On the Inclusion of Determinant Constraints in Lagrangian Duality for 3D SLAM

Roberto Tron*, David M. Rosen†, Luca Carlone‡
*Department of Computer and Information Science, University of Pennsylvania, USA. Email: tron@seas.upenn.edu
†Massachusetts Institute of Technology, Cambridge, MA, USA. Email: dmrosen@mit.edu
‡Georgia Institute of Technology, Atlanta, GA, USA. Email: luca.carlone@gatech.edu

***Abstract*—A recent paper in 3D Pose Graph Optimization (PGO) shows how a dual Lagrangian formulation of the problem can be used to verify (and possibly certify) the quality of a given solution [1]. A limitation of this approach is that, in the derivation, the authors relax the positive determinant constraint for the rotations. As a consequence, when the approach fails to certify an optimal solution (i.e., when the duality gap is non-zero), one cannot determine if this is due to the relaxation or if it is an intrinsic feature of the problem at hand.**

In this paper we extend the results of [1] by including the determinant constraints in the derivation of the dual, thus showing that their relaxation is unnecessary. We show experimentally that this complete formulation does not lead to tangible differences with respect to the original, relaxed version. This indicates that the reasons for failures in providing a certificate of optimality are intrinsic to the problem, and that the determinant constraints are somehow redundant in common PGO instances.

I. Introduction

Pose Graph Optimization (PGO) is one of the most popular formulations of the SLAM problem. The goal of PGO is to estimate a set of poses (sampled along the trajectory of a mobile robot) from relative measurements. This is done by solving a non-convex optimization problem, where the non-convexity is mainly due to the presence of unknown rotations.

Despite the empirical success of state-of-the-art techniques for PGO (see [1] for a literature review) current approaches are not able to guarantee the computation of a global optimal solution in general problem instances. This is due to the non-convexity of PGO, which implies that iterative optimization techniques can be trapped in local minima, resulting in wrong pose estimates. We argue that the transition of SLAM from research topic to industrial technology requires techniques with *guaranteed performance*. For instance, in autonomous vehicles, failure to produce a correct SLAM solution may put passengers' lives at risk. In other applications, SLAM failures can possibly cascade into path planning failures, preventing the reliable operation of mobile robots.

Driven by this motivation, previous works [1–3] use Lagrangian duality as a tool to obtain clear performance guarantees. The dual problem can be used for "verification" purposes [1, 2], i.e., given a PGO estimate (e.g., returned by an iterative solver), we can use the (convex) dual problem to evaluate the quality of this estimate and possibly certify its optimality. Moreover, the dual problem enables the computation of a guaranteed optimal solution in particular cases (when the duality gap is zero) [1, 2]. This second use currently seems less appealing, since it requires solving a large semidefinite program (SDP); despite the fact that SDPs are convex problems, current solvers do not scale well and prevent real time operation. On the other hand, the verification can be done without solving the SDP [1], and this makes duality appealing from a practical standpoint.

The present paper provides an extension of the formulation in [1], which derives a dual formulation for 3D PGO by relaxing the constraints that rotations have positive unit determinant. Effectively, this changes the domain of the optimization problem from rotation matrices (belonging to the group $SO(3)$) to orthogonal matrices (belonging to the group $\mathrm{O}(3)$). In this paper we demonstrate that this relaxation is unnecessary, and we show how to include the determinant constraints in the derivation of the dual (Sections III-IV). Moreover, we present a short summary of the results of [1] (Section V), omitting the proofs. We conclude the paper with an experimental comparison between the dual problem of [1] and the extended version proposed in this addendum (Section VI). Our Monte Carlo analysis shows that including the determinant constraints does not significantly impact the results, suggesting that these constraints are redundant in common PGO instances. We provide an intuitive explanation for this empirical observation in Section VII, and we draw conclusions in Section VIII.

II. 3D Pose Graph Optimization

PGO computes the *maximum likelihood* estimate for n poses $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n$, given m relative pose measurements $\bar{\boldsymbol{x}}_{ij}$ between pairs of poses i and j. In a 3D setup, both the unknown poses and the measurements are quantities in $\mathrm{SE}(3) \doteq \{(\boldsymbol{R},\boldsymbol{t}) : \boldsymbol{R} \in \mathrm{SO}(3), \boldsymbol{t} \in \mathbb{R}^3\}$. We use the notation $\boldsymbol{x}_i = (\boldsymbol{R}_i, \boldsymbol{t}_i)$ and $\bar{\boldsymbol{x}}_{ij} = (\bar{\boldsymbol{R}}_{ij}, \bar{\boldsymbol{t}}_{ij})$ to make explicit the rotation and the translation of each pose. PGO can be visualized as a directed graph $\mathcal{G}(\mathcal{V},\mathcal{E})$, in which we associate a node $i \in \mathcal{V} = \{1,\ldots,n\}$ to each pose $\boldsymbol{x}_i$ and an edge $(i,j) \in \mathcal{E}$ to each relative measurement $\bar{\boldsymbol{x}}_{ij}$.

We consider the following PGO formulation:

$$f^\star = \min_{\substack{\{\boldsymbol{t}_i \in \mathbb{R}^3\} \\ \{\boldsymbol{R}_i \in \mathrm{SO}(3)\}}} \sum_{(i,j)\in\mathcal{E}} \omega_t^2 \left\| \boldsymbol{t}_j - \boldsymbol{t}_i - \boldsymbol{R}_i \bar{\boldsymbol{t}}_{ij} \right\|^2 + \frac{\omega_R^2}{2} \left\| \boldsymbol{R}_j - \boldsymbol{R}_i \bar{\boldsymbol{R}}_{ij} \right\|_{\mathrm{F}}^2 \qquad \text{(PGO)} \quad (1)$$

in which we compute the translations $\{\boldsymbol{t}_i \in \mathbb{R}^3\}$ and the rotations $\{\boldsymbol{R}_i \in \mathrm{SO}(3)\}$ by minimizing the residual errors with respect to the given measurements $(\bar{\boldsymbol{R}}_{ij}, \bar{\boldsymbol{t}}_{ij}), \forall (i,j) \in \mathcal{E}$. In (1), $\|\cdot\|_{\mathrm{F}}^2$ is the (squared) Frobenius matrix norm (sum of the squares of the entries), and ω_t^2, ω_R^2 are the inverse of the translation and rotation measurement covariances, which are assumed isotropic (the derivation of the maximum likelihood estimator is given in [1]). The norm $\|\boldsymbol{R}_a - \boldsymbol{R}_b\|_{\mathrm{F}}^2$ is usually referred to as the *chordal distance* between two rotations $\boldsymbol{R}_a$ and $\boldsymbol{R}_b$ [4]. The main difference between (1) and formulations in related work is the use of the chordal distance in place of the commonly used *geodesic distance*. The chordal distance has been already proposed in a SLAM context in [5, 6]

The advantage of the formulation (1) is that it has a quadratic objective function. This facilitates the derivation of the Lagrangian dual problem (Section IV). Before presenting the dual, we reformulate (1) in a more convenient form. This is done in the following section.

III. 3D SLAM AS A QUADRATIC PROBLEM WITH QUADRATIC EQUALITY CONSTRAINS

In this section, we rewrite (1) in order to (i) have vector variables (the rotations $\boldsymbol{R}_i$ are matrices), (ii) express the constraints $\boldsymbol{R}_i \in \mathrm{SO}(3)$ as quadratic equality constraints, (iii) anchor one of the poses to the origin of the reference frame (this is standard in PGO solvers). This reformulation makes the derivation of the dual problem straightforward.

A. Vectorization

We define $\boldsymbol{r}_i \in \mathbb{R}^9$ as the vectorized version of $\boldsymbol{R}_i$: $\boldsymbol{r}_i \doteq [\boldsymbol{R}_i^{(1)} \ \ \boldsymbol{R}_i^{(2)} \ \ \boldsymbol{R}_i^{(3)}]^\mathsf{T}$, where $\boldsymbol{R}_i^{(k)}$ is the kth *row* of $\boldsymbol{R}_i$. We use the shorthand $\boldsymbol{r}_i = \mathrm{rows}(\boldsymbol{R}_i)$ to obtain the vector representation $\boldsymbol{r}_i$ of a 3×3 matrix $\boldsymbol{R}_i$. Using this parametrization, each summand in the objective in (1) becomes (using $\|\boldsymbol{R}\|_{\mathrm{F}} = \|\boldsymbol{R}^\mathsf{T}\|_{\mathrm{F}}$ in the first expression):

$$\begin{aligned} & \omega_t^2 \left\|\boldsymbol{t}_j - \boldsymbol{t}_i - \boldsymbol{R}_i \bar{\boldsymbol{t}}_{ij}\right\|^2 + \frac{\omega_R^2}{2} \left\|\boldsymbol{R}_j^\mathsf{T} - \bar{\boldsymbol{R}}_{ij}^\mathsf{T} \boldsymbol{R}_i^\mathsf{T}\right\|_{\mathrm{F}}^2 \\ = \quad & \omega_t^2 \left\|\boldsymbol{t}_j - \boldsymbol{t}_i - \boldsymbol{T}_{ij} \boldsymbol{r}_i\right\|^2 + \frac{\omega_R^2}{2} \left\|\boldsymbol{r}_j - \boldsymbol{Q}_{ij} \boldsymbol{r}_i\right\|^2 \end{aligned} \tag{2}$$

where $\boldsymbol{T}_{ij} \doteq \mathbf{I}_3 \otimes \bar{\boldsymbol{t}}_{ij}^\mathsf{T} \in \mathbb{R}^{3 \times 9}$, $\boldsymbol{Q}_{ij} \doteq \mathbf{I}_3 \otimes \bar{\boldsymbol{R}}_{ij}^\mathsf{T} \in \mathbb{R}^{9 \times 9}$, and $\otimes$ is the Kronecker product.

B. Constraints in PGO

We cannot choose arbitrary vectors $\boldsymbol{r}_i \in \mathbb{R}^9$, but have to limit ourself to choices of $\boldsymbol{r}_i$ that produce meaningful rows of a rotation matrix $\boldsymbol{R}_i \in \mathrm{SO}(3)$. The rotation group $\mathrm{SO}(3)$ is defined as $\mathrm{SO}(3) \doteq \{\boldsymbol{R} \in \mathbb{R}^{3\times 3} : \boldsymbol{R}^\mathsf{T}\boldsymbol{R} = \mathbf{I}_3, \det(\boldsymbol{R}) = 1\}$, which, written in terms of the rows of $\boldsymbol{R}_i$, becomes:

$$\boldsymbol{R}_i^\mathsf{T}\boldsymbol{R}_i = \mathbf{I}_3 \Leftrightarrow (\boldsymbol{R}_i^{(u)})^\mathsf{T}\boldsymbol{R}_i^{(v)} = \begin{cases} 1 & \text{if } u = v, \ \ u,v = \\ 0 & \text{if } u \neq v, \ \ 1,2,3 \end{cases} \tag{3}$$

$$\det(\boldsymbol{R}_i) = 1 \Leftrightarrow \boldsymbol{R}_i^{(1)} \times \boldsymbol{R}_i^{(2)} = \boldsymbol{R}_i^{(3)} \tag{4}$$

where $\times$ denotes the cross product. In other words, the rows of a rotation matrix have to be orthonormal (3), and have to satisfy the right-hand rule (4). The orthogonality constraints (3) can be written in terms of the vector $\boldsymbol{r}_i$:

$$(\boldsymbol{R}_i^{(u)})^\mathsf{T}\boldsymbol{R}_i^{(v)} = 0 \Leftrightarrow \boldsymbol{r}_i^\mathsf{T} E_{uv} \boldsymbol{r}_i = 0 \qquad u \neq v, \tag{5}$$

$$(\boldsymbol{R}_i^{(u)})^\mathsf{T}\boldsymbol{R}_i^{(u)} = 1 \Leftrightarrow \boldsymbol{r}_i^\mathsf{T} E_{uu} \boldsymbol{r}_i = 1 \tag{6}$$

where E_{uv} is a 9×9 selection matrix composed of 3×3 blocks that are zero everywhere except the 3×3 block in position (u, v), which is the identity matrix. To facilitate the following derivation, we include a slack variable y and a constraint:

$$y^2 = 1 \tag{7}$$

which allows writing (6) equivalently as:

$$\boldsymbol{r}_i^\mathsf{T} E_{uu} \boldsymbol{r}_i = 1 \Leftrightarrow \boldsymbol{r}_i^\mathsf{T} E_{uu} \boldsymbol{r}_i - y^2 = 0 \tag{8}$$

(the derivation is easier if most constraints are homogeneous).

While in the previous work [1] the authors relaxed the determinant constraint (4), in this paper we show that it is actually possible to include it in the derivation of the dual problem. For this purpose, we define $\boldsymbol{e}_u \in \mathbb{R}^3$ as the vector that is zero everywhere except the u-th entry which is 1, and we use the following equalities (note that $\boldsymbol{R}_i^{(k)}$ are treated as column vectors):

$$\begin{aligned} \boldsymbol{R}_i^{(1)} \times \boldsymbol{R}_i^{(2)} = \boldsymbol{R}_i^{(3)} &\Leftrightarrow \boldsymbol{e}_u^\mathsf{T}\boldsymbol{R}_i^{(1)} \times \boldsymbol{R}_i^{(2)} = \boldsymbol{e}_u^\mathsf{T}\boldsymbol{R}_i^{(3)}, \ \ {\scriptstyle u=1,2,3} \\ (\text{using } a \times b = [a]_\times b) &\Leftrightarrow \boldsymbol{e}_u^\mathsf{T}[\boldsymbol{R}_i^{(1)}]_\times \boldsymbol{R}_i^{(2)} = \boldsymbol{e}_u^\mathsf{T}\boldsymbol{R}_i^{(3)}, \\ (\text{using } [a]_\times b = -[b]_\times a) &\Leftrightarrow -(\boldsymbol{R}_i^{(1)})^\mathsf{T}[\boldsymbol{e}_u]_\times \boldsymbol{R}_i^{(2)} = \boldsymbol{e}_u^\mathsf{T}\boldsymbol{R}_i^{(3)}, \\ (\text{including } y) &\Leftrightarrow -(\boldsymbol{R}_i^{(1)})^\mathsf{T}[\boldsymbol{e}_u]_\times \boldsymbol{R}_i^{(2)} = y\boldsymbol{e}_u^\mathsf{T}\boldsymbol{R}_i^{(3)}, \\ (\text{using } r_i) &\Leftrightarrow \boldsymbol{r}_i^\mathsf{T}\boldsymbol{S}_u\boldsymbol{r}_i + y\boldsymbol{e}_{6+u}^\mathsf{T}\boldsymbol{r}_i = 0, \end{aligned} \tag{9}$$

where we assumed that $y = 1$ (we will comment more on this in Section III-D), $[\boldsymbol{a}]_\times \in \mathbb{R}^{3\times 3}$ is a skew-symmetric matrix built from the entries of a vector $\boldsymbol{a} \in \mathbb{R}^3$, and where $\boldsymbol{S}_u$ is a 9×9 selection matrix composed of 3×3 blocks that are zero everywhere except the 3×3 block in position $(1, 2)$, which is $[\boldsymbol{e}_u]_\times$; moreover, since the last three entries of $\boldsymbol{r}_i$ coincide with $\boldsymbol{R}_i^{(3)}$, in (9) we also used $\boldsymbol{e}_u^\mathsf{T}\boldsymbol{R}_i^{(3)} = \boldsymbol{e}_{6+u}^\mathsf{T}\boldsymbol{r}_i$.

Combining the objective function (2) and the constraints (5), (7), (8), (9), we rewrite the PGO problem (1) as:

$$\begin{aligned} f^\star = \min_{\{\boldsymbol{r}_i, \boldsymbol{t}_i\}} \quad & \sum_{(i,j)\in\mathcal{E}} \omega_t^2 \left\|\boldsymbol{t}_j - \boldsymbol{t}_i - \boldsymbol{T}_{ij}\boldsymbol{r}_i\right\|^2 + \frac{\omega_R^2}{2}\left\|\boldsymbol{r}_j - \boldsymbol{Q}_{ij}\boldsymbol{r}_i\right\|^2 \\ \text{subject to} \quad & \left.\begin{array}{lll} \boldsymbol{r}_i^\mathsf{T}\boldsymbol{E}_{uv}\boldsymbol{r}_i - y^2 & = 0, & u = v \\ \boldsymbol{r}_i^\mathsf{T}\boldsymbol{E}_{uv}\boldsymbol{r}_i & = 0, & u \neq v \\ \boldsymbol{r}_i^\mathsf{T}\boldsymbol{S}_u\boldsymbol{r}_i + y\boldsymbol{e}_{6+u}^\mathsf{T}\boldsymbol{r}_i & = 0 & \end{array}\right\} {\scriptstyle u,v=1,2,3 \atop i=1,\ldots,n} \\ & y^2 = 1 \end{aligned} \tag{10}$$

In order to write (10) in a more compact matrix notation, we define the vector $\breve{\boldsymbol{x}} = [\boldsymbol{t}_1^\mathsf{T}, \ldots, \boldsymbol{t}_n^\mathsf{T}, \boldsymbol{r}_1^\mathsf{T}, \ldots, \boldsymbol{r}_n^\mathsf{T}, y]^\mathsf{T} \in \mathbb{R}^{12n+1}$. Using this notation, (10) becomes:

$$\begin{aligned} f^\star = \min_{\breve{\boldsymbol{x}}} \quad & \|\breve{\boldsymbol{A}}\breve{\boldsymbol{x}}\|^2 \\ \text{subject to} \quad & \left.\begin{array}{lll} \breve{\boldsymbol{x}}^\mathsf{T}\breve{\boldsymbol{E}}_{iuv}\breve{\boldsymbol{x}} & = 0, & u = v \\ \breve{\boldsymbol{x}}^\mathsf{T}\breve{\boldsymbol{E}}_{iuv}\breve{\boldsymbol{x}} & = 0, & u \neq v \\ \breve{\boldsymbol{x}}^\mathsf{T}\,\breve{\boldsymbol{S}}_{iu}\,\breve{\boldsymbol{x}} & = 0 & \end{array}\right\} \begin{array}{l} u,v = 1,2,3 \\ i = 1,\ldots,n \end{array} \\ & \breve{\boldsymbol{x}}^\mathsf{T}\,\breve{\boldsymbol{U}}\,\breve{\boldsymbol{x}} = 1 \end{aligned} \tag{11}$$

where the matrices $\breve{\boldsymbol{A}}$ and $\breve{\boldsymbol{E}}_{iuv}$, and $\breve{\boldsymbol{S}}_{iu}$ are obtained by stacking the coefficient matrices in (10), with suitable zero blocks for padding, while $\breve{\boldsymbol{U}}$ is zero everywhere, except a single entry in the bottom-right corner, which is equal to 1.

C. Anchoring

Since absolute poses are not observable from relative measurements, we fix a pose to be our reference frame. Without loss of generality we fix the pose of the first node to the identity pose ($\boldsymbol{t}_1 = \mathbf{0}_3$ and $\boldsymbol{R}_1 = \mathbf{I}_3$, or, equivalently $\boldsymbol{r}_1 = \text{rows}(\mathbf{I}_3)$). This process is usually called *anchoring*.

Preserving an objective function that is homogeneous in the variables is convenient for the derivation of the dual problem; hence, rather than setting $\boldsymbol{r}_1 = \text{rows}(\mathbf{I}_3)$ (which would lead to a constant terms within the square in the objective), we set $\boldsymbol{r}_1 = y\ \text{rows}(\mathbf{I}_3)$ (more discussion on this in Section III-D). Anchoring the first pose modifies (11) as follows:

$$
\begin{aligned}
f^\star = \min_{\boldsymbol{x}} \quad & \|\boldsymbol{A}\boldsymbol{x}\|^2 && \text{(Primal problem)} \\
\text{subject to} \quad & \left.\begin{aligned} \boldsymbol{x}^\mathsf{T}\boldsymbol{E}_{iuv}\boldsymbol{x} &= 0, \quad u = v \\ \boldsymbol{x}^\mathsf{T}\boldsymbol{E}_{iuv}\boldsymbol{x} &= 0, \quad u \neq v \\ \boldsymbol{x}^\mathsf{T}\,\boldsymbol{S}_{iu}\,\boldsymbol{x} &= 0 \end{aligned}\right\} && \begin{aligned} u, v &= 1,2,3 \\ i &= 1,\ldots,n-1 \end{aligned} \\
& \boldsymbol{x}^\mathsf{T}\,\boldsymbol{U}\,\boldsymbol{x} = 1
\end{aligned}
\tag{12}
$$

where $\boldsymbol{x} \in \mathbb{R}^{12(n-1)+1}$ is obtained by removing the first pose from $\breve{\boldsymbol{x}}$, $\boldsymbol{A}$ is obtained by removing from $\breve{\boldsymbol{A}}$ the columns corresponding to the first pose and adjusting the last column to accommodate the terms multiplying y (essentially, $\text{rows}(\mathbf{I}_3)$); $\boldsymbol{E}_{iuv}$ (resp. $\boldsymbol{S}_{iu}$, $\boldsymbol{U}$) are the same as $\breve{\boldsymbol{E}}_{iuv}$ (resp. $\breve{\boldsymbol{S}}_{iu}$, $\breve{\boldsymbol{U}}$) but without the rows and columns corresponding to the first pose.

D. Effect of the slack variable y

Observe that the constraint $y^2 = 1$ implies that $y = \pm 1$. Hence, when $y = -1$, it seems that we get "wrong" constraints:

1) The constraint $\boldsymbol{r}_1 = y\ \text{rows}(\mathbf{I}_3)$ becomes $\boldsymbol{R}_1 = -\mathbf{I}_3$.
2) The constraint (10) implies $\det(\boldsymbol{R}_i) = -1$ for all $i = 1, \ldots, n$.

However, noting that the objective in (12) is homogeneous, we can easily see that $\|\boldsymbol{A}\boldsymbol{x}\| = \|-\boldsymbol{A}\boldsymbol{x}\|$. Hence, we can change the sign of y and of $\boldsymbol{R}_i$, $i = 1, \ldots, n$ to satisfy the right constraints without altering the cost. This means that, without loss of generality, we can assume $y = 1$. This makes the constraints in our homogeneous formulation equivalent to the original ones.

IV. The dual problem

Standard optimization theory tells us that to every constrained optimization problem (called the *primal* problem), we can associate a *dual* problem, and it provides useful results to relate the two problems [7, 8]. In this section we derive the Lagrangian dual problem of the primal problem (12).

The first step to derive the dual is to build the *Lagrangian*:

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \quad & \|\boldsymbol{A}\boldsymbol{x}\|^2 - \left(\sum_{i=1}^{n-1} \sum_{u,v=1,2,3} \lambda_{iuv}^{\perp}\, \boldsymbol{x}^\mathsf{T}\boldsymbol{E}_{iuv}\boldsymbol{x}\right) \\
& - \left(\sum_{i=1}^{n-1} \sum_{u=1,2,3} \lambda_{iu}^{\det}\, \boldsymbol{x}^\mathsf{T}\,\boldsymbol{S}_{iu}\,\boldsymbol{x}\right) - \lambda^y \left(\boldsymbol{x}^\mathsf{T}\,\boldsymbol{U}\,\boldsymbol{x} - 1\right)
\end{aligned}
\tag{13}
$$

where $\lambda_{iuv}^{\perp}$, $\lambda_{iu}^{\det}$, and λ^y are the *Lagrange multipliers* or *dual variables* associated to the orthogonality, determinant, and $y^2 = 1$ constraints in (12), respectively; we stack all the dual variables in a vector $\boldsymbol{\lambda}$. The Lagrangian can be understood as a function that includes the objective function in (12) and penalty terms corresponding to each constraint in (12). We note that the only difference (besides minor notation changes) with respect to [1] is the presence of the terms "$\lambda_{iu}^{\det}\boldsymbol{x}^\mathsf{T}\boldsymbol{S}_{iu}\boldsymbol{x}$", which are due to the determinant constraints.

To make the notation more compact, we define the matrix:

$$
\boldsymbol{W}(\boldsymbol{\lambda}) \doteq \boldsymbol{A}^\mathsf{T}\boldsymbol{A} - \sum_{i=1}^{n-1}\left(\sum_{\substack{u,v=\\1,2,3}} \lambda_{iuv}^{\perp}\boldsymbol{E}_{iuv} + \sum_{\substack{u=\\1,2,3}} \lambda_{iu}^{\det}\boldsymbol{S}_{iu}\right) - \lambda^y\boldsymbol{U} \tag{14}
$$

Using $\boldsymbol{W}(\boldsymbol{\lambda})$ we write the Lagrangian succinctly as:

$$
\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \boldsymbol{x}^\mathsf{T}\boldsymbol{W}(\boldsymbol{\lambda})\boldsymbol{x} + \lambda^y \tag{15}
$$

The *dual function* $d(\boldsymbol{\lambda})$ is the infimum of the Lagrangian with respect to $\boldsymbol{x}$:

$$
d(\boldsymbol{\lambda}) = \inf_{\boldsymbol{x}}\ \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \inf_{\boldsymbol{x}}\ \boldsymbol{x}^\mathsf{T}\boldsymbol{W}(\boldsymbol{\lambda})\boldsymbol{x} + \lambda^y \tag{16}
$$

For any choice of $\boldsymbol{\lambda}$ the dual function provides a lower bound for the optimal value of the primal problem [7, Section 5.1.3]. Therefore, the *Lagrangian dual problem* looks for a *maximum* of the dual function over $\boldsymbol{\lambda}$:

$$
d^\star \doteq \max_{\boldsymbol{\lambda}}\ d(\boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda}}\ \inf_{\boldsymbol{x}}\ \boldsymbol{x}^\mathsf{T}\boldsymbol{W}(\boldsymbol{\lambda})\boldsymbol{x} + \lambda^y \tag{17}
$$

The infimum over $\boldsymbol{x}$ of $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})$ drifts to $-\infty$ unless $\boldsymbol{W}(\boldsymbol{\lambda}) \succeq 0$. Therefore we can safely restrict the maximization to vectors $\boldsymbol{\lambda}$ that are such that $\boldsymbol{W}(\boldsymbol{\lambda}) \succeq 0$; these are called *dual-feasible*. Moreover, at any dual-feasible $\boldsymbol{\lambda}$, the $\boldsymbol{x}$ minimizing the Lagrangian are those that make $\boldsymbol{x}^\mathsf{T}\boldsymbol{W}(\boldsymbol{\lambda})\boldsymbol{x} = 0$ (recall that $\boldsymbol{x}^\mathsf{T}\boldsymbol{W}(\boldsymbol{\lambda})\boldsymbol{x}$ cannot be negative when $\boldsymbol{W}(\boldsymbol{\lambda}) \succeq 0$). Therefore, (17) reduces to the following *dual problem*

$$
\begin{aligned}
d^\star = \quad & \max_{\boldsymbol{\lambda}} && \lambda^y, && \text{(Dual problem)} \\
& \text{s.t.:} && \boldsymbol{W}(\boldsymbol{\lambda}) \succeq 0.
\end{aligned}
\tag{18}
$$

The importance of the dual problem is twofold. First, it holds

$$
d^\star \leq f^\star \tag{19}
$$

This property is called *weak duality*, see, e.g., [7, Section 5.2.2]. For particular problems the inequality (19) becomes an equality, and in such cases we say that *strong duality* holds. The difference between $f^\star$ and $d^\star$ is called *duality gap* and, by (19), the gap is always positive.

The second important property of the Lagrangian dual problem (18) is that, since $d(\boldsymbol{\lambda})$ is concave (point-wise minimum of affine functions), the dual (18) is always convex in $\boldsymbol{\lambda}$, regardless the convexity properties of the primal problem. The dual PGO problem (18) is a semidefinite program (SDP) and we can compute the optimal solution $\boldsymbol{\lambda}^\star$ of (18) using off-the-shelf solvers, e.g., [9, 10]. In the following section we discuss the relations between the primal and the dual problem, and elucidate on the practical use of the dual.

V. Relation between the Primal and the Dual Problem and Practical Use

In this section we provide a more practical view of the results of [1], applying them to the extended formulation proposed in this addendum. We do not give proofs, since the inclusion of the determinant constraints only added extra terms in the definition of the matrix $\boldsymbol{W}(\boldsymbol{\lambda})$, hence the proof of the results stated here remains identical to the ones given in [1].

As stated in the introduction, our objective is two-fold. We want to design verification techniques and possibly find globally optimal solutions for PGO. Let us start with verification.

Verification. Assume that we computed an estimate for the poses in the pose graph, namely $(\hat{\boldsymbol{R}}_i, \hat{\boldsymbol{t}}_i)$, $i = 1, \ldots, n$, using a state-of-the-art iterative solver, e.g., iSAM2 [11] or g2o [12]. Then we want to ask a basic question: *can we quantify how far is this estimate from the global minimum of the cost function, and possibly certify its optimality*?

We call this candidate solution $\hat{\boldsymbol{x}}$, assuming that the poses $(\hat{\boldsymbol{R}}_i, \hat{\boldsymbol{t}}_i)$, $i = 1, \ldots, n$, have been "vectorized" as in Section III. The following result provides a first tool for verification.

Proposition 1 (Verification of Primal Objective): Given a candidate solution $\hat{\boldsymbol{x}}$ for the primal problem (12), and calling $d^\star$ the optimal objective of the dual problem (18), if $f(\hat{\boldsymbol{x}}) = d^\star$, then the duality gap is zero and $\hat{\boldsymbol{x}}$ is an optimal solution of (12). Moreover, even if the duality gap is nonzero, $f(\hat{\boldsymbol{x}}) - d^\star \geq f(\hat{\boldsymbol{x}}) - f^\star$, meaning that $f(\hat{\boldsymbol{x}}) - d^\star$ is an upper-bound for the sub-optimality gap of $\hat{\boldsymbol{x}}$.

Proposition 1 (*cf.* [1, Proposition 2]) ensures that the candidate $\hat{\boldsymbol{x}}$ is optimal when $f(\hat{\boldsymbol{x}}) = d^\star$. Moreover, even in the case in which we get $f(\hat{\boldsymbol{x}}) > d^\star$, the quantity $f(\hat{\boldsymbol{x}}) - d^\star$ can be used as an indicator of how far $\hat{\boldsymbol{x}}$ is from the global optimum.

Our derivation also enables a more sophisticated verification technique. This technique, given in Proposition 3, is based on the following result (*cf.* [1, Lemma 1 and Proposition 3]).

Lemma 2 (Primal optimal solution and zero duality gap): If the duality gap is zero ($d^\star = f^\star$), then any primal optimal solution $\boldsymbol{x}^\star$ of (12) is in the null space of the matrix $\boldsymbol{W}(\boldsymbol{\lambda}^\star)$, where $\boldsymbol{\lambda}^\star$ is the solution of the dual (18), i.e., $\boldsymbol{W}(\boldsymbol{\lambda}^\star)\boldsymbol{x}^\star = \boldsymbol{0}$.

Proposition 3 (Verification of Primal Optimal Solution): Given a candidate solution $\hat{\boldsymbol{x}}$ for the primal problem (12), if the solution $\hat{\boldsymbol{\lambda}}$ of the linear system

$$\boldsymbol{W}(\hat{\boldsymbol{\lambda}})\hat{\boldsymbol{x}} = \boldsymbol{0} \qquad \text{(to be solved w.r.t. } \hat{\boldsymbol{\lambda}}) \tag{20}$$

is such that $\boldsymbol{W}(\hat{\boldsymbol{\lambda}}) \succeq 0$ and $d(\hat{\boldsymbol{\lambda}}) = f(\hat{\boldsymbol{x}})$, then the duality gap is zero and $\hat{\boldsymbol{x}}$ is a primal optimal solution.

This second verification technique is more convenient in practice, since it does not require solving the SDP (18), but only requires solving a sparse linear system and then verifying that the sparse matrix $\boldsymbol{W}(\hat{\boldsymbol{\lambda}})$ is positive definite (this can be checked by computing the smallest eigenvalue of $\boldsymbol{W}(\hat{\boldsymbol{\lambda}})$).

Optimal solution. In [1], it has been shown that, when the duality gap is zero, we can compute a guaranteed, globally optimal solution to PGO. The result stems directly from Lemma 2 and can be stated as follows (*cf.* [1, Proposition 5]).

Proposition 4: If the duality gap is zero and $\boldsymbol{\lambda}^\star$ is an optimal solution of (18), then an optimal solution $\boldsymbol{x}^\star$ of (12) can be computed by solving the following linear system:

$$\boldsymbol{W}(\boldsymbol{\lambda}^\star)\boldsymbol{x}^\star = \boldsymbol{0} \quad \text{(to be solved w.r.t. } \boldsymbol{x}^\star) \tag{21}$$

where the last entry in $\boldsymbol{x}^\star$ (corresponding to y) is fixed to 1.

Despite the appeal of having a (non-iterative) technique that computes a globally optimal solution for SLAM, the current use of (21) is pretty limited, as the computation of $\boldsymbol{x}^\star$ requires to compute $\boldsymbol{\lambda}^\star$ by solving the SDP (18). While SDPs are convex problems (hence can be solved in polynomial time), current SDP solvers are fairly slow and do not scale to large problems, as the ones usually encountered in SLAM.

VI. Experiments

Section V shows that when the duality gap is zero we are able to provide strong results on 3D PGO: we are able to design fast verification techniques that do not require solving the SDP underlying the dual problem (Proposition 3), and we can compute provably optimal solutions (Proposition 4).

In [1] we showed that the duality gap is zero in many large-scale real SLAM problems. Moreover, we provided a Monte Carlo analysis that highlights the influence of different parameters (noise level, number of poses) on the duality gap.

The goal of this experimental section is to compare the results of [1] and the extended formulation proposed in this addendum, and check if the addition of the determinant constraints "enlarged" the domain in which the duality gap is zero. We do this comparison on the same simulation setup of [1]. We use the cube dataset of Fig. 1(b4). In this dataset, the odometric trajectory is simulated as the robot travels on a 3D grid world, and random loop closures are added between nearby nodes, with probability 0.3. Relative pose measurements are obtained by contaminating the true relative poses with zero-mean Gaussian noise, with standard deviation σ_T and σ_R for the translational and rotational noise, respectively. Statistics are computed over 10 runs: for each run we create a cube with random connectivity and random measurement noise.

In our experiments we compute the "optimal" solution $f^\star$ of (1) by refining the *chordal* initialization of [5, 13] with 10 Gauss-Newton (GN) iterations. While one cannot guarantee *a priori* that this approach always produces the optimal estimate, using the results of this paper we will be able to check optimality a posteriori. For the solution of the dual problem (18), we use SDPA [10]. The solver returns the dual optimal objective $d^\star$ and the dual optimal variables $\boldsymbol{\lambda}^\star$.

Duality gap. Here we compare the value of the primal optimal objective $f^\star$, against the value of the dual optimal

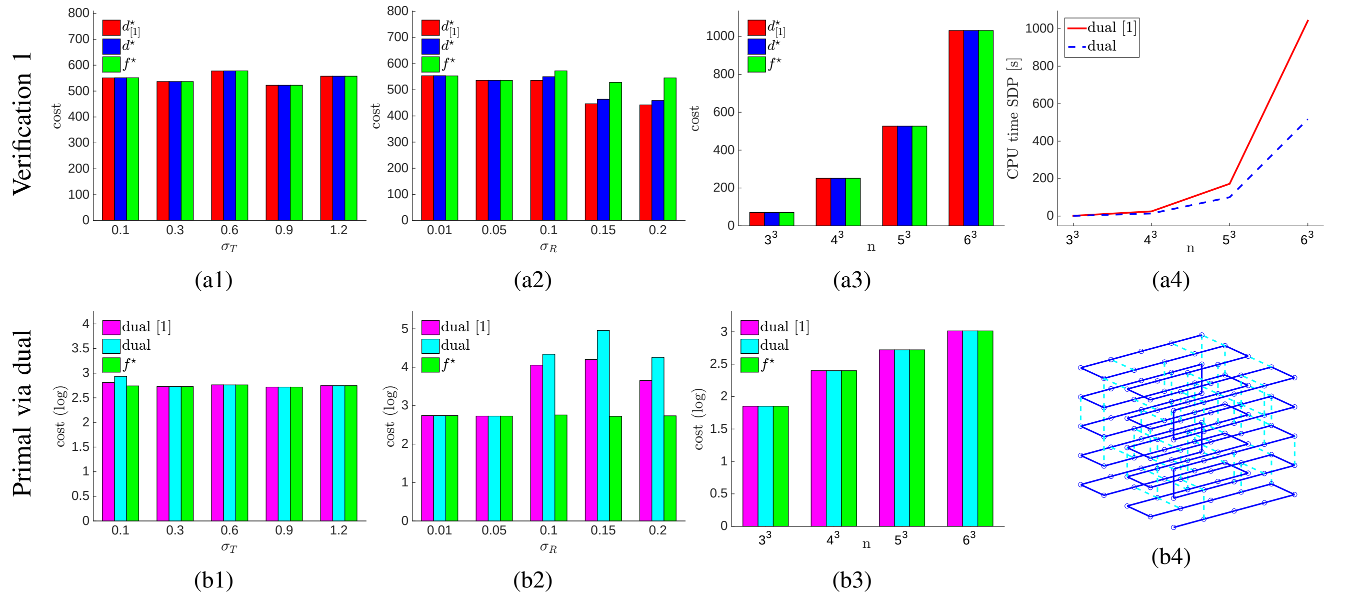

Fig. 1: (a1)-(a3): comparison between $f^\star$, $d^\star$ (eq. (18)), and $d^\star_{[1]}$ [1, eq. (22)], for different levels of translation noise σ_T (first column), rotation noise σ_R (second column), and size of the problem (third column). (a4): CPU time required to solve the SDP in (18) (dashed blue line) and in [1, eq. (22)] (solid red line). (b1)-(b3): comparison between $f^\star$, the primal optimal solution computed according to Proposition 4 ("dual"), and the primal solution computed as in [1, Proposition 5] ("dual [1]"), (b4): cube dataset used for the Monte Carlo simulations.

objective $d^\star$ in (18), and the dual objective of the relaxed formulation in [1, eq. (22)], which we denote as $d^\star_{[1]}$. Fig. 1(a1) shows $d^\star_{[1]}$, $d^\star$, and $f^\star$ for different translational noise levels, fixing $\sigma_R = 0.05$rad. The figure shows that $d^\star_{[1]} = d^\star = f^\star$ (zero duality gap) independently on the translational noise. The duality gap was already observed to be zero using the formulation [1], hence the inclusion of the determinant constraints does not imply particular benefits. The same conclusion can be drawn from Fig. 1(a3), which compares $d^\star_{[1]}$, $d^\star$ and $f^\star$ for different sizes of the cube dataset, fixing $\sigma_R = 0.05$rad and $\sigma_T = 0.1$m. Again, $d^\star_{[1]} = d^\star = f^\star$ (zero duality gap) independently of the size of the dataset, hence the relaxed formulation [1] already performed well.

Fig. 1(a2) shows $d^\star_{[1]}$, $d^\star$, and $f^\star$ for different rotational noise, fixing $\sigma_T = 0.1$m. In this case the duality gap $f^\star - d^\star$ is more sensitive to the noise level, and for large rotational noise both $d^\star_{[1]}$ and $d^\star$ become smaller than $f^\star$. While $d^\star$ is slightly better (i.e., larger) than $d^\star_{[1]}$, the difference is small, and the inclusion of the determinant constraints does not significantly enlarge the domain in which the duality gap is zero.

Fig. 1(a4) reports the average CPU time required to solve the SDP in eq. (18) (dashed blue line) and the SDP in [1, eq. (22)] (solid red line). Interestingly, the computation cost of solving the complete dual problem (including the determinant constraints) is slightly smaller than the cost of the relaxed formulation [1]. Therefore, the inclusion of the determinant constraints does not imply an extra computational burden.

Primal optimal solution via the dual. In this paragraph we compare the performance of the primal optimal solution computed according to Proposition 4 against the primal solution computed according to [1, Proposition 5]. Recall that the two formulations only differ by the extra terms we included in the matrix $\boldsymbol{W}(\boldsymbol{\lambda}^\star)$ and which correspond to the determinant constraints. The ideal outcome of our analysis is that the extended formulation performs better, since it can find more instances in which the duality gap is zero and Proposition 4 is guaranteed to produce an optimal solution. Unfortunately, the following experiments show that this is not the case.

Figs. 1(b1)-(b3) compare the cost of the solution obtained from (i) Proposition 4 (label: "dual"), (ii) Proposition 5 in [1] (label: "dual [1]"), and the primal solution $f^\star$. Fig. 1(b1) and Fig. 1(b3) compare the costs for increasing levels of translational noise and dataset size, respectively. We can see that the bars corresponding to the two dual problems lead to very similar costs, hence there is no clear advantage in including the determinant constraints in the formulation. We attribute the small mismatch between the heights of the bars on the left of Fig. 1(b1) to numerical errors, as inaccuracies in the solution of the SDP (18) (which may depend on the stopping conditions used in SDPA [10]) propagate to the primal solution computed according to Proposition 4.

Fig. 1(b2) compares the costs for increasing levels of rotation noise. In this case we see that the bars corresponding to the two dual problems become larger than $f^\star$ (i.e., they produce suboptimal solutions) for rotation noise larger than 0.1rad; this corresponds to the noise levels for which the duality gap becomes non-zero (*cf.* Fig. 1(a2)); indeed, when the duality gap is non-zero, the results in Proposition 4 and [1, Proposition 5] do not apply, hence we may get suboptimal solutions. Interestingly, the suboptimal solutions of Proposition 4 (which includes the determinant constraints) have larger cost with respect to the relaxed formulation [1, Proposition 5].

We conclude this section by remarking that, using Proposition 3, we can classify a candidate solution (e.g., distinguish optimal versus suboptimal estimates) whenever the duality gap is zero. To evaluate the efficacy of this classification, in [1], we provided an empirical analysis of the relaxed formulation by reporting precision-recall curves. We repeated the same tests for the extended formulation of this paper and we obtained exactly the same precision-recall curves, which essentially means that whenever the complete formulation proposed in this paper succeeded to certify optimality, also the relaxed formulation [1] was able to correctly classify the candidate. This again confirms that the introduction of the determinant constraints does not have significant impact in practice.

VII. Discussion

Our results seem to suggest that the inclusion of the determinant constraints, in practical instances of PGO, do not provide any significant advantage from the point of view of the duality gap. This, in turn, implies that the determinant constraints are not active at the optimal (primal) solution, and that the orthogonality constraints are sufficient to define the optimal solution. We argue that this is due to the fact that, in practical instances of the problem (in which rotation measurement noise is reasonably small) one cannot obtain better solutions by flipping the sign of some rotations and turning them into reflections. Intuitively, this would happen only with highly self-inconsistent sets of measurements which contain a large number of outliers (for which the least-squares formulation in (1) might be not suitable). Nonetheless, our analysis shows that the relaxation of the determinant constraints, in practice, is not the determining factor to cause a non-zero duality gap.

VIII. Conclusion

This technical addendum shows how to include determinant constraints in the derivation of the dual SLAM problem, originally proposed in [1]. Determinant constraints can be formulated as quadratic equality constraints and do not fundamentally alter the structure of the primal and dual problem. As such, the results given in [1] apply to the extended formulation proposed in this paper without any significant modification.

While this extension completes the derivation given in [1], current experiments do not highlight any tangible advantage from adding the determinant constraints: the range of operation in which the duality gap is empirically zero remains practically the same for both formulations. We conclude that the relaxation of the determinant constraints is not a cause of failures in the certification of optimality.

Future work includes performing further numerical evaluations (e.g., using datasets corrupted by outliers), and investigating alternative methods to reduce/quantify the duality gap.

References

[1] L. Carlone, D. Rosen, G. Calafiore, J.J. Leonard, and F. Dellaert. Lagrangian duality in 3D SLAM: Verification techniques and optimal solutions. In *the Intl. Conf. on Intelligent Robots and Systems, submitted, ArXiv preprint available at: http://arxiv.org/abs/1506.00746*, 2015.

[2] L. Carlone and F. Dellaert. Duality-based verification techniques for 2D SLAM. In *Intl. Conf. on Robotics and Automation (ICRA), accepted*, 2015.

[3] G. Calafiore, L. Carlone, and F. Dellaert. Pose graph optimization in the complex domain: Lagrangian duality, conditions for zero duality gap, and optimal solutions. In *arXiv preprint: 1505.03437*, 2015.

[4] R. Hartley, J. Trumpf, Y. Dai, and H. Li. Rotation averaging. *IJCV*, 103(3):267–305, 2013.

[5] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert. Initialization techniques for 3D SLAM: a survey on rotation estimation and its use in pose graph optimization. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[6] D.M. Rosen and J.J. Leonard. A convex relaxation for approximate global optimization in simultaneous localization and mapping. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[7] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[8] G.C. Calafiore and L. El Ghaoui. *Optimization Models*. Cambridge University Press, 2014.

[9] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming. URL http://cvxr.com/cvx.

[10] M. Yamashita, K. Fujisawa, and M. Kojima. Implementation and evaluation of SDPA 6.0 (semidefinite programming algorithm 6.0). *Optimization Methods and Software*, 18(4):491–505, August 2003.

[11] 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*, 31:217–236, Feb 2012.

[12] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. g2o: A general framework for graph optimization. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, May 2011.

[13] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.