2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Majorization Minimization Methods for Distributed Pose Graph Optimization with Convergence Guarantees

Taosha Fan Todd Murphey

***Abstract*— In this paper, we consider the problem of distributed pose graph optimization (PGO) that has extensive applications in multi-robot simultaneous localization and mapping (SLAM). We propose majorization minimization methods for distributed PGO and show that our methods are guaranteed to converge to first-order critical points under mild conditions. Furthermore, since our methods rely a proximal operator of distributed PGO, the convergence rate can be significantly accelerated with Nesterov's method, and more importantly, the acceleration induces no compromise of convergence guarantees. In addition, we also present accelerated majorization minimization methods for the distributed chordal initialization that have a quadratic convergence, which can be used to compute an initial guess for distributed PGO. The efficacy of this work is validated through applications on a number of 2D and 3D SLAM datasets and comparisons with existing state-of-the-art methods, which indicates that our methods have faster convergence and result in better solutions to distributed PGO.**

I. Introduction

In multi-robot simultaneous localization and mapping (SLAM) [1]–[3], each robot needs to estimate its own poses from noisy relative pose measurements under limited communication with the other robots, and such a problem can be formulated as distributed pose graph optimization (PGO), in which each robot can be represented as a node and two nodes (robots) are said to be neighbors if there exists a noisy relative pose measurement between them. In most cases, it is assumed that each node can only communicate with its neighbors, which suggests that distributed PGO is much more challenging than non-distributed PGO. Even though there exists numerous methods for PGO [4]–[12], most, if not all, of them are difficult to be distributed.

In the last decade, multi-robot SLAM has been becoming increasingly important, which further promotes the research of distributed PGO [13]–[15]. A distributed linear system solver is implemented in [13] to evaluate the chordal initialization [16] and the Gauss-Newton direction for distributed PGO. A heuristic extension of the alternating direction method of multipliers (ADMM) for distributed PGO is proposed in [14]. A multi-stage first-order method for distributed PGO using the Riemannian gradient is presented in [15]. However, [13], [14] have no guarantees to converge to critical points, whereas [15] needs strict presumptions that might fail to hold in practice.

Taosha Fan and Todd Murphey are with the Department of Mechanical Engineering, Northwestern University, Evanston, IL 60201, USA. E-mail: taosha.fan@u.northwestern.edu, t-murphey@northwestern.edu

This material is based upon work supported by the National Science Foundation under award DCSD-1662233.

In this paper, we propose majorization minimization methods [17] for distributed PGO that extend our previous work [18], in which proximal methods for PGO are proposed that converge to first-order critical points for both centralized and distributed PGO. In [18], each pose is represented as a single node and updated independently. Even though proximal methods for PGO in [18] converge fast for centralized PGO and apply to any distributed PGO, it might have slow convergence for multi-robot SLAM, in which each robot usually has more than one poses and it is more reasonable to represent poses of the same robot rather than each individual pose as a node. In this paper, poses of the same robot are represented as the same node and are updated as a whole, which makes use of more information in optimization and is expected to converge faster than [18] for distributed PGO in multi-robot SLAM. Furthermore, we redesign the accelerated algorithm using Nesterov's method for distributed PGO such that the inefficient objective evaluation is avoided and the inter-node communication is significantly reduced.

In contrast to existing state-of-the-art methods [13]–[15], our methods minimize upper bounds of PGO by solving independent optimization subproblems and are guaranteed to converge to first-order critical points. Furthermore, since our methods rely on proximal operators of PGO, Nesterov's method [19], [20] can be used for acceleration, and the acceleration has limited extra computation, and more importantly, induces no compromise of convergence guarantees. In addition, we also propose accelerated majorization minimization methods for the distributed chordal initialization that have a quadratic convergence to compute an initial guess for distributed PGO.

The rest of this paper is organized as follows: Section II introduces notation used in this paper. Section III formulates the problem of distributed PGO. Section IV presents an upper bound that majorizes distributed PGO. Section V and Section VI present and accelerate majorization minimization methods for distributed PGO, respectively. Section VII presents accelerated majorization minimization methods for the distributed chordal initialization. Section VIII implements our methods for distributed PGO in multi-robot SLAM and makes comparisons with existing state-of-the-art method [13]. Section IX concludes the paper.

II. Notation

$\mathbb{R}$ denotes real numbers; $\mathbb{R}^{m\times n}$ and $\mathbb{R}^n$ denote $m \times n$ matrices and $n \times 1$ vectors, respectively; and $SO(d)$ and $SE(d)$ denote special orthogonal groups and special Euclidean groups, respectively. For a matrix $X \in \mathbb{R}^{m\times n}$, the

notation $[X]_{ij}$ denotes its $(i,\,j)$-th entry or $(i,\,j)$-th block. The notation $\|\cdot\|$ denotes the Frobenius norm of matrices and vectors. For symmetric matrices $Y,\,Z\in\mathbb{R}^{n\times n}$, $Y\succeq Z$ (or $Z\preceq Y$) and $Y\succ Z$ (or $Z\prec Y$) indicate that $Y-Z$ is positive (or negative) semidefinite and definite, respectively. If $F:\mathbb{R}^{m\times n}\rightarrow\mathbb{R}$ is a function, $\mathcal{M}\subset\mathbb{R}^{m\times n}$ is a Riemannian manifold and $X\in\mathcal{M}$, the notation $\nabla F(X)$ and $\mathrm{grad}\,F(X)$ denote the Euclidean and Riemannian gradients, respectively.

III. Problem Formulation

Distributed PGO considers the problem of estimating unknown poses $g_1^\alpha,\,g_2^\alpha,\,\cdots,\,g_{n_\alpha}^\alpha\in SE(d)$ of each node $\alpha\in\mathcal{A}\triangleq\{1,\,2,\,\cdots,\,A\}$, in which $g_{(\cdot)}^\alpha=(t_{(\cdot)}^\alpha,\,R_{(\cdot)}^\alpha)$ with $t_{(\cdot)}^\alpha\in\mathbb{R}^d$ and $R_{(\cdot)}^\alpha\in SO(d)$ and n_α is the number of poses in node α, given intra-node noisy measurements $\tilde{g}_{ij}^{\alpha\alpha}\in SE(d)$ of

$$g_{ij}^{\alpha\alpha}\triangleq\left(g_i^\alpha\right)^{-1}g_j^\alpha\in SE(d)$$

within a single node α and inter-node noisy measurements $\tilde{g}_{ij}^{\alpha\beta}\in SE(d)$ of

$$g_{ij}^{\alpha\beta}\triangleq\left(g_i^\alpha\right)^{-1}g_j^\beta\in SE(d)$$

between different nodes $\alpha\neq\beta$.

For notational simplicity, we rewrite poses $g_1^\alpha,\,g_2^\alpha,\,\cdots,\,g_{n_\alpha}^\alpha\in SE(d)$ of node α as

$$X^\alpha\triangleq\begin{bmatrix}t^\alpha & R^\alpha\end{bmatrix}\in\mathcal{X}^\alpha\subset\mathbb{R}^{d\times(d+1)n_\alpha}\tag{1}$$

in which $t^\alpha\triangleq\begin{bmatrix}t_1^\alpha & \cdots & t_{n_\alpha}^\alpha\end{bmatrix}\in\mathbb{R}^{d\times n_\alpha}$, $R^\alpha\triangleq\begin{bmatrix}R_1^\alpha & \cdots & R_{n_\alpha}^\alpha\end{bmatrix}\in SO(d)^{n_\alpha}\subset\mathbb{R}^{d\times dn_\alpha}$, and

$$\mathcal{X}^\alpha\triangleq\mathbb{R}^{d\times n_\alpha}\times SO(d)^{n_\alpha}.$$

Moreover, we define $\mathcal{X}$ to be the product space such that $\mathcal{X}\triangleq\mathcal{X}^1\times\cdots\times\mathcal{X}^A\subset\mathbb{R}^{d\times(d+1)n}$ in which $n=\sum_{\alpha\in\mathcal{A}}n_\alpha$, and define $\vec{\mathcal{E}}^{\,\alpha\beta}$ to be the set of index pairs such that $(i,\,j)\in\vec{\mathcal{E}}^{\,\alpha\beta}$ if and only if there exists a noisy measurement $\tilde{g}_{ij}^{\alpha\beta}\in SE(d)$, and define $\mathcal{N}_-^\alpha$ (respectively, $\mathcal{N}_+^\alpha$) to be the subset of $\mathcal{A}$ such that a node $\beta\in\mathcal{N}_-^\alpha$ (respectively, $\beta\in\mathcal{N}_+^\alpha$) if and only if $\vec{\mathcal{E}}^{\,\alpha\beta}\neq\emptyset$ (respectively, $\vec{\mathcal{E}}^{\,\beta\alpha}\neq\emptyset$) and $\beta\neq\alpha$.

Following [4], it is possible to formulate distributed PGO as maximum likelihood estimation

$$\min_{X\in\mathcal{X}}F(X).\tag{2}$$

in which $X\triangleq\begin{bmatrix}X^1 & \cdots & X^A\end{bmatrix}\in\mathcal{X}$. In Eq. (2), the objective function $F(X)$ is defined to be

$$\begin{aligned}F(X)\triangleq&\sum_{\alpha\in\mathcal{A}}\sum_{(i,j)\in\vec{\mathcal{E}}^{\,\alpha\alpha}}\frac{1}{2}\Big[\kappa_{ij}^{\alpha\alpha}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\alpha}-R_j^\alpha\|^2+\\&\tau_{ij}^{\alpha\alpha}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\alpha}+t_i^\alpha-t_j^\alpha\|^2\Big]+\\&\sum_{\substack{\alpha,\beta\in\mathcal{A},\\\alpha\neq\beta}}\sum_{(i,j)\in\vec{\mathcal{E}}^{\,\alpha\beta}}\frac{1}{2}\Big[\kappa_{ij}^{\alpha\beta}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\beta}-R_j^\beta\|^2+\\&\tau_{ij}^{\alpha\beta}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\beta}+t_i^\alpha-t_j^\beta\|^2\Big],\end{aligned}\tag{3}$$

in which $\kappa_{ij}^{\alpha\alpha}$, $\kappa_{ij}^{\alpha\beta}$, $\tau_{ij}^{\alpha\alpha}$ and $\tau_{ij}^{\alpha\beta}$ are weights that are related with measurement noise [4]. Furthermore, it is straightforward to show that there exists a positive-semidefinite data matrix $\widetilde{M}\in\mathbb{R}^{(d+1)n\times(d+1)n}$ such that Eq. (3) is equivalent to [4]

$$F(X)\triangleq\frac{1}{2}\,\mathrm{trace}(X\widetilde{M}X^\top).\tag{4}$$

In the following sections, we will present majorization minimization methods to solve distributed PGO of Eqs. (2) and (4), which is the major contribution of this paper.

IV. The Majorization of Distributed PGO

In this section, following a similar procedure to [18], we will propose a function $E(X|X^{(k)})$ that is a proximal operator majorizing $F(X)$ in Eqs. (3) and (4), and such a proximal operator is critical to our proposed majorization minimization methods for distributed PGO.

For any matrices $B,\,C\in\mathbb{R}^{m\times n}$, it is known that

$$\frac{1}{2}\|B-C\|^2\leq\|B-P\|^2+\|C-P\|^2\tag{5}$$

for any $P\in\mathbb{R}^{m\times n}$, in which "=" holds if

$$P=\frac{1}{2}B+\frac{1}{2}C.$$

If we assume that $X^{(k)}=\begin{bmatrix}X^{1(k)} & \cdots & X^{A(k)}\end{bmatrix}\in\mathcal{X}$ with $X^{\alpha(k)}\in\mathcal{X}^\alpha$ is the current iterate of Eq. (2), then implementing Eq. (5) on each inter-robot measurement cost

$$\frac{1}{2}\kappa_{ij}^{\alpha\beta}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\beta}-R_j^\beta\|^2+\frac{1}{2}\tau_{ij}^{\alpha\beta}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\beta}+t_i^\alpha-t_j^\beta\|^2$$

between node α and β $(\alpha\neq\beta)$ in Eq. (3), we obtain an upper bound of Eq. (3) as

$$\begin{aligned}E(X|X^{(k)})\triangleq&\sum_{\alpha\in\mathcal{A}}\sum_{(i,j)\in\vec{\mathcal{E}}^{\,\alpha\alpha}}\frac{1}{2}\Big[\kappa_{ij}^{\alpha\alpha}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\alpha}-R_j^\alpha\|^2+\\&\tau_{ij}^{\alpha\alpha}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\alpha}+t_i^\alpha-t_j^\alpha\|^2\Big]+\\&\sum_{\substack{\alpha,\beta\in\mathcal{A},\\\alpha\neq\beta}}\sum_{(i,j)\in\vec{\mathcal{E}}^{\,\alpha\beta}}\Big[\kappa_{ij}^{\alpha\beta}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\beta}-P_{ij}^{\alpha\beta(k)}\|^2+\\&\tau_{ij}^{\alpha\beta}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\beta}+t_i^\alpha-p_{ij}^{\alpha\beta(k)}\|^2+\\&\kappa_{ij}^{\alpha\beta}\|R_j^\beta-P_{ij}^{\alpha\beta(k)}\|^2+\tau_{ij}^{\alpha\beta}\|t_j^\beta-p_{ij}^{\alpha\beta(k)}\|^2\Big],\end{aligned}\tag{6}$$

in which

$$P_{ij}^{\alpha\beta(k)}=\frac{1}{2}R_i^{\alpha(k)}\widetilde{R}_{ij}^{\alpha\beta}+\frac{1}{2}R_j^{\beta(k)}\tag{7}$$

and

$$p_{ij}^{\alpha\beta(k)}=\frac{1}{2}R_i^{\alpha(k)}\tilde{t}_{ij}^{\alpha\beta}+\frac{1}{2}t_j^{\beta(k)}.\tag{8}$$

As a matter of fact, we might decompose $E(X|X^{(k)})$ into

$$E(X|X^{(k)})=\sum_{\alpha\in\mathcal{A}}E^\alpha(X^\alpha|X^{(k)}),\tag{9}$$

in which each $E^\alpha(X^\alpha|X^{(k)})$ only depends on X^α in a single node α

$$E^\alpha(X^\alpha|X^{(k)}) \triangleq \sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\alpha\alpha}} \frac{1}{2}\Big[\kappa_{ij}^{\alpha\alpha}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\alpha} - R_j^\alpha\|^2 + \tau_{ij}^{\alpha\alpha}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\alpha} + t_i^\alpha - t_j^\alpha\|^2\Big] + \sum_{\beta\in\mathcal{N}_-^\alpha}\sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\alpha\beta}} \Big[\kappa_{ij}^{\alpha\beta}\|R_i^\alpha\widetilde{R}_{ij}^{\alpha\beta} - P_{ij}^{\alpha\beta(k)}\|^2 + \tau_{ij}^{\alpha\beta}\|R_i^\alpha\tilde{t}_{ij}^{\alpha\beta} + t_i^\alpha - p_{ij}^{\alpha\beta(k)}\|^2\Big] + \sum_{\beta\in\mathcal{N}_+^\alpha}\sum_{(j,i)\in\overrightarrow{\mathcal{E}}^{\beta\alpha}} \Big[\kappa_{ji}^{\beta\alpha}\|R_i^\alpha - P_{ji}^{\beta\alpha(k)}\|^2 + \tau_{ji}^{\beta\alpha}\|t_i^\alpha - p_{ji}^{\beta\alpha(k)}\|^2\Big]. \tag{10}$$

Moreover, $E(X|X^{(k)})$ is a proximal operator of $F(X)$ as the following proposition states.

Proposition 1. For all $\alpha\in\mathcal{A}$, there exists constant positive-semidefinite matrices $\widetilde{\Omega}^\alpha \in \mathbb{R}^{(d+1)n_\alpha\times(d+1)n_\alpha}$ such that $E(X|X^{(k)})$ is equivalent to

$$E(X|X^{(k)}) \triangleq \frac{1}{2}\langle\widetilde{\Omega}(X-X^{(k)}), X-X^{(k)}\rangle + \langle\nabla F(X^{(k)}), X-X^{(k)}\rangle + F(X^{(k)}), \tag{11}$$

in which $\widetilde{\Omega}\in\mathbb{R}^{(d+1)n\times(d+1)n}$ is a block diagonal matrix

$$\widetilde{\Omega} \triangleq \operatorname{diag}\big\{\widetilde{\Omega}^1, \cdots, \widetilde{\Omega}^A\big\} \in \mathbb{R}^{(d+1)n\times(d+1)n},$$

and $\nabla F(X^{(k)}) \triangleq X^{(k)}\widetilde{M} \in \mathbb{R}^{d\times(d+1)n}$ is the Euclidean gradient of $F(X)$ at $X^{(k)}\in\mathcal{X}$. Furthermore, $\widetilde{\Omega}\succeq\widetilde{M}$ and $E(X|X^{(k)}) \geq F(X^{(k)})$ in which "=" holds if $X = X^{(k)}$.

Proof. See [21, Appendix A]. □

In the next section, we will present majorization minimization methods for distributed PGO using $E(X|X^{(k)})$ in Eqs. (6) and (11).

V. The Majorization Minimization Method for Distributed PGO

From Proposition 1, it is known $\widetilde{\Omega}\succeq\widetilde{M}$, and then it can be shown that if $\xi\in\mathbb{R}$ and $\xi\geq 0$, there exists a block diagonal matrix $\widetilde{\Gamma}\triangleq\widetilde{\Omega}+\xi\cdot\mathbf{I}\in\mathbb{R}^{(d+1)n\times(d+1)n}$ such that

$$\widetilde{\Gamma} \triangleq \operatorname{diag}\big\{\widetilde{\Gamma}^1, \cdots, \widetilde{\Gamma}^A\big\} \succeq \widetilde{M}, \tag{12}$$

in which

$$\widetilde{\Gamma}^\alpha \triangleq \widetilde{\Omega}^\alpha + \xi\cdot\mathbf{I}^\alpha \in \mathbb{R}^{(d+1)n_\alpha\times(d+1)n_\alpha} \tag{13}$$

and $\mathbf{I}^\alpha\in\mathbb{R}^{(d+1)n_\alpha\times(d+1)n_\alpha}$ is the identity matrix. Then, we obtain

$$G(X|X^{(k)}) \triangleq \frac{1}{2}\langle\widetilde{\Gamma}(X-X^{(k)}), X-X^{(k)}\rangle + \langle\nabla F(X^{(k)}), X-X^{(k)}\rangle + F(X^{(k)}) \tag{14}$$

such that $G(X|X^{(k)})\geq F(X)$ in which "=" holds if $X = X^{(k)}$. More importantly, if there exists $X^{(k+1)}$ with

$$G(X^{(k+1)}|X^{(k)}) \leq G(X^{(k)}|X^{(k)}), \tag{15}$$

then it can be concluded that

$$F(X^{(k+1)}) \leq G(X^{(k+1)}|X^{(k)}) \leq G(X^{(k)}|X^{(k)}) = F(X^{(k)}).$$

In general, such a $X^{(k+1)}$ satisfying Eq. (15) can be found by solving

$$\min_{X\in\mathcal{X}} G(X|X^{(k)}). \tag{16}$$

Furthermore, since $\widetilde{\Gamma}$ is a block diagonal matrix and

$$\nabla F(X^{(k)}) \triangleq \big[\nabla_1 F(X^{(k)}) \quad \cdots \quad \nabla_A F(X^{(k)})\big], \tag{17}$$

in which $\nabla_\alpha F(X)\in\mathbb{R}^{d\times(d+1)n_\alpha}$ is the Euclidean gradient of $F(X)$ with respect to $X^\alpha\in\mathcal{X}^\alpha$, we might decompose $G(X|X^{(k)})$ into

$$G(X|X^{(k)}) = \sum_{X^\alpha\in\mathcal{X}^\alpha} G^\alpha(X^\alpha|X^{(k)}), \tag{18}$$

such that each $G^\alpha(X^\alpha|X^{(k)})$ is only related with X^α and $X^{\alpha(k)}\in\mathcal{X}^\alpha$

$$G^\alpha(X^\alpha|X^{(k)}) \triangleq \frac{1}{2}\langle\widetilde{\Gamma}^\alpha(X^\alpha - X^{\alpha(k)}), X^\alpha - X^{\alpha(k)}\rangle + \langle\nabla_\alpha F(X^{(k)}), X^\alpha - X^{\alpha(k)}\rangle + \overline{G}^{\alpha(k)}, \tag{19}$$

in which

$$\begin{aligned}\overline{G}^{\alpha(k)} \triangleq & \sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\alpha\alpha}} \frac{1}{2}\Big[\kappa_{ij}^{\alpha\alpha}\|R_i^{\alpha(k)}\widetilde{R}_{ij}^{\alpha\alpha} - R_j^{\alpha(k)}\|^2 + \tau_{ij}^{\alpha\alpha}\|R_i^{\alpha(k)}\tilde{t}_{ij}^{\alpha\alpha} + t_i^{\alpha(k)} - t_j^{\alpha(k)}\|^2\Big] + \\ & \sum_{\substack{\beta\in\mathcal{A},\\ \alpha\neq\beta}}\sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\alpha\beta}} \frac{1}{4}\Big[\kappa_{ij}^{\alpha\beta}\|R_i^{\alpha(k)}\widetilde{R}_{ij}^{\alpha\beta} - R_j^{\beta(k)}\|^2 + \tau_{ij}^{\alpha\beta}\|R_i^{\alpha(k)}\tilde{t}_{ij}^{\alpha\beta} + t_i^{\alpha(k)} - t_j^{\beta(k)}\|^2\Big] + \\ & \sum_{\substack{\beta\in\mathcal{A},\\ \alpha\neq\beta}}\sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\beta\alpha}} \frac{1}{4}\Big[\kappa_{ij}^{\beta\alpha}\|R_i^{\beta(k)}\widetilde{R}_{ij}^{\beta\alpha} - R_j^{\alpha(k)}\|^2 + \tau_{ij}^{\beta\alpha}\|R_i^{\beta(k)}\tilde{t}_{ij}^{\beta\alpha} + t_i^{\beta(k)} - t_j^{\alpha(k)}\|^2\Big].\end{aligned} \tag{20}$$

As a result, it is straightforward to show that Eq. (16) is equivalent to solving A independent optimization subproblems of smaller size

$$\min_{X^\alpha\in\mathcal{X}^\alpha} G^\alpha(X^\alpha|X^{(k)}), \qquad \forall\alpha = 1, \cdots, A. \tag{21}$$

If $X^{\alpha(k+1)}$ is computed from Eq. (21), it can be concluded that $F(X^{(0)})$, $F(X^{(1)})$, $\cdots$ are non-increasing as long as $G^\alpha(X^{\alpha(k+1)}|X^{(k)}) \leq G^\alpha(X^{\alpha(k)}|X^{(k)})$ for each node α.

Following Eqs. (16) and (21), we obtain the MM − PGO method for distributed PGO (Algorithm 1). The MM − PGO method can be classified as the majorization minimization method [17] that is widely used in machine learning, signal processing, applied mathematics, etc. The MM − PGO method can be distributed without introducing any extra computational workloads as long as each node α can communicate with its neighbor node $\beta\in\mathcal{N}_-^\alpha\cup\mathcal{N}_+^\alpha$, i.e., each pose g_i^α in node α only needs pose g_j^β in node β for which

Algorithm 1 The MM − PGO Method

1: **Input**: An initial iterate $X^{(0)} \in \mathcal{X}$.
2: **Output**: A sequence of iterates $\{X^{(k)}\}$.
3: $\nabla F(X^{(0)}) \leftarrow X^{(0)}\widetilde{M}$
4: **for** $k = 0, 1, 2, \cdots$ **do**
5: **for** $\alpha = 1, \cdots, A$ **do**
6: $X^{\alpha(k+1)} \leftarrow \arg\min_{X^\alpha \in \mathcal{X}^\alpha} G^\alpha(X^\alpha | X^{(k)})$
7: **end for**
8: $X^{(k+1)} \leftarrow \begin{bmatrix} X^{1(k+1)} & \cdots & X^{A(k+1)} \end{bmatrix}$
9: $\nabla F(X^{(k+1)}) \leftarrow X^{(k+1)}\widetilde{M}$
10: **end for**

either $(i, j) \in \overrightarrow{\mathcal{E}}^{\alpha\beta}$ or $(j, i) \in \overrightarrow{\mathcal{E}}^{\beta\alpha}$ to evaluate $\nabla F(X^{(k)})$ in line 9 of Algorithm 1. Even though $\overline{G}^{\alpha(k)}$ is included in $G^\alpha(X^\alpha|X^{(k)})$, it does not have to be explicitly evaluated when we minimize $G^\alpha(X^\alpha|X^{(k)})$. Furthermore, as is shown below, if Assumption 1 holds, we obtain Proposition 2 that the MM − PGO method is guaranteed to converge to first-order critical points of distributed PGO under mild conditions.

Assumption 1. For $X^{\alpha(k+1)} \leftarrow \arg\min_{X^\alpha \in \mathcal{X}^\alpha} G^\alpha(X^\alpha|X^{(k)})$ in the MM − PGO method, it is assumed that

$$G^\alpha(X^{\alpha(k+1)}|X^{(k)}) \leq G^\alpha(X^{\alpha(k)}|X^{(k)})$$

and

$$\operatorname{grad} G^\alpha(X^{\alpha(k+1)}|X^{(k)}) = \mathbf{0}$$

for all $\alpha = 1, \cdots, A$.

Proposition 2. If Assumption 1 holds, then for a sequence of iterates $\{X^{(k)}\}$ generated by Algorithm 1, we obtain

(1) $F(X^{(k)})$ is non-increasing;
(2) $F(X^{(k)}) \to F^\infty$ as $k \to \infty$;
(3) $\|X^{(k+1)} - X^{(k)}\| \to 0$ as $k \to \infty$ if $\widetilde{\Gamma} \succ \widetilde{M}$;
(4) $\operatorname{grad} F(X^{(k)}) \to \mathbf{0}$ as $k \to \infty$ if $\widetilde{\Gamma} \succ \widetilde{M}$;
(5) $\|X^{(k+1)} - X^{(k)}\| \to 0$ as $k \to \infty$ if $\xi > 0$;
(6) $\operatorname{grad} F(X^{(k)}) \to \mathbf{0}$ as $k \to \infty$ if $\xi > 0$.

Proof. See [21, Appendix B]. □

It should be noted that the MM − PGO method differs from [13]. The distributed PGO method in [13] relies on iterative distributed linear system solvers to evaluate the Gauss-Newton direction and then update the estimate using a single Gauss-Newton step, whereas the MM − PGO method in our paper minimizes an upper bound of PGO that is guaranteed to improve the current estimate and no Gauss-Newton directions are evaluated. Furthermore, to our knowledge, the distributed method in [13] is not non-increasing and has no convergence guarantees.

VI. The Accelerated Majorization Minimization Method for Distributed PGO

In the last thirty years, a number of accelerated first-order optimization methods have been proposed [19], [20]. Even though most of these accelerated methods were originally developed for convex optimization, it has been recently found that they empirically have a good performance for nonconvex optimization as well [22], [23].

Algorithm 2 The AMM − PGO Method

1: **Input**: An initial iterate $X^{(0)} \in \mathcal{X}$.
2: **Output**: A sequence of iterates $\{X^{(k)}\}$.
3: $\nabla F(X^{(0)}) \leftarrow X^{(0)}\widetilde{M}$
4: **for** $\alpha = 1, \cdots, A$ **do**
5: evaluate $\overline{G}^{\alpha(0)}$ using Eq. (20)
6: $s^{\alpha(0)} \leftarrow 1$
7: $X^{\alpha(-1)} \leftarrow X^{\alpha(0)}$, $\nabla F(X^{\alpha(-1)}) \leftarrow \nabla F(X^{\alpha(0)})$
8: **end for**
9: **for** $k = 0, 1, 2, \cdots$ **do**
10: **for** $\alpha = 1, \cdots, A$ **do**
11: $s^{\alpha(k+1)} \leftarrow \frac{\sqrt{4{s^{\alpha(k)}}^2+1}+1}{2}$, $\gamma^{\alpha(k)} \leftarrow \frac{s^{\alpha(k)}-1}{s^{\alpha(k+1)}}$
12: $Y^{\alpha(k)} \leftarrow X^{\alpha(k)} + \gamma^{\alpha(k)} \cdot \left(X^{\alpha(k)} - X^{\alpha(k-1)}\right)$
13: $\nabla F(Y^{\alpha(k)}) \leftarrow \nabla F(X^{\alpha(k)}) + \gamma^{\alpha(k)} \cdot \left(\nabla F(X^{\alpha(k)}) - \nabla F(X^{\alpha(k-1)})\right)$
14: $Z^{\alpha(k+1)} \leftarrow \arg\min_{Z^\alpha \in \mathcal{X}^\alpha} G^\alpha(Z^\alpha|Y^{\alpha(k)})$
15: **if** $G^\alpha(Z^{\alpha(k+1)}|X^{(k)}) > \overline{G}^{\alpha(k)}$ **then**
16: $X^{\alpha(k+1)} \leftarrow \arg\min_{X^\alpha \in \mathcal{X}^\alpha} G^\alpha(X^\alpha|X^{(k)})$
17: $s^{\alpha(k+1)} \leftarrow \max\{\frac{1}{2}s^{\alpha(k+1)}, 1\}$
18: **else**
19: $X^{\alpha(k+1)} \leftarrow Z^{\alpha(k+1)}$
20: **end if**
21: **end for**
22: $X^{(k+1)} \leftarrow \begin{bmatrix} X^{1(k+1)} & \cdots & X^{A(k+1)} \end{bmatrix}$
23: $\nabla F(X^{(k+1)}) \leftarrow X^{(k+1)}\widetilde{M}$
24: **for** $\alpha = 1, \cdots, A$ **do**
25: evaluate $\overline{G}^{\alpha(k+1)}$ using Eq. (20)
26: **end for**
27: **end for**

From Eq. (14), it can be seen that $G(X|X^{(k)})$ is a proximal operator of $F(X)$, which suggests that the MM − PGO method is a proximal method, and most importantly, it is possible to exploit existing accelerated schemes for proximal methods [19], [20].

Similar to [18], we might extend the MM − PGO method to obtain the accelerated majorization minimization method for distributed PGO using Nesterov's method [19], [20]. The resulting algorithm is referred as the AMM − PGO method (Algorithm 2). For the AMM − PGO method, each node α only needs pose estimates $g_j^{\beta(k)}$ of its neighbor node β to evaluate $\nabla F(X^{(k)})$ and $\overline{G}^{\alpha(k)}$ in lines 3, 5, 23 and 25 of Algorithm 2. The AMM − PGO method is equivalent to the MM − PGO method when $s^{\alpha(k)} = 1$, and is more governed by Nesterov's momentum as $s^{\alpha(k)}$ increases.

From Algorithm 2, the AMM – PGO method introduces Nesterov's momentum in lines 11 to 13 for acceleration, and adopts a restart in lines 15 to 20 to guarantee the convergence and improve the overall performance. In Algorithm 2, there is no need to evaluate the objective of PGO, i.e., Eqs. (3) and (4), which differs from the algorithm in [18], and thus, it is well-suited for distributed PGO. As Proposition 3 states, the AMM – PGO method has $F(X^{(k)})$ non-increasing and is guaranteed to converge to first-order critical points as long as Assumptions 1 and 2 hold and $\xi > 0$.

Assumption 2. For $Z^{\alpha(k+1)} \leftarrow \arg\min_{Z^\alpha \in \mathcal{X}^\alpha} G^\alpha(Z^\alpha | Y^{\alpha(k)})$ in the AMM – PGO method, it is assumed that

$$\operatorname{grad} G^\alpha(Z^{\alpha(k+1)} | Y^{\alpha(k)}) = \mathbf{0}$$

for all $\alpha = 1, \cdots, A$.

Proposition 3. If Assumptions 1 and 2 hold, then for a sequence of iterates $\{X^{(k)}\}$ generated by Algorithm 2, we obtain

(1) $F(X^{(k)})$ is non-increasing;
(2) $F(X^{(k)}) \to F^\infty$ as $k \to \infty$;
(3) $\|X^{(k+1)} - X^{(k)}\| \to 0$ as $k \to \infty$ if $\widetilde{\Gamma} \succ \widetilde{M}$;
(4) $\operatorname{grad} F(X^{(k)}) \to \mathbf{0}$ as $k \to \infty$ if $\widetilde{\Gamma} \succ \widetilde{M}$;
(5) $\|X^{(k+1)} - X^{(k)}\| \to 0$ as $k \to \infty$ if $\xi > 0$;
(6) $\operatorname{grad} F(X^{(k)}) \to \mathbf{0}$ as $k \to \infty$ if $\xi > 0$.

Proof. See [21, Appendix C]. □

VII. The Majorization Minimization Method for the Distributed Chordal Initialization

In PGO, the chordal initialization is one of the most popular initialization techniques [16], however, the distributed chordal initialization remains challenging [13]. In this section, we will present majorization minimization methods for the distributed chordal initialization that have a quadratic convergence.

The chordal initialization relaxes $SO(d)^n$ to $\mathbb{R}^{d\times dn}$ and solves the convex optimization problem

$$\min_{R\in\mathcal{R}} F_R(R) \tag{22}$$

In Eq. (22), $F_R(R)$, R and $\mathcal{R}$ are respectively defined to be

$$F_R(R) \triangleq \sum_{\alpha\in\mathcal{A}} \sum_{(i,j)\in\vec{\mathcal{E}}^{\alpha\alpha}} \frac{1}{2}\kappa_{ij}^{\alpha\alpha}\|R_i^\alpha \widetilde{R}_{ij}^{\alpha\alpha} - R_j^\alpha\|^2 + \sum_{\substack{\alpha,\beta\in\mathcal{A},\\ \alpha\neq\beta}} \sum_{(i,j)\in\vec{\mathcal{E}}^{\alpha\beta}} \frac{1}{2}\kappa_{ij}^{\alpha\beta}\|R_i^\alpha \widetilde{R}_{ij}^{\alpha\beta} - R_j^\beta\|^2, \tag{23}$$

and $R \triangleq \begin{bmatrix} R^1 & \cdots & R^A \end{bmatrix} \in \mathbb{R}^{d\times dn}$ in which $R^\alpha \triangleq \begin{bmatrix} R_1^\alpha & \cdots & R_{n_\alpha}^\alpha \end{bmatrix} \in \mathbb{R}^{d\times dn_\alpha}$, and $\mathcal{R} \triangleq \mathcal{R}^1 \times \cdots \times \mathcal{R}^A$ in which $\mathcal{R}^1 \triangleq \{R^1 \in \mathbb{R}^{d\times dn_1} | R_1^1 = \mathbf{I} \in \mathbb{R}^{d\times d}\}$ and $\mathcal{R}^\alpha \triangleq \mathbb{R}^{d\times dn_\alpha}$ if $\alpha \neq 1$. From Eq. (5), if $R^{(k)}$ is the current estimate of R, we might obtain an upper bound $G_R(R|R^{(k)})$ of $F(R)$

Algorithm 3 The AMM – Chordal Method

Input: An initial iterate $R^{(0)} \in \mathcal{R}$.
Output: A sequence of iterates $\{R^{(k)}\}$.
evaluate $\nabla F_R(R^{(0)})$
for $\alpha = 1, \cdots, A$ **do**
  $G_R^{\alpha(0)} \leftarrow G_R^\alpha(X|X^{\alpha(0)})$
  $s^{\alpha(0)} \leftarrow 1$
  $R^{\alpha(-1)} \leftarrow R^{\alpha(0)}$, $\nabla F_R(R^{\alpha(-1)}) \leftarrow \nabla F_R(R^{\alpha(0)})$
end for
for $k = 0, 1, 2, \cdots$ **do**
  for $\alpha = 1, \cdots, A$ **do**
    $s^{\alpha(k+1)} \leftarrow \frac{\sqrt{4s^{\alpha(k)2}+1}+1}{2}$, $\gamma^{\alpha(k)} \leftarrow \frac{s^{\alpha(k)}-1}{s^{\alpha(k+1)}}$
    $Y^{\alpha(k)} \leftarrow R^{\alpha(k)} + \gamma^{\alpha(k)} \cdot \left(R^{\alpha(k)} - R^{\alpha(k-1)}\right)$
    $\nabla F_R(Y^{\alpha(k)}) \leftarrow \nabla F_R(R^{\alpha(k)}) + \gamma^{\alpha(k)} \cdot \left(\nabla F_R(R^{\alpha(k)}) - \nabla F_R(R^{\alpha(k-1)})\right)$
    $R^{\alpha(k+1)} \leftarrow \arg\min_{Z^\alpha\in\mathcal{R}^\alpha} G_R^\alpha(Z^\alpha|Y^{\alpha(k)})$
  end for
  $R^{(k+1)} \leftarrow \begin{bmatrix} R^{1(k+1)} & \cdots & R^{A(k+1)} \end{bmatrix}$
  evaluate $\nabla F_R(R^{(k+1)})$
end for

$$G_R(R|R^{(k)}) \triangleq \sum_{\alpha\in\mathcal{A}} \sum_{(i,j)\in\vec{\mathcal{E}}^{\alpha\alpha}} \frac{1}{2}\kappa_{ij}^{\alpha\alpha}\|R_i^\alpha \widetilde{R}_{ij}^{\alpha\alpha} - R_j^\alpha\|^2 + \sum_{\substack{\alpha,\beta\in\mathcal{A},\\ \alpha\neq\beta}} \sum_{(i,j)\in\vec{\mathcal{E}}^{\alpha\beta}} \left[\kappa_{ij}^{\alpha\beta}\|R_i^\alpha \widetilde{R}_{ij}^{\alpha\beta} - P_{ij}^{\alpha\beta(k)}\|^2 + \kappa_{ij}^{\alpha\beta}\|R_j^\beta - P_{ij}^{\alpha\beta(k)}\|^2\right] + \frac{1}{2}\xi\|R - R^{(k)}\|^2, \tag{24}$$

in which $P_{ij}^{\alpha\beta(k)} \in \mathbb{R}^{d\times d}$ is defined as Eq. (7) and $\xi \geq 0$.

In a similar way to $G(X|X^{(k)})$ in Eq. (14), it can be shown that $G_R(R|R^{(k)})$ is a proximal operator of $F_R(R)$ at $R^{(k)}$ and there exists $G^\alpha(R^\alpha|R^{\alpha(k)})$ such that $G_R(R|R^{(k)}) = \sum_{i=1}^A G_R^\alpha(R^\alpha|R^{\alpha(k)})$ and

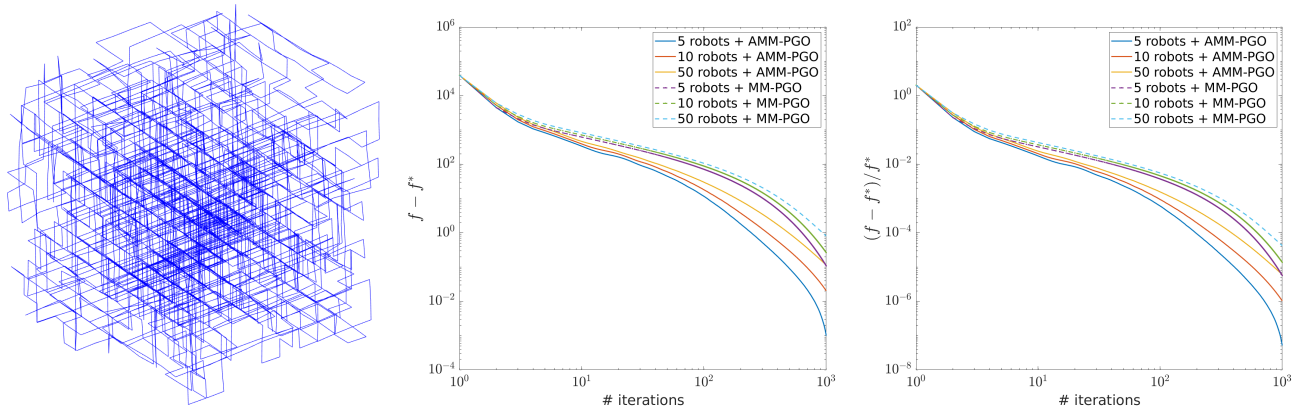
$$\min_{R\in\mathcal{R}} G_R(R|R^{(k)})$$

is equivalent to solving A independent convex optimization subproblems

$$\min_{R^\alpha\in\mathcal{R}^\alpha} G_R^\alpha(R^\alpha|R^{\alpha(k)}).$$

From Nesterov's method [19], [20], we obtain the AMM – Chordal method (Algorithm 3), which is an accelerated majorization minimization method to solve the distributed chordal initialization of Eq. (22). Furthermore, since the chordal initialization is a convex optimization problem, the AMM – Chordal method quadratically converges to the global optimum of Eq. (22) as follows.

Proposition 4. The AMM – Chordal method has a convergence rate of $O(1/k^2)$ to the global optimum of the distributed choral initialization of Eq. (22).



(a) (b) (c)

Fig. 1: The comparisons of the MM − PGO and AMM − PGO methods on 20 Cube datasets with 5, 10 and 50 robots in which the maximum number of iterations is 1000. Each Cube dataset has $12 \times 12 \times 12$ grids of side length of 1 m, 3600 poses, probability of loop closure of 0.1, an translational RSME of $\sigma_t = 0.02$ m and an angular RSME of $\sigma_R = 0.02\pi$ rad. The results are (a) an example of the Cube dataset, (b) suboptimality gap $f - f^*$ and (c) relative suboptimality gap $(f - f^*)/f^*$. In (b) and (c), f is the objective attained by the MM − PGO and AMM − PGO methods and f^* is the globally optimal objective attained by SE-Sync [4].

Proof. See [21, Appendix D]. □

The resulting solution to the chordal initialization might not satisfy the orthogonal constraints, and we need to project each R_i^α from $\mathbb{R}^{d\times d}$ to $SO(d)$ using the singular value decomposition [24] to get the initial guess $R^{(0)} \in SO(d)^n$ of the rotation $R \in SO(d)^n$.

It is possible to further obtain an initial guess $t^{(0)} \in t^{d\times n}$ of the translation $t \triangleq \begin{bmatrix} t^1 & \cdots & t^A \end{bmatrix} \in t^{d\times n}$ with $t^\alpha \triangleq \begin{bmatrix} t_1^\alpha & \cdots & t_{n_\alpha}^\alpha \end{bmatrix} \in \mathbb{R}^{d\times n_\alpha}$ by substituting $R^{(0)}$ into Eq. (3) and solving the optimization problem

$$\min_{t\in\mathcal{T}} F_t(t), \tag{25}$$

in which

$$F_t(t) \triangleq \sum_{\alpha\in\mathcal{A}} \sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\alpha\alpha}} \frac{1}{2}\tau_{ij}^{\alpha\alpha}\|R_i^{\alpha(0)}\tilde{t}_{ij}^{\alpha\alpha} + t_i^\alpha - t_j^\alpha\|^2 + \sum_{\substack{\alpha,\beta\in\mathcal{A},\\ \alpha\neq\beta}} \sum_{(i,j)\in\overrightarrow{\mathcal{E}}^{\alpha\beta}} \frac{1}{2}\tau_{ij}^{\alpha\beta}\|R_i^{\alpha(0)}\tilde{t}_{ij}^{\alpha\beta} + t_i^\alpha - t_j^\beta\|^2, \tag{26}$$

and $\mathcal{T} \triangleq \mathcal{T}^1 \times \cdots \times \mathcal{T}^A \subset \mathbb{R}^{d\times A}$ with $\mathcal{T}^1 \triangleq \{t^1 \in \mathbb{R}^{d\times n_1} | t_1^1 = \mathbf{0} \in \mathbb{R}^d\}$ and $\mathcal{T}^\alpha \triangleq \mathbb{R}^{d\times n_\alpha}$ if $\alpha \neq 1$. Following a similar procedure to Eq. (22), Eq. (25) can be solved with the majorization minimization method, from which we obtain an initial guess $t^{(0)} \in \mathbb{R}^{d\times n}$ of the translation $t \in \mathbb{R}^{d\times n}$.

VIII. Numerical Experiments

In this section, we evaluate the performance of our proposed majorization minimization (MM − PGO and AMM − PGO) methods for distributed PGO on the simulated Cube datasets and a number of 2D and 3D SLAM benchmark datasets [4]. We also make comparisons with the distributed Gauss-Seidel (DGS) method in [13], which is the state-of-the-art method for distributed PGO. We use the certifiably-correct algorithm SE-Sync [4] to provide the ground truth and globally optimal objective for all the datasets. All the experiments have been performed on a laptop with an Intel i7-8750H CPU and 32GB of RAM running Ubuntu 18.04 and using g++ 7.8 as C++ compiler. For both MM − PGO and AMM − PGO methods, ξ in Eq. (13) is chosen to be 0.001, and the DGS method uses the default settings.

A. Cube Datasets

In this section, we evaluate the convergence of the MM − PGO and AMM − PGO methods on 20 simulated Cube datasets with 5, 10 and 50 robots.

In the experiments, a Cube dataset (Fig. 1(a)) has $12 \times 12 \times 12$ cube grids with 1 m side length, and a path of 3600 poses along the rectilinear edge of the cube grid, and odometric measurements between all the pairs of sequential poses, and loop-closure measurements between nearby but non-sequential poses that are randomly available with a probability of 0.1. We generate the odometric and loop-closure measurements according to the noise models in [4] with an expected translational RMSE of $\sigma_t = 0.02$ m and an expected angular RMSE of $\sigma_R = 0.02\pi$ rad.

The results of a maximum of 1000 iterations are as shown in Fig. 1, which has the suboptimality gap $f - f^*$ and the relative suboptimality gap $(f - f^*)/f^*$ in (b) and (c), respectively. In Fig. 1, f is the objective attained by the the MM − PGO and AMM − PGO methods and f^* is the globally optimal objective attained by SE-Sync [4]. It can be seen from Fig. 1 that both MM − PGO and AMM − PGO methods have better convergence as the number of robots decreases, which is not surprising since $E(X|X^{(k)})$ in Eqs. (6) and (9) results in a tighter approximation of distributed PGO in Eqs. (3) and (4) with fewer robots. In addition, the AMM − PGO method always outperforms the MM − PGO method in terms of the convergence, which suggests that Nesterov's method accelerates distributed PGO. In particular, it should be noted that the AMM − PGO method with 50 robots converges almost faster than the MM − PGO method

with 5 robots, which further indicates that the AMM – PGO method is well suited for distributed PGO considering the fact that no convergence guarantees are compromised and only limited extra computation is introduced in acceleration.

B. SLAM Benchmark Datasets

In this section, we compare the MM – PGO and AMM – PGO methods with the distributed Gauss-Seidel (DGS) method [13], which is the state-of-the-art method for distributed PGO on a number of 2D and 3D SLAM benchmark datasets. It should be noted that originally the MM – PGO and AMM – PGO methods and the DGS method adopt different algorithms to initialize the rotation $R \in SO(d)^n$, and the MM – PGO and AMM – PGO methods initialize the translation $t \in \mathbb{R}^{d\times n}$, whereas the DGS method does not. Therefore, in order to make the comparisons fair, we initialize the MM – PGO and AMM – PGO methods and the DGS method with the centralized chordal initialization for both the rotation $R \in SO(d)^n$ and the translation $t \in \mathbb{R}^{d\times n}$.

In the experiments, the DGS method is assigned an ordering according to which the poses of each robot are updated, which improves the convergence performance. Even though such an ordering reduces the number of iterations, parts of the robots have to stay idle until poses of the other robots are updated, which might increase the overall computational time in the end, and thus, is not that desirable in distributed PGO. In contrast, the MM – PGO and AMM – PGO methods update the poses of all the robots at the same time and no ordering is needed.

The MM – PGO and AMM – PGO methods and the DGS method are evaluated with 10 robots. The results are shown as Tables I and II, in which f^* is the objective value of the globally optimal objective attained by SE-Sync [4] and f is the objective attained by each method with the given number of iterations, i.e., 100, 250 and 1000. For each dataset and each number of iterations, the best and second results are colored in red and blue, respectively. From Tables I and II, the AMM – PGO method outperforms the DGS method [13] on all the datasets except the intel dataset, for which the chordal initialization is sufficient for one Gauss-Newton step to attain the global optimum. Even though the MM – PGO method is inferior to the AMM – PGO method, it still has a better performance on most of the datasets than the DGS method. Furthermore, the MM – PGO and AMM – PGO are theoretically guaranteed to improve the estimates as the number of iterations increases, whereas the DGS method, which is equivalent to a one-step Gauss-Newton method, is not — on the ais2klinik dataset, the DGS method has the objective of 250 iterations greater than that of 100 iterations — and as a matter of fact, as discussed in [25], the convergence of the Gauss-Newton method without stepsize tuning can not be guaranteed.

IX. Conclusion

In this paper, we have presented majorization minimization methods for distributed PGO that has important applications in multi-robot SLAM. We have proved that our methods are guaranteed to converge to first-order critical points under mild conditions. Furthermore, we have accelerated majorization minimization methods for distributed PGO using Nesterov's method with no compromise of convergence guarantees. We have also presented majorization minimization methods for the distributed chordal initialization that converge quadratically. The efficacy of this work has been validated through applications on a number of 2D and 3D SLAM datasets and comparisons with existing state-of-the-art methods [13], which indicates that our methods converge faster and result in better solutions to distributed PGO.

References

[1] A. Cunningham, M. Paluri, and F. Dellaert, "DDF-SAM: Fully distributed SLAM using constrained factor graphs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.

[2] J. Dong, E. Nelson, V. Indelman, N. Michael, and F. Dellaert, "Distributed real-time cooperative localization and mapping using an uncertainty-aware expectation maximization approach," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[3] T. Cieslewski, S. Choudhary, and D. Scaramuzza, "Data-efficient decentralized visual SLAM," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[4] 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," *arXiv preprint arXiv:1612.07386*, 2016.

[5] F. Dellaert, "Factor graphs and GTSAM: A hands-on introduction," Georgia Institute of Technology, Tech. Rep., 2012.

[6] L. Carlone, G. C. Calafiore, C. Tommolillo, and F. Dellaert, "Planar pose graph optimization: Duality, optimal solutions, and verification," *IEEE Transactions on Robotics*, 2016.

[7] L. Carlone, D. M. Rosen, G. Calafiore, J. J. Leonard, and F. Dellaert, "Lagrangian duality in 3D SLAM: Verification techniques and optimal solutions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[8] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011.

[9] G. Grisetti, C. Stachniss, and W. Burgard, "Nonlinear constraint network optimization for efficient map learning," *IEEE Transactions on Intelligent Transportation Systems*, 2009.

[10] J. Briales and J. Gonzalez-Jimenez, "Cartan-sync: Fast and global SE(d)-synchronization," *IEEE Robotics and Automation Letters*, 2017.

[11] T. Fan, H. Wang, M. Rubenstein, and T. Murphey, "Efficient and guaranteed planar pose graph optimization using the complex number representation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2019.

[12] ——, "CPL-SLAM: Efficient and certifiably correct planar graph-based SLAM using the complex number representation," *IEEE Transactions on Robotics*, 2020.

[13] S. Choudhary, L. Carlone, C. Nieto, J. Rogers, H. I. Christensen, and F. Dellaert, "Distributed mapping with privacy and communication constraints: Lightweight algorithms and object-based models," *The International Journal of Robotics Research*, 2017.

[14] S. Choudhary, L. Carlone, H. I. Christensen, and F. Dellaert, "Exactly sparse memory efficient slam using the multi-block alternating direction method of multipliers," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[15] R. Tron and R. Vidal, "Distributed 3-D localization of camera sensor networks from 2-D image measurements," *IEEE Transactions on Automatic Control*.

[16] 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 International Conference on Robotics and Automation (ICRA)*, 2015.

[17] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, 2004.

[18] T. Fan and T. Murphey, "Generalized proximal methods for pose graph optimization," in *International Symposium on Robotics Research (ISRR)*, 2019.

Dataset	# poses	# edges	f^*	# iterations	f		
					MM − PGO [ours]	AMM − PGO [ours]	DGS [13]
ais2klinik	15115	16727	1.885×10^2	100	2.012×10^2	1.982×10^2	8.646×10^2
				250	1.992×10^2	1.962×10^2	9.315×10^2
				1000	1.961×10^2	1.933×10^2	3.350×10^2
city	10000	20687	6.386×10^2	100	6.556×10^2	6.524×10^2	7.989×10^2
				250	6.529×10^2	6.484×10^2	7.055×10^2
				1000	6.473×10^2	6.418×10^2	6.562×10^2
CSAIL	1045	1172	3.170×10^1	100	3.170×10^1	3.170×10^1	3.248×10^1
				250	3.170×10^1	3.170×10^1	3.179×10^1
				1000	3.170×10^1	3.170×10^1	3.171×10^1
M3500	3500	5453	1.939×10^2	100	1.952×10^2	1.947×10^2	1.956×10^2
				250	1.947×10^2	1.944×10^2	1.946×10^2
				1000	1.943×10^2	1.940×10^2	1.943×10^2
intel	1728	2512	5.235×10^1	100	5.257×10^1	5.252×10^1	5.255×10^1
				250	5.252×10^1	5.248×10^1	5.244×10^1
				1000	5.243×10^1	5.240×10^1	5.238×10^1
MITb	808	827	6.115×10^1	100	6.347×10^1	6.228×10^1	9.244×10^1
				250	6.220×10^1	6.153×10^1	7.453×10^1
				1000	6.136×10^1	6.117×10^1	6.887×10^1

TABLE I: Results of the 2D SLAM Benchmark datasets with 10 robots, in which the best and second results are colored in red and blue, respectively.

Dataset	# poses	# edges	f^*	# iterations	f		
					MM − PGO [ours]	AMM − PGO [ours]	DGS [13]
sphere	2500	4949	1.687×10^3	100	1.691×10^3	1.687×10^3	1.688×10^3
				250	1.687×10^3	1.687×10^3	1.688×10^3
				1000	1.687×10^3	1.687×10^3	1.687×10^3
torus	5000	9048	2.423×10^4	100	2.424×10^4	2.423×10^4	2.425×10^4
				250	2.423×10^4	2.423×10^4	2.425×10^4
				1000	2.423×10^4	2.423×10^4	2.424×10^4
grid	8000	22236	8.432×10^4	100	8.433×10^4	8.432×10^4	8.433×10^4
				250	8.432×10^4	8.432×10^4	8.433×10^4
				1000	8.432×10^4	8.432×10^4	8.433×10^4
garage	1661	6275	1.263×10^0	100	1.279×10^0	1.275×10^0	1.319×10^0
				250	1.274×10^0	1.270×10^0	1.287×10^0
				1000	1.269×10^0	1.266×10^0	1.273×10^0
cubicle	5750	16869	7.171×10^2	100	7.228×10^2	7.204×10^2	7.317×10^2
				250	7.206×10^2	7.189×10^2	7.231×10^2
				1000	7.185×10^2	7.176×10^2	7.205×10^2
rim	10195	29743	5.461×10^3	100	5.779×10^3	5.674×10^3	6.114×10^2
				250	5.695×10^3	5.573×10^3	6.035×10^3
				1000	5.549×10^3	5.486×10^3	5.932×10^3

TABLE II: Results of the 3D SLAM Benchmark datasets with 10 robots, in which the best and second results are colored in red and blue, respectively.

[19] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence O (1/kˆ 2)," in *Doklady AN USSR*, vol. 269, 1983, pp. 543–547.

[20] ——, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

[21] T. Fan and T. Murphey, "Majorization minimization methods to distributed pose graph optimization with convergence guarantees," 2019. [Online]. Available: https://northwestern.box.com/s/x5rchjz7dd1maf3xsjtznq8sh8j785zw

[22] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, 2016.

[23] C. Jin, P. Netrapalli, and M. I. Jordan, "Accelerated gradient descent escapes saddle points faster than gradient descent," in *Conference On Learning Theory*, 2018.

[24] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 4, pp. 376–380, 1991.

[25] D. M. Rosen, M. Kaess, and J. J. Leonard, "RISE: An incremental trust-region method for robust online sparse least-squares estimation," *IEEE Transactions on Robotics*, 2014.