From Angular Manifolds to the Integer Lattice: Guaranteed Orientation Estimation with Application to Pose Graph Optimization

Luca Carlone, *Member* Andrea Censi, *Member*

***Abstract*—Pose graph optimization from relative measurements is challenging because of the angular component of the poses: the variables live on a manifold product with nontrivial topology, and the likelihood function is non-convex and has many local minima. Due to these issues, iterative solvers are not robust to large amounts of noise. This paper describes a global estimation method, called MOLE2D, for the estimation of the nodes' orientation in a pose graph. We demonstrate that the original nonlinear optimization problem on the manifold product is equivalent to an unconstrained quadratic optimization problem on the integer lattice. Exploiting this insight, we show that, in general, the maximum likelihood estimate alone cannot be considered a reliable estimator. Therefore, MOLE2D returns a *set* of point estimates, for which we can derive precise probabilistic guarantees. Experiments show that the method is able to tolerate extreme amounts of noise, far above all noise levels of sensors used in applications. Using MOLE2D's output to bootstrap the initial guess of iterative pose graph optimization methods improves their robustness and makes them avoid local minima even for high levels of noise.**

***Index Terms*—pose graph optimization, SLAM, mobile robots, orientation estimation, SO(2) manifold, multi-hypothesis estimation, integer quadratic programming**

I. Introduction

A *pose graph* is a model commonly used to formalize the Simultaneous Localization and Mapping (SLAM) problem [1], [2]. Each node in the graph represents the pose of a mobile robot at a given time. Two nodes share an edge if a relative measurement between the two poses is available. Relative measurements might be obtained by means of proprioceptive sensors (e.g., wheel odometry) or exteroceptive-sensor-based techniques (e.g., scan matching or visual odometry). The objective of *pose graph optimization* is to estimate the poses that maximize the likelihood of the relative measurements. Once the poses have been estimated, it is possible to construct a map of the environment by placing all measurements in the same global coordinate frame. Finding the global minimum of the likelihood is difficult mainly because of the angular component: nodes' orientations belong to a product of manifolds that have a nontrivial topology ($\mathrm{SO}(2)^n$ or $\mathrm{SO}(3)^n$, with n the number of observable poses). In fact, if the orientations were known, pose optimization would be a linear problem [3]. With unknown orientations, the problem is nonlinear, non-convex, with multiple local minima even in simple instances [4].

The state-of-the-art techniques for pose optimization, such as g2o [5] and Toro [6], are iterative optimization methods that minimize a cost function starting from an initial guess. None can guarantee convergence to a global minimum, and it is observed that they get trapped in local minima in presence of large measurement noise [6]. This paper focuses on the problem of estimating the nodes' orientations from pairwise relative angular measurements in the planar case (orientations in $\mathrm{SO}(2)$), which is of interest in several application scenarios, ranging from domestic environments to factory floors. We describe MOLE2D, a multi-hypothesis global optimization method that does not suffer from local minima, even with extreme amounts of noise, and has precise probabilistic guarantees. Experiments show that, once the nodes' orientations have been estimated with our method, they can be used as a first guess for traditional solvers to obtain a robust and accurate solution for pose graph optimization.

Related work in robotics: The formulation of SLAM as an optimization problem on a graph traces back to Lu and Milios [1]. Gutmann and Konolige [7] describe how to build a pose graph in incremental fashion from laser scan measurements. A large amount of subsequent work focuses on speeding up computation. Duckett *et al.* [8] use a Gauss-Seidel relaxation to minimize residual errors. Konolige [9] describes a reduction scheme to improve efficiency of nonlinear optimization. Thrun and Montemerlo [2] describe a conjugate gradient-based optimization that enables large-scale estimation. Frese *et al.* [10] propose a multilevel relaxation that considerably reduces the computation time by applying a multi-grid algorithm. Olson *et al.* [11] propose an alternative parametrization for the problem, which entails several advantages in terms of computation and robustness. Grisetti *et al.* [6] extend such framework, proposing a method (Toro) that is based on stochastic gradient descent and uses a tree-based parametrization. Kaess *et al.* [12]–[14] present an elegant formalization of SLAM using a *Bayes tree* model and investigate incremental estimation techniques. Several recent papers focus on the manifold structure of the problem: the domain of the problem is a product of manifolds $\mathrm{SE}(2)$ or $\mathrm{SE}(3)$, and this aspect requires a suitable treatment when using iterative optimization techniques [15], or closed-form problem-specific methods [16]–[18]. Kuemmerle *et al.* [5] describe the g2o framework for solving general optimization problems with variables belonging to manifolds. Olson and Agarwal [19], and Sünderhauf and Protzel [20], [21] extend this framework, with the purpose of increasing robustness to outliers. Rosen *et al.* [22] propose the use of a trust-region method to enhance convergence of iterative optimization. The idea of improving convergence by bootstrapping iterative solvers with suitable estimates has appeared in different forms in literature [23]–[25]. The theoretical analysis of the problem is slightly behind applications; see Knuth and Barooah [26], Huang *et al.* [27], and Wang *et al.* [4].

Related work in other fields: Other fields outside of robotics

L. Carlone (*corresponding author*) is with the College of Computing, Georgia Institute of Technology, Atlanta, GA, USA. luca.carlone@gatech.edu

A. Censi is with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA. censi@mit.edu

deal with problems formally equivalent to pose graph optimization, such as attitude synchronization [28] and calibration of camera networks [29]. Often these problems are formulated in a distributed setting, where a set of agents must estimate a local state (pose, position, orientation, etc.) based on inter-agent measurements (relative distance, relative bearing, etc.). For example, Barooah and Hespanha [3], [30] consider the problem of estimating positions of robots in a team from relative position measurements, assuming known orientations. Knuth and Barooah [31] focus on distributed computation. The case in which the nodes positions have to be estimated from bearing measurements was pioneered by Stanfield [32] and further developed in more recent work [33]–[35]. Another common setup is the one in which nodes positions are estimated from pairwise distance measurements [36]–[39].

Relation with previous work: Previous work by the first author and colleagues [25], [40] proposed a fast approximated method for computing orientations in a pose graph, and then solving for the translations given the orientations. Those works motivated this paper, but it follows a different route. This paper presents a formal treatment of the orientation-only estimation problem; rather than proposing an approximation, we care about finding a *guaranteed* orientation estimate. In hindsight, the results of this paper allow us to conclude that the rounding operation proposed to solve the wraparound problem in [25] is only a heuristic to solve a quadratic integer program, which does not necessarily lead to the optimal solution [41]. More in detail, the proof of equivalence to a quadratic integer program, the MOLE2D algorithm, and the experimental results are original and have not been published in previous work.

Paper outline: Our results derive from the joint application of graph theory, differential geometry, and integer programming. Section II recalls the necessary preliminaries.

Section III introduces the usual maximum likelihood formalization for the orientation estimation problem, with extra care to assumptions and problem symmetries. Section IV proves that the maximum likelihood orientation estimation problem, with domain $\mathrm{SO}(2)^n$ (where n is the number of observable nodes), is equivalent to an *unconstrained*, *quadratic*, *integer* optimization problem on $\mathbb{Z}^\ell$, where ℓ is the number of cycles.

Section V shows that the maximum likelihood orientation estimator may suffer from a bias, which intuitively corresponds to choosing the wrong number of windings for one or more loops in the graph. For this reason, it is not necessarily the best choice for the problem at hand.

Section VI and VII describe the MOLE2D algorithm (*Multi-hypothesis Orientation-from-Lattice Estimation in 2D*), which returns a *set* of hypotheses for the nodes' orientations. It has precise probabilistic guarantees: at least one hypothesis is "close" to the actual nodes' orientations within a given confidence level. The algorithm returns a small set of plausible hypotheses, if the "frame of reference", described by the *cycle basis matrix*, is chosen appropriately. Choosing the *minimum cycle basis* minimizes the expected number of hypotheses.

Section VIII discusses the performance of MOLE2D on standard SLAM datasets. For the case of orientation estimation, we explore the trade-off in performance implied by the choice of the cycle basis matrix used by MOLE2D. The results confirm the theoretical predictions. In common problem instances the set of estimates contains a single element, because the problem is very well constrained. For the case of pose optimization, we show that simply substituting the orientation estimate computed by MOLE2D in place of the odometric initial guess greatly enhances the robustness to noise in g2o.

Datasets, source code, and the technical report [42], with extended proofs are available at [43].

II. Preliminaries

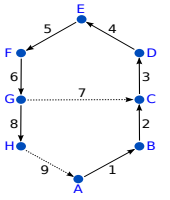
A. Computational graph theory

Chen [44] is a popular reference for computational graph theory. Our notation (Table I) is compatible with Kavitha *et al.* [45], from which we take most results about cycle bases. A *directed graph* $\mathcal{G}$ is a pair $(\mathcal{V}, \mathcal{E})$, where the *vertices* or *nodes* $\mathcal{V}$ are a finite set of elements, and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of *edges*. Each edge is an ordered pair $e = (i, j)$. We say that e is *incident* on nodes i and j, *leaves* node i, called *tail*, and is *directed towards* node j, called *head*. A *weighted graph* has also a nonnegative weight w_{ij} associated to any edge $e = (i, j)$. The number of nodes and edges are denoted with $n + 1$ and m. Our graph has $n + 1$ nodes, rather than n, because only n independent variables will be observable, and this choice will simplify the notation later.

The *incidence matrix* $\overline{\boldsymbol{A}}$ of a directed graph is a $(n+1) \times m$ matrix with elements in $\{-1, 0, +1\}$ that exhaustively describes the graph topology. Each column of $\overline{\boldsymbol{A}}$ corresponds to an edge and has exactly two non-zero elements. For the column corresponding to edge $e = (i, j)$, there is a -1 on the i-th row and a $+1$ on the j-th row (Figure 1).

The *reduced incidence matrix* $\boldsymbol{A}$ is obtained from $\overline{\boldsymbol{A}}$ by removing one row. Without loss of generality, in this paper we assume that it is the first row, which corresponds to the first node that is set to the origin of the reference frame. If $\overline{\boldsymbol{A}}$ has dimensions $(n + 1) \times m$, $\boldsymbol{A}$ has dimension $n \times m$.

A *spanning tree* is a subgraph with n edges that contains all the nodes in the graph. For a given spanning tree, the edges of the original graph that do not belong to the spanning tree are called *chords*. A *cycle* is a subgraph in which every node appears in an even number of edges. A *circuit* is a cycle in which every node appears exactly in two edges. Once the m



$$\overline{\boldsymbol{A}} = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & +1 \\ +1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & +1 & -1 & 0 & 0 & 0 & +1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & +1 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & +1 & -1 \end{pmatrix}$$

$$\boldsymbol{C}_1 = \begin{pmatrix} +1 & +1 & +1 & +1 & +1 & +1 & 0 & +1 & +1 \\ +1 & +1 & 0 & 0 & 0 & 0 & -1 & +1 & +1 \end{pmatrix}$$

$$\boldsymbol{C}_2 = \begin{pmatrix} +1 & +1 & 0 & 0 & 0 & 0 & -1 & +1 & +1 \\ 0 & 0 & +1 & +1 & +1 & +1 & +1 & 0 & 0 \end{pmatrix}$$

Fig. 1. To illustrate the definition of cycle basis, we use a toy graph with vertex set $\mathcal{V} = \{A, \ldots, H\}$ and edge set $\mathcal{E} = \{1, \ldots, 9\}$. We assume that each edge has unitary weight. The topology is entirely described by the incidence matrix of the graph, while the reduced incidence matrix $\boldsymbol{A}$ is obtained from $\overline{\boldsymbol{A}}$ by deleting the first row. A spanning tree is given by edges $\{1, 2, 3, 4, 5, 6, 8\}$; the corresponding chords are edge 7 and edge 9 (reported as dashed lines in the figure). $\boldsymbol{C}_1$ is a cycle basis matrix for the graph, whose first circuit includes edges $\{1, 2, 3, 4, 5, 6, 8, 9\}$ and second circuit includes edges $\{1, 2, 7, 8, 9\}$. $\boldsymbol{C}_2$ is a *minimum* cycle basis, whose first circuit includes edges $\{1, 2, 7, 8, 9\}$ and second circuit includes edges $\{3, 4, 5, 6, 7\}$.

TABLE I
SYMBOLS USED IN THIS PAPER

Graph

$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	Directed graph
m	Number of edges
$n+1$	Number of nodes
n	Number of observable variables
$\mathcal{V}$	Vertex set; $\lvert\mathcal{V}\rvert = n+1$
$\mathcal{E}$	Edge set; $\lvert\mathcal{E}\rvert = m$
$e = (i,j) \in \mathcal{E}$	Edge between nodes i and j
ℓ	Number of cycles; $\ell = m - n$
w_{ij}	Weight associated to edge (i,j)
$\overline{\boldsymbol{A}} \in \mathbb{R}^{(n+1)\times m}$	Incidence matrix of $\mathcal{G}$
$\boldsymbol{A} \in \mathbb{R}^{n\times m}$	Reduced incidence matrix of $\mathcal{G}$

Cycle bases

$\mathcal{C}_\mathcal{G}$	The set of all cycle basis for $\mathcal{G}$
$\boldsymbol{c} \in \{-1,0,+1\}^m$	Row vector describing a circuit
$\boldsymbol{W}(\boldsymbol{c}, \boldsymbol{w})$	Weight of a cycle $\boldsymbol{c}$
$\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{w})$	Weight of a cycle basis $\boldsymbol{C}$
$\boldsymbol{C} \in \mathbb{Z}^{\ell\times m}$	Matrix describing a cycle basis
$\boldsymbol{C}_\perp, \boldsymbol{C}_\mathsf{T}$	Ordering of $\boldsymbol{C}$ according to a spanning tree T
MCB	Minimum cycle basis for a given weight function
FCB	Fundamental cycle basis built from a spanning tree

Geometry of angles

$\mathrm{SO}(2)$	2D rotation matrices
$\mathrm{Exp} : \mathbb{R} \to \mathrm{SO}(2)$	Exponential map
$\mathrm{Log} : \mathrm{SO}(2) \to \mathcal{P}(\mathbb{R})$	Logarithmic map
$\mathrm{Log}_0 : \mathrm{SO}(2) \to \mathbb{R}$	Principal logarithmic map
$\langle\cdot\rangle_{2\pi} : \mathbb{R} \to (-\pi, +\pi]$	2π modulus operation

Orientation estimation (intrinsic formalization)

$\boldsymbol{r}_i^\circ \in \mathrm{SO}(2)$	Unknown orientation of node i
$\boldsymbol{r}_i \in \mathrm{SO}(2)$	Optimization variables for orientation of node i
$\boldsymbol{d}_{ij} \in \mathrm{SO}(2)$	Relative orientation measurement
$\boldsymbol{\varepsilon}_{ij} \in \mathrm{SO}(2)$	Measurement error
$\epsilon_{ij} \in \mathbb{R}$	Gaussian noise producing $\boldsymbol{\varepsilon}_{ij}$
σ_{ij}	Standard deviation of ϵ_{ij}

Orientation estimation (in $(-\pi, +\pi]$ coordinates)

$\boldsymbol{\theta}^\circ \in (-\pi, +\pi]^n$	Unknown orientations
$\boldsymbol{\theta} \in (-\pi, +\pi]^n$	Optimization variables for orientations
$\boldsymbol{\delta} \in (-\pi, +\pi]^m$	Relative orientation measurements
$\boldsymbol{P}_\delta \in \mathbb{R}^{m\times m}$	Measurement covariance

Mixed-integer formalization in $\boldsymbol{k}$ and $\boldsymbol{x}$

$\boldsymbol{x} \in \mathbb{R}^n$	Real-valued optimization variables for orientations
$\boldsymbol{k} \in \mathbb{Z}^m$	Regularization vector
$\boldsymbol{\theta}^{\star\lvert\boldsymbol{k}}$	Estimate of $\boldsymbol{\theta}^\circ$ given $\boldsymbol{k} \in \mathbb{Z}^m$

Reduced formalization in cycle space

$\boldsymbol{\gamma} \in \mathbb{Z}^\ell$	Integer vector living on the cycles
$\hat{\boldsymbol{\gamma}} \in \mathbb{Z}^m$	Estimator for $\boldsymbol{\gamma}$
$\boldsymbol{x}^{\star\lvert\boldsymbol{\gamma}}$	Real-valued estimate of $\boldsymbol{\theta}^\circ$ given $\boldsymbol{\gamma}$
$\boldsymbol{\theta}^{\star\lvert\boldsymbol{\gamma}} = \langle \boldsymbol{x}^{\star\lvert\boldsymbol{\gamma}} \rangle_{2\pi}$	Wrapped estimate of $\boldsymbol{\theta}^\circ$ given $\boldsymbol{\gamma}$
$\boldsymbol{\theta}^\star = \boldsymbol{\theta}^{\star\lvert\boldsymbol{\gamma}^\star}$	Max. likelihood estimate of $\boldsymbol{\theta}^\circ$
Γ	Estimates for $\boldsymbol{\gamma}$ returned by INTEGER-SCREENING
Θ	Estimates of $\boldsymbol{\theta}^\circ$ returned by MOLE2D

Miscellanea

$\mathcal{P}(S)$	Power set of the set S
$\lvert S\rvert$	Cardinality of the set S
$\mathbf{I}_n$	$n \times n$ identity matrix
$\mathbf{0}_n$	Column vector of zeros of dimension n
$\mathbf{1}_n$	Column vector of ones of dimension n
$\lfloor\cdot\rfloor$	Floor operator
$\mathsf{Trace}\{\boldsymbol{P}\}$	Trace of the matrix $\boldsymbol{P}$
$\chi^2_{\ell,\alpha}$	Quantile of the χ^2 distribution with ℓ degrees of freedom and upper tail probability equal to α

edges have been given an arbitrary order, a *directed circuit* can be described by a vector of m elements, in which the e-th element is $+1$ or -1 if edge e is traversed respectively forwards (from tail to head) or backwards, and 0 if it does not appear. Therefore, a circuit can be represented by a vector in $\{-1,0,+1\}^m$. Because an edge can be traversed twice or more by a cycle, a cycle is represented by a vector $\boldsymbol{c} \in \mathbb{Z}^m$.

If the graph is weighted, the weight of a cycle $\boldsymbol{c} \in \mathbb{Z}^m$ is the sum of the weights of the edges that it traverses:

$$\boldsymbol{W}(\boldsymbol{c}, \boldsymbol{w}) = \sum_{(i,j)\in\mathcal{E}} w_{ij}|c_{ij}|.$$

A *cycle basis* of a graph is a minimal set of circuits such that any cycle in the graph can be written as a combination of the circuits in the basis. We define $\mathcal{C}_\mathcal{G}$ as the set of all cycle basis of the graph. The number of independent circuits in the cycle basis is called *cyclomatic number* and it is equal to $\ell = m - n$. A *cycle basis matrix* is a matrix $\boldsymbol{C} \in \mathbb{Z}^{\ell\times m}$, such that each row $\boldsymbol{c}^{(t)}$ describes one of the circuits (Figure 1):

$$\boldsymbol{C} = \begin{pmatrix} \boldsymbol{c}^{(1)} \\ \vdots \\ \boldsymbol{c}^{(\ell)} \end{pmatrix} \in \mathbb{Z}^{\ell\times m}.$$

In order to simplify the notation, and without loss of generality, we order the edges such that the first n edges belong to a spanning tree T and the remaining ℓ edges are chords with respect to T. This allows to write any cycle basis matrix $\boldsymbol{C}$ as

$$\boldsymbol{C} = (\boldsymbol{C}_\mathsf{T}\ \boldsymbol{C}_\perp), \tag{1}$$

where $\boldsymbol{C}_\mathsf{T} \in \mathbb{Z}^{\ell\times n}$ contains the columns in $\boldsymbol{C}$ corresponding to edges in T, and $\boldsymbol{C}_\perp \in \mathbb{Z}^{\ell\times\ell}$ contains the columns in $\boldsymbol{C}$ corresponding to chords with respect to T.

The weight of a cycle basis is the sum of the cycles' weights:

$$\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{w}) = \sum_{t=1}^{\ell} \boldsymbol{W}(\boldsymbol{c}^{(t)}, \boldsymbol{w}).$$

The *minimum cycle basis* (MCB) is a basis (not necessarily unique) that has minimum weight:

$$\mathsf{MCB}(\mathcal{G}, \boldsymbol{w}) \doteq \operatorname*{arg\,min}_{\boldsymbol{C}\in\mathcal{C}_\mathcal{G}} \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{w}).$$

In the paper we only consider the weight function $\boldsymbol{w}$ that associates to each edge the variance of the corresponding measurement. Therefore, we use the notation "MCB" omitting the dependence on the graph and on the weight function, and implying that we consider a minimum *uncertainty* cycle basis. Similarly, when we talk about *minimum spanning tree*, we refer to a minimum *uncertainty* spanning tree.

B. Modulus algebra

The map $\langle\cdot\rangle_{2\pi}$ is a function from $\mathbb{R}$ to the interval $(-\pi, +\pi]$:

$$\langle\cdot\rangle_{2\pi} : \mathbb{R} \to (-\pi, +\pi],$$

which can be written explicitly as

$$\langle\omega\rangle_{2\pi} \doteq \omega + 2\pi \left\lfloor \frac{\pi - \omega}{2\pi} \right\rfloor, \tag{2}$$

where $\lfloor \cdot \rfloor$ is the floor operator. For a given ω, there exists only one integer k_ω, called *regularization term* [46] such that $\langle \omega \rangle_{2\pi} = \omega + 2\pi k_\omega$, and it is given by $k_\omega \doteq \left\lfloor \frac{\pi - \omega}{2\pi} \right\rfloor$. An equivalent definition for k_ω is

$$k_\omega = \operatorname*{arg\,min}_{k \in \mathbb{Z}} |\omega + 2\pi k|,$$

and it also holds that

$$|\langle \omega \rangle_{2\pi}| = \min_{k \in \mathbb{Z}} |\omega + 2\pi k|. \tag{3}$$

The modulus is not distributive with respect to addition, but it holds that

$$\langle \omega_1 + \omega_2 \rangle_{2\pi} = \langle \langle \omega_1 \rangle_{2\pi} + \langle \omega_2 \rangle_{2\pi} \rangle_{2\pi}. \tag{4}$$

C. Differential geometry of angles

The *exponential map* for the manifold $\mathrm{SO}(2)$ is a map from the *tangent space* at the identity $\mathrm{so}(2) \simeq \mathbb{R}$ to the manifold:

$$\mathrm{Exp} : \mathbb{R} \to \mathrm{SO}(2).$$

This map is onto (surjective) but not 1-to-1 (bijective).

The *logarithmic map* is the *right inverse* of the exponential map, and it maps an angle in $\mathrm{SO}(2)$ to all possible elements in the tangent space that have the same exponential:

$$\mathrm{Log} : \mathrm{SO}(2) \to \mathcal{P}(\mathbb{R}).$$

Here, "$\mathcal{P}(\mathbb{R})$" denotes the *power set* of $\mathbb{R}$. Note that the fact that the exponential map is not invertible is an intrinsic property and does not depend on a particular parametrization. The logarithmic map satisfies the Lie group property

$$\mathrm{Log}(\boldsymbol{s}^{-1}) = -\mathrm{Log}(\boldsymbol{s}) \tag{5}$$

and the Abelian property

$$\mathrm{Log}(\boldsymbol{s}_1 \boldsymbol{s}_2) = \mathrm{Log}(\boldsymbol{s}_1) + \mathrm{Log}(\boldsymbol{s}_2). \tag{6}$$

The *principal logarithm map* Log_0 is a 1-to-1 function that chooses one particular element on the tangent space:

$$\mathrm{Log}_0 : \mathrm{SO}(2) \to \mathbb{R},$$

namely, the closest to the origin. A property of the principal logarithm is that it can be written with the map $\langle \cdot \rangle_{2\pi}$:

$$\mathrm{Log}_0(\boldsymbol{s}) = \langle \mathrm{Log}(\boldsymbol{s}) \rangle_{2\pi}. \tag{7}$$

If $\mathrm{SO}(2)$ is parametrized by angular coordinates in $(-\pi, +\pi]$, then the coordinate version of Exp is simply the modulus $\langle \cdot \rangle_{2\pi}$, while the principal logarithm maps a rotation matrix to the corresponding angle of rotation in $(-\pi, +\pi]$.

D. Wrapped Gaussian distribution on the circle

The *wrapped* Gaussian distribution on the circle is the generalization of a Gaussian distribution [47], [48], in the sense that it is the solution of the heat equation on the circle, and has several other analogous properties. It can be obtained by applying the exponential map to a Gaussian variable that lives on the tangent space:

$$\boldsymbol{\varepsilon} = \mathrm{Exp}(\epsilon), \quad \text{with } \epsilon \sim \mathcal{N}(0, \sigma^2). \tag{8}$$

We call σ the *scaling parameter* for the wrapped distribution.

The probability density function for a wrapped Gaussian $\mathcal{W}_{\sigma^2} : \mathrm{SO}(2) \to \mathbb{R}^+$ can be written as

$$\mathcal{W}_{\sigma^2}(\boldsymbol{\varepsilon}) = \tfrac{1}{\sigma\sqrt{2\pi}} \sum_{k=-\infty}^{+\infty} \exp\left(\frac{-(\mathrm{Log}_0(\boldsymbol{\varepsilon}) + 2\pi k)^2}{2\sigma^2} \right).$$

Note that Log_0 returns a value in $[-\pi, +\pi)$.

A wrapped Gaussian may show a very different behaviour w.r.t. a Gaussian density. For instance, as the noise increases, a Gaussian density would tend pointwise to 0, while the wrapped Gaussian distribution tends to the uniform distribution on the circle: $\lim_{\sigma^2 \to \infty} \mathcal{W}_{\sigma^2} = 1/2\pi$. Other properties are instead maintained, such as the closure with respect to convolution [48].

Lemma 1. *If $\boldsymbol{s}_1 \sim \mathcal{W}_{\sigma_1^2}$ and $\boldsymbol{s}_2 \sim \mathcal{W}_{\sigma_2^2}$, then the product $\boldsymbol{s}_1 \boldsymbol{s}_2$ has distribution $\mathcal{W}_{\sigma_1^2 + \sigma_2^2}$.*

III. Problem statement: Maximum Likelihood Orientation Estimation

In this section and in Section IV we show how to compute the maximum likelihood estimate for nodes' orientations. Then, in Section V, we discuss limitations of a maximum likelihood estimator applied to the problem at hand before generalizing our results to produce a guaranteed multi-hypothesis estimator.

Let $\mathcal{G}$ be a directed graph with $n+1$ nodes and m edges, and call $\mathcal{E}$ the set of edges in $\mathcal{G}$. Each node is assigned an unknown orientation $\boldsymbol{r}_i^\circ \in \mathrm{SO}(2)$. For any edge $(i,j) \in \mathcal{E}$, the observation $\boldsymbol{d}_{ij} \in \mathrm{SO}(2)$ is a noisy measurement of the relative orientation:

$$\boldsymbol{d}_{ij} = (\boldsymbol{r}_i^\circ)^{-1} \boldsymbol{r}_j^\circ \, \boldsymbol{\varepsilon}_{ij} \in \mathrm{SO}(2), \tag{9}$$

where $\boldsymbol{r}_i^\circ$ is the true (unknown) orientation of the i-th node and $\boldsymbol{\varepsilon}_{ij}$ is a random variable on $\mathrm{SO}(2)$ that represents measurement noise, and that we assume to be distributed according to a *wrapped* Gaussian with scaling parameter $\sigma_{ij} > 0$. We refer to σ_{ij}^2 as the *variance*, as it parametrizes the Gaussian noise before wrapping, see (8). The first formalization of the problem is intrinsic on the manifold $\mathrm{SO}(2)$.

Problem 1 (Intrinsic formulation of maximum likelihood orientation estimation in the absolute frame)**.** *Given the set of relative observations $\{\boldsymbol{d}_{ij}\} \in \mathrm{SO}(2)^m$, for $(i,j) \in \mathcal{E}$ and the corresponding variances $\sigma_{ij}^2 > 0$, find the set of minimizers $S^1 \subset \mathrm{SO}(2)^{n+1}$ that satisfies*

$$S^1 = \operatorname*{arg\,min}_{\{\boldsymbol{r}_i\} \in \mathrm{SO}(2)^{n+1}} \sum_{(i,j) \in \mathcal{E}} -\log \mathcal{W}_{\sigma_{ij}^2}(\boldsymbol{d}_{ij}^{-1} \boldsymbol{r}_i^{-1} \boldsymbol{r}_j). \tag{10}$$

The symbol $\boldsymbol{r}_i$ denotes the optimization variable associated to the orientation of the i-th node, while $\boldsymbol{r}_i^\circ$ is the true value.

Remark 2 (Use of the wrapped Gaussian)**.** *By modeling the noise as a wrapped Gaussian in Problem 1 we made explicit an assumption often kept implicit in related work focusing on the optimization perspective (e.g., [5], [15]), for which one just needs to define an objective function without necessarily defining a generative model for the noise. For small measurement errors, assuming that the objective function is quadratic on the tangent space implies that the noise is distributed according to*

a wrapped Gaussian (Lemma 6). Other modeling possibilities include the Von Mises or Bingham distributions [49]. Using the wrapped Gaussian for angles has the same justifications why we use the Gaussian distribution on Euclidean space: for theoretical reasons, as it is privileged among all distributions because it is the one that maximizes entropy for a given variance; for simplicity, as (wrapped) Gaussian distributions have the semigroup property (Lemma 1) so they allow for finite-dimensional parametrization; for practical reasons, as linear(ized) models play well with Gaussian distributions and sometimes allow for closed forms.

Remark 3 (Independence assumption)**.** *Problem 1 assumes that the orientation measurements are indepedendent. This assumption would be violated if the same raw measurement is used for computing two relative orientation measurements, as it happens in scan matching. In general, assuming uncorrelated measurements leads to estimators which are still consistent though not optimal. Just like most related work, this choice is done here mainly for notational simplicity, as we can write the likelihood as a sum of terms each involving only one measurement. This is convenient in the first part of the paper as we go through multiple reformulations of the problem. Once we get to vectorial notations, then we can easily accommodate correlated measurements; for instance, Problem 5 is valid for any possibly nondiagonal covariance matrix.*

A. Observability and Symmetries

All optimization problems in this paper are posed as finding a *set* of minimizers, rather than the "optimal solution". The set of minimizers is indicated as S^i for Problem i. Only in some cases we will be able to conclude there is a unique solution, and hence S^i has only one element. We need to be careful about keeping track of the set of minimizers, because the first part of the paper (Section IV) consists in transforming one problem to another, sometimes changing the domain or introducing extra variables. To facilitate bookkeeping, we use the concept of *symmetry*: a symmetry of an optimization problem is an invertible transformation of the unknown variables that preserves the value of the objective function. Speaking of symmetries is a formal way of speaking of unobservability from the algebraic/geometric point of view. Problem 1 has one symmetry that corresponds to the well-known fact that the absolute orientations are not observable from only relative measurements: the relative measurements do not change if the nodes' orientations are rotated by the same amount.

Formally, for any rotation matrix $\boldsymbol{s} \in \mathrm{SO}(2)$, the objective function (10) is *invariant* to the invertible function

$$\begin{aligned} f_{\boldsymbol{s}}: \quad \mathrm{SO}(2)^{n+1} &\to \mathrm{SO}(2)^{n+1} \\ \{\boldsymbol{r}_i\} &\mapsto \{\boldsymbol{s}\,\boldsymbol{r}_i\}. \end{aligned}$$

To avoid this ambiguity, and following standard convention, we fix the orientation of the first node to the arbitrary value $\boldsymbol{r}_0 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$. The problem can be then restated using only n variables.

Problem 2. *(Intrinsic formulation of maximum likelihood orientation estimation) Given a set of relative observations $\boldsymbol{d}_{ij} \in \mathrm{SO}(2)^m$, for $(i,j) \in \mathcal{E}$, and the variances $\sigma_{ij}^2 > 0$, find the set of minimizers $S^2 = \{\boldsymbol{r}_i^\star\} \subset \mathrm{SO}(2)^n$ that satisfies*

$$S^2 = \operatorname*{arg\,min}_{\{\boldsymbol{r}_i\} \in \mathrm{SO}(2)^n} \sum_{(i,j)\in\mathcal{E}} -\log \mathcal{W}_{\sigma_{ij}^2}(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j), \qquad (11)$$

having fixed $\boldsymbol{r}_0 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$.

The function has a minimum, because it is defined on the compact set $\mathrm{SO}(2)^n$ and is bounded from below. The minimum is unique in a noiseless setup and for a connected graph.

Proposition 4. *(Observability of orientations) In the noiseless case, $|S^2| = 1$ if and only if the graph is connected.*

The proof is straightforward and proceeds by looking at the measurements along a spanning tree, which uniquely identify the optimal solution [42]. Henceforth the graph $\mathcal{G}$ is assumed to be connected. It has not been known whether the solution is unique in the noisy case; we will show that this is true with probability 1 (Proposition 13).

B. Choosing coordinates

As a first step, we make a choice of coordinates for the nodes' orientations and measurements. We include this passage in the Problem Statement section since it leads to the problem formulation that is commonly adopted in literature.

We use the coordinates $\boldsymbol{\theta}^\circ \in (-\pi, +\pi]^n$ for the true unknown orientations, the coordinates $\boldsymbol{\theta} \in (-\pi, +\pi]^n$ for the optimization variables, and the coordinates $\boldsymbol{\delta} \in (-\pi, +\pi]^m$ for the relative measurements. Formally, these are defined as the principal logarithm of quantities that live on $\mathrm{SO}(2)$:

$$\theta_i^\circ \doteq \mathrm{Log}_0(\boldsymbol{r}_i^\circ), \quad \theta_i \doteq \mathrm{Log}_0(\boldsymbol{r}_i), \quad \delta_{ij} \doteq \mathrm{Log}_0(\boldsymbol{d}_{ij}). \quad (12)$$

The $\boldsymbol{\theta}^\circ$ is the "true" and unknown quantity that we want to estimate. We now want to write the objective function (11) as a least-squares cost, depending on the coordinates $\boldsymbol{\theta}$ and $\boldsymbol{\delta}$. It is only possible to write the likelihood as a quadratic function if σ_{ij} is small enough; as it grows, the likelihood of the measurement tends to a constant that cannot be represented as a quadratic function.

Assumption 5. *The uncertainty of a single measurement does not "spill over" the $\pm\pi$ boundaries: $3\,\sigma_{ij} \ll \pi$.*

This assumption is typically respected in robotics because the measurements are more precise than the given threshold.

Moreover, with a preliminary transformation, we can assume without loss of generality that Assumption 5 is satisfied: because the convolution of two wrapped Gaussians is still a wrapped Gaussian (Lemma 1), we just need to replace one edge with a large variance σ_{ij}^2 with two edges with smaller variances whose sum is σ_{ij}^2.

With this assumptions we can write the likelihood as a quadratic function (but, still, with the modulus operation).

Lemma 6 (Quadratic approximation in coordinates)**.** *If Assumption 5 is satisfied, then*

$$-\log \mathcal{W}_{\sigma_{ij}^2}(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j) \simeq \tfrac{1}{2\sigma_{ij}^2} \left|\langle \theta_j - \theta_i - \delta_{ij} \rangle_{2\pi}\right|^2 + c, \quad (13)$$

with the constant c equal to $\log\left(\sigma_{ij}\sqrt{2\pi}\right)$.

Proof: From the definition of $\boldsymbol{d}_{ij}$ in (9) it follows that

$$
\begin{aligned}
&-\log \mathcal{W}_{\sigma_{ij}^2}(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j) \\
&= -\log\left(\tfrac{1}{\sigma_{ij}\sqrt{2\pi}}\sum_{k=-\infty}^{+\infty}\exp\left(-\frac{(\mathrm{Log}_0(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j)+2\pi k)^2}{2\sigma_{ij}^2}\right)\right), \\
&\qquad \text{(The other summands are negligible for } 3\sigma_{ij} \ll \pi.\text{)} \\
&\simeq -\log\left(\tfrac{1}{\sigma_{ij}\sqrt{2\pi}}\exp\left(\frac{-(\mathrm{Log}_0(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j))^2}{2\sigma_{ij}^2}\right)\right), \\
&\simeq \tfrac{1}{2\sigma_{ij}^2}|\mathrm{Log}_0(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j)|^2 + c.
\end{aligned}
$$

The rest of the proof consists of algebraic manipulation based on properties that we have already introduced:

$$
\begin{aligned}
&|\mathrm{Log}_0(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j)|^2 = \left|\langle \mathrm{Log}(\boldsymbol{d}_{ij}^{-1}\boldsymbol{r}_i^{-1}\boldsymbol{r}_j)\rangle_{2\pi}\right|^2 \\
&\qquad \text{(Using the property of } \mathrm{Log}_0 \text{ in (7).)} \\
&= \left|\langle \mathrm{Log}(\boldsymbol{d}_{ij}^{-1}) + \mathrm{Log}(\boldsymbol{r}_i^{-1}) + \mathrm{Log}(\boldsymbol{r}_j)\rangle_{2\pi}\right|^2 \\
&\qquad \text{(Using the property of SO(2) in (6).)} \\
&= \left|\langle -\mathrm{Log}(\boldsymbol{d}_{ij}) - \mathrm{Log}(\boldsymbol{r}_i) + \mathrm{Log}(\boldsymbol{r}_j)\rangle_{2\pi}\right|^2 \\
&\qquad \text{(Using the property of SO(2) in (5).)} \\
&= \left|\langle -\langle \mathrm{Log}(\boldsymbol{d}_{ij})\rangle_{2\pi} - \langle \mathrm{Log}(\boldsymbol{r}_i)\rangle_{2\pi} + \langle \mathrm{Log}(\boldsymbol{r}_j)\rangle_{2\pi}\rangle_{2\pi}\right|^2 \\
&\qquad \text{(Using the property (4).)}
\end{aligned}
$$

$$
= |\langle \mathrm{Log}_0(\boldsymbol{d}_{ij}) - \mathrm{Log}_0(\boldsymbol{r}_i) + \mathrm{Log}_0(\boldsymbol{r}_j)\rangle_{2\pi}|^2 = |\langle \delta_{ij} + \theta_j - \theta_i\rangle_{2\pi}|^2.
$$

(Using the property (7) and the definition (12).) ∎

Based on Lemma 6, we can restate Problem 2 as the minimization of a least-squares cost.

Problem 3 (Angular coordinates formulation of maximum likelihood orientation estimation)**.** *Given the observations $\delta_{ij} \in (-\pi, +\pi]$, for $(i,j) \in \mathcal{E}$, and the variances $\sigma_{ij}^2 > 0$, find the set of minimizers $S^3 \subset (-\pi, +\pi]^n$ that satisfies*

$$
S^3 = \operatorname*{arg\,min}_{\boldsymbol{\theta}\in(-\pi,+\pi]^n} \sum_{(i,j)\in\mathcal{E}} \tfrac{1}{\sigma_{ij}^2} |\langle \theta_j - \theta_i - \delta_{ij}\rangle_{2\pi}|^2, \qquad (14)
$$

having fixed the first node's orientation to $\theta_0 = 0$.

For clarity, the constant term in (13) is omitted in (14).

1) From Problem 3 to Problem 2: The conversion between the two solutions sets is just a change of coordinates:

$$
S^2 = \mathrm{Exp}(S^3), \qquad S^3 = \mathrm{Log}_0(S^2).
$$

2) Symmetries of Problem 3: If Assumption 5 is satisfied, Problem 3 is just a restatement in different coordinates of Problem 2, so they have the same symmetries. In the noiseless case, the solution is unique (Proposition 4). We still do not know what happens in the noisy case, but we can anticipate (Proposition 13) that, for general data, the solution is unique.

Besides the modulus operation, Problem 3 might appear to be similar to a linear estimation problem, but this is not the case: due to the nonlinearity, the cost function is non-convex and has several local minima (see Figure 2 in [42]).

TABLE II
RELATIONS AMONG THE PROBLEMS DEFINED IN THIS PAPER

problem	*variables*	*solution set*	*symmetry group*
Problem 1	$\{\boldsymbol{r}_i\} \in \mathrm{SO}(2)^{n+1}$	S^1	$\mathrm{SO}(2)$
Fixing reference frame			
Problem 2	$\{\boldsymbol{r}_i\} \in \mathrm{SO}(2)^n$	S^2	none
Choice of coordinates		$\mathrm{Log}_0 \downarrow \uparrow \mathrm{Exp}$	
Problem 3	$\boldsymbol{\theta} \in (-\pi, +\pi]^n$	S^3	none
Real-valued parametrization		$\varphi_4^3 \uparrow$	
Problem 4	$\boldsymbol{x} \in \mathbb{R}^n$	S^4	$\mathbb{Z}^n$
Introduction of regularization terms		$\varphi_5^4 \uparrow$	
Problem 5	$(\boldsymbol{x}, \boldsymbol{k}) \in \mathbb{R}^n \times \mathbb{Z}^m$	S^5	$\mathbb{Z}^n$
Separability of error function		$\varphi_6^5 \uparrow$	
Problem 6	$\boldsymbol{k} \in \mathbb{Z}^m$	S^6	$\mathbb{Z}^n$
Minimality of parametrization		$\varphi_7^6 \uparrow$	
Problem 7	$\boldsymbol{\gamma} \in \mathbb{Z}^\ell$	S^7	none

IV. Estimation on SO(2): from angles to integers

This section shows that the nonlinear, non-convex, constrained Problem 3 is equivalent to an unconstrained quadratic integer optimization problem. We will convert Problem 3, which is defined on $(-\pi, +\pi]^n$, where n is the number of observable nodes, through a series of intermediate formulations, until we arrive at Problem 7, which is defined on the integers $\mathbb{Z}^\ell$, where ℓ is the number of *cycles* in the graph. A schematic representation of the relations among the problems presented in this paper is shown in Table II.

The final result of this section, Theorem 15 on page 10, says that the solution of Problem 3 is almost surely unique, and that we can obtain such solution by solving Problem 7.

There are several intermediate reformulations. The set S^i is the set of minimizers of the i-th problem. At each step we keep track of the size of this set by characterizing its symmetry group. This is important because some of the intermediate formulations, namely Problem 4 to Problem 6, have multiple solutions. These are not ambiguities of the original problem, but rather artifacts of our choice of using a redundant representation.

Section IV-A formulates Problem 4, whose optimization variable $\boldsymbol{x}$ is defined on $\mathbb{R}^n$, hence not constrained in $(-\pi, +\pi]^n$, as in Problem 3. Here, we work on a larger domain, and hence we introduce an additional ambiguity, which is that each entry of $\boldsymbol{x}$ is determined only modulo 2π.

Section IV-B formulates Problem 5, which is defined

on $(\boldsymbol{x}, \boldsymbol{k}) \in \mathbb{R}^n \times \mathbb{Z}^m$. The new variable $\boldsymbol{k} \in \mathbb{Z}^m$ can be interpreted as regularization terms that we introduce to keep track of "angular excess" on each edge.

Section IV-C shows that, given the value of $\boldsymbol{k}$, the value of $\boldsymbol{x}$ can be recovered in a closed form using linear estimation: in fact, if we knew $\boldsymbol{k}$, the problem would be linear.

Section IV-D formulates Problem 6, which is defined only on $\boldsymbol{k} \in \mathbb{Z}^m$. The insight is that the integer and the real part of the problem can be solved separately in a two-stage procedure.

Section IV-E formulates Problem 7, which is defined on an integer variable $\boldsymbol{\gamma} \in \mathbb{Z}^\ell$. While $\boldsymbol{k}$ lives on the edges, $\boldsymbol{\gamma}$ lives on the cycles of the graph and it is a minimal parametrization.

Section IV-F puts together the chain of implications, and shows that the solution set S^7 can be mapped surjectively to S^3, and we can easily compute S^3 once we know S^7.

A. Real-valued formulation

The first step is the reformulation of Problem 3 as an *unconstrained* optimization problem for real variables $\boldsymbol{x} \in \mathbb{R}^n$.

Problem 4 (Real-valued formulation of maximum likelihood orientation estimation)**.** *Given the observations $\delta_{ij} \in (-\pi, +\pi]$, $(i,j) \in \mathcal{E}$ and the corresponding variances $\sigma_{ij}^2 > 0$, find the set $S^4 \subset \mathbb{R}^n$ that satisfies*

$$S^4 = \underset{\boldsymbol{x} \in \mathbb{R}^n}{\arg\min} \sum_{(i,j)\in\mathcal{E}} \tfrac{1}{\sigma_{ij}^2} |\, \langle \theta_j - \theta_i - \delta_{ij} \rangle_{2\pi} \,|^2, \qquad (15)$$

having fixed $\theta_0 = 0$.

1) From Problem 4 to Problem 3: While Problem 4 has multiple solutions due to the symmetry, they are all equivalent when projected down to the manifold using the map

$$\begin{aligned} \varphi_4^3 : \mathbb{R}^n &\rightarrow (-\pi, +\pi]^n \\ \boldsymbol{x} &\mapsto \langle \boldsymbol{x} \rangle_{2\pi} . \end{aligned}$$

Proposition 7. $\varphi_4^3(S^4) = S^3$.

Proof: Using property (4) and noticing that Problem 3 is a constrained version of Problem 4 we can easily demonstrate that Problem 3 and Problem 4 attain the same optimal objective. We use the notation $J(\boldsymbol{x})$, to denote the value of the objective function of the two problems for a given vector $\boldsymbol{x}$. Now, we have to show that (i) for any $\boldsymbol{x}^\star \in S^4$, the variable $\langle \boldsymbol{x}^\star \rangle_{2\pi}$ is in S^3, and (ii) for any solution $\boldsymbol{\theta}^\star \in S^3$ there exists at least one $\boldsymbol{x}^\star \in S^4$, such that $\langle \boldsymbol{x}^\star \rangle_{2\pi} = \boldsymbol{\theta}^\star$. The implication (i) is a direct consequence of fact that the problems attain the same optimal objective: for any $\boldsymbol{x}^\star \in S^4$, property (4) assures that $J(\boldsymbol{x}^\star) = J(\langle \boldsymbol{x}^\star \rangle_{2\pi})$, hence $\langle \boldsymbol{x}^\star \rangle_{2\pi}$ attains the optimal objective and satisfies the constraints of Problem 3; therefore $\langle \boldsymbol{x}^\star \rangle_{2\pi} \in S^3$. Regarding the implication (ii), we note that, for any solution $\boldsymbol{\theta}^\star \in S^3$, we can simply pick $\boldsymbol{x}^\star = \boldsymbol{\theta}^\star$, which implies $\langle \boldsymbol{x}^\star \rangle_{2\pi} = \langle \boldsymbol{\theta}^\star \rangle_{2\pi} = \boldsymbol{\theta}^\star$ (the modulus operation produces no effect since $\boldsymbol{\theta}^\star \in (-\pi, +\pi]^n$). An extended version of this proof can be found in [42]. ∎

Using this result, we can solve the unconstrained Problem 4, obtaining a solution $\boldsymbol{x}^\star \in \mathbb{R}^n$, and then compute an optimal solution of Problem 3 as $\boldsymbol{\theta}^\star \doteq \langle \boldsymbol{x}^\star \rangle_{2\pi} \in (-\pi, +\pi]^n$.

2) Symmetries of Problem 4: Note that this problem has more solutions than Problem 3. This is an artifact of the real-valued parametrization. In particular, if $\boldsymbol{x} \in \mathbb{R}^n$ is a solution, also $\boldsymbol{x} - 2\pi\boldsymbol{p}$ is a solution, for any integer vector $\boldsymbol{p} \in \mathbb{Z}^n$.

B. Mixed-integer formulation

Problem 4 is an optimization problem in real variables, but its residual errors are still nonlinear and difficult to minimize. We now get to the core idea of this paper: instead of solving a *non-convex* problem in *real* variables, we choose to solve a *convex (quadratic)* problem in *mixed (integer and real)* variables. The "trick" is that one can get rid of the modulus in the expression of the residuals: by using the property (3), the terms in the error function (15) can be written as

$$|\, \langle \theta_j - \theta_i - \delta_{ij} \rangle_{2\pi} \,|^2 = \min_{k_{ij} \in \mathbb{Z}} |\theta_j - \theta_i - \delta_{ij} + 2\pi k_{ij}|^2.$$

This can be written in a more compact form using the *reduced incidence matrix* $\boldsymbol{A}$ of the graph (Section II-A). Suppose that there is an ordering of the edges from 1 to m, so that the measurements can be written as an m-dimensional vector $\boldsymbol{\delta} \in (-\pi, +\pi]^m$. Define accordingly the *regularization vector* $\boldsymbol{k} = (k_1 \ \cdots \ k_m)^\mathsf{T} \in \mathbb{Z}^m$, and the *measurement covariance* $\boldsymbol{P}_\delta = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_m^2) \in \mathbb{R}^{m \times m}$. We reformulate the problem using this vector notation.

Problem 5 (Mixed-integer formulation of maximum likelihood orientation estimation)**.** *Given the vector $\boldsymbol{\delta} \in (-\pi, +\pi]^m$ and the diagonal positive definite matrix $\boldsymbol{P}_\delta \in \mathbb{R}^{m \times m}$, find the set of minimizers $S^5 \subset \mathbb{R}^n \times \mathbb{Z}^m$ that satisfy*

$$S^5 = \underset{(\boldsymbol{x},\boldsymbol{k}) \in \mathbb{R}^n \times \mathbb{Z}^m}{\arg\min} \left\| \boldsymbol{A}^\mathsf{T}\boldsymbol{x} - \boldsymbol{\delta} + 2\pi\boldsymbol{k} \right\|^2_{\boldsymbol{P}_\delta^{-1}} . \qquad (16)$$

Problem 5 has quadratic objective and includes both continuous and discrete variables, hence belongs to the class of *mixed-integer convex programs* [50].

1) From Problem 5 to Problem 4: Because $\boldsymbol{k}$ is a slack variable, we obtain S^4 from S^5 with the the projection map

$$\begin{aligned} \varphi_5^4 : \mathbb{R}^n \times \mathbb{Z}^m &\rightarrow \mathbb{R}^n \\ (\boldsymbol{x}, \boldsymbol{k}) &\mapsto \boldsymbol{x}. \end{aligned}$$

Proposition 8. $\varphi_5^4(S^5) = S^4$.

Proof: We show how to transform Problem 4 into Problem 5. Using property (3) we can rewrite (15) as

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} \sum_{(i,j)\in\mathcal{E}} \tfrac{1}{\sigma_{ij}^2} \min_{k_{ij} \in \mathbb{Z}} |\theta_j - \theta_i - \delta_{ij} + 2\pi k_{ij}|^2,$$

which corresponds to

$$\min_{\boldsymbol{x} \in \mathbb{R}^n,\ k_{ij} \in \mathbb{Z}, (i,j)\in\mathcal{E}} \sum_{(i,j)\in\mathcal{E}} \tfrac{1}{\sigma_{ij}^2} |\theta_j - \theta_i - \delta_{ij} + 2\pi k_{ij}|^2. \qquad (17)$$

Therefore, k_{ij} are only slack variables that implicitly represent the modulus operator, and finding the optimal $\boldsymbol{x}$ for (17) is the same as finding the optimal $\boldsymbol{x}$ for Problem 4. ∎

2) Symmetries of Problem 5: Note that we are now working on a larger space $\mathbb{R}^n \times \mathbb{Z}^m$: we are overparametrizing the problem in order to make the corresponding cost function quadratic. Therefore, we might have enlarged the number of solutions. In fact, we introduced the following symmetry. For any vector $\boldsymbol{p} \in \mathbb{R}^n$, such that $\boldsymbol{A}^\mathsf{T}\boldsymbol{p}$ is an integer vector, the following transformation leaves the error function invariant:

$$(\boldsymbol{x}, \boldsymbol{k}) \mapsto (\boldsymbol{x} - 2\pi\boldsymbol{p}, \boldsymbol{k} + \boldsymbol{A}^\mathsf{T}\boldsymbol{p}). \tag{18}$$

Because $\boldsymbol{A}^\mathsf{T}$ has full column rank, this is the only symmetry.

C. Solving for $\boldsymbol{x}$ given known $\boldsymbol{k}$

Before further manipulation of Problem 5, we introduce formulas for the estimation of $\boldsymbol{x}$ in the case $\boldsymbol{k}$ was known.

Once the regularization vector $\boldsymbol{k}$ is known, say $\boldsymbol{k} = \bar{\boldsymbol{k}}$, the optimization problem becomes an unconstrained quadratic problem in $\boldsymbol{x} \in \mathbb{R}^n$:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\|\boldsymbol{A}^\mathsf{T}\boldsymbol{x} - \boldsymbol{\delta} + 2\pi\bar{\boldsymbol{k}}\right\|^2_{\boldsymbol{P}_\delta^{-1}}.$$

This problem can be solved in a closed form. Denote by $\boldsymbol{x}^{\star|\boldsymbol{k}}$ the optimal $\boldsymbol{x}$ for a fixed $\boldsymbol{k}$:

$$\boldsymbol{x}^{\star|\boldsymbol{k}} = (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}(\boldsymbol{\delta} - 2\pi\boldsymbol{k}). \tag{19}$$

D. Separating the integer-valued and the real-valued problems

This section shows that the cost function (16) is separable into two terms, enabling a two-stage optimization in which the cost is first optimized with respect to $\boldsymbol{k}$ and then the optimal choice of $\boldsymbol{x}$ is computed in closed form.

The following lemma gives the separability result. It uses a cycle basis matrix $\boldsymbol{C}$ of the graph $\mathcal{G}$, and it is valid for *any* cycle basis. In Section VI-C we discuss the implications of the choice of a particular cycle basis matrix.

Lemma 9. *For any given cycle basis matrix $\boldsymbol{C}$, minimizing the cost* (16) *is the same as minimizing*

$$\|\boldsymbol{x} - \boldsymbol{x}^{\star|\boldsymbol{k}}\|^2_{(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})} + \|2\pi\boldsymbol{C}\boldsymbol{k} - \boldsymbol{C}\boldsymbol{\delta}\|^2_{(\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T})^{-1}}, \tag{20}$$

where $\boldsymbol{x}^{\star|\boldsymbol{k}}$ is a function of $\boldsymbol{k}$ and is given in (19).

Proof: The proof consists of straightforward algebraic manipulations. For compactness, we name the matrices

$$\boldsymbol{X} \doteq \boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}, \quad \boldsymbol{Y} \doteq \boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T})^{-1}\boldsymbol{C}.$$

Expanding the right hand side of (16), and neglecting the terms that do not depend on the optimization variables, we obtain

$$\begin{aligned} f_1(\boldsymbol{x}, \boldsymbol{k}) \doteq \quad & \|\boldsymbol{x}\|^2_{\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}} - 2\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} + 4\pi\boldsymbol{k}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} \\ & + 4\pi^2\|\boldsymbol{k}\|^2_{\boldsymbol{P}_\delta^{-1}} - 4\pi\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{k}. \end{aligned} \tag{21}$$

To show that minimizing $f_1(\boldsymbol{x}, \boldsymbol{k})$ in (21) is the same as minimizing (20), we rewrite the latter as

$$\begin{aligned} & \|\boldsymbol{x}\|^2_{\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}} - 2\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} + 4\pi\boldsymbol{k}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} \\ & + \|\boldsymbol{\delta}\|^2_{\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1}} + 4\pi^2\|\boldsymbol{k}\|^2_{\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1}} - 4\pi\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1}\boldsymbol{k} \\ & + 4\pi^2\|\boldsymbol{k}\|^2_{\boldsymbol{Y}} - 4\pi\boldsymbol{\delta}^\mathsf{T}\boldsymbol{Y}\boldsymbol{k} + \|\boldsymbol{\delta}\|^2_{\boldsymbol{Y}}. \end{aligned} \tag{22}$$

Because $\|\boldsymbol{\delta}\|^2_{\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1}}$ and $\|\boldsymbol{\delta}\|^2_{\boldsymbol{Y}}$ are constant and do not depend on $\boldsymbol{x}$ and $\boldsymbol{k}$, minimizing (22) is the same as minimizing

$$\begin{aligned} f_2(\boldsymbol{x}, k) \doteq \quad & \|\boldsymbol{x}\|^2_{\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}} - 2\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} + 4\pi\boldsymbol{k}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} \\ & + 4\pi^2\|\boldsymbol{k}\|^2_{\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1}} - 4\pi\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1}\boldsymbol{k} \\ & + 4\pi^2\|\boldsymbol{k}\|^2_{\boldsymbol{Y}} - 4\pi\boldsymbol{\delta}^\mathsf{T}\boldsymbol{Y}\boldsymbol{k}. \end{aligned} \tag{23}$$

Comparing (21) and (23), one concludes that the first three terms in $f_1(\boldsymbol{x}, \boldsymbol{k})$ and $f_2(\boldsymbol{x}, \boldsymbol{k})$ coincide, and it only remains to show equality for the last terms. Rewrite (23) as

$$\begin{aligned} & \|\boldsymbol{x}\|^2_{\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}} - 2\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} + 4\pi\boldsymbol{k}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} \\ & + 4\pi^2\boldsymbol{k}^\mathsf{T}(\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1} + \boldsymbol{Y})\boldsymbol{k} - 4\pi\boldsymbol{\delta}^\mathsf{T}(\boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1} + \boldsymbol{Y})\boldsymbol{k}. \end{aligned} \tag{24}$$

Since $\boldsymbol{P}_\delta$ is positive definite, the technical result of Lemma 22 (in appendix) implies that

$$\boldsymbol{P}_\delta^{-1} = \boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1} + \boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T})^{-1}\boldsymbol{C}.$$

Hence $\boldsymbol{P}_\delta^{-1} = \boldsymbol{P}_\delta^{-1}\boldsymbol{X}\boldsymbol{P}_\delta^{-1} + \boldsymbol{Y}$, and (24) becomes

$$\begin{aligned} & \|\boldsymbol{x}\|^2_{\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}} - 2\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} + 4\pi\boldsymbol{k}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}\boldsymbol{x} \\ & + 4\pi^2\boldsymbol{k}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{k} - 4\pi\boldsymbol{\delta}^\mathsf{T}\boldsymbol{P}_\delta^{-1}\boldsymbol{k}, \end{aligned}$$

which can be easily seen to coincide with (21). Since the objective functions $f_1(\boldsymbol{x}, k)$ and $f_2(\boldsymbol{x}, k)$ coincide, problems (16) and (20) have the same solutions. ∎

A consequence of writing the error function as in (20) is that a separability principle holds: we can obtain the maximum likelihood solution using a two-stage approach: first we estimate the $\boldsymbol{k}$, and then we estimate $\boldsymbol{x}$ given $\boldsymbol{k}$. This aspect is formalized later, in Proposition 10. Intuitively, the cost function (20) comprises two terms, the first that can be made equal to zero choosing $\boldsymbol{x} = \boldsymbol{x}^{\star|\boldsymbol{k}}$, and the second, that does not depend on $\boldsymbol{x}$ and can be minimized by working on $\boldsymbol{k}$. Since we already have a closed-form expression for $\boldsymbol{x}$ given $\boldsymbol{k}$ the only problem that we have to solve is finding $\boldsymbol{k}$.

Problem 6 (Integer formulation of maximum likelihood orientation estimation in edge space)**.** *Given the vector $\boldsymbol{\delta} \in (-\pi, +\pi]^m$ and the diagonal positive definite matrix $\boldsymbol{P}_\delta \in \mathbb{R}^{m\times m}$, and a cycle basis matrix $\boldsymbol{C} \in \mathbb{Z}^{\ell\times m}$, find the set of minimizers $S^6 \subset \mathbb{Z}^m$ that satisfy*

$$S^6 = \arg\min_{\boldsymbol{k}\in\mathbb{Z}^m} \left\|\boldsymbol{C}\boldsymbol{k} - \tfrac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta}\right\|^2_{(\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T})^{-1}}. \tag{25}$$

Notice that (25) is the same as the second summand in (20), and we only rearranged the 2π term.

1) From Problem 6 to Problem 5: Proposition 10 assures that solving Problem 6 is the same as solving Problem 5, if we convert the solutions using the map

$$\begin{aligned} \varphi_6^5 : \mathbb{Z}^m \quad & \to \quad \mathbb{R}^n \times \mathbb{Z}^m \\ \boldsymbol{k} \quad & \mapsto \quad (\boldsymbol{x}^{\star|\boldsymbol{k}}, \boldsymbol{k}). \end{aligned}$$

Proposition 10. $\varphi_6^5(S^6) = S^5$.

Proof: Since $\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T}$ is positive definite, the first term in (20) is non-negative. This implies that, for any $\boldsymbol{k}$, the minimum is attained for $\boldsymbol{x} = \boldsymbol{x}^{\star|\boldsymbol{k}}$ (which annihilates the first summand in the objective function). Moreover, the second summand in (20) does not depend on $\boldsymbol{x}$. ∎

Summarizing the chain of implications presented so far, we conclude that for any solution $\boldsymbol{k}^\star$ of Problem 6 we can obtain a solution $(\boldsymbol{x}^\star, \boldsymbol{k}^\star)$ of Problem 4. Moreover, from $\boldsymbol{x}^\star$ we can easily obtain the solution of our original problem (Problem 3) by applying the modulus operation to $\boldsymbol{x}^\star$.

2) Symmetries of Problem 6: Because (16) and (20) are completely equivalent, they have the same symmetries. However, it is interesting to find the symmetries of Problem 6 directly. Notice that the reorganization of the terms made the term $\boldsymbol{C}\boldsymbol{k}$ explicit. Because the $\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}$ is positive definite, the only symmetries are described by the kernel of $\boldsymbol{C}$. For any integer vector $\boldsymbol{q} \in \ker \boldsymbol{C}$, this transformation does not change the value of the objective function:

$$\boldsymbol{k} \mapsto \boldsymbol{k} + \boldsymbol{q}, \qquad \text{for } \boldsymbol{q} \in \ker \boldsymbol{C} \cap \mathbb{Z}^m.$$

Recall that $\boldsymbol{C}$ is a full-row-rank $\ell \times m$ matrix, where ℓ is the dimension of the cycle space. Its kernel $\ker \boldsymbol{C}$ has thus dimension $m - \ell$, which is equal to n. As it happens, $\boldsymbol{A}^\mathsf{T}$ is an orthogonal complement of $\boldsymbol{C}$, so that it provides a base for its kernel. Therefore, any $\boldsymbol{q} \in \ker \boldsymbol{C} \cap \mathbb{Z}^m$ can be written as $\boldsymbol{q} = \boldsymbol{A}^\mathsf{T}\boldsymbol{p}$, for some $\boldsymbol{p} \in \mathbb{Z}^n$. Therefore, the symmetry can be written as

$$\boldsymbol{k} \mapsto \boldsymbol{k} + \boldsymbol{A}^\mathsf{T}\boldsymbol{p}, \qquad \text{for } \boldsymbol{p} \in \mathbb{Z}^n, \tag{26}$$

which confirms the symmetry in (18).

E. From $\boldsymbol{k}$ towards a minimal parameterization $\boldsymbol{\gamma}$

The cycle basis matrix $\boldsymbol{C}$ is a "fat" $\ell \times m$ matrix, because the number of cycles ℓ is much less than the number of edges m. Therefore, there will be an infinite number of $\boldsymbol{k}^\star$ such that the product $\boldsymbol{C}\boldsymbol{k}^\star$ attains the minimum of (25). This infinite number is precisely described by the symmetry (26). Consequently, we will have an infinite number of optimal orientation estimates $\boldsymbol{x}^{\star|\boldsymbol{k}^\star}$. Fortunately, the next proposition assures that the infinite cardinality of solutions is an artifact created when passing from $\mathrm{SO}(2)$ to the reals. In particular, we show that all vectors $\boldsymbol{k}$ having the same product $\boldsymbol{C}\boldsymbol{k}$ lead to orientation estimates $\boldsymbol{x}^{\star|\boldsymbol{k}}$ that differ by integer multiples of 2π and they give equivalent solutions.

Proposition 11 (Equivalence of $\boldsymbol{k}$ satisfying $\boldsymbol{C}\boldsymbol{k} = \boldsymbol{\gamma}$)**.** *Consider a $\boldsymbol{k}$ and the corresponding orientation estimate $\boldsymbol{x}^{\star|\boldsymbol{k}}$; then any $\bar{\boldsymbol{k}}$ such that $\boldsymbol{C}\bar{\boldsymbol{k}} = \boldsymbol{C}\boldsymbol{k}$ satisfies*

$$\boldsymbol{x}^{\star|\bar{\boldsymbol{k}}} = \boldsymbol{x}^{\star|\boldsymbol{k}} + 2\pi\boldsymbol{D}(\boldsymbol{k} - \bar{\boldsymbol{k}}),$$

where $\boldsymbol{D}$ is a suitable integer matrix.

Proof: Consider two regularization vectors $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ such that $\boldsymbol{C}\boldsymbol{k}_1 = \boldsymbol{C}\boldsymbol{k}_2 = \boldsymbol{\gamma}$. Recall that $\boldsymbol{x}^{\star|\boldsymbol{k}_1} = (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}(\boldsymbol{\delta} - 2\pi\boldsymbol{k}_1)$ and similarly $\boldsymbol{x}^{\star|\boldsymbol{k}_2} = (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}(\boldsymbol{\delta} - 2\pi\boldsymbol{k}_2)$. Define $\boldsymbol{\delta}^{\star|\boldsymbol{k}_1} = \boldsymbol{A}^\mathsf{T}\boldsymbol{x}^{\star|\boldsymbol{k}_1}$ and $\boldsymbol{\delta}^{\star|\boldsymbol{k}_2} = \boldsymbol{A}^\mathsf{T}\boldsymbol{x}^{\star|\boldsymbol{k}_2}$. If the orientation of the first node is set to zero, $\boldsymbol{\delta}^{\star|\boldsymbol{k}_1}$ and $\boldsymbol{\delta}^{\star|\boldsymbol{k}_2}$ uniquely identify $\boldsymbol{x}^{\star|\boldsymbol{k}_1}$ and $\boldsymbol{x}^{\star|\boldsymbol{k}_2}$, since $\boldsymbol{x}^{\star|\boldsymbol{k}_i}$ can be rewritten as an integer-valued linear combination of $\boldsymbol{\delta}^{\star|\boldsymbol{k}_i}, i = 1, 2$; In general, we have $\theta^{\star|\boldsymbol{k}_i} = \boldsymbol{D}\delta^{\star|\boldsymbol{k}_i}, i = 1, 2$, where $\boldsymbol{D}$ is an integer-valued matrix ($\boldsymbol{D}$ is a left integer pseudoinverse of $\boldsymbol{A}^\mathsf{T}$). Therefore, determining $\boldsymbol{\delta}^{\star|\boldsymbol{k}_i}$ is the same as determining $\boldsymbol{x}^{\star|\boldsymbol{k}_i}$, $i = 1, 2$.

The difference $\boldsymbol{\delta}^{\star|\boldsymbol{k}_2} - \boldsymbol{\delta}^{\star|\boldsymbol{k}_1}$ can be written as

$$\begin{aligned}\boldsymbol{\delta}^{\star|\boldsymbol{k}_2} - \boldsymbol{\delta}^{\star|\boldsymbol{k}_1} &= 2\pi\boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}(\boldsymbol{k}_1 - \boldsymbol{k}_2) \qquad (27)\\ &\quad \text{(By Lemma 22)}\\ &= 2\pi(\boldsymbol{k}_1 - \boldsymbol{k}_2) - 2\pi\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T})^{-1}(\boldsymbol{C}\boldsymbol{k}_1 - \boldsymbol{C}\boldsymbol{k}_2).\end{aligned}$$

Since by assumption $\boldsymbol{C}\boldsymbol{k}_1 = \boldsymbol{C}\boldsymbol{k}_2$, the second term in (27) disappears, and one obtains $\boldsymbol{\delta}^{\star|\boldsymbol{k}_2} - \boldsymbol{\delta}^{\star|\boldsymbol{k}_1} = 2\pi(\boldsymbol{k}_1 - \boldsymbol{k}_2)$. Therefore, elements of $\boldsymbol{\delta}^{\star|\boldsymbol{k}_1}$ and $\boldsymbol{\delta}^{\star|\boldsymbol{k}_2}$ only differ by multiples of 2π; then, $\boldsymbol{x}^{\star|\boldsymbol{k}_2} - \boldsymbol{x}^{\star|\boldsymbol{k}_2} = \boldsymbol{D}(\boldsymbol{\delta}^{\star|\boldsymbol{k}_2} - \boldsymbol{\delta}^{\star|\boldsymbol{k}_1}) = 2\pi\boldsymbol{D}(\boldsymbol{k}_1 - \boldsymbol{k}_2)$, and since $\boldsymbol{D}$ is an integer-valued matrix, also the elements of $\boldsymbol{x}^{\star|\boldsymbol{k}_1}$ and $\boldsymbol{x}^{\star|\boldsymbol{k}_2}$ only differ by multiples of 2π. ∎

The previous result enables to solve Problem 6 directly on the slack variable

$$\boldsymbol{\gamma} = \boldsymbol{C}\boldsymbol{k} \in \mathbb{Z}^m.$$

In fact, all $\boldsymbol{k}$ producing the same $\boldsymbol{\gamma} = \boldsymbol{C}\boldsymbol{k}$ are equivalent, in the sense specified in Proposition 11. Because $\boldsymbol{C}$ is an integer matrix and $\boldsymbol{k}$ is an integer vector, also $\boldsymbol{\gamma}$ is an integer vector.

The final formulation of the problem uses only $\boldsymbol{\gamma}$.

Problem 7 (Integer formulation of maximum likelihood orientation estimation in cycle space)**.** *Given the vector $\boldsymbol{\delta} \in (-\pi, +\pi]^m$ and the diagonal positive definite matrix $\boldsymbol{P}_\delta \in \mathbb{R}^{m\times m}$, and a cycle basis matrix $\boldsymbol{C} \in \mathbb{Z}^{\ell\times m}$, find the set of minimizers $S^7 \subset \mathbb{Z}^\ell$ that satisfy*

$$S^7 = \operatorname*{arg\,min}_{\boldsymbol{\gamma}\in\mathbb{Z}^\ell} \|\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}\|^2_{\boldsymbol{P}_\gamma^{-1}} \tag{28}$$

with $\hat{\boldsymbol{\gamma}} = \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta}$, and $\boldsymbol{P}_\gamma = \frac{1}{4\pi^2}\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}$.

1) From Problem 7 to Problem 6: Given a $\boldsymbol{\gamma}$, there is a simple way to compute a $\boldsymbol{k}$ satisfying $\boldsymbol{C}\boldsymbol{k} = \boldsymbol{\gamma}$, assuming that the rows of $\boldsymbol{C}$ are ordered appropriately as in (1).

Lemma 12. *Given a vector $\boldsymbol{\gamma} \in \mathbb{Z}^\ell$ and a cycle basis matrix written as $\boldsymbol{C} = (\boldsymbol{C}_\mathsf{T}\ \boldsymbol{C}_\perp)$, an integer solution to $\boldsymbol{C}\boldsymbol{k} = \boldsymbol{\gamma}$ can be computed as*

$$\boldsymbol{k} = \begin{pmatrix} \boldsymbol{0}_n \\ \boldsymbol{C}_\perp^{-1}\boldsymbol{\gamma} \end{pmatrix}. \tag{29}$$

Proof: From Liebchen [51, Lemma 3 and Theorem 7] it follows that $\boldsymbol{C}_\perp$ is invertible and $\det(\boldsymbol{C}_\perp) = \pm 1$. Moreover, because $\boldsymbol{C}_\perp$ is an integer matrix with unitary determinant, necessarily $\boldsymbol{C}_\perp^{-1}$ is itself integer (see Schrijver [52], or just think that the inverse is the adjoint matrix over the determinant). Therefore, $\boldsymbol{C}_\perp^{-1}\boldsymbol{\gamma}$ is an integer vector. Finally, we can show that $\boldsymbol{C}\boldsymbol{k} = (\boldsymbol{C}_\mathsf{T}\ \boldsymbol{C}_\perp)(\boldsymbol{0}_n^\mathsf{T}\ (\boldsymbol{C}_\perp^{-1}\boldsymbol{\gamma})^\mathsf{T})^\mathsf{T} = \boldsymbol{C}_\perp(\boldsymbol{C}_\perp^{-1}\boldsymbol{\gamma}) = \boldsymbol{\gamma}$. ∎
We use the notation

$$\boldsymbol{C}^\dagger = \begin{pmatrix} \boldsymbol{0}_n \\ \boldsymbol{C}_\perp^{-1} \end{pmatrix}$$

to remark that $\boldsymbol{C}^\dagger$ is a right (integer) pseudoinverse of $\boldsymbol{C}$.

2) Symmetries of Problem 7: The objective function (28) is convex so it would be tempting to just say that there is only one minimum. However, we should be careful because the intuitions of convex optimization often fail in integer programming. For example, Figure 3 in [42] shows a case in which a convex objective function has two integer solutions.

What we can say is that this cannot happen for general data.

Proposition 13. $\left|S^7\right| = 1$ *with probability 1.*

Proof: The set S^7 is the set of minimizers of (28), which has an objective function of the form

$$\|\boldsymbol{\gamma} - \boldsymbol{\mu}\|^2_{\boldsymbol{P}}, \tag{30}$$

with $\boldsymbol{P} \in \mathbb{R}^{\ell\times\ell}$ a positive definite matrix, and $\boldsymbol{\mu} \in \mathbb{R}^\ell$ a random variable which depends on the measurements, and can be seen to be a Gaussian vector. Consider the set of values $M \subset \mathbb{R}^\ell$ such that if $\boldsymbol{\mu} \in M$, then (30) has multiple solutions for $\boldsymbol{\gamma}$. Then it is easy to show that the set M has measure 0 in $\mathbb{R}^\ell$. (For example, we can see how for an infinitesimal perturbation of the mean of the parabola in Figure 3 in [42], the solution set goes from $\{0, 1\}$ to either $\{0\}$ or $\{1\}$.) For demonstrating this claim, consider a generic $\boldsymbol{\mu} \in M$. Since $\boldsymbol{\mu} \in M$ there exist $v \geq 2$ discrete values $\{\boldsymbol{\gamma}^1, \dots, \boldsymbol{\gamma}^v\} \in \mathbb{Z}^\ell$ such that

$$\left\|\boldsymbol{\gamma}^1 - \boldsymbol{\mu}\right\|^2_{\boldsymbol{P}} = \cdots = \|\boldsymbol{\gamma}^v - \boldsymbol{\mu}\|^2_{\boldsymbol{P}}. \tag{31}$$

If we fix $\{\boldsymbol{\gamma}^1, \dots, \boldsymbol{\gamma}^v\}$, and take $\boldsymbol{\mu}$ as the independent variable, the constraints in (31) define an algebraic variety of dimension at most $\ell - 1$, which has measure 0 in $\mathbb{R}^\ell$. ■

F. Inception

Let us link Problem 7 with the other problems presented so far. From a solution $\boldsymbol{\gamma}$ of Problem 7 we can find a solution $\boldsymbol{k}$ of Problem 6 using the formula (29). We call this map φ^6_7:

$$\begin{aligned} \varphi^6_7 : \mathbb{Z}^\ell &\to \mathbb{Z}^m \\ \boldsymbol{\gamma} &\mapsto \boldsymbol{C}^\dagger\boldsymbol{\gamma}. \end{aligned}$$

However, there are much fewer $\boldsymbol{\gamma} \in \mathbb{Z}^\ell$ than $\boldsymbol{k} \in \mathbb{Z}^m$, therefore, using this map on S^7 we will not be able to cover all of S^6: $\varphi^6_7(S^7) \subsetneq S^6$. Likewise, further projecting down to using φ^5_6 we will still lose solutions: $\varphi^5_6 \circ \varphi^6_7(S^7) \subsetneq S^5$, where the symbol $\circ$ denotes the function composition operator. Another step is not enough, as we still have $\varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(S^7) \subsetneq S^4$. We have to go four levels deep. When we finally arrive back to Problem 3, we do obtain all solutions.

Proposition 14 (Inception)**.** $\varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(S^7) = S^3$.

Proof: We first prove the relation $\varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(S^7) \subseteq S^3$. Consider a $\boldsymbol{\gamma}^\star \in S^7$. By construction, $\boldsymbol{k}^\star = \varphi^6_7(\boldsymbol{\gamma}^\star)$ attains a minimum of Problem 6, since $\boldsymbol{\gamma}$ is only a slack variable that substitutes $\boldsymbol{Ck}$. Therefore, $\boldsymbol{k}^\star \in S^6$. Then, applying in sequence Proposition 10, Proposition 7, and Proposition 8, we can easily conclude that $\varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(\boldsymbol{\gamma}^\star) \in S^3$.

Now, we want to prove $\varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(S^7) \supseteq S^3$, that, together with the relation proved in the previous paragraph, demonstrates Proposition 14. Pick a $\boldsymbol{\theta}^\star \in S^3$. As shown in the proof of Proposition 7, $\boldsymbol{x}^\star = \boldsymbol{\theta}^\star$ also belongs to S^4, and it is such that (a) $\varphi^3_4(\boldsymbol{x}^\star) = \boldsymbol{\theta}^\star$. We have already seen in Section IV-B that introducing $\boldsymbol{k}$ is a way to implicitly represent the modulus. If we choose $\boldsymbol{k}^\star = \lfloor(\pi\mathbf{1}_m - \boldsymbol{A}^\mathsf{T}\boldsymbol{x}^\star + \boldsymbol{\delta})/2\pi\rfloor$ (compare with equations (2) and (16)), then the pair $(\boldsymbol{x}^\star, \boldsymbol{k}^\star)$ solves Problem 5; moreover, $(\boldsymbol{x}^\star, \boldsymbol{k}^\star)$ is such that $\varphi^4_5(\boldsymbol{x}^\star, \boldsymbol{k}^\star) = \boldsymbol{x}^\star$, that, using (a), implies (b) $\varphi^3_4 \circ \varphi^4_5(\boldsymbol{x}^\star, \boldsymbol{k}^\star) = \boldsymbol{\theta}^\star$. According to Proposition 10, if $(\boldsymbol{x}^\star, \boldsymbol{k}^\star) \in S^5$, then $\boldsymbol{k}^\star \in S^6$, and then, from (b), we get (c) $\varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6(\boldsymbol{k}^\star) = \boldsymbol{\theta}^\star$. Finally, if $\boldsymbol{k}^\star$ is an optimal solution for Problem 6, we can pick $\boldsymbol{\gamma}^\star = \boldsymbol{Ck}^\star$ and guarantee that $\boldsymbol{\gamma}^\star$ is an optimal solution for Problem 7 (there is only a change of variables between the two problems), i.e., $\boldsymbol{\gamma}^\star \in S^7$. Concluding, for a given $\boldsymbol{\theta}^\star \in S^3$, we found a $\boldsymbol{\gamma}^\star \in S^7$, that is such that $\boldsymbol{\theta}^\star = \varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(\boldsymbol{\gamma}^\star)$. ■

The following theorem, whose proof is a direct consequence of Proposition 13 and 14, summarizes what we have learned.

Theorem 15 (Mixed-integer maximum likelihood estimation on $SO(2)$)**.** *The solution of Problem 3 is almost surely unique, and, given the solution $\boldsymbol{\gamma}^\star$ of*

$$\boldsymbol{\gamma}^\star = \operatorname*{arg\,min}_{\boldsymbol{\gamma}\in\mathbb{Z}^\ell} \left\|\boldsymbol{\gamma} - \tfrac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta}\right\|^2_{\boldsymbol{P}_\gamma^{-1}},$$

the solution of Problem 3 can be computed as

$$\boldsymbol{\theta}^\star = \left\langle (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\left(\boldsymbol{\delta} - 2\pi\boldsymbol{C}^\dagger\boldsymbol{\gamma}^\star\right)\right\rangle_{2\pi}, \tag{32}$$

with $\boldsymbol{C}^\dagger \doteq \begin{pmatrix} \mathbf{0}_n \\ \boldsymbol{C}_\perp^{-1} \end{pmatrix}$.

Equation (32) is a closed-form expression for the mapping $\varphi^3_4 \circ \varphi^4_5 \circ \varphi^5_6 \circ \varphi^6_7(\gamma^\star)$, hence the result is a more explicit version of Proposition 14.

In contrast with iterative optimization techniques, the proposed computation of $\boldsymbol{\theta}^{\star|\boldsymbol{\gamma}^\star}$ does not suffer from local minima, assuming that we are able to compute $\boldsymbol{\gamma}^\star$.

Two further challenges stand in the way. First, Problem 7 is NP-hard [50]. Several algorithms have been proposed in literature to solve integer quadratic programming (see, e.g., [41], [53], [54] and references therein); however, for the hardness of the problem, one cannot expect to solve exactly and *quickly* large-scale problems; for example, Chang and Golub [55] report computational times above 10 seconds in problem instances with less than 50 variables. Similar numerical results are reported by Jazaeri *et al.* [54]. Second, as explained in the next section, computing only the maximum likelihood estimate does not guarantee to have an accurate orientation estimate.

V. Limitations of the maximum likelihood estimate

Section IV has shown that the maximum likelihood estimation problem on the orientations, which lie on the product manifold $\mathrm{SO}(2)^n$, is equivalent to an optimization problem in the variable $\boldsymbol{\gamma}$, which lies on the integers lattice $\mathbb{Z}^\ell$. This section shows that, if the maximum likelihood solution $\boldsymbol{\gamma}^\star$ is different than the "true" value $\boldsymbol{\gamma}^\circ$, then there is a bias in the estimate of $\boldsymbol{x}$ (Lemma 16), which raises the mean square estimation error.

A. Distribution of maximum likelihood estimate

To define the "true" value of $\boldsymbol{\gamma}$ we rewrite the measurement model as

$$\boldsymbol{\delta} = \langle \boldsymbol{A}^\mathsf{T}\boldsymbol{\theta}^\circ + \boldsymbol{\epsilon}\rangle_{2\pi},$$

where $\boldsymbol{\theta}^\circ$ is the "true" value of the nodes' orientations. From (2) the measurement model can be written as

$$\boldsymbol{\delta} = \boldsymbol{A}^\mathsf{T}\boldsymbol{\theta}^\circ + 2\boldsymbol{k}^\circ\pi + \boldsymbol{\epsilon}, \tag{33}$$

where $\boldsymbol{k}^\circ$ is the "true" regularization vector, and has the expression $\boldsymbol{k}^\circ \doteq \left\lfloor(\pi\mathbf{1}_m - \boldsymbol{A}^\mathsf{T}\boldsymbol{\theta}^\circ - \boldsymbol{\epsilon})/2\pi\right\rfloor$. Define the "true" value of $\boldsymbol{\gamma}$ as

$$\boldsymbol{\gamma}^\circ \doteq \boldsymbol{C}\boldsymbol{k}^\circ.$$

For a given $\boldsymbol{\gamma}$, define the *real-valued estimate* $\boldsymbol{x}^{\star|\boldsymbol{\gamma}}$ as

$$\boldsymbol{x}^{\star|\boldsymbol{\gamma}} \doteq (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}(\boldsymbol{\delta} - 2\pi\boldsymbol{C}^\dagger\boldsymbol{\gamma}) \in \mathbb{R}^n, \quad (34)$$

and the corresponding (wrapped) estimate $\boldsymbol{\theta}^{\star|\boldsymbol{\gamma}}$ as:

$$\boldsymbol{\theta}^{\star|\boldsymbol{\gamma}} \doteq \langle \boldsymbol{x}^{\star|\boldsymbol{\gamma}} \rangle_{2\pi} \in (-\pi, +\pi]^n.$$

We call $\boldsymbol{x}^{\star|\boldsymbol{\gamma}^\star}$ the *real-valued maximum likelihood estimate* (it is $\boldsymbol{x}^{\star|\boldsymbol{\gamma}}$ computed for $\boldsymbol{\gamma}^\star$). Comparing with (32), the reader can notice that *real-valued maximum likelihood estimate* is simply the maximum likelihood orientation estimate before applying the modulus. We can give a full characterization of the distribution of the real-valued maximum likelihood estimator.

Lemma 16. *The real-valued maximum likelihood estimator $\boldsymbol{x}^{\star|\boldsymbol{\gamma}^\star}$ can be written as*

$$\begin{aligned} \boldsymbol{x}^{\star|\boldsymbol{\gamma}^\star} \quad = \quad & \boldsymbol{\theta}^\circ + & (35)\\ & 2\pi\boldsymbol{p} + & (36)\\ & (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{\epsilon} + & (37)\\ & 2\pi(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{C}^\dagger(\boldsymbol{\gamma}^\circ - \boldsymbol{\gamma}^\star), & (38) \end{aligned}$$

where the term (35) *is the true value of the orientations; the term* (36) *contains some integer vector $\boldsymbol{p}$ and has no effect once the modulus is applied to $\boldsymbol{x}^{\star|\boldsymbol{\gamma}^\star}$; the term* (37) *contains the noise which would appear even if the problem were linear; and, finally, the term* (38) *contains an* additional *bias, which is proportional to the mismatch between $\boldsymbol{\gamma}^\star$ and $\boldsymbol{\gamma}^\circ$. In particular, if $\boldsymbol{\gamma}^\star = \boldsymbol{\gamma}^\circ$, then the vector $\boldsymbol{x}^{\star|\boldsymbol{\gamma}}$ is Normally distributed with mean $\boldsymbol{\theta}^\circ + 2\pi\boldsymbol{p}$ and covariance matrix $(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}$.*

Proof: Substitute $\boldsymbol{\delta}$ from (33) in the expression of the real-valued estimator to obtain

$$\boldsymbol{x}^{\star|\boldsymbol{\gamma}} = (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\left(\boldsymbol{A}^\mathsf{T}\boldsymbol{\theta}^\circ + \boldsymbol{\epsilon} + 2\pi(\boldsymbol{k}^\circ - \boldsymbol{C}^\dagger\boldsymbol{\gamma})\right).$$

Rewrite $\boldsymbol{k}^\circ = \boldsymbol{C}^\dagger\boldsymbol{\gamma}^\circ + \boldsymbol{k}^\perp$, as to separate $\boldsymbol{k}^\circ$ into two vectors: the first is $\boldsymbol{C}^\dagger\boldsymbol{\gamma}^\circ$, which, by construction, satisfies $\boldsymbol{C}(\boldsymbol{C}^\dagger\boldsymbol{\gamma}^\circ) = \boldsymbol{\gamma}^\circ$. The second belongs to the kernel of $\boldsymbol{C}$, and then satisfies $\boldsymbol{C}\boldsymbol{k}^\perp = \boldsymbol{0}_\ell$. Note that $\boldsymbol{k}^\perp$ is an integer vector, since both $\boldsymbol{k}^\circ$ and $\boldsymbol{C}^\dagger\boldsymbol{\gamma}^\circ$ are integer vectors by construction. According to the result in Proposition 11, the term $\boldsymbol{k}^\perp$ only adds multiples of 2π to the estimate

$$\boldsymbol{x}^{\star|\boldsymbol{\gamma}} = (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\left(\boldsymbol{A}^\mathsf{T}\boldsymbol{\theta}^\circ + \boldsymbol{\epsilon} + 2\pi\boldsymbol{C}^\dagger(\boldsymbol{\gamma}^\circ - \boldsymbol{\gamma})\right) + 2\pi\boldsymbol{p},$$

with $\boldsymbol{p} = \boldsymbol{D}\boldsymbol{k}^\perp$, for some integer matrix $\boldsymbol{D}$. Multiplying the parentheses gives the desired result for $\boldsymbol{\gamma} = \boldsymbol{\gamma}^\star$. ∎

One consequence of this result is that the maximum likelihood estimate of $\boldsymbol{x}$ computed from $\boldsymbol{\gamma}^\star$ is not necessarily the one with the minimum estimation error, because one cannot guarantee in general that the optimal solution $\boldsymbol{\gamma}^\star$ of Problem 7 is such that $\boldsymbol{\gamma}^\star = \boldsymbol{\gamma}^\circ$. Section V-B in [42] shows a simple example in which the maximum likelihood estimator fails to retrieve $\boldsymbol{\gamma}^\circ$. Consequently, in order to attain a small estimation error, one should look for $\boldsymbol{\gamma}^\circ$, instead of simply computing $\boldsymbol{\gamma}^\star$.

VI. A Multi-hypothesis Estimator for $\boldsymbol{\gamma}^\circ$

This section describes an algorithm that we call INTEGER-SCREENING which is able to find a set Γ of integer vectors that contains $\boldsymbol{\gamma}^\circ$ with a desired probability.

Section VI-A describes how to derive an estimator for $\boldsymbol{\gamma}^\circ$ which allows building a confidence set. Section VI-B describes the INTEGER-SCREENING algorithm, which builds a set of integer vectors Γ that contains $\boldsymbol{\gamma}^\circ$ with desired probability. Section VI-C discusses the influence of the cycle basis matrix in the construction of Γ and formally proves that the minimum cycle basis matrix MCB is the optimal choice that minimizes the expected size of the set Γ.

A. An estimator of $\boldsymbol{\gamma}^\circ$

From the knowledge of a cycle basis matrix $\boldsymbol{C}$, the measurements $\boldsymbol{\delta}$, and the covariance matrix $\boldsymbol{P}_\delta$ we can design an estimator $\hat{\boldsymbol{\gamma}}$ for the unknown integer vector $\boldsymbol{\gamma}^\circ$.

Proposition 17. *The real-valued vector*

$$\hat{\boldsymbol{\gamma}} \doteq \tfrac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta} \quad (39)$$

is a Normally distributed estimator of $\boldsymbol{\gamma}^\circ$, with covariance matrix

$$\boldsymbol{P}_\gamma \doteq \tfrac{1}{4\pi^2}\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}. \quad (40)$$

Proof: Multiply both members of (33) by $\boldsymbol{C}$ to obtain

$$\boldsymbol{C}\boldsymbol{\delta} = \boldsymbol{C}(\boldsymbol{A}^\mathsf{T}\boldsymbol{\theta}^\circ + 2\pi\boldsymbol{k}^\circ + \boldsymbol{\epsilon}).$$

By Lemma 21, the term $\boldsymbol{C}\boldsymbol{A}^\mathsf{T}$ is equal to zero. Reordering the remaining terms, we get $\boldsymbol{C}\boldsymbol{k}^\circ = \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta} - \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\epsilon}$. Using the definition $\boldsymbol{\gamma}^\circ = \boldsymbol{C}\boldsymbol{k}^\circ$ given in Section V, this implies $\boldsymbol{\gamma}^\circ = \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta} - \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\epsilon}$, from which the thesis follows. ∎

The availability of this estimator allows the computation of the set Γ, as described in the following section.

B. The INTEGER-SCREENING algorithm

The INTEGER-SCREENING algorithm (Algorithm 1) computes a finite set of integer vectors that contains $\boldsymbol{\gamma}^\circ$ with a user-specified probability. The algorithm is based on two simple ideas: *marginalization* and *conditioning*. We use *marginalization* to exclude non-plausible values for the elements of $\boldsymbol{\gamma}^\circ$. Since $\hat{\boldsymbol{\gamma}}$ is Normally distributed, i.e., $\hat{\boldsymbol{\gamma}} \sim \mathcal{N}(\boldsymbol{\gamma}^\circ, \boldsymbol{P}_\gamma)$, also the marginal distribution of the i-th element $\hat{\gamma}_i$ is a Gaussian with mean γ_i°. Therefore we easily derive a confidence interval for the single element, based on a given confidence level. If the interval contains only one integer, then we can assign that value to the element with the specified confidence. Once we are sure of the value of one element, say $\gamma_i^\circ = u_i$, we use *conditioning* to reduce the plausible values of the others, by conditioning on $\gamma_i^\circ = u_i$. These two ideas suggest an iterative algorithm that looks for elements of $\boldsymbol{\gamma}^\circ$ that can be determined unambiguously, and then uses those constraints to further shrink the uncertainty on the remaining elements.

The input to Algorithm 1 consists of the real vector $\hat{\boldsymbol{\gamma}}$, the positive semidefinite matrix $\boldsymbol{P}_\gamma$, and the confidence level α. It is assumed that $\hat{\boldsymbol{\gamma}}$ is a Normally distributed estimator of $\boldsymbol{\gamma}^\circ \in \mathbb{Z}^\ell$ and that $\boldsymbol{P}_\gamma$ is its covariance. The output of the algorithm is a set Γ of integer vectors that has probability no smaller than α of containing $\boldsymbol{\gamma}^\circ$; the probability bound is derived in Proposition 18.

Throughout the execution, the set $\mathcal{U}^{(k)}$ contains the indices of the elements of $\boldsymbol{\gamma}^\circ$ that are uniquely identified at iteration k,

Algorithm 1: INTEGER-SCREENING

lstlisting

1 **input**:
2 a real vector $\hat{\boldsymbol{\gamma}} \in \mathbb{R}^\ell$
3 a positive definite matrix $\boldsymbol{P}_\gamma \in \mathbb{R}^{\ell \times \ell}$
4 a confidence level $\alpha \in (0, 1)$
5 **precondition**: $\hat{\boldsymbol{\gamma}} \sim \mathcal{N}(\boldsymbol{\gamma}^\circ, \boldsymbol{P}_\gamma)$
6 **output**: a subset of $\mathbb{Z}^\ell$ containing $\boldsymbol{\gamma}^\circ$ with probability at least α
7 **variables**:
8 $\mathcal{U}^{(k)} \subseteq \{1, \dots, \ell\}$ # Indices determined at iter. k
9 $\mathcal{R}^{(k)} \subseteq \{1, \dots, \ell\}$ # Indices that are still ambiguous at iter. k
10 $\langle \boldsymbol{\zeta}_{\mathcal{R}^{(k)}}, \boldsymbol{P}_{\mathcal{R}^{(k)}} \rangle$ # Current estimate of γ_i°, $i \in \mathcal{R}^{(k)}$
11 $\mathcal{I}_i^{(k)} \subset \mathbb{R}$ # Current confidence interval for γ_i°, $i = \{1, \dots, \ell\}$
12 $\Gamma_i^{(k)} \subset \mathbb{Z}$ # Current admissible values for γ_i°, $i = \{1, \dots, \ell\}$
13
14 $\mathcal{R}^{(1)} \leftarrow \{1, \dots, \ell\}$
15 $\langle \boldsymbol{\zeta}_{\mathcal{R}^{(1)}}, \boldsymbol{P}_{\mathcal{R}^{(1)}} \rangle \leftarrow \langle \hat{\boldsymbol{\gamma}}, \boldsymbol{P}_\gamma \rangle$
16 **for** k **in** $\{1, 2, \dots\}$:
17 # Compute confidence sets using marginal probabilities
18 $\mathcal{U}^{(k)} = \emptyset$
19 **for** i **in** $\mathcal{R}^{(k)}$:
20 $\eta = \alpha^{\frac{1}{\ell}}$
21 $b_i \leftarrow \sqrt{\boldsymbol{P}_{ii}^{(k)} \chi^2_{1,\eta}}$, with $\eta = \alpha^{1/\ell}$
22 $\mathcal{I}_i^{(k)} \leftarrow [\zeta_i^{(k)} - b_i, \zeta_i^{(k)} + b]$
23 $\Gamma_i^{(k)} \leftarrow \mathcal{I}_i^{(k)} \cap \mathbb{Z}$
24 # Check if the set contains a single integer
25 **if** $|\Gamma_i^{(k)}| = 1$:
26 $\mathcal{U}^{(k)} \leftarrow \mathcal{U}^{(k)} \cup \{i\}$
27 **end**
28 # Preserve the previous confidence sets
29 **for** i **in** $\{1, \dots, \ell\} \setminus \mathcal{R}^{(k)}$:
30 $\mathcal{I}_i^{(k)} \leftarrow \mathcal{I}_i^{(k-1)}$
31 $\Gamma_i^{(k)} \leftarrow \Gamma_i^{(k-1)}$
32 **end**
33 # Update the set of indices that are still ambiguous
34 $\mathcal{R}^{(k+1)} \leftarrow \mathcal{R}^{(k)} \setminus \mathcal{U}^{(k)}$
35 **if** $\mathcal{U}^{(k)} = \emptyset$: # Break if there is no progress
36 **break**
37 # Otherwise condition on the elements that we determined
38 $\langle \boldsymbol{\zeta}_{\mathcal{R}^{(k+1)}}, \boldsymbol{P}_{\mathcal{R}^{(k+1)}} \rangle \leftarrow \text{CONDITION}(\langle \boldsymbol{\zeta}_{\mathcal{R}^{(k)}}, \boldsymbol{P}_{\mathcal{R}^{(k)}} \rangle, \boldsymbol{\gamma}_{\mathcal{U}^{(k)}} = \Gamma_{\mathcal{U}^{(k)}})$
39 **end**
40 $\Gamma \leftarrow \Gamma_1^{(k)} \times \Gamma_2^{(k)} \times 1 \dots \Gamma_\ell^{(k)}$
41 **return** Γ

External functions:

- CONDITION$(\langle \boldsymbol{\zeta}, \boldsymbol{P} \rangle, \boldsymbol{\gamma}_i = u_i)$ computes the conditional distribution given the constraints that some components are known.

and, conversely, the set $\mathcal{R}^{(k)}$ that contains the indices that are still ambiguous. At the beginning (line 14), $\mathcal{R}^{(k)} = \{1, \dots, \ell\}$ as no element has been identified.

We use $\boldsymbol{\gamma}^\circ_{\mathcal{U}^{(k)}}$ to indicate the subvector of $\boldsymbol{\gamma}^\circ$ at the indices given by $\mathcal{U}^{(k)}$, and $\boldsymbol{\gamma}^\circ_{\mathcal{R}^{(k)}}$ to denote the elements of $\boldsymbol{\gamma}^\circ$ at the indices given by $\mathcal{R}^{(k)}$. The algorithm updates two variables $\boldsymbol{\zeta}_{\mathcal{R}^{(k)}}$ and $\boldsymbol{P}_{\mathcal{R}^{(k)}}$, preserving the invariant

$$\boldsymbol{\zeta}_{\mathcal{R}^{(k)}} \sim \mathcal{N}(\boldsymbol{\gamma}^\circ_{\mathcal{R}^{(k)}}, \boldsymbol{P}_{\mathcal{R}^{(k)}}),$$

i.e., they describe a Normally distributed estimator of the elements $\boldsymbol{\gamma}^\circ_{\mathcal{R}^{(k)}}$ that have not been identified yet. The invariant holds at the beginning as the variables are initialized to $\boldsymbol{\zeta}_{\mathcal{R}^{(k)}} = \hat{\boldsymbol{\gamma}}$ and $\boldsymbol{P}_{\mathcal{R}^{(k)}} = \boldsymbol{P}_\gamma$ (line 15).

At a generic iteration k, the algorithm computes the confidence set for each γ_i°, $i \in \mathcal{R}^{(k)}$. Since $\boldsymbol{\zeta}_{\mathcal{R}^{(k)}}$ is Normally distributed, also each component $\zeta_i^{(k)}$ is Normally distributed with mean γ_i° and variance given by the i-th element of the covariance $\boldsymbol{P}_{ii}^{(k)}$. Therefore, with probability η, it holds that

$$\gamma_i^\circ \in \left[\ \zeta_i^{(k)} - b, \zeta_i^{(k)} + b \ \right] \doteq \mathcal{I}_i^{(k)},$$

where $b = \sqrt{\boldsymbol{P}_{ii}^{(k)} \chi^2_{1,\eta}}$, and $\chi^2_{1,\eta}$ is the quantile of the χ^2 distribution with 1 degree of freedom and upper tail probability equal to η (lines 21–22). Note that the confidence interval depends on $\eta = \alpha^{\frac{1}{\ell}}$. The relation between α and η is justified by Proposition 18.

Then, the algorithm computes all the integers within the interval $\mathcal{I}_i^{(k)}$, obtaining $\Gamma_i^{(k)} = \mathcal{I}_i^{(k)} \cap \mathbb{Z}$ (line 23). If the set $\Gamma_i^{(k)}$ contains a single integer, say u_i, then with probability η it holds that $\gamma_i^\circ = u_i$, and the algorithm adds the index i to the set of uniquely determined elements $\mathcal{U}^{(k)}$ (line 26).

After checking all sets $\Gamma_i^{(k)}$, $i \in \mathcal{R}^{(k)}$ (line 27), we have the set $\mathcal{U}^{(k)}$, that contains all the elements in $\boldsymbol{\gamma}^\circ$ that we uniquely determined at the current iteration. Clearly, these indices can be removed from the ones that are still ambiguous (line 34). Moreover, we can exploit this information to infer the value of the remaining elements of $\boldsymbol{\gamma}^\circ$, by computing the density $\mathbb{P}(\boldsymbol{\zeta}_{\mathcal{R}^{(k)}} | \boldsymbol{\zeta}_{\mathcal{U}^{(k)}} = \Gamma_{\mathcal{U}^{(k)}})$, which is the conditional density of the elements that are still ambiguous, given the elements uniquely defined (line 38). Since the original density is a Gaussian, also the conditional density is a Gaussian, with mean $\boldsymbol{\zeta}_{\mathcal{R}^{(k+1)}}$ and covariance $\boldsymbol{P}_{\mathcal{R}^{(k+1)}}$. Therefore, at the end of the iteration, we have a unique value for the elements in $\mathcal{U}^{(k)}$ and a probabilistic description (i.e., mean and covariance) of the elements in $\mathcal{R}^{(k+1)}$. Since the conditioning may have shrunk the uncertainty on some element, we proceed to the next iteration. If the set $\mathcal{U}^{(k)}$ is empty, it means that we are not able to make any progress and the loop exits (line 36). Notice that when conditioning on some component of $\boldsymbol{\gamma}^\circ$ we reduce the size of the mean vector and the covariance matrix. At iteration k, it holds $\boldsymbol{\zeta}_{\mathcal{R}^{(k)}} \in \mathbb{R}^{|\mathcal{R}^{(k)}|}$ and $\boldsymbol{P}_{\mathcal{R}^{(k)}} \in \mathbb{R}^{|\mathcal{R}^{(k)}| \times |\mathcal{R}^{(k)}|}$. The algorithm performs at most $K \leq \ell$ iterations because, at each iteration, at least one additional element of $\boldsymbol{\gamma}^\circ \in \mathbb{R}^\ell$ is determined. After the algorithm stops we have a collection of confidence sets $\Gamma_i^{(K)} \subset \mathbb{Z}$, $i \in \{1, \dots, \ell\}$. If the admissible elements γ_i° belongs to $\Gamma_i^{(K)}$, then it must hold that

$$\boldsymbol{\gamma}^\circ \in \Gamma_1^{(K)} \times \Gamma_2^{(K)} \times \ldots \times \Gamma_\ell^{(K)} \doteq \Gamma,$$

where $\times$ denotes the Cartesian product of sets (line 40). Proposition 18 bounds the probability of $\boldsymbol{\gamma}^\circ \in \Gamma$.

Proposition 18 (Correctness of INTEGER-SCREENING)**.** *The integer vector $\boldsymbol{\gamma}^\circ$ is in the set Γ returned by Algorithm 1 with probability no smaller than α.*

Proof: The complete proof is given in [42]. The proof proceeds as follows. By construction, the sets $\mathcal{U}^{(1)}, \cdots, \mathcal{U}^{(k)}, \mathcal{R}^{(k+1)}$ are disjoint and, at the end of each iteration $k = \{1, \dots, K\}$ it holds:

$$|\mathcal{U}^{(1)}| + \cdots + |\mathcal{U}^{(k)}| + |\mathcal{R}^{(k+1)}| = \ell.$$

At iteration k, the intervals $\mathcal{I}_i^{(k)}$ are built from the marginals of a Normal distribution. Therefore, the technical result of Lemma 23 (in Appendix) guarantees that:

$$\mathbb{P}\Big(\bigwedge_{i\in\{1,\dots,\ell\}} \gamma_i^\circ \in \mathcal{I}_i^{(k)}\Big) \geq \eta^{|\mathcal{U}^{(1)}|+\dots+|\mathcal{U}^{(k)}|+|\mathcal{R}^{(k+1)}|} = \eta^\ell$$

which leads to the desired result for $\eta^\ell = \alpha$. ∎

C. Optimal choice of the cycle basis matrix

So far we have not discussed how to choose the cycle basis matrix $\boldsymbol{C}$ and if there is a particular selection that turns out to be more convenient in the screening of admissible vectors $\boldsymbol{\gamma}^\circ$. The choice of $\boldsymbol{C}$ influences the quality of the estimator $\hat{\boldsymbol{\gamma}}$, appearing in both the expression of the estimate and in its covariance $\boldsymbol{P}_\gamma$, as per (39) and (40). Therefore, we are now interested in investigating the choice of $\boldsymbol{C}$ that leads to the most informative estimate of $\boldsymbol{\gamma}^\circ$.

Proposition 19 (Optimal choice of cycle basis matrix)**.** *Choosing $\boldsymbol{C}$ to be the minimum (uncertainty) cycle basis matrix* MCB *makes $\hat{\boldsymbol{\gamma}}$ a minimum variance unbiased estimator of $\boldsymbol{\gamma}^\circ$ within the class of estimators $\{\hat{\boldsymbol{\gamma}} = \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta} \ : \ \boldsymbol{C} \in \mathcal{C}_\mathcal{G}\}$.*

Proof: We already know that $\hat{\boldsymbol{\gamma}}$ is unbiased. For an unbiased estimator, the variance is equal to the mean square error. For a Normally distributed estimator, the mean square error is proportional to the trace of the covariance matrix. Therefore, we have to show that choosing $\boldsymbol{C} = \mathsf{MCB}$ minimizes $\mathsf{Trace}(\boldsymbol{P}_\gamma) = \frac{1}{4\pi^2}\mathsf{Trace}(\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T})$. The t-th term on the main diagonal of $\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}$ is $\boldsymbol{c}_t\boldsymbol{P}_\delta\boldsymbol{c}_t^\mathsf{T} = \sum_{(i,j)\in\boldsymbol{c}_t}\sigma_{ij}^2$, where the notation $(i,j) \in \boldsymbol{c}_t$ means "for all edges that belong to the t-th cycle". The previous expression coincides with the weight of the cycle $\boldsymbol{c}_t$ under the weight function $\boldsymbol{w} : (i,j) \to \sigma_{ij}^2$.

Therefore, the trace of $\boldsymbol{P}_\gamma = \boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}$ is equal to the sum of the weights of the cycles in the cycle basis, which by definition is $\boldsymbol{W}(\boldsymbol{C},\boldsymbol{w})$. Therefore the minimum cycle basis, which minimizes $\boldsymbol{W}(\boldsymbol{C},\boldsymbol{w})$, also minimizes $\mathsf{Trace}(\boldsymbol{P}_\gamma)$. ∎

What is the practical advantage of having a minimum variance estimator in our problem? The confidence set Γ that we build in Algorithm 1 is directly influenced by the variance of the estimator $\hat{\boldsymbol{\gamma}}$; in Algorithm 1 the diagonal elements of $\boldsymbol{P}_\gamma$ define the width of the confidence intervals, therefore, since the minimum cycle basis matrix minimizes the sum of the diagonal elements of $\boldsymbol{P}_\gamma$ (i.e., its trace), then it also minimizes the widths of the confidence intervals used for the INTEGER-SCREENING algorithm. Therefore, it enables the determination of a small set Γ of admissible integer vectors.

VII. The MOLE2D Orientation Estimation algorithm

We can now summarize the findings we presented so far in a single algorithm, that allows computing a (multi-hypothesis) estimate of robot orientations, which we call MOLE2D (*Multi-hypothesis Orientation-from-Lattice Estimation in 2D*).

Pseudocode for MOLE2D is given in Algorithm 2. The input to the algorithm is the reduced incidence matrix $\boldsymbol{A}$ (which describes the graph), the measurements $\boldsymbol{\delta}$, their covariance $\boldsymbol{P}_\delta$, and a parameter α which gives the desired confidence level. The output is the set of estimates Θ.

Algorithm 2: MOLE2D

lstlisting
1 **input**:
2 reduced incidence matrix $\boldsymbol{A}$ (topology of the graph)
3 measurements $\boldsymbol{\delta}$
4 covariance $\boldsymbol{P}_\delta$
5 confidence level $\alpha \in (0,1)$
6 **output**: set of estimates of $\boldsymbol{\theta}^\circ$
7
8 # Compute a cycle basis matrix in canonical form
9 $\boldsymbol{C} = (\boldsymbol{C}_\mathsf{T}^\mathsf{T}\ \boldsymbol{C}_\perp^\mathsf{T})^\mathsf{T} \leftarrow$ COMPUTE-CYCLE-BASIS$(\boldsymbol{A}, \boldsymbol{P}_\delta)$
10 # Compute set Γ, containing admissible vectors for $\boldsymbol{\gamma}^\circ$
11 $\hat{\boldsymbol{\gamma}} \leftarrow \frac{1}{2\pi}\boldsymbol{C}\boldsymbol{\delta}$
12 $\boldsymbol{P}_\gamma \leftarrow \frac{1}{4\pi^2}\boldsymbol{C}\boldsymbol{P}_\delta\boldsymbol{C}^\mathsf{T}$
13 $\Gamma \leftarrow$ INTEGER-SCREENING$(\hat{\boldsymbol{\gamma}}, \boldsymbol{P}_\gamma, \alpha)$
14 **for** $\boldsymbol{\gamma}$ **in** Γ:
15 $\quad \boldsymbol{x}^{\star|\gamma} = (\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\Big(\boldsymbol{\delta} - 2\pi\begin{pmatrix}\boldsymbol{0}_n \\ \boldsymbol{C}_\perp^{-1}\end{pmatrix}\boldsymbol{\gamma}\Big)$
16 $\quad \boldsymbol{\theta}^{\star|\gamma} = \langle\, \boldsymbol{x}^{\star|\gamma} \,\rangle_{2\pi}$
17 **end**
18 **return** $\Theta = \{\boldsymbol{\theta}^{\star|\gamma}\}$.

External functions:
- COMPUTE-CYCLE-BASIS$(\boldsymbol{A}, \boldsymbol{P}_\delta)$ computes a cycle basis for the graph, possibly informed by the covariance matrix $\boldsymbol{P}_\delta$.

The first step (line 9) is the computation of a cycle basis of the graph. Any cycle basis will do, but Section VI-C tells us that the choice of the cycle basis can be improved if informed by the covariance $\boldsymbol{P}_\delta$, and the best choice is the minimum cycle basis matrix of the graph. Without loss of generality, we assume that the rows of the cycle basis are ordered according to $(\boldsymbol{C}_\mathsf{T}; \boldsymbol{C}_\perp)$, as described in (1).

The next step (lines 11–13) consists in the computation of a set Γ which contains $\boldsymbol{\gamma}^\circ$ with confidence α using the INTEGER-SCREENING algorithm, described in Section VI-B.

The "for" loop in line 14 computes, for each integer vector $\boldsymbol{\gamma} \in \Gamma$, the corresponding real-valued estimate $\boldsymbol{x}^{\star|\gamma}$ (line 15), and obtain the wrapped estimate $\boldsymbol{\theta}^{\star|\gamma}$ by applying the modulus to $\boldsymbol{x}^{\star|\gamma}$ (line 16). The collection of wrapped estimates is then returned in the set Θ.

In the rest of this section we show that the MOLE2D algorithm solves the two limitations of the maximum likelihood estimator of Section IV (computation efficiency and need of an assessment for the resulting estimate). In particular, in Section VII-A we show that one of the orientation hypotheses returned by the algorithm is "close" to $\boldsymbol{\theta}^\circ$ (with desired probability); then, in Section VII-B we show that the algorithm includes only worst-case polynomial operations.

A. Assessment of the estimator

This section is aimed at evaluating the quality of our estimator. More precisely, we want to assess how the set of estimates $\Theta = \{\boldsymbol{\theta}^{\star|\gamma}\}$, which is the output of Algorithm 2, relates with $\boldsymbol{\theta}^\circ$. The assessment of the estimator is given in the following proposition.

Proposition 20. *Consider the set of estimators Θ returned by Algorithm 2. Then, with probability no smaller than α, one*

of the estimators in Θ, say $\boldsymbol{\theta}^{\star|\gamma^j} \in \Theta$, satisfies the following statements:

1) *the real-valued estimator $\boldsymbol{x}^{\star|\gamma^j}$ is Normally distributed with mean $\boldsymbol{\theta}^\circ + 2\pi\boldsymbol{p}$ and covariance matrix $(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}$;*
2) *the wrapped estimator $\boldsymbol{\theta}^{\star|\gamma^j} = \langle \boldsymbol{x}^{\star|\gamma^j} \rangle_{2\pi}$ is distributed according to a wrapped Gaussian with mean $\boldsymbol{\theta}^\circ$ and covariance matrix $(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}$.*

Proof: The statements follow from Lemma 16 and Proposition 18. Lemma 16 assures that the event $\boldsymbol{\gamma}^\circ \in \Gamma$ is the same as the event "at least one real-valued (respectively, wrapped) orientation estimate is distributed according to a Gaussian (respectively, wrapped Gaussian)", and Proposition 18 assures that such event happens with probability α. ∎

A straightforward consequence of Proposition 20 is that, in the case $|\Gamma| = 1$, the single estimate contained in Θ is distributed according to a wrapped Gaussian around $\boldsymbol{\theta}^\circ$. Therefore, in the case of $|\Gamma| = 1$, we can draw conclusions that are peculiar of linear estimators and are very rare in nonlinear estimation problems (e.g., Normality). In the experimental section we will show that most of the problem instances that constitute a benchmark for state-of-the-art approaches to SLAM satisfy the condition $|\Gamma| = 1$. Therefore, in common problem instances, our multi-hypothesis estimator returns a single *guaranteed* orientation estimate.

B. Complexity

This section helps assessing the worst-case performance of Algorithm 2. The complexity of computing $\boldsymbol{C}$ (line 9) heavily depends on the choice of the cycle basis. In the experimental section we consider four possible cycle bases, listed here from computationally cheap to expensive:

$\mathsf{FCB_o}$: This is the *fundamental cycle basis* built from the *odometric spanning tree*. Call $\mathsf{T_o}$ the odometric spanning tree, which is also a spanning path for the graph. Each cycle of $\mathsf{FCB}(\mathsf{T_o})$ comprises a chord in the graph with respect to $\mathsf{T_o}$, say (i, j), and the unique path in $\mathsf{T_o}$ from node i to node j. The construction of $\mathsf{FCB_o}$ implies a complexity $(n\,\ell)$: the odometric spanning path can be considered a given of the problem and the complexity reduces to fill in the matrix $\boldsymbol{C} \in \mathbb{R}^{\ell \times m}$, which has at most $n + 1$ nonzero elements in each row.

$\mathsf{FCB_m}$: This is the *fundamental cycle basis* built from the *minimum uncertainty spanning tree*. The fundamental cycle basis built from the minimum spanning tree requires the computation of the minimum spanning tree, which amounts to $(m + n)$, and the construction of the matrix $\boldsymbol{C}$ $((n\,m))$, therefore the overall cost is $(n\,m)$.

$\mathsf{MCB_a}$: A $(2\nu - 1)$-*approximation* of the minimum cycle basis is computed using the algorithm proposed by Kavitha *et al.* [56], which implies a complexity $(n^{3+\frac{2}{\nu}})$.

MCB: The *minimum uncertainty cycle basis*, is computed using the method by Mehlhorn and Michail [57], which implies a complexity (m^3).

The computation of the estimator $\hat{\boldsymbol{\gamma}}$ (line 11) requires at most $\ell\,m$ operations, while the covariance matrix requires $\ell^2 m$ operations (exploiting the fact that $\boldsymbol{P}_\delta$ is diagonal). For the INTEGER-SCREENING algorithm (line 13), the worst case is when only one index is added to the set of uniquely determined elements at each iteration. Conditioning is an operation that has cubic operation for general matrices ($O(\ell^3)$). Therefore, in the worst case, the complexity is $\mathcal{O}(\ell^4)$ (the algorithm performs ℓ conditioning).

Finally, to obtain Θ one has to compute $\boldsymbol{x}^{\star|\boldsymbol{\gamma}}$ and $\boldsymbol{\theta}^{\star|\boldsymbol{\gamma}}$ for each $\boldsymbol{\gamma} \in \Gamma$. The expression of $\boldsymbol{x}^{\star|\boldsymbol{\gamma}}$ is given in (34) and contains the two matrix inverses $\boldsymbol{C}_\perp^{-1}$ and $(\boldsymbol{A}\boldsymbol{P}_\delta^{-1}\boldsymbol{A}^\mathsf{T})^{-1}$; in practice, one would not perform these matrix inversions, but would rather solve two linear systems, with overall complexity (m^3) (recall that $\ell < m$ and $n \leq m$). The complexity of computing $\boldsymbol{\theta}^{\star|\boldsymbol{\gamma}}$ from $\boldsymbol{x}^{\star|\boldsymbol{\gamma}}$ (i.e., applying the modulus operation) is (n).

Assuming that the cardinality of Γ does not grow with the problem dimension, the worst-case complexity of Algorithm 2 amounts to (m^4). The cardinality of Γ depends on the size of the loops and on the measurement uncertainty rather than on problem dimension: we already observed in the proof of Proposition 19 that the diagonal elements of $\boldsymbol{P}_\delta$ (that determine Γ) are essentially the sum of the variances of the measurements along each cycle.

The worst-case complexity is a poor indicator of the actual complexity of algorithm, for two main reasons. First, the matrices involved in the various steps of the algorithm are sparse, therefore the computation of $\boldsymbol{\theta}^{\star|\boldsymbol{\gamma}}$ (line 14) has a complexity that is far below the upper-bound. Second, if we are careful about the choice of the cycle basis matrix from which $\boldsymbol{P}_\gamma$ is computed, the INTEGER-SCREENING algorithm is able to compute a small set Γ in few iterations, therefore the average complexity is essentially that of doing one conditioning.

VIII. Experimental evaluation

This section presents an experimental analysis of the proposed approach and its application to pose graph optimization.

Section VIII-A discusses the performance of MOLE2D algorithm in the problem of orientation estimation. The objective is to evaluate how important is the choice of the cycle basis matrix in practice, what is the cardinality of the set of candidate vectors Γ in real applications (recall that $|\Gamma| = |\Theta|$), and how fast is the algorithm on common problem instances. Section VIII-B discusses the use of the orientation estimate produced by MOLE2D as the initial guess for g2o.

The experiments used three standard datasets:

INTEL: This dataset, acquired at the *Intel Research Lab* in Seattle, includes odometry and range-finder data. Relative pose constraints are derived from scan matching. Data processing details are given in previous work [25].

MITb: This dataset was acquired at the MIT *Killian Court*. Data processing details are given in previous work [40].

M3500: This simulated dataset, also known as *Manhattan world*, was created by Olson *et al.* [11].

To test MOLE2D in more challenging scenarios, we obtained other datasets by adding extra Gaussian noise (with standard deviation σ) to the M3500 orientation measurements. These new datasets are called M3500a ($\sigma = 0.1\,\mathrm{rad}$), M3500b ($\sigma = 0.2\,\mathrm{rad}$), and M3500c ($\sigma = 0.3\,\mathrm{rad}$).

TABLE III
PERFORMANCE OF INTEGER-SCREENING

	cycle basis	K		u (%)	d (%)	$\lvert\Gamma\rvert$
INTEL	$\mathsf{FCB_o}$	1		100.00	n/a	1
	$\mathsf{FCB_m}$	1		100.00	n/a	1
	$\mathsf{MCB_a}$	1		100.00	n/a	1
	MCB	1		100.00	n/a	1
MIT	$\mathsf{FCB_o}$	2	iter. 1	(80.00)	(25.78)	(16)
			iter. 2	20.00	n/a	1
	$\mathsf{FCB_m}$	1		100.00	n/a	1
	$\mathsf{MCB_a}$	1		100.00	n/a	1
	MCB	1		100.00	n/a	1
M3500	$\mathsf{FCB_o}$	5	iter. 1	(52.92)	(1.69)	($>10^{100}$)
			iter. 2	(21.54)	(15.61)	($>10^{100}$)
			iter. 3	(20.06)	(100.00)	($>10^{50}$)
			iter. 4	(5.12)	(100.00)	(972)
			iter. 5	0.36	n/a	1
	$\mathsf{FCB_m}$	2	iter. 1	(98.62)	(2.41)	($>10^{9}$)
			iter. 2	1.38	n/a	1
	$\mathsf{MCB_a}$	2	iter. 1	(99.95)	(0.48)	(3)
			iter. 2	0.05	n/a	1
	MCB	2	iter. 1	(99.95)	(0.44)	(3)
			iter. 2	0.05	n/a	1
M3500a	$\mathsf{FCB_o}$	6		–	–	$>10^{30}$
	$\mathsf{FCB_m}$	3		–	–	8
	$\mathsf{MCB_a}$	2		–	–	1
	MCB	2		–	–	1
M3500b	$\mathsf{FCB_o}$	29		–	–	$>10^{40}$
	$\mathsf{FCB_m}$	4		–	–	27
	$\mathsf{MCB_a}$	3		–	–	3
	MCB	3		–	–	3
M3500c	$\mathsf{FCB_o}$	9		–	–	$>10^{100}$
	$\mathsf{FCB_m}$	7		–	–	$>10^{4}$
	$\mathsf{MCB_a}$	4		–	–	16
	MCB	4		–	–	16

This table reports for each choice of the cycle basis matrix:

1) the number of iterations K performed in the INTEGER-SCREENING, reporting details of each iteration when significative;
2) the percentage of elements that are uniquely determined at k-th iteration, i.e., $u \doteq |\mathcal{U}^{(k)}|/\ell$ (in percentage);
3) the *density* d of the matrix to be inverted to compute the conditional Gaussian probability in line 38 of INTEGER-SCREENING. The *density* is defined as the number of non-zero elements in the matrix over the total number of elements (in percentage);
4) the number of admissible vectors $\Gamma^{(k)}$ at iteration k, and the cardinality of the resulting set Γ.

The symbol "n/a" denotes that there is no matrix to invert (the algorithm exits the main loop because all elements of γ° have been determined). The statistics of the intermediate iterations of the algorithm are reported in parenthesis. In the cells with "−" we omitted the details of the iterations for brevity.

A. *Effect of different cycle bases on orientation estimation and practical computational cost*

Here we only consider the orientation measurements in the pose graph and the corresponding covariance matrix. Regarding the MOLE2D algorithm, we chose a confidence level $\alpha = 0.99$. The test are performed on a desktop computer with processor Intel i7, 3.4 GHz. For the computation of the cycle basis matrices, we used Michail's C++ implementation [58], and we chose $\nu = 2$ for the $(2\nu - 1)$-*approximation* [56]. The rest of the MOLE2D algorithm is instead implemented in Matlab, which makes extremely simple sparse matrix manipulation.

In the scenarios INTEL, MITb, and M3500, the INTEGER-SCREENING algorithm is able to identify a single possible value for $\boldsymbol{\gamma}^\circ$, regardless the choice of the cycle basis matrix (Table III, last column). In the scenarios characterized by extreme noise levels (M3500a–c), the choice of the cycle basis truly matters. If one uses the fundamental cycle basis $\mathsf{FCB_o}$ the size of Γ is too big to be tractable; the explosion of $|\Gamma|$ is partially mitigated by the use of $\mathsf{FCB_m}$, that, however, fails to produce a reasonably small number of vectors in Γ in the scenario M3500c. Using a minimum cycle basis gives a small cardinality of Γ, respectively, 1, 3, and 16, for the cases M3500a, M3500b, and M3500c, with no observed difference between the exact minimum cycle basis MCB and the approximation $\mathsf{MCB_a}$.

TABLE IV
COMPUTATION TIME FOR MOLE2D (SECONDS)

	phase	Computation of $\hat{\boldsymbol{\gamma}}$ and $\boldsymbol{P}_\gamma$	INTEGER-SCREENING	Computation of Θ from Γ	Total
	lines	11–12	13	14–16	
INTEL	$\mathsf{FCB_o}$	0.07	0.04	≤0.01	0.10
	$\mathsf{FCB_m}$	≤0.01	0.03	≤0.01	0.04
	$\mathsf{MCB_a}$	≤0.01	0.05	≤0.01	0.05
	MCB	≤0.01	0.04	≤0.01	0.04
MIT	$\mathsf{FCB_o}$	≤0.01	0.04	≤0.01	0.04
	$\mathsf{FCB_m}$	≤0.01	0.03	≤0.01	0.03
	$\mathsf{MCB_a}$	≤0.01	0.03	≤0.01	0.03
	MCB	≤0.01	0.03	≤0.01	0.03
M3500	$\mathsf{FCB_o}$	0.72	0.79	≤0.01	1.52
	$\mathsf{FCB_m}$	≤0.01	0.47	≤0.01	0.47
	$\mathsf{MCB_a}$	≤0.01	0.21	≤0.01	0.22
	MCB	≤0.01	0.21	≤0.01	0.22
M3500a	$\mathsf{FCB_o}$	0.72	0.80	(Γ too large to continue)	
	$\mathsf{FCB_m}$	≤0.01	0.36	0.04	0.4
	$\mathsf{MCB_a}$	≤0.01	0.21	≤0.01	0.22
	MCB	≤0.01	0.21	≤0.01	0.22
M3500b	$\mathsf{FCB_o}$	0.71	1.19	(Γ too large to continue)	
	$\mathsf{FCB_m}$	≤0.01	0.51	0.12	0.64
	$\mathsf{MCB_a}$	≤0.01	0.23	0.03	0.26
	MCB	≤0.01	0.23	0.03	0.26
M3500c	$\mathsf{FCB_o}$	0.72	0.72	(Γ too large to continue)	
	$\mathsf{FCB_m}$	≤0.01	0.48	(Γ too large to continue)	
	$\mathsf{MCB_a}$	≤0.01	0.23	0.15	0.38
	MCB	≤0.01	0.23	0.14	0.37

As predicted by Proposition 19, the minimum cycle bases minimize the number of iterations in the INTEGER-SCREENING (Table III, second column). Moreover, the minimum cycle bases are able to determine most of the components of $\boldsymbol{\gamma}^\circ$ (e.g., 95%) in the first iteration (Table III, third column), and they require inverting lower-density matrices (Table III, fourth column). All these elements provide a computational advantage when using the minimum cycle bases in the INTEGER-SCREENING (Table IV, third column). The minimum cycle basis matrices are more sparse, and this also constitutes an advantage in

TABLE V
COMPUTATION TIME FOR CYCLE BASIS MATRICES (SECONDS)

	n	m	$\mathsf{FCB_o}$	$\mathsf{FCB_m}$	$\mathsf{MCB_a}$	MCB
INTEL	1228	1505	≤0.01	≤0.01	0.09	0.20
MITb	808	828	≤0.01	≤0.01	0.01	0.01
M3500	3500	5599	≤0.01	0.30	1.11	1.54

the computation of $\hat{\gamma}$ and $\boldsymbol{P}_\gamma$ (Table IV, second column). Finally, since the minimum cycle bases produce a smaller set of hypotheses Γ, they require solving a smaller number of linear systems for computing Θ from Γ (Table IV, fourth column).

While the minimum cycle bases have better performance in MOLE2D, they are more expensive to compute (Table V). In conclusion, if the noise is moderate the FCB_m offers a good compromise between performance and computational effort. For extreme noise, the approximate minimum cycle basis matrix MCB_a is a choice that assures a similar performance to the MCB while being cheaper to compute.

B. Robustness of MOLE2D*-based pose graph optimization*

This section shows how the use of MOLE2D can improve pose graph optimization, by comparing Toro [6], g2o [5], and g2o bootstrapped with the orientation estimate provided by MOLE2D; this last algorithm is denoted with MOLE2D+g2o.

The objective function in pose graph optimization is referred to as the χ^2 cost (see equation (2) in [15]), and it is a generalization of the cost in Problem 3 to include translation errors. Convergence to a global minimum of the cost function corresponds to small χ^2 costs, while convergence to local minima leads to larger χ^2 values.

In MOLE2D+g2o we use the MOLE2D algorithm to compute nodes' orientations from relative orientation measurements, and then we substitute this estimate as a first guess of the orientations for g2o; the translation guess is the one from odometry. Following the recommendation of the previous section, we used the approximate minimum cycle basis matrix within the MOLE2D algorithm. If MOLE2D returns more than one hypothesis, we run MOLE2D+g2o for each possible initial guess and choose the one that achieves the smallest cost.

The results are summarized in Figure 2 and 3. Figure 2 shows a graphical representation of the estimated pose graph for one run in each scenario with the corresponding χ^2 value. Figure 3 shows the sample distribution of the χ^2 cost for the M3500 data for 100 realizations of the orientation noise.

The results confirm what is already known about Toro and g2o: for low noise g2o converges to a smaller χ^2 cost than Toro, since Toro's approximations prevent reaching the minimum. This happens in the "easy" scenarios (e.g. INTEL or M3500 in Figure 2) and in several runs with $\sigma = 0.05\,\mathrm{rad}$ (first column in Figure 3). In scenarios with moderate noise, Toro has better convergence properties: generally, gradient methods are known to have a larger basin of convergence [6]. Both methods fail for larger noise as they get stuck in local minima (e.g., MITb, M3500a–c in Figure 2). Local minima correspond to incorrect wraparounds in long loops (Figure 2m).

The results show that combining MOLE2D with MOLE2D+g2o gives a method that is both robust and precise. The third column in Figure 2 shows that this bootstrapping greatly improves the robustness of the iterative solver. In all cases the combination of MOLE2D and g2o attains the smallest observed χ^2 value, and, visually, qualitatively more correct maps. This finding is confirmed by the statistics in Figure 3 where in all cases MOLE2D+g2o is able to reach the smallest χ^2 cost. The median

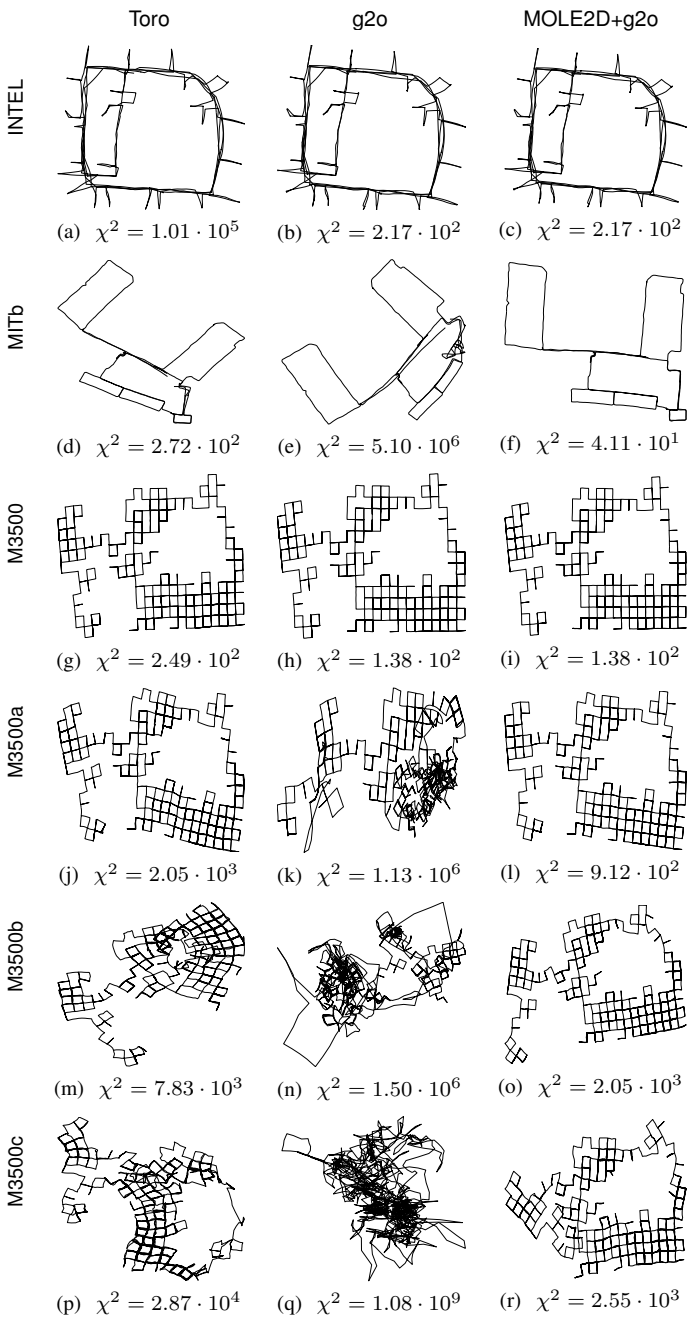


Fig. 2. Estimated pose graphs and corresponding χ^2 values. The first column reports the results obtained from Toro. The second shows the results obtained from g2o. The third column reports the results obtained by bootstrapping g2o with the orientation estimate of MOLE2D (MOLE2D+g2o).

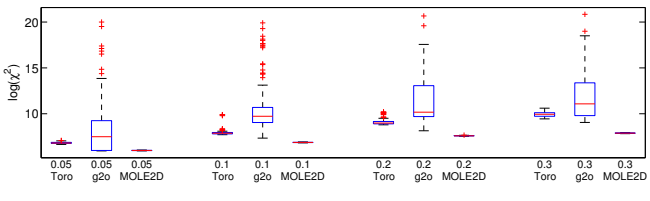


Fig. 3. Distribution of the χ^2 cost for 100 trials and different levels of noise ($\sigma \in \{0.05, 0.1, 0.2, 0.3\}$) in the M3500 dataset for Toro, g2o and MOLE2D+g2o. g2o was set to perform 20 iterations. The plot is drawn using Matlab's boxplot. The red line is the median; the blue box margins are the 25th and 75th percentiles $q_{0.25}$ and $q_{0.75}$; the black whiskers are at $q_{0.25} - w$ and $q_{0.75} + w$ where $w = 1.5(q_{0.75} - q_{0.25})$. All points outside of this range are considered outliers and plotted individually with red crosses.

number of orientation hypotheses reported by MOLE2D was $|\Gamma|= 1$ for $\sigma = 0.05\,\text{rad}$ and $\sigma = 0.1\,\text{rad}$, while it was $|\Gamma|= 3$ for $\sigma = 0.2\,\text{rad}$ and $|\Gamma|= 48$ for $\sigma = 0.3\,\text{rad}$.

IX. Conclusions

This paper considered the problem of estimating the orientations of nodes in a pose graph from relative orientation measurements. The reformulation of maximum likelihood orientation estimation in terms of quadratic integer programming allows us concluding that the maximum likelihood estimate is almost surely unique, and gives us a way to compute a global solution. Starting from this observation we devised a multi-hypothesis estimator that enables efficient computation and has guaranteed performance (at least one of the computed estimates is guaranteed to be "close" to the actual orientation of the nodes). We elucidated on the theoretical derivation with numerical experiments on real and simulated data. Finally, we showed that the proposed approach can be used to bootstrap state-of-the-art techniques for pose graph optimization and allows a remarkable boost in their performance, extending their applicability. Future work includes the analysis of the estimation problem in a 3D setup, which is nontrivial because $\mathrm{SO}(3)$ is not Abelian. A second line of research consists in deriving probabilistic guarantees on the *pose* estimate (the results of this paper only guarantee the quality of the *orientation* estimate).

Appendix

Lemma 21 (Orthogonal complements [59])**.** *For a connected graph $\mathcal{G}$, the transpose of the cycle basis matrix $\boldsymbol{C}^\mathsf{T}$ is an orthogonal complement of the transpose of the reduced incidence matrix $\boldsymbol{A}^\mathsf{T}$, in the sense that: 1) $(\boldsymbol{A}^\mathsf{T} \;\; \boldsymbol{C}^\mathsf{T})$ is a square matrix of full rank; and 2) $\boldsymbol{C}\boldsymbol{A}^\mathsf{T} = \boldsymbol{0}_{\ell\times n}$.*

Lemma 22. *Given a cycle basis matrix $\boldsymbol{C}$ and a reduced incidence matrix $\boldsymbol{A}$ of a connected graph $\mathcal{G}$, for any symmetric positive definite matrix $\boldsymbol{P}$, it holds that*

$$\boldsymbol{P}^{-1}\boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{P}^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}^{-1} + \boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^\mathsf{T})^{-1}\boldsymbol{C} = \boldsymbol{P}^{-1}.$$

Proof: Because $\boldsymbol{P}$ is symmetric and positive definite, there exists two symmetric and positive definite matrices $\boldsymbol{N}$ and $\boldsymbol{M}$ such that $\boldsymbol{M}^2 = \boldsymbol{P}$, $\boldsymbol{N}^2 = \boldsymbol{P}^{-1}$, and $\boldsymbol{N} = \boldsymbol{M}^{-1}$.

Following Meyer [60, equation (5.13.3)], the *orthogonal projector* of $\boldsymbol{N}\boldsymbol{A}^\mathsf{T}$ is $\boldsymbol{N}\boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{N}\boldsymbol{N}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{N}$ and the orthogonal projector of $\boldsymbol{M}\boldsymbol{C}^\mathsf{T}$ is $\boldsymbol{M}\boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{M}\boldsymbol{M}\boldsymbol{C}^\mathsf{T})^{-1}\boldsymbol{C}\boldsymbol{M}$. Because $\boldsymbol{N}$ and $\boldsymbol{M}$ are full rank and $\boldsymbol{C}^\mathsf{T}$ is an orthogonal complement of $\boldsymbol{A}^\mathsf{T}$ (Lemma 21), also $\boldsymbol{N}\boldsymbol{A}^\mathsf{T}$ is an orthogonal complement of $\boldsymbol{M}\boldsymbol{C}^\mathsf{T}$. Meyer [60, equation (5.13.6)] describes a relation between the projectors of two matrices that are orthogonal complements. For our matrices, the relation is

$$\boldsymbol{N}\boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{N}\boldsymbol{N}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{N} + \boldsymbol{M}\boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{M}\boldsymbol{M}\boldsymbol{C}^\mathsf{T})^{-1}\boldsymbol{C}\boldsymbol{M} = \mathbf{I}_m,$$

where $\mathbf{I}_m$ is the identity of size m. Pre-and post-multiplying by $\boldsymbol{N}$ and recalling that $\boldsymbol{M}^2 = \boldsymbol{P}$ and $\boldsymbol{N}^2 = \boldsymbol{P}^{-1}$ we get

$$\boldsymbol{P}^{-1}\boldsymbol{A}^\mathsf{T}(\boldsymbol{A}\boldsymbol{P}^{-1}\boldsymbol{A}^\mathsf{T})^{-1}\boldsymbol{A}\boldsymbol{P}^{-1} + \boldsymbol{N}\boldsymbol{M}\boldsymbol{C}^\mathsf{T}(\boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^\mathsf{T})^{-1}\boldsymbol{C}\boldsymbol{M}\boldsymbol{N} = \boldsymbol{P}^{-1}.$$

The result follows by substituting $\boldsymbol{N} = \boldsymbol{M}^{-1}$. ∎

Lemma 23 (Multiple confidence intervals)**.** *Let $\boldsymbol{x} \in \mathbb{R}^n$ be a Normally distributed random variable with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{P}$. Given the confidence intervals*

$$\mathcal{I}_i = \left[\mu_i - \sqrt{\boldsymbol{P}_{ii}\chi^2_{1,\eta}}\,,\; \mu_i + \sqrt{\boldsymbol{P}_{ii}\chi^2_{1,\eta}}\right], \quad i = \{1,\ldots,n\},$$

then $\mathbb{P}\left(x_1 \in \mathcal{I}_1 \wedge \ldots \wedge x_n \in \mathcal{I}_n\right) \geq \eta^n$.

Proof: The lemma can be seen as a direct consequence of Theorem 1 in [61] (see also [62]). ∎

References

[1] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, pp. 333–349, 1997.

[2] S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *Int. J. Robot. Res.*, vol. 25, pp. 403–429, 2006.

[3] P. Barooah and J. Hespanha, "Estimation on graphs from relative measurements," *Control Systems Magazine*, vol. 27, no. 4, pp. 57–74, 2007.

[4] H. Wang, G. Hu, S. Huang, and G. Dissanayake, "On the Structure of Nonlinearities in Pose Graph SLAM," in *Robotics: Science and Systems*, 2012.

[5] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *Int. Conf. on Robotics and Automation*, pp. 3607–3613, 2011.

[6] G. Grisetti, C. Stachniss, and W. Burgard, "Non-linear constraint network optimization for efficient map learning," *IEEE Trans. on Intelligent Transportation Systems*, vol. 10, no. 3, pp. 428–439, 2009.

[7] J. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *Proc. of the IEEE Int. Symp. CIRA*, pp. 318–325, 1999.

[8] T. Duckett, S. Marsland, and J. Shapiro, "Fast, on-line learning of globally consistent maps," *Autonomous Robots*, vol. 12, no. 3, pp. 287–300, 2002.

[9] K. Konolige, "Large-scale map-making," in *Proc. of the AAAI National Conf. on Artificial Intelligence*, pp. 457–463, 2004.

[10] U. Frese, P. Larsson, and T. Duckett, "A multilevel relaxation algorithm for simultaneous localization and mapping," *IEEE Trans. on Robotics*, vol. 21, no. 2, pp. 196–207, 2005.

[11] E. Olson, J. Leonard, and S. Teller, "Fast iterative alignment of pose graphs with poor estimates," in *Int. Conf. on Robotics and Automation*, pp. 2262–2269, 2006.

[12] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM:incremental smoothing and mapping," *IEEE Trans. on Robotics*, vol. 24, no. 6, pp. 1365–1378, 2008.

[13] 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 *Int. Conf. on Robotics and Automation*, pp. 3281–3288, 2011.

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

[15] G. Grisetti, R. Kuemmerle, C. Stachniss, U. Frese, and C. Hertzberg, "Hierarchical optimization on manifolds for online 2D and 3D mapping," in *Int. Conf. on Robotics and Automation*, pp. 273–278, 2010.

[16] D. L. Rizzini, "A closed-form constraint networks solver for maximum likelihood mapping," in *European Conf. on Mobile Robots*, pp. 223–228, 2010.

[17] G. Dubbelman, I. Esteban, and K. Schutte, "Efficient trajectory bending with applications to loop closure," in *Int. Conf. on Intelligent Robots and Systems*, pp. 1–7, 2010.

[18] G. Dubbelman, P. Hansen, B. Browning, and M. Dias, "Orientation only loop-closing with closed-form trajectory bending," in *Int. Conf. on Robotics and Automation*, pp. 815–821, 2012.

[19] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," in *Robotics: Science and Systems*, 2012.

[20] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose graph slam," in *Int. Conf. on Intelligent Robots and Systems*, pp. 1879–1884, 2012.

[21] N. Sünderhauf and P. Protzel, "Towards a robust back-end for pose graph slam," in *Int. Conf. on Robotics and Automation*, pp. 1254–1261, 2012.

[22] D. Rosen, M. Kaess, and J. Leonard, "An incremental trust-region method for robust online sparse least-squares estimation," in *Int. Conf. on Robotics and Automation*, pp. 1262–1269, 2012.

[23] F. Dellaert and A. Stroupe, "Linear 2D localization and mapping for single and multiple robots," in *Int. Conf. on Robotics and Automation*, pp. 688–694, 2002.

[24] M. Bosse and R. Zlot, "Keypoint design and evaluation for place recognition in 2d lidar maps," *Robotics and Autonomous System J.*, vol. 57, no. 12, pp. 1211–1224, 2009.

[25] L. Carlone, R. Aragues, J. Castellanos, and B. Bona, "A linear approximation for graph-based simultaneous localization and mapping," in *Robotics: Science and Systems*, 2011.

[26] J. Knuth and P. Barooah, "Error growth in position estimation from noisy relative pose measurements," *Robotics and Autonomous System J.*, vol. 61, no. 3, pp. 229–244, 2013.

[27] S. Huang, Y. Lai, U. Frese, and G. Dissanayake, "How far is SLAM from a linear least squares problem?," in *Int. Conf. on Intelligent Robots and Systems*, pp. 3011–3016, 2010.

[28] J. Thunberg, E. Montijano, and X. Hu, "Distributed attitude synchronization control," in *IEEE Conf. on Decision and Control*, pp. 1962–1967, 2011.

[29] R. Tron and R. Vidal, "Distributed image-based 3-D localization of camera sensor networks," in *IEEE Conf. on Decision and Control*, pp. 901–908, 2009.

[30] P. Barooah, "Estimation and control with relative measurements: Algorithms and scaling laws," *Ph.D. Thesis, Center for Control, Dynamical-systems, and Computation, University of California Santa Barbara*, 2007.

[31] J. Knuth and P. Barooah, "Distributed collaborative 3d pose estimation of robots from heterogeneous relative measurements: an optimization on manifold approach," *submitted to Robotica, preprint available online: http://plaza.ufl.edu/knuth/Publications/*, 2013.

[32] R. Stanfield, "Statistical theory of DF finding," *Journal of IEE*, vol. 94, no. 5, pp. 762–770, 1947.

[33] W. Foy, "Position-location solutions by Taylor-series estimation," in *IEEE Transaction on Aerospace and Electronic Systems AES-12 (2)*, pp. 187–194, 1976.

[34] G. Mao, B. Fidan, and B. Anderson, "Wireless sensor network localization techniques," *Computer Networks*, vol. 51, no. 10, pp. 2529–2553, 2007.

[35] G. Piovan, I. Shames, B. Fidan, F. Bullo, and B. Anderson, "On frame and orientation localization for relative sensing networks," *Automatica*, vol. 49, no. 1, pp. 206–213, 2011.

[36] P. Biswas, T. Liang, K. Toh, T. Wang, and Y. Ye, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," *IEEE Transactions on Automation Science and Engineering*, vol. 3, no. 4, pp. 360–371, 2006.

[37] G. Calafiore, L. Carlone, and M. Wei, "A distributed technique for localization of agent formations from relative range measurements," *IEEE Trans. on Systems, Man and Cybernetics, Part A*, vol. 42, no. 5, pp. 1083–4427, 2012.

[38] T. Eren, D. Goldenberg, W. Whiteley, Y. Yang, A. Morse, B. Anderson, and P. Belhumeur, "Rigidity, computation, and randomization in network localization," in *IEEE INFOCOM*, vol. 4, pp. 2673–2684, 2004.

[39] I. Shames, B. Fidan, and B. Anderson, "Minimization of the effect of noisy measurements on localization of multi-agent autonomous formations," *Automatica*, vol. 45, no. 4, pp. 1058–1065, 2009.

[40] L. Carlone, R. Aragues, J. Castellanos, and B. Bona, "A first-order solution to simultaneous localization and mapping with graphical models," in *Int. Conf. on Robotics and Automation*, pp. 1764–1771, 2011.

[41] A. Hassibi and S. Boyd, "Integer parameter estimation in linear models with applications to GPS," *IEEE Trans. On Signal Processing*, vol. 46, no. 11, pp. 2938–2952, 1998.

[42] L. Carlone and A. Censi, "From angular manifolds to the integer lattice: Guaranteed orientation estimation with application to pose graph optimization," ArXiv preprint: http://arxiv.org/abs/1211.3063, 2012.

[43] L. Carlone and A. Censi, "From angular manifolds to the integer lattice: Guaranteed orientation estimation with application to pose graph optimization," supplementary material, http://www.lucacarlone.com/index.php/resources/research/mole2d, 2013.

[44] W. Chen, *Graph Theory and Its Engineering Applications*. Advanced Series in Electrical and Computer Engineering, 1997.

[45] 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.*, vol. 3, no. 4, pp. 199–243, 2009.

[46] H. Luogeng, *Starting with the unit circle*. Springer Verlag, 1981.

[47] G. S. Chirikjian and A. B. Kyatkin, *Engineering applications of noncommutative harmonic analysis*. CRC Press, 2001.

[48] G. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 1: Classical Results and Geometric Methods*. Applied and Numerical Harmonic Analysis, Birkhäuser, 2009.

[49] K. V. Mardia and P. E. Jupp, *Directional Statistics*. John Wiley & Sons, 1999.

[50] G. Nemhauser and L. Wolsey, *Integer and combinatorial optimization*. Wiley New York, 1988.

[51] C. Liebchen, "Finding short integral cycle bases for cyclic timetabling," *Lecture Notes in Computer Science*, vol. 2832, pp. 715–726, 2003.

[52] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, 2003.

[53] H. Lenstra, "Integer programming with a fixed number of variables," *Math. Oper. Res.*, vol. 8, pp. 538–548, 1983.

[54] S. Jazaeri, A. Amiri-Simkooei, and M. Sharifi, "Fast integer least-squares estimation for GNSS high-dimensional ambiguity resolution using lattice theory," *Journal of Geodesy*, vol. 86, no. 2, pp. 123–136, 2012.

[55] X. Chang and G. Golub, "Solving ellipsoid-constrained integer least squares problems," *SIAM J. Matrix Analysis Applications*, pp. 1071–1089, 2009.

[56] T. Kavitha, K. Mehlhorn, and D. Michail, "New approximation algorithms for minimum cycle bases of graphs," *Lecture Notes in Computer Science*, vol. 4393, pp. 512–523, 2007.

[57] K. Mehlhorn and D. Michail, "Minimum cycle bases: Faster and simpler," *ACM Transactions on Algorithms*, vol. 6, no. 1, pp. 1–13, 2009.

[58] D. Michail, "Minimum cycle basis library." http://www.mpi-inf.mpg.de/~michail/subpages/mcb/doxygen/.

[59] W. Russell, D. Klein, and J. Hespanha, "Optimal estimation on the graph cycle space," in *American Control Conf.*, pp. 1918–1924, 2010.

[60] C. Meyer, *Matrix Analysis and Applied Linear Algebra*. SIAM, 2000.

[61] Z. Sidak, "Rectangular confidence regions for the means of multivariate Normal distributions," *Journal of the American Statistical Association*, vol. 62, no. 318, pp. 626–633, 1967.

[62] O. Dunn, "Estimation of the means of dependent variables," *Annals of Mathematical Statistics*, vol. 29, no. 4, pp. 1095–1111, 1958.