

Linear Estimation on SO(2) for Graph-based Simultaneous Localization and Mapping

Luca Carlone and Andrea Censi

Abstract—In this paper we discuss the problem of estimating orientation of nodes in a *pose graph* from relative measurements. We formalize some intuitions of previous work showing that, when mapping the maximum likelihood problem from the manifold SO(2) to a vector space, it is necessary to include integer-valued unknowns (*regularization terms*). We show that, in general, the introduction of such regularization terms makes the solution of the problem challenging, since the maximum likelihood problem becomes a quadratic optimization problem with integer constraints. However, we propose a technique for reducing the possible choices of the regularization terms by discarding choices with negligible probability of being correct. Experimental results show that, in common problems, a single choice has non-negligible probability of being correct. Once a correct regularization term is retrieved, the linearity of the framework assures the consistency of the estimation errors with the corresponding covariance matrix, hence providing probabilistic assessments on the quality of the orientation estimate. As a by-product, we show that, exploiting the results of the present paper, it is possible to estimate a correct pose graph configuration also when state-of-the-art approaches are likely to be stuck in a local minimum.

I. INTRODUCTION

Simultaneous Localization And Mapping (SLAM) has been recognized to be a central topic or research within the robotic community. Its importance stems from the fact that the construction of a world model and the estimation of robot pose is a crucial prerequisite for performing tasks in unstructured environments. Most of the techniques for SLAM are based on nonlinear estimation methods and the nonlinearity of the underlying problem makes challenging to provide analytical assessments on the results of the estimation problem (i.e., accuracy or consistency of the estimate with respect to the actual nonlinear solution). In this context we are interested in a batch SLAM problem in which measurements acquired by the robot are modeled as a network of constraints in a *pose graph*; in this case the problem can be formulated in terms of nonlinear optimization and the maximum likelihood solution can be retrieved using iterative nonlinear techniques. The seminal paper [14] paved the way for the solution of this batch problem. The framework of Lu and Milios was extended to the general case of a graph containing both robot poses and landmark positions in [20]. Thrun and Montemerlo showed that it is possible to marginalize out variables corresponding to landmarks, hence reducing the problem to the pose estimation setup discussed in [14]. Konolige investigated a reduction scheme for the purpose of improving the computational effort of nonlinear optimization [11]. Frese *et al.* proposed a multilevel relaxation approach

for batch SLAM [4], allowing to considerably reduce the computational time of optimization by applying a multi-grid algorithm. A further breakthrough in the literature of graph-based approaches came with the use of *incremental pose parametrization*, proposed by Olson *et al.* in [18]. Grisetti *et al.* extended such framework, taking advantage of the use of stochastic gradient descent in planar and three-dimensional scenarios [6]. Some original attempts to exploit the mathematical structure of the nonlinear pose graph optimization have been recently proposed [5], [8], [7]. The aforementioned techniques are iterative, in the sense that, at each iteration, they solve a convex approximation of the original problem, and use such local solution to update the estimate. This process is then repeated until the optimization variable converges to a minimum of the cost function.

Although state-of-the-art approaches for graph-based SLAM appear to work well in common problems instances, none of them can guarantee to attain a global minimum of the objective function. Therefore, they cannot provide assessments on the quality of the resulting estimate. In graph-based SLAM the assessments we are looking for answer the following question: how close is the estimate from the actual pose graph configuration? This question is hard to answer in general for the nonlinear nature of the problem. Since it is well-known that the nonlinearity of the problem lies in the orientation variables, in this work we investigate if it is possible to retrieve an estimate of the orientations of nodes in a pose graph, with probabilistic guarantees of its quality (i.e., consistency of the estimation errors). The derivation of this paper is widely motivated by the recent result [2], which shows that it is possible to compute an accurate estimate of the pose graph configuration once a reliable estimate of nodes' orientation is available; in particular, [2] highlights that, when rephrasing the orientation estimation problem in a linear framework, it is necessary to correct relative orientation measurements with some multiples of 2π (*regularization terms*) for the result to be meaningful. In this work we consider the same formulation of [2] and we focus on the first phase of the SLAM approach, i.e., the *linear orientation estimation*. We then formally show that the multiples of 2π may appear when mapping the original problem, defined on the manifold SO(2), to a vector space. In general, the introduction of such regularization terms makes the solution of the problem challenging, since the maximum likelihood problem becomes a quadratic optimization problem with integer constraints. Intuitively, there are infinite choices of the regularization terms that has to be considered. However, as a further contribution of this work, we propose a *screening* that allows to reduce the possible choices of the regularization terms by discarding choices with negligible probability of being correct. As a result, we show that it is possible to retrieve a finite set of choices, that contains the correct regularization terms with arbitrary high probability. Experimental results show

This work was funded by MACP4LOG grant (RU/02/26) from Piemonte Region, Italy, and by PRIN grant n. 20087W5P2K from the Italian Ministry of University and Research.

L. Carlone is with CSPP, Laboratorio di Meccatronica, Politecnico di Torino, Italy. luca.carlone@polito.it

A. Censi is with the Control and Dynamical Systems department, California Institute of Technology, United States. andrea@cds.caltech.edu

that, in common problem instances, a single choice has non-negligible probability of being correct; moreover, also after inflating a large amount of noise in the orientation measurements, a suitable choice of the matrices involved in the screening is still able to determine few admissible choices for the regularization terms. As a by-product, we show that, exploiting the results of this paper, the approach proposed in [2] can estimate a correct pose graph configuration also when state-of-the-art approaches are stuck in a local minimum. Finally, from the derivation reported in this paper it turns out that the *regularization* procedure proposed in [2] corresponds to select the maximum likelihood regularization terms given measurements (then such terms are used for computing the maximum likelihood solution for nodes' orientation); however, we here show that a formal approach requires solving the maximum likelihood problem on both variables (regularization terms and orientations) at the same time.

Notation and Preliminaries. \mathbf{I}_n denotes the $n \times n$ identity matrix, $\mathbf{0}_n$ denotes a (column) vector of all zeros of dimension n . The cardinality of a generic set S is written as $|S|$. A *directed* graph \mathcal{G} is a pair $(\mathcal{V}, \mathcal{E})$, where \mathcal{V} is a finite set of elements, called *vertices* or *nodes*, and \mathcal{E} is a set containing ordered pairs of nodes. A generic element $e \in \mathcal{E}$, referred to as *edge*, is in the form $e = (i, j)$, meaning that edge e , incident on nodes i and j , leaves i (*tail*) and is directed towards node j (*head*) [9]. The number of nodes and edges are denoted with $n + 1$ (the reason for this choice will be clear later) and m , respectively, i.e., $|\mathcal{V}| = n + 1$ and $|\mathcal{E}| = m$. The *incidence matrix* \mathcal{A} of a directed graph is a matrix in $\mathbb{R}^{(n+1) \times m}$ in which each column contains the information of an edge in \mathcal{E} ; in particular the column corresponding to the edge $e = (i, j)$, has the i -th element equal to -1 , the j -th element equal to $+1$ and all the others equal to zero. A *spanning tree* of \mathcal{G} is a subgraph with n edges that contains all the nodes in \mathcal{G} . The edges of \mathcal{G} that do not belong to a given spanning tree of the graph are referred to as *chords* [3]. The *exponential map* (for the circle group), $\Phi : \mathbb{R} \rightarrow \text{SO}(2)$, maps any real number into an element of the special orthogonal group $\text{SO}(2)$. In particular the exponential map of $\theta \in \mathbb{R}$ is $\check{\theta} \doteq \Phi(\theta) \doteq \text{mod}_{2\pi}(\theta)$, where $\text{mod}_{2\pi}$ is the modulo 2π operator. Accordingly, we define the *inverse map* as $\Phi^{-1} : \text{SO}(2) \rightarrow \mathbb{R}$, that can be written explicitly as $\theta \doteq \Phi^{-1}(\check{\theta}) \doteq \check{\theta} - 2k\pi$, with $\check{\theta} \in \text{SO}(2)$ and $k \in \mathbb{Z}$.

II. PROBLEM STATEMENT AND MOTIVATIONS

Let $\mathcal{V} = \{v_0, \dots, v_n\}$ be a set of $n + 1$ nodes (representing subsequent poses assumed by a mobile robot traveling in a planar setup) and $\check{\Theta} = \{\check{\theta}_0, \dots, \check{\theta}_n\}$, $\check{\theta}_i \in \text{SO}(2)$, $i = 0, 1, \dots, n$ denote the corresponding set of absolute orientations assumed by the robot. Suppose that it is possible to measure the relative orientation between some nodes, say nodes (i, j) ; in particular node i can measure the orientation of j in its local reference frame \mathcal{R}_i , i.e., $\check{\delta}_{ij} = \Phi(\check{\theta}_j - \check{\theta}_i + \epsilon_{ij}) \in \text{SO}(2)$, where ϵ_{ij} is a zero mean Gaussian noise with standard deviation σ_{ij} . Now the objective is to find a maximum likelihood estimate of the unknown $\check{\Theta}$. The problem can be naturally modeled using graph formalism: each node in the set \mathcal{V} corresponds to a vertex of a directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (often referred to as *pose graph*), where \mathcal{E} (graph edges) is the set containing the unordered node pairs (i, j) such that a relative pose measurement exists between i and j . By convention, if an edge is directed from node i to node j , the corresponding relative measurement is expressed in the reference frame of node i . Moreover, the first node is

assumed to have 0 orientation (i.e., it is the reference frame in which the other orientations have to be estimated).

In graph-SLAM problems, for every $(i, j) \in \mathcal{E}$, also the relative position information is usually available [18]; however, in this work we will focus on the orientation measurements, since, as pointed out in [2], once a good estimate of robot orientation is available, it is possible for approximate the nonlinear pose optimization with a linear estimation problem. Moreover, the relative position measurements belong to a vector space and linear estimation theory easily applies, whereas orientations belong to a manifold, $\text{SO}(2)$ in our planar setup, and the application of linear estimation needs be carefully investigated. This reasons motivate the following development.

III. LINEAR ESTIMATION ON $\text{SO}(2)$

Let us rewrite the measurement model:

$$\check{\delta}_{ij} = \Phi(\check{\theta}_j - \check{\theta}_i + \epsilon_{ij}) \in \text{SO}(2), \quad (i, j) \in \mathcal{E}. \quad (1)$$

Notice that we can apply the following substitution without altering the result:

$$\check{\delta}_{ij} = \Phi(\Phi^{-1}(\check{\theta}_j) - \Phi^{-1}(\check{\theta}_i) + \epsilon_{ij}), \quad (i, j) \in \mathcal{E}, \quad (2)$$

since the exponential map is invariant under addition of multiples of 2π . Therefore, recalling that $\Phi^{-1} : \text{SO}(2) \rightarrow \mathbb{R}$, we can rephrase the problem in the unknown $\theta_i \in \mathbb{R}$, $i = 0, 1, \dots, n$ as follows:

$$\check{\delta}_{ij} = \Phi(\theta_j - \theta_i + \epsilon_{ij}), \quad (i, j) \in \mathcal{E}. \quad (3)$$

Finally we can formulate the measurement model as a linear function by applying the inverse map Φ^{-1} to both members of (3):

$$\check{\delta}_{ij} - 2k_{ij}\pi = \theta_j - \theta_i + \epsilon_{ij}, \quad (i, j) \in \mathcal{E}. \quad (4)$$

In order to write the measurement model in a more compact form, let us number the relative orientation measurements from 1 to m . Define accordingly $\check{\delta} = [\check{\delta}_1 \ \check{\delta}_2 \ \dots \ \check{\delta}_m]^\top$ (*measurement vector*), $k = [k_1 \ k_2 \ \dots \ k_m]^\top$ (*regularization vector*), $\epsilon = [\epsilon_1 \ \epsilon_2 \ \dots \ \epsilon_m]^\top$ (*noise vector*), and $P_\delta = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_m^2)$ (*measurement covariance*). Furthermore, let $\theta = [\theta_0 \ \dots \ \theta_n]^\top \in \mathbb{R}^{n+1}$ be the unknown *nodes' orientation*. Then (4) can be rewritten as:

$$A^\top \theta = \check{\delta} - 2k\pi - \epsilon, \quad k \in \mathbb{Z}^m, \quad (5)$$

where A is the *incidence matrix* of the graph \mathcal{G} . The problem of retrieving a maximum likelihood estimate of robot orientations can be then formulated as the minimization of the following cost function:

$$\begin{aligned} \min_{\theta, k} \|A^\top \theta - \check{\delta} + 2k\pi\|_{P_\delta}^2 \\ \text{subject to: } k \in \mathbb{Z}^m, \end{aligned} \quad (6)$$

where the *Mahalanobis distance* is defined as $\|x\|_P \doteq x^\top P^{-1}x$. The previous problem is hard to solve, due to its combinatorial nature. However, as pointed out in [2], once the regularization vector k is known, the estimation problem becomes linear and can be solved in closed-form. In particular, if we know the regularization vector, say $k = k^*$, then the solution of the problem can be computed as $\theta^* = (AP_\delta^{-1}A^\top)^{-1}AP_\delta^{-1}(\check{\delta} - 2k^*\pi)$. Moreover, for the linearity of the estimation problem we can guarantee that the covariance of the estimation errors is $P_\theta = (AP_\delta^{-1}A^\top)^{-1}$ [16] (a

probabilistic assessment of the quality of the estimate). If we denote $\theta^* = [\theta_0^* \dots \theta_n^*] \in \mathbb{R}^{n+1}$, it is then easy to verify from equations (1) and (2), that the corresponding solution in $\text{SO}(2)$ is given by $\Theta^* = \{ \text{mod } 2\pi\theta_0^*, \dots, \text{mod } 2\pi\theta_n^* \}$.

In the following derivation we show that it is possible to mitigate the combinatorial complexity of retrieving the correct solution, thus assuring with arbitrary high probability to estimate the correct orientations. In particular, we show that it is possible to compute a finite set $\mathcal{K} = \{k_1, k_2, \dots, k_V\} \subset \mathbb{Z}^m$, such that $k^* \in \mathcal{K}$ with high probability. Therefore, we will transform (6) into:

$$\begin{aligned} \min_{\theta, k} \quad & \|A^\top \theta - \check{\delta} + 2k\pi\|_{P_\delta}^2 \\ \text{subject to:} \quad & k \in \mathcal{K}. \end{aligned} \quad (7)$$

If $|\mathcal{K}|$ is small, we can solve (7) by computing the solution θ_k^* for each $k \in \mathcal{K}$, and the maximum likelihood estimate will be the pair (θ_k^*, k) minimizing the objective function in (7). The rest of this article addresses the problem of computing the set \mathcal{K} .

IV. MITIGATING THE COMBINATORIAL COMPLEXITY

For limiting the possible choices of k we should exploit problem constraints. Let us start from the measurement model (5). If we define $\delta \doteq A^\top \theta$, $\delta \in \mathbb{R}^m$ needs to satisfy the *zero-sum property*, i.e., after mapping nodes' orientation in a vector space, the actual relative orientations between nodes have to sum up to zero along cycles in the graph [19]. A necessary and sufficient condition for the zero-sum property to be satisfied is that $C\delta = C(A^\top \theta) = \mathbf{0}_\ell$, where $C \in \mathbb{Z}^{\ell \times m}$ is a *cycle basis matrix* of graph \mathcal{G} , and $\ell = m - n$. Let us multiply both members of equation (5) by C :

$$C(A\theta) = C(\check{\delta} - 2k\pi - \epsilon), \quad k \in \mathbb{Z}^m. \quad (8)$$

We already observed that the first member needs be zero, hence we can write (8) as:

$$\begin{aligned} 2\pi Ck &= C\check{\delta} - C\epsilon, \quad k \in \mathbb{Z}^m \\ \iff Ck &= \frac{1}{2\pi}C\check{\delta} - \frac{1}{2\pi}C\epsilon, \quad k \in \mathbb{Z}^m. \end{aligned} \quad (9)$$

Since C is an integer-valued matrix and $k \in \mathbb{Z}^m$, also Ck needs be integer-valued and we can apply the change of variables $\gamma \doteq Ck$ in (9):

$$\begin{cases} \gamma = \frac{1}{2\pi}C\check{\delta} - \frac{1}{2\pi}C\epsilon \\ \gamma \in \mathbb{Z}^\ell \end{cases}. \quad (10)$$

In the previous expression we can identify two conditions on γ : (i) a probabilistic condition (ϵ is a random vector) that imposes γ to be distributed according to a Gaussian with mean $\mu_\gamma = \frac{1}{2\pi}C\check{\delta}$ and covariance matrix $P_\gamma = \frac{1}{4\pi^2}CP_\delta C^\top$; (ii) a deterministic condition that imposes γ to be an integer. In the following section we propose a simple procedure for computing the vectors γ that satisfy equation (10) with arbitrary high probability. Then, in Section IV-C we discuss how to retrieve the set of regularization vectors \mathcal{K} from γ .

A. First Screening on γ : Marginal Probabilities

We can express equation (10) as follows:

$$\begin{cases} \gamma \sim \mathcal{N}(\mu_\gamma, P_\gamma) \\ \gamma \in \mathbb{Z}^\ell \end{cases}, \quad 1$$

¹For simplicity we use the notation $\gamma \sim \mathcal{N}(\mu_\gamma, P_\gamma)$ meaning that $\text{prob}(\gamma|\check{\delta})$ is Normally distributed according to $\mathcal{N}(\mu_\gamma, P_\gamma)$

where $\mathcal{N}(\mu_\gamma, P_\gamma)$ denotes a Normal density with mean μ_γ and covariance matrix P_γ . Let γ_i and μ_{γ_i} denote the i -th element of γ and μ_γ , respectively, and $\sigma_{\gamma_i}^2$ denote the i -th diagonal element of the covariance matrix P_γ . Then, considering the marginal probabilities of each element of γ , we can write:

$$\begin{cases} \gamma_i \sim \mathcal{N}(\mu_{\gamma_i}, \sigma_{\gamma_i}^2) \\ \gamma_i \in \mathbb{Z} \end{cases}. \quad (11)$$

From the first condition in equation (11), we have that, for a desired confidence level $(1 - \alpha)$, γ_i belongs to the interval $\check{\Gamma}_i = [\mu_{\gamma_i} - \sqrt{\chi_{1, (1-\alpha)}^2 \sigma_{\gamma_i}^2}, \mu_{\gamma_i} + \sqrt{\chi_{1, (1-\alpha)}^2 \sigma_{\gamma_i}^2}]$, with probability $(1 - \alpha)$, where $\chi_{\beta, (1-\alpha)}^2$ is the quantile of the χ^2 distribution with β degrees of freedom and upper tail probability equal to α [1]. Considering also the second condition in (11) we obtain that the following set has to contain with probability $(1 - \alpha)$ the correct value of γ_i , $i = 1, 2, \dots, \ell$:

$$\Gamma_i = \check{\Gamma}_i \cap \mathbb{Z} \quad (12)$$

It is easy to see that each set Γ_i contains all the integers in the interval defined by $\check{\Gamma}_i$. Therefore we restricted the possible selections of γ from the infinite set \mathbb{Z}^ℓ to the finite set $\Gamma = \Gamma_1 \times \Gamma_2 \times \dots \times \Gamma_\ell$; the previous Cartesian product means that the set Γ contains all the vectors in the form $\gamma = [\gamma_1 \ \gamma_2 \ \dots \ \gamma_\ell]$, such that $\gamma_i \in \Gamma_i$.

This screening can be made even more effective by devising an iterative procedure for selecting the set of γ having non-negligible probability of being correct. Let us call $\Gamma_i[1]$ the set of integers obtained from equation (12), for $i = 1, 2, \dots, \ell$. It is possible that, for some i , $\Gamma_i[1]$ contains a single integer, say $\Gamma_i[1] = \{u_i\}$; therefore, the corresponding correction term γ_i can be uniquely determined. Let us call $\mathcal{U}[1]$ the set of indices corresponding to the entries of γ that can be uniquely determined, i.e., $\mathcal{U}[1] = \{i : |\Gamma_i[1]| = 1\}$. Accordingly, we call $\check{\mathcal{U}}[1]$ the set of indices of the elements of γ that still present ambiguities, and we call $\gamma[2]$ the vector containing elements $\gamma_i, i \in \check{\mathcal{U}}[1]$. We can improve the estimate for the elements of $\gamma[2]$, computing the conditional probability $\text{prob}(\gamma[2] | \{\gamma_i = u_i, i \in \mathcal{U}[1]\})$. The previous probability is a Normal distribution, say $\mathcal{N}(\mu_\gamma[2], P_\gamma[2])$, and can be easily obtained from $\mathcal{N}(\mu_\gamma, P_\gamma)$ by conditioning on $\gamma_i = u_i, i \in \mathcal{U}[1]$. Then, we can recompute the $(1 - \alpha)$ -admissible sets as:

$$\begin{cases} \Gamma_i[2] = \check{\Gamma}_i[2] \cap \mathbb{Z} & i \in \check{\mathcal{U}}[1] \\ \Gamma_i[2] = \Gamma_i[1] & i \in \mathcal{U}[1] \end{cases},$$

with $\check{\Gamma}_i[2] = [\mu_{\gamma_i}[2] - \sqrt{\chi_{1, (1-\alpha)}^2 \sigma_{\gamma_i}[2]}, \mu_{\gamma_i}[2] + \sqrt{\chi_{1, (1-\alpha)}^2 \sigma_{\gamma_i}[2]}]$. In general, at iteration j , the $(1 - \alpha)$ -admissible sets become:

$$\begin{cases} \Gamma_i[j] = \check{\Gamma}_i[j] \cap \mathbb{Z} & i \in \check{\mathcal{U}}[j-1] \\ \Gamma_i[j] = \Gamma_i[j-1] & i \in \mathcal{U}[j-1] \end{cases},$$

where $\check{\Gamma}_i[j]$ are the $(1 - \alpha)$ -confidence intervals, defined over the Gaussian density $\text{prob}(\gamma[j] | \{\gamma_i = u_i, i \in \mathcal{U}[j-1]\})$. Therefore, at each iteration, we shrink the intervals for the elements of γ that are still ambiguous. The iterative procedure stops when no further reduction of the cardinality of the sets $\Gamma_i, i = 1, 2, \dots, \ell$ is possible. It can be easily seen that an upper-bound on the number of iterations is ℓ : at each iteration, at least one element in $\gamma \in \mathbb{Z}^\ell$ needs be included in $\mathcal{U}[j]$, otherwise the algorithm halts; moreover, after fixing

all ℓ elements no further iteration is needed. In practice, the number of iterations in this first screening is small (see the experimental section). As outcome of the screening on marginal probabilities we have restricted each γ_i to some integer values, $\Gamma_i[N]$, $i = 1, 2, \dots, \ell$, where N is the number of iterations performed by the algorithm described so far. Therefore, the only $(1 - \alpha)$ -admissible values for γ have to be found in $\Gamma[N] = \Gamma_1[N] \times \Gamma_2[N] \times \dots \times \Gamma_\ell[N]$.

B. Second Screening on γ : Joint Probability

In the previous section we used the marginal probabilities of $\mathcal{N}(\mu_\gamma, P_\gamma)$ for selecting the set each γ_i has to belong to. In particular, after N iterations $\Gamma_i[N] = \{u_i\}$, $i \in \mathcal{U}[N]$ (elements with a single possible choice), whereas $|\Gamma_i[N]| > 1$, $i \in \bar{\mathcal{U}}[N]$. Let us call $\gamma[N+1]$ the vector comprising γ_i , $i \in \mathcal{U}[N]$. The vector $\gamma[N+1]$ can assume values in the set $\times_{i \in \bar{\mathcal{U}}[N]} \Gamma_i[N]$, where the symbol $\times_{i \in \bar{\mathcal{U}}[N]}$ denotes the cartesian product of the sets whose index is in $\bar{\mathcal{U}}[N]$. Computing the Cartesian product we obtain a collection of vectors $\times_{i \in \bar{\mathcal{U}}[N]} \Gamma_i[N] = \{\gamma^1, \gamma^2, \dots, \gamma^W\}$ that have non-negligible probability of being the correct choice of $\gamma[N]$. By construction, all vectors in $\{\gamma^1, \gamma^2, \dots, \gamma^W\}$ are integer-valued. As an output of this first screening we also obtained a Gaussian density associated with the elements of $\gamma[N]$, i.e., $\mathcal{N}(\mu_\gamma[N], P_\gamma[N])$. From the Gaussian density, for a given confidence level, say $(1 - \alpha)$, we can build an ellipsoid $\bar{\Gamma}[N]$ with center $\mu_\gamma[N]$ and shape matrix $\chi_{|\bar{\mathcal{U}}[N]|, 1-\alpha}^2 P_\gamma[N]$ that has to contain the correct value of $\gamma[N]$ with probability $(1 - \alpha)$. Therefore the second screening simply selects among vectors $\{\gamma^1, \gamma^2, \dots, \gamma^W\}$ the ones falling in the $(1 - \alpha)$ -confidence ellipsoid. We shall call $\Gamma[N+1] \subseteq \{\gamma^1, \gamma^2, \dots, \gamma^W\}$ the set of vectors falling in the confidence ellipsoid.

Also in this case the procedure can be made iterative. After computing $\Gamma[N+1]$, it is possible that, for some j , the j -th entry of all vectors in $\Gamma[N+1]$ is equal: this means that the value of the j -th element can be univocally determined. Therefore we can compute a new distribution for the undetermined elements, say $\mathcal{N}(\mu_\gamma[N+2], P_\gamma[N+2])$, conditioning on the value of the j -th element. Then, the $(1 - \alpha)$ -confidence ellipsoid is computed from $\mathcal{N}(\mu_\gamma[N+2], P_\gamma[N+2])$ and the test is repeated on the undetermined elements in γ . The second screening stops when there exists no index j such that all vectors in $\Gamma[N+M]$ have the same j -th entry, with M denoting the number of iterations performed by the second screening. Also in this case, an upper-bound on the number of iterations is ℓ : at each iteration, at least one element in $\gamma \in \mathbb{Z}^\ell$ needs to be univocally determined, otherwise the algorithm halts. As outcome of the second screening we obtain a collection of vectors $\Gamma[N+M] = \{\gamma^1, \gamma^2, \dots, \gamma^V\}$ that have non-negligible probability of representing the correct choice for γ .

Remark 1: The first screening is performed element-wise on γ , whereas the second is done on integer-valued vectors. Before the first screening γ can assume all the values in \mathbb{Z}^ℓ ; after the first screening, due to the boundedness of the intervals $\bar{\Gamma}_i$ (for $\alpha > 0$), each element can assume only a finite number of values, hence the possible choices of γ becomes finite. This enables to perform the second screening: the second screening requires to check all admissible vectors γ and it could be performed in finite-time only when the possible choices for γ are finite. \square

C. Computation of the Set \mathcal{K}

After computing the finite set $\Gamma[N+M]$, we need to retrieve the admissible regularization vectors, i.e., the set \mathcal{K} . By definition, $\mathcal{K} = \{k : Ck = \gamma, \gamma \in \Gamma[N+M]\}$, therefore we need to compute the regularization vector k for each γ . Since the system $Ck = \gamma$ is underdetermined, for each choice of γ there exists an infinite number of vectors k satisfying $Ck = \gamma$. However, for each k we will be lately required to solve a linear estimation problem like $\min_\theta \|A^\top \theta - \check{\delta} + 2k\pi\|_{P_\delta}^2$, hence we need $|\mathcal{K}|$ finite and possibly small. The following result allows to consider a single k for each choice of γ , assuring that $|\mathcal{K}|$ is finite.

Proposition 1: Given a vector $\bar{\gamma} \in \mathbb{Z}^\ell$, every regularization vector \bar{k} satisfying $C\bar{k} = \bar{\gamma}$ will produce the same solution Θ^* , where $\Theta^* = \{ \text{mod } 2\pi\theta_0^*, \dots, \text{mod } 2\pi\theta_n^* \}$ and $\theta^* = [\theta_0^* \dots \theta_n^*] \in \mathbb{R}^{n+1}$ solves the linear estimation problem $\min_\theta \|A^\top \theta - \check{\delta} + 2\bar{k}\pi\|_{P_\delta}^2$.

Proof. See Appendix. \square

Then we only have to solve a single linear estimation problem for each $\bar{\gamma}$. Moreover, the following results provide a straightforward way to compute a k , satisfying $Ck = \bar{\gamma}$.

Lemma 1: Let \mathcal{G} be a pose graph and order the edges in the graph such that the first n edges belong to a spanning tree T of the graph and the remaining ℓ edges are chords with respect to T ; moreover let us write the cycle basis matrix $C = [C_T \ C_L]$, where $C_L \in \mathbb{Z}^{\ell \times n}$ contains the columns in C corresponding to edges in T and $C_L \in \mathbb{Z}^{\ell \times \ell}$ contains the columns in C corresponding to chords with respect to T . Then, C_L is invertible.

Proof. See Lemma 3 in [13]. \square

Proposition 2: Given a vector $\bar{\gamma} \in \mathbb{Z}^\ell$, a graph \mathcal{G} , and a cycle basis matrix $C = [C_T \ C_L]$ defined as in Lemma 1, a solution to $C\bar{k} = \bar{\gamma}$ can be computed as $\bar{k} = [\mathbf{0}_n^\top (C_L^{-1}\bar{\gamma})^\top]^\top$.

Proof. First of all, notice that the existence of \bar{k} is guaranteed by Lemma 1. We can develop the product $C\bar{k}$ as follows: $C\bar{k} = [C_T \ C_L][\mathbf{0}_n^\top (C_L^{-1}\bar{\gamma})^\top]^\top = C_L(C_L^{-1}\bar{\gamma}) = \bar{\gamma}$, which proves our claim. \square

V. OPTIMAL CHOICE OF THE CYCLE BASIS MATRIX

So far we have not discussed how to choose the cycle basis matrix C and if there is a particular selection that turns out to be more convenient in the screening of admissible γ . Since the possibility of reducing the cardinality of the set Γ heavily relies on the covariance matrix P_γ , in this section we discuss a choice of C that allows to minimize some measure on P_γ . In this context we consider the trace of P_γ as measure to be minimized by the choice of C . The trace is the sum of the diagonal elements of P_γ , and the diagonal elements directly influence the length of the confidence intervals in the first screening. Moreover, the trace is proportional to the sum of the squares of the lengths of the semi-axes of the $(1 - \alpha)$ -ellipsoid $\bar{\Gamma}$ that is used in the second screening. Therefore we want to determine:

$$C^* = \arg \min_C \text{tr} P_\gamma(C) \quad (13)$$

Before establishing the key result in Proposition 3 we need to introduce the following concepts (with slight abuse of notation we will identify with C both the cycle basis and the cycle basis matrix).

Definition 1 (Weights of cycles and cycle basis): Let us consider a cycle basis C , comprising cycles $\check{c}_1, \check{c}_2, \dots, \check{c}_\ell$;

then the *weight of a cycle* is $\Omega(\check{c}_t) = \sum_{(i,j) \in \check{c}_t} \omega_{ij}$, where $(i, j) \in \check{c}_t$ means that the sum comprises edges (i, j) belonging to cycle \check{c}_t and ω_{ij} is a weight associated with edge (i, j) . Moreover the *weight of a cycle basis* is $\Omega(C) = \sum_{t=1}^{\ell} \Omega(\check{c}_t)$. \square

Definition 2 (Minimum cycle basis): Given a graph \mathcal{G} and a weight function $\psi : \mathcal{E} \ni (i, j) \rightarrow \omega_{ij} \in \mathbb{R}$ that associates a weight to each edge in \mathcal{G} , a cycle basis with minimum weight is said to be a *minimum cycle basis* with respect to the pair (\mathcal{G}, ψ) . \square

We denote the minimum cycle basis with $\text{MCB}(\mathcal{G}, \psi)$. The following result establishes the optimality of the minimum cycle basis for problem (13).

Proposition 3 (Optimal choice of the cycle basis matrix): Let us consider the pose graph \mathcal{G} and the following weight function $\psi : (i, j) \rightarrow \sigma_{ij}^2$; then the minimum cycle basis with respect to the pair (\mathcal{G}, ψ) minimizes $\text{tr}P_\gamma$ in (13).

Proof. We here want to demonstrate that $\text{MCB}(\mathcal{G}, \psi) = \arg \min_C \text{tr}P_\gamma(C)$. By definition $P_\gamma = \frac{1}{4\pi^2} CP_\delta C^\top$. First of all we notice that the factor $\frac{1}{4\pi^2}$ does not influence the solution of the problem, hence the claim can be rewritten as $\text{MCB}(\mathcal{G}, \psi) = \arg \min_C \text{tr}CP_\delta C^\top$. Let us define c_t as the t -th row in C and with \check{c}_t the corresponding cycle in the graph. By simple inspection we notice that the t -th term on the main diagonal of $CP_\delta C^\top$ is $c_t P_\delta c_t^\top = \sum_{(i,j) \in \check{c}_t} \sigma_{ij}^2$. The previous expression coincides with the *weight of the t -th cycle* under the weight function ψ , see Definition 1. Therefore, the trace of $CP_\delta C^\top$ is the sum of the weights of the cycles in the cycle basis, which by definition is $\Omega(C)$. We hence established that $\min_C \text{tr}P_\gamma(C) = \min_C \Omega(C)$; by definition $\text{MCB}(\mathcal{G}, \psi)$ attains the minimum of $\Omega(C)$, therefore it holds $\text{MCB}(\mathcal{G}, \psi) \doteq \arg \min_C \Omega(C) = \arg \min_C \text{tr}P_\gamma(C)$. \square

VI. EXPERIMENTS AND DISCUSSION

In this section we present an experimental validation of the derivation presented so far. The two screenings are applied in cascade: (i) the first screening takes as input the relative orientation measurements δ and the corresponding covariance matrix P_δ and provides as output the set $\Gamma[N] \subset \mathbb{Z}^\ell$; (ii) the second screening takes as input the set $\Gamma[N]$ and returns as output the set $\Gamma[N+M] \subseteq \Gamma[N]$. Both screenings rely on the cycle basis matrix C , from which it is possible to compute μ_γ and P_γ . Then, for each $\gamma \in \Gamma[N+M]$ we have to retrieve the corresponding k . The collection of k defines the set \mathcal{K} which is the output of our algorithm.

The results in this section are aimed at the understanding of the following points:

- 1) how important is the choice of the cycle basis matrix;
- 2) what is the cardinality of \mathcal{K} in real applications;
- 3) what is the computational effort for computing \mathcal{K} ;

Let us start by the computation of the cycle basis matrix. There is a vast literature on minimum cycle basis and in this context we used the algorithm described in [15] and the C++ implementation available at [17] for computing the minimum cycle basis matrix, $\text{MCB}(\mathcal{G}, \psi)$. For sake of clarity in the following we omit the dependence on the graph and on the weight function and we simply write MCB . With the purpose of speeding-up the computation we also consider an algorithm for computing a $(2\eta - 1)$ -approximation of the cycle basis [10], (in our tests $\eta = 2$). We shall call the approximate minimum cycle basis matrix MCB_a . Besides the minimum cycle basis matrix, for our numerical evaluation, we consider the *fundamental cycle matrix* with respect to

	FCM(T_o)	FCM(T_m)	MCB	MCB _a
INTEL	< 0.01	< 0.01	0.2	0.09
MIT	< 0.01	< 0.01	0.01	0.01
M3500	< 0.01	0.3	1.54	1.11

TABLE I
CPU TIME FOR COMPUTING THE CYCLE BASIS MATRIX FOR EACH BENCHMARKING SCENARIO. TIME IS EXPRESSED IN SECONDS. CODE IS IMPLEMENTED IN C++.

a spanning tree T : in the fundamental cycle basis, each cycle comprises a chord in the graph with respect to T , say (i, j) , and the unique path in T from node i to node j . In particular, we consider the *fundamental cycle matrix* built on the *odometric spanning tree* $\text{FCM}(T_o)$ and the *fundamental cycle matrix* built on the *minimum uncertainty spanning tree* $\text{FCM}(T_m)$. The odometric spanning tree T_o is the spanning tree comprising the edges corresponding to the odometric constraints [2]; since T_o is given in common SLAM problems, the computation of $\text{FCM}(T_o)$ is inexpensive. The minimum uncertainty spanning tree, in this context, is the minimum weight spanning tree over graph \mathcal{G} [3], where the edges weights are defined by the weight function $\psi : (i, j) \rightarrow \sigma_{ij}^2$. We test the four aforementioned algorithms on well-known SLAM benchmarking scenarios [12], namely the *Intel Research Lab* (INTEL), the *MIT Killian Court* (MIT), and the *Manhattan world* (M3500). In each scenario we consider the pose graph and we compute the cycle basis matrix $\text{FCM}(T_o)$, $\text{FCM}(T_m)$, MCB and MCB_a . The CPU times required for computing the matrices are reported in Table I. Now we can evaluate the effectiveness of the first screening. In Table II we report for each scenario and for each choice of the cycle basis matrix:

- 1) the number of iterations N performed in the first screening, reporting details of each iteration when significant;
- 2) the percentage of elements that are univocally determined at j -th iteration, i.e., $\bar{u}(j) \doteq |\mathcal{U}[j]|/\ell \cdot 100$;
- 3) the *density* d of the matrix to be inverted to compute the conditional Gaussian probability $\text{prob}(\gamma[j] \mid \{\gamma_i = u_i, i \in \mathcal{U}[j-1]\})$. The *density* is defined as the number of non-zero elements in the matrix over the total number of elements;
- 4) the cardinality of the set $\Gamma[N]$;
- 5) the CPU time required for completing the first screening in our Matlab implementation; this time includes the CPU time required for computing μ_γ and P_γ .

We can notice that in common SLAM scenarios the first screening is sufficient in identifying a single possible value for γ , i.e., $|\Gamma[N]| = 1$; therefore, in these cases, no second screening is needed. Regardless the choice of the cycle basis matrix, the first iteration of the first screening allows to uniquely identify most of the elements in γ . However, the minimum cycle basis (approximate or exact) can easily select few admissible vectors γ after the first iteration, whereas the number of admissible γ after the first iteration may be still prohibitive for the fundamental cycle bases. The fundamental cycle bases usually requires more iterations for completing the first phase. Moreover, they usually require

		FIRST SCREENING ON γ					
		N	\bar{u}	d	$ \Gamma[N] $	CPU	
INTEL	FCB(T_o)	1	100	–	1	0.18	
	FCB(T_m)	1	100	–	1	0.07	
	MCB	1	100	–	1	0.07	
	MCB _{wa}	1	100	–	1	0.07	
MIT	FCB(T_o)	2	1	80	0.29	16	0.07
		2	20	–	1		
	FCB(T_m)	1	100	–	1	0.06	
	MCB	1	100	–	1	0.06	
MCB _a	1	100	–	1	0.06		
M3500	FCB(T_o)	4	1	52.92	0.02	$> 10^{200}$	3.12
			2	22.42	0.16	$> 10^{200}$	
			3	19.19	1	$> 10^{60}$	
			4	5.47	–	1	
	FCB(T_m)	2	1	98.62	0.02	$> 10^9$	0.67
			2	1.38	–	1	
	MCB	2	1	99.95	0.004	3	0.43
			2	0.05	–	1	
MCB _a	2	1	99.95	0.005	3	0.43	
		2	0.05	–	1		

TABLE II

N : NUMBER OF ITERATIONS; $\bar{u}(j)$: PERCENTAGE OF ELEMENTS THAT ARE UNIVOCALLY DETERMINED AT j -TH ITERATION; d : DENSITY OF THE MATRIX TO BE INVERTED AT EACH ITERATION; $|\Gamma[N]|$: CARDINALITY OF THE SET $\Gamma[N]$; CPU: TIME REQUIRED FOR COMPLETING THE FIRST SCREENING (MATLAB).

to manage matrices with higher density, when computing $\text{prob}(\gamma[j] \mid \{\gamma_i = u_i, i \in \mathcal{U}[j-1]\})$. Both reasons justify the higher CPU time required by the FCB. For testing the proposed approach in more challenging scenarios we considered the scenario M3500 and we added further random noise on the orientation measurements. In the scenario M3500b we added zero mean Gaussian noise with standard deviation 0.25 (angles are expressed in radians), whereas in the scenario lately referred to as M3500c we added zero mean Gaussian noise with standard deviation 0.3 to each orientation measurement. The corresponding statistics regarding the first screening are reported in Table III. In this case the choice of the cycle basis appears to be even more important: besides the computational effort, that is remarkably lower when using MCB (due to the smaller number of iterations and to the sparsity of the involved matrices), it is possible to notice that the fundamental cycle basis FCB(T_o) is not able to select a sufficiently small number of hypotheses for γ (the cardinality of $\Gamma[N]$ remains larger than 10^{200}), making the application of the second screening prohibitive. The matrix FCB(T_m)

is able to restrict the choice of γ to few vectors (8 in both scenarios). In the scenario M3500b, both MCB are able to determine a single admissible vector γ , making superfluous the application of the second screening. In M3500c, instead, two hypotheses survived the first screening. We can now analyze the second screening, considering scenarios M3500b and M3500c. The corresponding statistics are summarized in Table IV. For the FCB(T_o) it was not possible to perform the second screening due to the overwhelming number of hypotheses to be tested. After one iteration, using FCB(T_m) the number of hypotheses is reduced from 8 to 1 in the scenario M3500b, and from 8 to 2 in M3500c. For the MCB (exact and approximate) the screening did not reduce the number of hypotheses, i.e., $\Gamma[N+M] = \Gamma[N]$. In all cases the computational effort is an order of magnitude lower than the effort devoted to the first screening.

After computing the set $\Gamma[N+M]$, we need to retrieve the set \mathcal{K} , according to Section IV-C. In all the tests presented so far the time for retrieving \mathcal{K} was found to be negligible with respect to the cost of the other phases (< 0.01 seconds). The only scenario in which the cardinality of set \mathcal{K} is greater than one is M3500c (according to Section IV-C $|\mathcal{K}| = |\Gamma[N+M]|$); in this case, the set contains two hypotheses, i.e., $\mathcal{K} = \{k_1, k_2\}$. In our tests, FCB(T_m), MCB and MCB_a produced the same two hypotheses.

As answer to the three main questions motivating this experimental section we can then state that (i) the choice of the cycle basis is crucial in assuring to retrieve a small set \mathcal{K} with limited computational effort; (ii) in common SLAM scenarios the cardinality of \mathcal{K} is one, and, when using MCB (exactly or approximate), $|\mathcal{K}|$ remains small also after inflating a large amount of noise in the orientation measurements; (iii) regarding the computational effort, there is a clear tradeoff between the effort of computing a cycle basis matrix (larger for MCB) and the effort of performing the two screenings (larger for FCB). From the results presented so far it turns out that a good compromise can be obtained using the $(2\eta-1)$ -approximate minimum cycle basis which assures similar performance with respect to its exact counterpart, thus being computationally cheaper to compute.

According to the derivation in [2], after selecting the correct regularization vector k it is possible to compute the pose graph configuration from relative pose measurements. In Figure 1 we report the estimated pose graphs for each scenario and for each $k \in \mathcal{K}$. It is possible to notice that in all cases the \mathcal{K} contains the correct regularization vector. As a side comment we notice that we are already considering scenarios with severe orientation noise in which state-of-the-art techniques are likely to be stuck in local minima; for instance, in Figure 2 we report the estimated pose-graphs with TORO [6] for the scenarios M3500b and M3500c. In the proposed approach, when the cardinality of \mathcal{K} is one, the approach assures the consistency of the orientation errors (roughly speaking we can guarantee to be close to the actual nodes' orientation); moreover, when increasing the noise affecting orientation measurements, the approach only enlarge the number of possible choices for the regularization vectors, thus assuring with high probability that one of the choices is correct.

VII. CONCLUSION

In this work we show that is possible to recast the problem of estimating orientations in $SO(2)$, from relative measurements, into a linear framework. We formalize the

		FIRST SCREENING ON γ					
		N	\bar{u}	d	$ \Gamma[N] $	CPU	
M3500b	FCB(T_o)	9	63.10	*	$> 10^{200}$	3.22	
	FCB(T_m)	$\left. \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array} \right\}$	1	74.00	0.008	$> 10^{200}$	1.53
			2	21.65	0.39	$> 10^{25}$	
			3	4.2	0.73	27	
			4	0	-	8	
	MCB	$\left. \begin{array}{l} 1 \\ 2 \\ 3 \end{array} \right\}$	1	98.21	0.004	$> 10^{10}$	0.43
			2	1.74	0.38	3	
			3	0.05	-	1	
	MCB $_a$	$\left. \begin{array}{l} 1 \\ 2 \\ 3 \end{array} \right\}$	1	98.05	0.05	$> 10^{12}$	0.44
			2	1.89	0.38	3	
			3	0.05	-	1	
	M3500c	FCB(T_o)	16	50.31	*	$> 10^{200}$	4.24
FCB(T_m)		$\left. \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \right\}$	1	65.51	0.006	$> 10^{200}$	0.85
			2	22.21	0.16	$> 10^{80}$	
			3	9.93	0.53	$> 10^{14}$	
			4	2.20	0.44	27	
			5	0	-	8	
MCB		$\left. \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array} \right\}$	1	95.45	0.004	$> 10^{28}$	0.43
			2	4.35	0.50	32	
			3	0.15	0.55	2	
			4	0	-	2	
MCB $_a$		$\left. \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array} \right\}$	1	95.04	0.04	$> 10^{30}$	0.45
			2	4.76	0.48	32	
	3		0.15	0.55	2		
	4		0	-	2		

TABLE III

N : NUMBER OF ITERATIONS; $\bar{u}(j)$: PERCENTAGE OF ELEMENTS THAT ARE UNIVOCALLY DETERMINED AT j -TH ITERATION; d : DENSITY OF THE MATRIX TO BE INVERTED AT EACH ITERATION; $|\Gamma[N]|$: CARDINALITY OF THE SET $\Gamma[N]$; CPU: TIME REQUIRED FOR COMPLETING THE FIRST SCREENING (MATLAB). * WE OMIT THE DETAILS FOR SPACE REASONS.

		SECOND SCREENING ON γ					
		$ \Gamma[N] $	M	\bar{u}	d	$ \Gamma[N+M] $	CPU
M3500b	FCB $_{ost}$	$> 10^{200}$	*	*	*	*	*
	FCB $_{mst}$	8	1	0.41	-	1	0.03
	MCB	1	-	-	-	1	-
	MCB $_a$	1	-	-	-	1	-
M3500c	FCB $_{ost}$	$> 10^{200}$	*	*	*	*	*
	FCB $_{mst}$	8	1	0.31	-	2	0.03
	MCB	2	1	0	-	2	0.02
	MCB $_a$	2	1	0	-	2	0.02

TABLE IV

M : NUMBER OF ITERATIONS; $\bar{u}(j)$: PERCENTAGE OF ELEMENTS THAT ARE UNIVOCALLY DETERMINED AT j -TH ITERATION; d : DENSITY OF THE MATRIX TO BE INVERTED AT EACH ITERATION; $|\Gamma[N]|$, $|\Gamma[N+M]|$: CARDINALITY OF THE SETS $\Gamma[N]$ AND $\Gamma[N+M]$, RESPECTIVELY; CPU: TIME REQUIRED FOR COMPLETING THE SECOND SCREENING (MATLAB). * NO SCREENING POSSIBLE, DUE TO THE CARDINALITY OF $\Gamma[N]$.

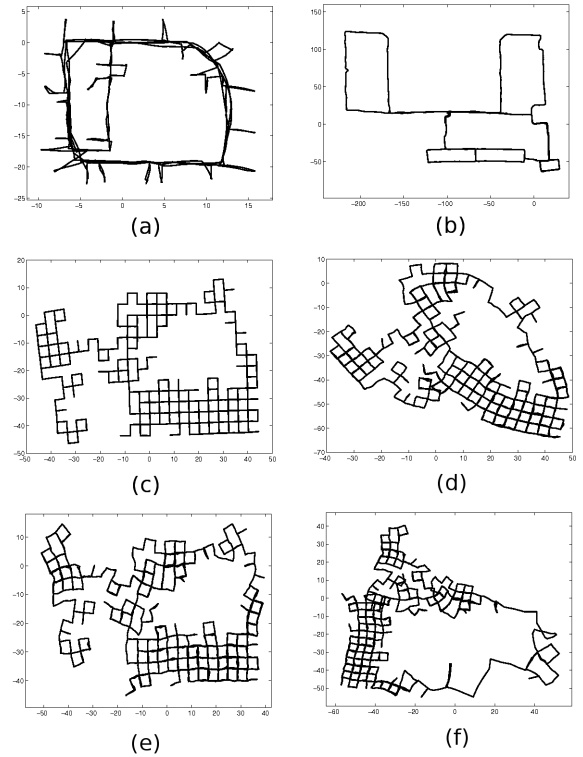


Fig. 1. Estimated pose graph configuration with the approach proposed in [2] for each admissible value k ; when relevant we also report the optimal objective value ($\chi^2(k)$) for the configuration [5]. (a) Intel research lab; (b) MIT Killian Court; (c) Manhattan World; (d) Manhattan World with noise inflation (M3500b); (e) Manhattan World with noise inflation (M3500c) and $k = k_1$, $\chi^2(k_1) = 3.17 \cdot 10^3$; (f) Manhattan World with noise inflation (M3500c) and $k = k_2$, $\chi^2(k_2) = 1.32 \cdot 10^5$ (local minimum).

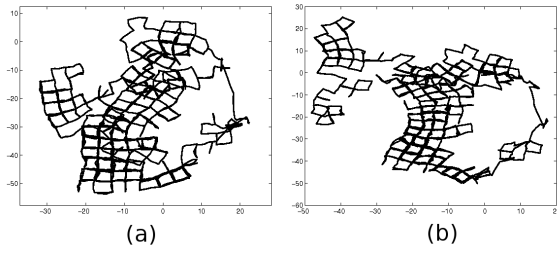


Fig. 2. Estimated pose graph configuration with TORO. (a) Manhattan World with noise inflation (M3500b): $\chi^2 = 1.57 \cdot 10^4$ (local minimum); (b) Manhattan World with noise inflation (M3500c): $\chi^2 = 2.61 \cdot 10^4$ (local minimum).

intuitions in [2] showing that, when mapping the maximum likelihood problem from the manifold $SO(2)$ to a vector space, it is necessary to include integer-valued unknowns (*regularization terms*). In general, the introduction of such regularization terms makes the solution of the problem challenging, since infinite possible regularization terms have to be considered. However, we propose a *screening* that allows to reduce the possible choices of the regularization terms by discarding choices with negligible probability of being correct. Experimental results show that, in common problems, a single choice has non-negligible probability of being correct. Once a correct regularization term is retrieved, the linearity of the framework assures the consistency of the estimation errors with the corresponding covariance matrix, hence providing probabilistic assessments on the quality of the orientation estimate. Moreover, also after inflating a large amount of noise in the orientation measurements, a suitable choice of the matrices involved in the screening can be still effective in determining few admissible choices for the regularization terms. As a by-product, we show that, exploiting the results of this paper, the approach proposed in [2] can estimate a correct pose graph configuration also when state-of-the-art approaches are stuck in a local minimum.

APPENDIX

This appendix contains the proof of Proposition 1. Let us consider two regularization vectors k_1 and k_2 such that $Ck_1 = Ck_2 = \bar{\gamma}$. Let us call $\theta^*[1]$ and $\theta^*[2]$ the solutions to the linear estimation problems $\min_{\theta} \|A^\top \theta - \check{\delta} + 2k_1\pi\|_{P_\delta}^2$ and $\min_{\theta} \|A^\top \theta - \check{\delta} + 2k_2\pi\|_{P_\delta}^2$, respectively. Moreover, we recall that, by convention, the orientation of the first node is set to 0. Let us define $\delta^*[1] = A^\top \theta^*[1]$ and $\delta^*[2] = A^\top \theta^*[2]$. First of all, observe that, after setting the orientation of the first node to zero, $\delta^*[1]$ and $\delta^*[2]$ uniquely identify $\theta^*[1]$ and $\theta^*[2]$, since $\theta^*[i]$ can be rewritten as an integer-valued linear combination of δ_i^* , $i = 1, 2$; for instance, the orientation of node j can be rewritten as $\theta_j^*[i] = \sum_{r \in p(0,j)} \lambda_r \delta_r^*[i]$, where $p(0,j)$ is the set of edges along a path connecting node 0 and node j and λ_r is $+1$ if the edges r is traversed forwards along the path, -1 otherwise. In general, we have $\theta^*[i] = D\delta^*[i]$, $i = 1, 2$, where D is an integer-valued matrix.

Now we recall from [19] that the linear estimation problem $\min_{\theta} \|A^\top \theta - \check{\delta} + 2k_i\pi\|_{P_\delta}^2$, $i = 1, 2$, can be solved directly in function of $\delta^*[i]$, $i = 1, 2$ as follows:

$$\delta^*[i] = \begin{cases} \check{\delta} + 2k_i\pi & \text{if } m = n \\ \check{\delta} + 2k_i\pi - P_\delta C^\top (CP_\delta C^\top)^{-1} C(\check{\delta} + 2k_i\pi) & \text{if } m > n \end{cases},$$

where C is the cycle basis matrix. Let us write the quantity $\delta^*[1] - \delta^*[2]$:

$$\begin{cases} 2\pi(k_1 - k_2) & \text{if } m = n \\ 2\pi(k_1 - k_2) - P_\delta C^\top (CP_\delta C^\top)^{-1} C(2\pi(k_1 - k_2)) & \text{if } m > n \end{cases},$$

Recalling that $Ck_1 = Ck_2 = \bar{\gamma}$, both cases reduce to $\delta^*[1] - \delta^*[2] = 2\pi(k_1 - k_2)$. Therefore, elements of $\delta^*[1]$ and $\delta^*[2]$ only differ by multiples of 2π ; then, $\theta^*[1] - \theta^*[2] = D(\delta^*[1] - \delta^*[2]) = 2\pi D(k_1 - k_2)$, and since D is an integer-valued matrix, also the elements of $\theta^*[1]$ and $\theta^*[2]$ only differ by multiples of 2π . Therefore, $\check{\theta}^*[1] = \{ \text{mod } 2\pi \theta_0^*[1] \dots \text{mod } 2\pi \theta_n^*[1] \} = \{ \text{mod } 2\pi \theta_0^*[2] \dots \text{mod } 2\pi \theta_n^*[2] \} = \check{\theta}^*[2]$, which concludes the proof. \square

REFERENCES

- [1] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley and Sons, 2001.
- [2] Luca Carlone, Rosario Aragues, Jose Castellanos, and Basilio Bona. A linear approximation for graph-based simultaneous localization and mapping. In *Proc. of Robotics: Science and Systems*.
- [3] W. Chen. *Graph Theory and Its Engineering Applications*. Advanced Series in Electrical and Computer Engineering, 1997.
- [4] U. Frese, P. Larsson, and T. Duckett. A multilevel relaxation algorithm for simultaneous localization and mapping. *IEEE Trans. on Robotics*, 21(2):196–207, 2005.
- [5] G. Grisetti, R. Kuemmerle, C. Stachniss, U. Frese, and C. Hertzberg. A hierarchical optimization on manifolds for online 2D and 3D mapping. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2010.
- [6] G. Grisetti, C. Stachniss, and W. Burgard. Non-linear constraint network optimization for efficient map learning. *IEEE Trans. on Intelligent Transportation Systems*, 10(3):428–439, 2009.
- [7] S. Huang, Y. Lai, U. Frese, and G. Dissanayake. How far is SLAM from a linear least squares problem? In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010.
- [8] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2011.
- [9] T. Kavitha, C. Liebchen, K. Mehlhorn, D. Michail, R. Rizzi, T. Ueckerdt, and K. Zweig. Cycle bases in graphs: Characterization, algorithms, complexity, and applications. *Computer Science Rev.*, 3(4):199–243, 2009.
- [10] T. Kavitha, K. Mehlhorn, and D. Michail. New approximation algorithms for minimum cycle bases of graphs. *Lecture Notes in Computer Science*, 4393:512–523, 2007.
- [11] K. Konolige. Large-scale map-making. In *Proc. of the AAAI National Conf. on Artificial Intelligence*, 2004.
- [12] R. Kümmerle, B. Steder, C. Dornhege, M. Ruhnke, G. Grisetti, C. Stachniss, and A. Kleiner. Slam benchmarking webpage. <http://ais.informatik.uni-freiburg.de/slamevaluation>, 2009.
- [13] C. Liebchen. Finding short integral cycle bases for cyclic timetabling. *Lecture Notes in Computer Science*, 2832:715–726, 2003.
- [14] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.
- [15] K. Mehlhorn and D. Michail. Minimum cycle bases: Faster and simpler. *ACM Transactions on Algorithms*, 6(1):1–13, 2009.
- [16] J.M. Mendel. *Lessons in estimation theory for signal processing, communications, and control*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [17] D. Michail. Minimum cycle basis library, <http://www.mpi-inf.mpg.de/~michail/subpages/mcb/doxygen/index.html>.
- [18] E. Olson, J.J. Leonard, and S. Teller. Fast iterative optimization of pose graphs with poor initial estimates. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 2262–2269, 2006.
- [19] W.J. Russell, D.J. Klein, and J.P. Hespanha. Optimal estimation on the graph cycle space. In *American Control Conference*, 2010.
- [20] S. Thrun and M. Montemerlo. The GraphSLAM algorithm with applications to large-scale mapping of urban structures. *Int. J. Robot. Res.*, 25:403–429, 2006.