, no. 2, pp. 143–171, 2013.
- [19] E. Olson, J. Leonard, and S. Teller, "Fast iterative alignment of pose graphs with poor initial estimates," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Orlando, FL, May 2006, pp. 2262–2269.
- [20] L. Carlone and A. Censi, "From angular manifolds to the integer lattice: Guaranteed orientation estimation with application to pose graph optimization," *IEEE Trans. Robotics*, vol. 30, no. 2, pp. 475–492, Apr. 2014.
- [21] M. Horowitz, N. Matni, and J. Burdick, "Convex relaxations of SE(2) and SE(3) for visual pose estimation," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Hong Kong, China, Jun. 2014.