2019

Collaborative Perception From Data Association To Localization

Spyridon Leonardos
University of Pennsylvania, s.leon.10@hotmail.com

Collaborative Perception From Data Association To Localization

Abstract

During the last decade, visual sensors have become ubiquitous. One or more cameras can be found in devices ranging from smartphones to unmanned aerial vehicles and autonomous cars. During the same time, we have witnessed the emergence of large scale networks ranging from sensor networks to robotic swarms.

Assume multiple visual sensors perceive the same scene from different viewpoints. In order to achieve consistent perception, the problem of correspondences between observed features must be first solved. Then, it is often necessary to perform distributed localization, i.e. to estimate the pose of each agent with respect to a global reference frame. Having everything set in the same coordinate system and everything having the same meaning for all agents, operation of the agents and interpretation of the jointly observed scene become possible.

The questions we address in this thesis are the following: first, can a group of visual sensors agree on what they see, in a decentralized fashion? This is the problem of collaborative data association. Then, based on what they see, can the visual sensors agree on where they are, in a decentralized fashion as well? This is the problem of cooperative localization.

The contributions of this work are five-fold. We are the first to address the problem of consistent multiway matching in a decentralized setting. Secondly, we propose an efficient decentralized dynamical systems approach for computing any number of smallest eigenvalues and the associated eigenvectors of a weighted graph with global convergence guarantees with direct applications in group synchronization problems, e.g. permutations or rotations synchronization. Thirdly, we propose a state-of-the art framework for decentralized collaborative localization for mobile agents under the presence of unknown cross-correlations by solving a minimax optimization problem to account for the missing information. Fourthly, we are the first to present an approach to the 3-D rotation localization of a camera sensor network from relative

bearing measurements. Lastly, we focus on the case of a group of three visual sensors. We propose a novel Riemannian geometric representation of the trifocal tensor which relates projections of points and lines in three overlapping views. The aforementioned representation enables the use of the state-of-the-art optimization methods on Riemannian manifolds and the use of robust averaging techniques for estimating the trifocal tensor.

Degree Type
Dissertation

Degree Name
Doctor of Philosophy (PhD)

Graduate Group
Computer and Information Science

First Advisor
Konstantinos Daniilidis

Keywords
Collaborative perception, Data association, Distributed localization, Multi-agent systems

Subject Categories
Computer Sciences

COLLABORATIVE PERCEPTION
FROM DATA ASSOCIATION TO LOCALIZATION

Spyridon Leonardos

A DISSERTATION

in

Computer and Information Science

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2019

Supervisor of Dissertation

Kostas Daniilidis, Ruth Yalom Stone Professor of Computer and Information Science

Graduate Group Chairperson

Rajeev Alur, Zisman Family Professor of Computer and Information Science

Dissertation Committee

Jean H. Gallier, Professor of Computer and Information Science

George J. Pappas, Joseph Moore Professor of Electrical and Systems Engineering

Victor M. Preciado, Assoc. Professor of Electrical and Systems Engineering

Roberto Tron, Assist. Professor of Mechanical Engineering (Boston University)

COLLABORATIVE PERCEPTION
FROM DATA ASSOCIATION TO LOCALIZATION

Dedicated to Vaso

Acknowledgments

First of all, I would like to thank my advisor Dr. Kostas Daniilidis for his guidance throughout my entire doctoral studies. I will be eternally grateful for the opportunity he gave me, for his support during the ups and downs of the Ph.D. and for providing me with the freedom to pursue my ideas.

I am also grateful to Dr. Jean Gallier for serving as the chair of my committee, for introducing me to several fields of mathematics and for all the discussions we have had throughout my Ph.D. I owe a lot to Dr. Roberto Tron for helping me shape my research interests during my first research steps, for our collaboration, for serving in my committee and for proofreading my dissertation. I am grateful to Dr. Victor Preciado for our collaboration and for serving in my committee. I would also like to thank Dr. George Pappas for serving in my committee and for his valuable feedback.

Finally, I would like to thank my wife, Vaso, for continuously supporting and encouraging me during my doctoral studies and my parents for their support.

Abstract

During the last decade, visual sensors have become ubiquitous. One or more cameras can be found in devices ranging from smartphones to unmanned aerial vehicles and autonomous cars. During the same time, we have witnessed the emergence of large scale networks ranging from sensor networks to robotic swarms.

Assume multiple visual sensors perceive the same scene from different viewpoints. In order to achieve consistent perception, the problem of correspondences between observed features must be first solved. Then, it is often necessary to perform distributed localization, i.e. to estimate the pose of each agent with respect to a global reference frame. Having everything set in the same coordinate system and everything having the same meaning for all agents, operation of the agents and interpretation of the jointly observed scene become possible.

The questions we address in this thesis are the following: first, can a group of visual sensors agree on what they see, in a decentralized fashion? This is the problem of collaborative data association. Then, based on what they see, can the visual sensors agree on where they are, in a decentralized fashion as well? This is the problem of cooperative localization.

The contributions of this work are five-fold. We are the first to address the problem of consistent multiway matching in a decentralized setting. Secondly, we propose an efficient decentralized dynamical systems approach for computing any number of smallest eigenvalues and the associated eigenvectors of a weighted graph with global convergence guarantees

with direct applications in group synchronization problems, e.g. permutations or rotations synchronization. Thirdly, we propose a state-of-the art framework for decentralized collaborative localization for mobile agents under the presence of unknown cross-correlations by solving a minimax optimization problem to account for the missing information. Fourthly, we are the first to present an approach to the 3-D rotation localization of a camera sensor network from relative bearing measurements. Lastly, we focus on the case of a group of three visual sensors. We propose a novel Riemannian geometric representation of the trifocal tensor which relates projections of points and lines in three overlapping views. The aforementioned representation enables the use of the state-of-the-art optimization methods on Riemannian manifolds and the use of robust averaging techniques for estimating the trifocal tensor.

Contents

List of Tables

List of Figures

List of Symbols

$\mathbb{R}^n$	Vector space of real n-vectors
$\mathbb{R}^{m\times n}$	Set of $m \times n$ matrices with real entries
$\mathbb{R}_+$	Set of nonnegative real numbers
$\mathbf{1}_n$	The n-vector of all ones
$\mathbf{1}_{m\times n}$	The $m \times n$ matrix of all ones
$\mathbf{0}_{m\times n}$	The $m \times n$ matrix of all zeros
$(X)_{ij}$	The (i, j)-th entry of matrix X
$[X]_{ij}$	The (i, j)-th block of matrix X
$A(\mathcal{G})$	Adjacency matrix of graph $\mathcal{G}$
$\Delta(\mathcal{G})$	Degree matrix of graph $\mathcal{G}$
$\mathcal{L}(\mathcal{G})$	Laplacian matrix of graph $\mathcal{G}$
$\lvert A\rvert$	Cardinality of set A
$\odot$	Elementwise (Hadamard) product
$\otimes$	Kronecker product
$\mathrm{Proj}_{\mathcal{C}}(x)$	Closest point to x in the set $\mathcal{C}$
$(x)_+$	Positive part of a scalar/vector, equal to $\max\{0, x\}$

Chapter 1

Introduction

1.1 Collaborative data association

Estimating correspondences between feature points, regions or objects observed in different images has been a long standing problem in computer vision and robotics with various applications such as structure from motion, image registration, shape analysis and object matching. Most of the efforts in previous works have been dedicated to improving the quality of the correspondences by designing new feature detectors, descriptors, and outlier rejection algorithms in a pairwise setting. However, the problem setting in practice is often multiway if more than one view of a scene or an object is available.

Multiway matching refers to the problem of establishing correspondences among a collection of images from noisy pairwise correspondences. It is a more recent problem compared to two-way matching and it has received increasing amount of attention during the last few years. Multiway matching has been successfully applied to computing consistent pointwise maps among a collection of shapes [93, 64, 54, 53] and to estimating consistent associations among a collection of images between either traditional feature descriptors [101, 23, 156, 148, 147] such as SIFT [81] and shape context features [10] or semantic descriptors [156, 140].

Figure 1.1: The first problem considered in this work: multiway matching with cycle consistency. Cycle consistency is satisfied on the left example and violated on the right example.

A necessary condition for good matching of multiple views is the cycle consistency, meaning that the composition of correspondences along a cycle of views should be equal to the identity. In practice, cycle consistency is not satisfied if pairs of views are matched separately, which is usually the case, due to the presence of outliers. Nguyen *et al.*[93] and Zach *et al.*[150] were the first to propose the use the cycle consistency to identify the correctness of pairwise correspondences.

Later, it was proposed [64, 54, 101] that finding cyclically consistent correspondences from noisy pairwise correspondences can be formulated as quadratic integer programming which in turn, can be relaxed into a generalized Rayleigh quotient problem whose solution is easily obtainable by the leading eigenvectors of the matrix of pairwise correspondences. Although spectral relaxations are easily implementable and come with some theoretical guarantees, in practice, they lack robustness since the total number of features has to be accurately known. This shortcoming was remedied by semidefinite programming relaxations proposed by Huang and Guibas [53] and Chen *et al.*[23]. These works provide theoretical guarantees under certain assumptions on outliers generation but do not scale well. Zhou *et al.*[156] improved the scalability of the semidefinite programming relaxations by reformulating them as rank minimization and developed a more efficient algorithm by dropping the positive-definite constraint on the matrix of pairwise correspondences. More recently, Tron *et al.*[140] proposed a generalization of the Quick Shift algorithm [142] for multiway matching. Maset *et*

al.[85] suggested a practical modification to the spectral relation which improves performance but consistency is no longer satisfied.

All of the aforementioned works address problem in a centralized setting, i.e. all pairwise measurements are available and optimized jointly. To the best of our knowledge, only our work and the concurrent work of [52] address the problem of multiway matching in a decentralized setting. This is the first main contribution of this work.

Computing the eigenvectors associated with a number of extreme eigenvalues of the adjacency matrix of a graph, is crucial in the spectral relaxation solutions of group synchronization problems such as permutations synchronization [101, 75] and rotations synchronization [7, 137]. This motivates us to consider the decentralized estimation of a number of extreme (either smallest or largest) eigenvalues and associated eigenvectors of a, possibly weighted, graph.

Apart from group synchronization problems, the eigenvalues and eigenvectors of matrices representing graphs are crucial in the analysis of a wide variety of networked systems problems. For example, the eigenvalue spectra of a graph are of relevance in connectivity maintenance [29, 152, 151, 149, 114], event-triggered consensus problems [4], decentralized network design [109], and distributed optimization for fast averaging [16, 100].

Common methods to compute the eigenvectors of a symmetric matrix in a centralized setting are (*i*) the Power Method, for one-dimensional invariant subspace computation, as well as (*ii*) the Orthogonal Iteration Method, for computing higher-dimensional invariant subspaces [43]. The Orthogonal Iteration method consists of a power step and an orthonormalization step based on the QR-decomposition. In many applications, the matrix representing the graph structure is not known by a centralized agent; in contrast, the agents in the network have access to the rows and/or columns of the matrix representing their local graph neighborhood. In this decentralized setting, the power step of the orthogonal iteration can be implemented using local information solely. However, the orthonormalization step in the QR

decomposition requires that all the nodes have access to the upper triangular factor of the QR decomposition. Kempe and McSherry [60] were the first to propose a decentralized variant of the Orthogonal Iteration, namely the Decentralized Orthogonal Iteration. Despite being one of the most widely used methods for decentralized eigenvector estimation, the Decentralized Orthogonal Iteration presents a major drawback. At each iteration, Decentralized Orthogonal Iteration requires a very accurate estimation of an aggregate quantity, namely the upper triangular factor of the QR decomposition of the matrix under consideration.

Apart from general decentralized algorithms to compute the eigendecomposition of a matrix, we also find in the literature specialized methods for estimating the algebraic connectivity (i.e. the smallest nontrivial eigenvalue) of the Laplacian matrix of a graph [89], [78], [5]. Furthermore, we also find a method for estimating all the eigenvalues of the Laplacian [37] by building a networked dynamical system whose vibrating frequencies correspond to the eigenvalues of the Laplacian matrix of the network. However, all of these methods are specific for the Laplacian of a graph, whereas the methods of the work herein presented are applicable to any symmetric matrix whose sparsity pattern matches that of a given communication graph.

1.2 Cooperative localization

The next problem we consider is decentralized cooperative state estimation of mobile agents. State estimation is one of the fundamental problems in control theory and robotics. The most common state estimators are undoubtedly the Kalman filter [59], which is optimal for the case of linear systems, and its generalizations for nonlinear systems: the Extended Kalman Filter (EKF) [120] and the Uscented Kalman Filter (UKF) [57]. In multi-agent systems, the task of state estimation takes a collaborative form in the sense that it involves inter-agent measurements and constraints. The paradigm we consider in this work, is decentralized cooperative localization of mobile agents [36, 113] using relative position measurements.

One of the main challenges of decentralized cooperative localization of mobile agents is that the group state estimates become highly correlated as information flows through the network. Ignoring these correlations has grave consequences: estimates become optimistic and result in divergence of the estimator.

The most widely used algorithm for information fusion under the presence of unknown cross-correlations is the Covariance Intersection (CI), introduced by Julier and Uhlmann [58]. In its simplest form, the Covariance Intersection algorithm is designed to fuse two random vectors whose correlation is not known by forming a convex combination of the two estimates in the information space. Covariance Intersection produces estimates that are provably consistent, in the sense that estimated error covariance is an upper bound of the true error covariance and can be generalized to fuse partially observable estimates. Optimality of Covariance Intersection was discussed in [22, 110]. Despite the aforementioned advantages, it has been observed [6, 146, 90, 63] that Covariance Intersection produces estimates that are too conservative, which may decrease the accuracy and convergence speed of the overall estimator. To alleviate the conservativeness of Covariance Intersection several other methods have been recently proposed such as the Inverse Covariance Intersection [95, 94], which is less conservative than Covariance Intersection but consistency is no longer guaranteed in general, and the Ellipsoidal Intersection [118, 117] which computes the largest volume ellipsoid within the intersection of the two given confidence ellipsoids. However, neither of the two aforementioned methods generalize for partially observable estimates. A minimax approach for fusing two, possibly partially observable estimates, of a random variable was proposed in [39], concurrently with the method presented in this work.

Approaches to decentralized cooperative localization can roughly be divided into two categories. Adopting the terminology of [63], the first category includes tightly coupled approaches [91, 113, 129, 77, 103, 63] in the sense that each agent broadcasts its information to the entire team. However, these approaches result in higher computational, memory and communication costs compared to the loosely coupled approaches.

The second category of works for decentralized cooperative localization includes loosely coupled approaches in the sense that only one or both agents involved with a relative measurement update their estimates. As mentioned before, the main difficulty of these approaches arises from the fact that estimates become highly correlated as information flows through the network. Several of these approaches [21, 6] use Covariance Intersection for localization using relative pose measurements and decentralized state estimation respectively. Other approaches [79, 144] use the Split Covariance Intersection method, a variation of Covariance Intersection which exploits partial independence assumptions, for intelligent transportation vehicle localization. Other approaches that fall in this category include [9], in which consistency is enforced in a decentralized manner by maintaining an exponential number of estimates to keep track of the dependencies, and [27] which proposes an approach very similar to CI.

The approach adopted in this work is a loosely coupled approach within an EKF framework. The open problem we address in this work, is how to update the state estimates upon taking relative measurements while on the one hand, being less conservative than Covariance Intersection, and on the other hand, taking the (unknown) correlations into account.

Next, we address the rotation localization problem of a sensor network in a minimal setup, where we assume that pairs of agents can measure a sparse subset of their relative bearings (without relative distance or rotation measurements), possibly in addition to the bearings with respect to a few external common points in the environment (for instance, the agents, in addition to detecting each other in the images they acquire, they might also detect a common object). We do not assume any common shared knowledge among the agents (e.g., a magnetic north or gravity direction), although we assume that agents that can measure each other can also communicate. Finally, we focus only on the 3-D rotation localization problem, since the 2-D version has been already solved, and translations can be retrieved relatively easily once rotations are known (specific references for these facts are reviewed below).

The problem of localizing a set of nodes using bearing-only information has received significant attention in the last decade. Note that the localization problem, in its simplest forms (e.g., if only first order dynamics is considered), is equivalent to the formation control problem [134], so in this section we consider both lines of work.

The simplest version of the problem considers only translations, by assuming that all the nodes have a common rotational reference frame. Since the constraints involved are linear, typical solutions can be mostly formulated and analyzed using graph theory and linear algebra; as a result, it is now well understood that the localizability of the problem can be captured through the notion of rigidity or combinatorial conditions on the graph topology (e.g., see [155, 132, 35] and references therein). Another, more challenging version of the localization problem considers the recovery of the relative rotations between the agents. In this case, the majority of existing works assume that, in addition to relative bearings, the agents measure full relative rotations. We note that, with this assumption, the localization problem can be transformed into a consensus problem on the position of a common reference frame [135], so that solutions to the latter can be used for the former. Even with these considerations, however, the topological obstructions introduced by the space of rotations make the development of almost globally convergent algorithms nontrivial. For instance, some existing solutions ensure only local convergence [127, 97]; to obtain almost global solutions it is necessary to resort to reshaping functions [130], or to consider extrinsic solutions, where the states of the nodes are not forced to be rotations, based on relaxations to singular vectors [71], the convex hull of the space of rotations [86], or QR decompositions [126]. A few papers consider the rotation localization problem in succession or conjunction with the translation counterpart [31, 135, 65]. As already stated, however, all these works assume that the full relative rotations are available. This information can be easily extracted in the 2-D case [106], and in the 2.5-D case where the nodes can measure a common direction (e.g., gravity,[134, 116]), as long as the graph contains at least an undirected spanning tree. The case of 3-D bearing-only (without relative rotations) rotation localization has been considered in [106]; in that paper, however, only small graphs (three or four nodes) were

considered, without providing any concrete algorithm. If we assume the availability of at least five external reference points between pairs of agents, then [135] also implicitly provides a solution to the 3-D bearing-only rotation localization problem by recovering relative rotations through *epipolar geometry* [82].

The last topic we consider in this work is the localization of a group of three visual sensors. Localizing such a sensor network is equivalent to estimating the trifocal tensor. The trifocal tensor relates projection of points and lines in three overlapping views of a rigid scene in the world. The trifocal tensor was first introduced in the context of calibrated geometry to describe relations between projections of lines by Spetsakis and Aloimonos [122] and Weng *et al.*[145]. Later, Hartley [49, 48] generalized the trifocal tensor for the uncalibrated case and Shashua [115] investigated trilinear relations of matched points in three perspective views.

There has been numerous works on minimal parametrizations of the projective trifocal tensor [128, 111, 102, 96, 20, 107]. In most formulations, one of the three cameras local reference frame is chosen as the global reference frame. A symmetric formulation (where every camera has a similar role) was recently proposed by Ponce and Hebert [107] who minimally parametrized the trifocal tensor by providing necessary and sufficient conditions for three visual rays to converge in terms of three epipolar and one or two trifocal constraints. Symmetric trilinear constraints were also introduced in [108].

Symmetric representations for the two view counterpart of the trifocal tensor, the essential matrix, have been introduced in [40, 50, 124] and are based on the singular value decomposition (SVD) of essential matrices. Geometric insights and further properties of symmetric representations for the space of essential matrices endowed with a Riemannian manifold structure were recently presented by Tron and Daniilidis [137]. However, the study of the space of essential matrices as a Riemannian manifold can be traced back to Soatto *et al.*[119] who formulated structure from motion as a filtering problem on the essential manifold. Later, Ma *et al.*[83] proposed a Riemannian Newton algorithm on the essential manifold for the

problem of structure and motion estimation which was later generalized by Vidal *et al.*[143] for multiple views. To the best of our knowledge, analogous representations and manifold structures for the space of calibrated trifocal tensors have not been investigated before.

In this work, we propose a parametrization of the trifocal tensor for calibrated cameras with non-colinear pinholes based on a quotient Riemannian manifold. This parametrization is almost symmetric (we use a preferred camera only for the translations), and is derived from a particular choice of the global reference frame.

1.3 Contributions

In the first main contribution of this work, we propose a distributed optimization method for solving the permutation synchronization problem, which is a specific instance of multiway matching. Then, we generalize our approach for the general case of multiway matching. The proposed optimization methods are based on distributed projected gradient descent with constant step size. We rigorously analyze the convergence properties of the proposed algorithms. We provide experimental evidence supporting that the proposed approaches, albeit decentralized, have performance comparable with the state of the art centralized approaches.

In the second main contribution, we propose a dynamical systems approach for the problem of distributedly estimating any number of smallest eigenvalues and the associated eigenvectors of a weighted graph. The proposed approach is fully decentralized and has global convergence guarantees. In contrast to approaches based on the Orthogonal Iteration, the orthogonality constraints that must be satisfied among the eigenvectors of a symmetric matrix are asymptotically satisfied by the dynamical system herein proposed. Thus, the main computational burden of the Decentralized Orthogonal Iteration, namely, the orthonormalization step, is not present in our approach. We demonstrate the validity of our approach through rigorous theoretical analysis and experimental evaluation.

In the third main contribution, we, first, address the problem of fusing two random vectors with unknown cross-correlations by formulating as a minimax optimization problem. Then, we extend our minimax formulation to linear measurement models and propose a numerical method for computing the optimal estimate. As a direct application, we consider the problem of decentralized cooperative localization for a group of mobile agents. The proposed estimator takes cross-correlations into account while being less conservative than the widely used Covariance Intersection. We demonstrate the superiority of the proposed method compared to Covariance Intersection with numerical examples and simulations.

The fourth contribution of this work consists of providing the first, to the best of our knowledge, solution to the problem of distributed rotation localization of a network from relative bearing measurements. We provide a distributed algorithm, based on distributed Riemannian gradient descent, that can work on any localizable network and we prove stronger localizability results than those provided in [106].

In the last main contribution, we propose a novel parametrization of the trifocal tensor obtained from a quotient Riemannian manifold. We show how it can be used for refining estimates of the tensor from image data through state of the art techniques for optimization on manifolds [1]. In addition, the Riemannian structure provides a notion of distance between trifocal tensors. We show that this distance can be computed efficiently, and that it produces meaningful results in a sample Structure-from-Motion problem.

1.4 Organization of this work

The remainder of this work is organized as follows. Some background material is first reviewed in Chapter 2. Chapter 3 contains the proposed distributed optimization algorithm for solving permutation synchronization. The proposed distributed optimization approach for the general case of multiway matching is the objective of Chapter 4. The proposed dynamical systems approach to distributedly computing the eigenvalues and associated eigen-

vectors of a weighted graph is presented in Chapter 5. Chapter 6 includes our approach to decentralized state estimation. In Chapter 7 we present our approach to distributed rotation localization from bearing measurements. Chapter 8 includes a novel Riemannian manifold representation of the trifocal tensor. Conclusions and potential future directions are presented in Chapter 9.

Chapter 2

Background

2.1 Graph theory

In this section, we review some elementary facts from graph theory. For an in depth analysis, we refer the reader to [42, 87]. An *undirected graph* or simply a *graph* is denoted by the pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, 2, \ldots, n\}$ is the set of vertices and $\mathcal{E} \subseteq [\mathcal{V}]^2$ is the set of edges, where $[\mathcal{V}]^2$ denotes the set of unordered pairs of elements of $\mathcal{V}$. The neighborhood $\mathcal{N}_i$ of the vertex i is the subset of $\mathcal{V}$ defined by

$$\mathcal{N}_i = \{j \in \mathcal{V} \mid \{i, j\} \in \mathcal{E}\}. \tag{2.1}$$

A *path* is a sequence $i_0, i_1, \ldots, i_m$ of distinct vertices such that $\{i_{k-1}, i_k\} \in \mathcal{E}$ for all $k = 1, \ldots, m$. A *cycle of length* n is a sequence $i_0, i_1, \ldots, i_{n-1}, i_n$ of vertices such that $\{i_{k-1}, i_k\} \in \mathcal{E}$ for all $k = 1, \ldots, n$, $i_n = i_0$ and $i_0, i_1, \ldots, i_{n-1}$ are distinct. A graph is *connected* if there is a path between any two vertices. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, its *adjacency matrix* is defined

by

$$(A(\mathcal{G}))_{ij} = \begin{cases} 1, & \text{if } \{i,j\} \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

The *degree matrix* $\Delta(\mathcal{G})$ is the diagonal matrix such that

$$(\Delta(\mathcal{G}))_{ii} = |\mathcal{N}_i| = \sum_{j \in \mathcal{N}_i} (A(\mathcal{G}))_{ij}, \tag{2.3}$$

where $|\cdot|$ denotes the cardinality of a set.

A *directed graph* or *digraph* is denoted by the pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. A *weighted digraph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ is a graph along with a function $w : \mathcal{E} \to \mathbb{R}_+$. The adjacency matrix of a weighted digraph is defined by

$$(A(\mathcal{G}))_{ij} = \begin{cases} w(j,i), & \text{if } (j,i) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

Intuitively, if $(A(\mathcal{G}))_{ij} > 0$ there is information flow from vertex j to vertex i. The neighborhood $\mathcal{N}_i$ of the vertex i is the subset of $\mathcal{V}$ defined by

$$\mathcal{N}_i = \{j \in \mathcal{V} \mid (j,i) \in \mathcal{E}\}. \tag{2.5}$$

The *degree matrix* $\Delta(\mathcal{G})$ is the diagonal matrix that contains the in-degrees on its diagonal, that is

$$(\Delta(\mathcal{G}))_{ii} = d_{\text{in}}(i) = \sum_{j \in \mathcal{N}_i} (A(\mathcal{G}))_{ij}. \tag{2.6}$$

The *maximum degree* of a graph $\mathcal{G}$ is defined by

$$d_{\max}(\mathcal{G}) = \max_{i \in \mathcal{V}} \{d_{\text{in}}(i)\}. \tag{2.7}$$

A *directed path* is a sequence $i_0, i_1, \ldots, i_m$ of distinct vertices such that $(i_{k-1}, i_k) \in \mathcal{E}$ for

all $k = 1, \ldots, m$. A digraph is *strongly connected* if there is a directed path between any two vertices. A digraph is *balanced* if the in-degree $d_{in}(i)$ and the out-degree $d_{\text{out}}(i) = \sum_{(i,j)\in\mathcal{E}}(A(\mathcal{G}))_{ji}$ are equal for all $i \in \mathcal{V}$. A digraph is a *rooted out-branching tree* if it has a vertex to which all other vertices are path connected and does not contain any cycles.

The *graph Laplacian* $\mathcal{L}(\mathcal{G})$ is defined as

$$\mathcal{L}(\mathcal{G}) = \Delta(\mathcal{G}) - A(\mathcal{G}). \tag{2.8}$$

By construction, $\mathcal{L}(\mathcal{G})\mathbf{1} = 0$. A digraph on n vertices contains a rooted-out branching if and only the rank of its Laplacian is $n - 1$. Moreover, $\mathcal{L}(\mathcal{G})$ is positive semidefinite if the graph is undirected. Using Gershgorin discs theorem [51], it can be easily verified that the eigenvalues of the adjacency matrix lie in the interval $[-d_{\max}, d_{\max}]$ and the eigenvalues of the Laplacian lie in the interval $[0, 2d_{\max}]$. The graph $\mathcal{G}$ is connected if and only if

$$\lambda_2(\mathcal{L}(\mathcal{G})) > 0, \tag{2.9}$$

where $\lambda_1(A) \leq \lambda_2(A) \leq \ldots \leq \lambda_n(A)$ denote the eigenvalues of a matrix $A \in \mathbb{R}^{n\times n}$ in ascending order. We also use $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ to denote the smallest and the largest eigenvalues, respectively, of a matrix $A \in \mathbb{R}^{n\times n}$. A lower bound on the second eigenvalue of the graph Laplacian [28],[88] is given by

$$\lambda_2(\mathcal{L}(\mathcal{G})) \geq \frac{4}{n\ \text{diam}(\mathcal{G})}. \tag{2.10}$$

where $\text{diam}(\mathcal{G})$ denotes the diameter of the graph $\mathcal{G}$, defined as the maximum length of the shortest path connecting two vertices in $\mathcal{G}$.

A matrix closely related to the graph Laplacian is the Perron matrix P_ϵ defined by

$$P_\epsilon(\mathcal{G}) = I - \epsilon\mathcal{L}(\mathcal{G}), \tag{2.11}$$

with $0 < \epsilon < 1/d_{\max}(\mathcal{G})$.

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $|\mathcal{V}| = n$, a symmetric matrix $A \in \mathbb{R}^{n\times n}$ is said to be *distributed across* $\mathcal{G}$ if $j \notin \mathcal{N}_i \cup \{i\}$ implies $(A)_{ij} = 0$ for all $i, j \in \{1, 2, \ldots, n\}$. A symmetric block matrix $A \in \mathbb{R}^{m\times m}$ with blocks $[A]_{ij} \in \mathbb{R}^{m_i \times m_j}$ is said to be *distributed across* $\mathcal{G}$ if $j \notin \mathcal{N}_i \cup \{i\}$ implies $[A]_{ij} = \mathbf{0}_{m_i \times m_j}$ for all $i, j \in \{1, 2, \ldots, n\}$. Common examples of matrices distributed across network include the graph Laplacian, the adjacency matrix of a graph and the Perron matrix.

2.2 Stochastic matrices and permutations

In this section, we introduce some notions and notations regarding stochastic matrices and permutations that will be heavily used throughout this work. Stochastic matrices and their properties have been well studied in the area of distributed dynamical systems [56, 98, 13] and in probability theory in the context of Markov chains [8, 13]. A nonnegative matrix is *stochastic* if all its row sums are equal to 1. The spectral radius $\rho(A)$ of a stochastic matrix A is equal to 1 and it is an eigenvalue of A. A nonnegative matrix is *doubly stochastic* if both its row sums and column sums are equal to 1. We denote by $\mathcal{D}_n$ the set of $n \times n$ doubly stochastic matrices, i.e.

$$\mathcal{D}_n = \{X \in \mathbb{R}^{n\times n} : X \geq 0, X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} = \mathbf{1}\}. \tag{2.12}$$

The set of $m \times n$ partial doubly stochastic matrices, denoted by $\mathcal{D}_{m,n}$, is the set of all $m \times n$ stochastic matrices with column sums at most 1, i.e.

$$\mathcal{D}_{m,n} = \{X \in \mathbb{R}^{m\times n} : X \geq 0, X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} \leq \mathbf{1}\}, \tag{2.13}$$

with $m \leq n$. A doubly stochastic matrix is a permutation matrix if its elements are either 0 or 1. The set of $n \times n$ permutation matrices is denoted by $\mathcal{P}_n$ and defined by

$$\mathcal{P}_n = \{X \in \{0,1\}^{n \times n} : X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} = \mathbf{1}\}. \tag{2.14}$$

The set of $n \times n$ permutation matrices is a group under matrix multiplication. The inverse of a permutation matrix is given by its transpose. An $m \times n$ stochastic matrix, with $m \leq n$, is a *partial permutation matrix* if its elements are either 0 or 1. The set of $m \times n$ partial permutation matrices is denoted by $\mathcal{P}_{m,n}$, i.e.

$$\mathcal{P}_{m,n} = \{X \in \{0,1\}^{m \times n} : X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} \leq \mathbf{1}\}. \tag{2.15}$$

Let $[n] \doteq \{1, 2, \ldots, n\}$ for some positive integer n. A mapping $\pi : [n] \to [n]$ is a *permutation* of $[n]$ if it is bijective. The set of all permutations of $[n]$ forms a group under composition, termed the *symmetric group* $\mathfrak{S}_n$. A permutation $\pi \in \mathfrak{S}_n$ is represented by an $n \times n$ permutation matrix $\Pi \in \mathcal{P}_n$ such that

$$(\Pi)_{ij} = \begin{cases} 1, & \text{if } \pi(j) = i, \\ 0, & \text{otherwise,} \end{cases} \tag{2.16}$$

or equivalently,

$$\Pi e_j = e_{\pi(j)}, \tag{2.17}$$

where e_i is the ith canonical basis vector. The simplest choice for a distance on $\mathfrak{S}_n$ is given by

$$d(\pi_1, \pi_2) \doteq n - \operatorname{tr}(\Pi_1^T \Pi_2) = \frac{1}{2}\|\Pi_1 - \Pi_2\|_F^2, \tag{2.18}$$

where Π_1, Π_2 are the matrix representations of permutations π_1 and π_2 respectively. The distance function defined above is simply the number of labels assigned differently by permutations π_1 and π_2. Observe that the distance as defined in (2.18), is invariant to left and

right translations, that is

$$d(\pi_1, \pi_2) = d(\pi_0 \circ \pi_1, \pi_0 \circ \pi_2) = d(\pi_1 \circ \pi_0, \pi_2 \circ \pi_0). \tag{2.19}$$

for all $\pi_0 \in \mathfrak{S}_n$. Finally, a mapping $\pi : [m] \to [n]$, $m \leq n$, is a *partial permutation* if it is injective. A partial permutation is represented by an $m \times n$ partial permutation matrix.

2.3 Consensus algorithms

Reaching agreement or consensus is one of the fundamental and most well studied problems in multi-agent systems. Consensus algorithms, that is algorithms for reaching agreement usually in a decentralized fashion, have been extensively studied in the control community [13, 56, 98]. In its simplest form, a consensus algorithm is a decentralized protocol in which a group of agents, modeled as vertices of a graph, try to reach agreement by communicating only with a small subset of the group. This small subset is usually defined as the set of closest agents in a Euclidean distance sense.

Formally, let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote the underlying communication graph. Let $x_i(t) \in \mathbb{R}$ denote the state of agent i at time time t. Then, the simplest discrete time consensus protocol is given by

$$x_i(t+1) = \sum_{j \in \mathcal{N}_i \cup \{i\}} w_{ij} x_j(t), \tag{2.20}$$

where $w_{ij} \geq 0$ and $\sum_j w_{ij} = 1$. Different choices for the coefficients $\{w_{ij}\}_{i,j=1}^n$ result in different protocols. The choice $w_{ii} = 1 - \epsilon(\Delta(\mathcal{G}))_{ii}$ and $w_{ij} = -\epsilon(A(\mathcal{G}))_{ij}$ for $i \neq j$ results in the following protocol

$$\mathbf{x}(t+1) = P_\epsilon(\mathcal{G})\mathbf{x}(t), \tag{2.21}$$

where $\mathbf{x} = [x_1 \ \cdots \ x_n]^T$ and $0 < \epsilon < 1/d_{\max}(\mathcal{G})$.

The convergence properties of the consensus protocol (2.21) depend on the connectivity of

the communication graph $\mathcal{G}$. Specifically, if $\mathcal{G}$ is undirected and connected then,

$$\lim_{k\to\infty} P_\epsilon(\mathcal{G})^k = \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T, \tag{2.22}$$

where n is the number of agents and $P_\epsilon(\mathcal{G})^k$ denotes the kth power of matrix $P_\epsilon(\mathcal{G})$. Thus,

$$\lim_{t\to\infty} x(t) = (\mathbf{1}_n^T x(0)/n)\mathbf{1}_n, \tag{2.23}$$

that is the states of all agents asymptotically converge to the the average of the initial states.

In the case of a directed communication graph $\mathcal{G}$, we have that a necessary condition for protocol (2.21) to converge to agreement is the existence of a rooted out-branching as a subgraph. Specifically, if $\mathcal{G}$ contains a rooted out-branching as a subgraph, then for any initial condition $x(0)$

$$\lim_{t\to\infty} x(t) = (q_1^T x(0)/\sqrt{n})\mathbf{1}_n, \tag{2.24}$$

where q_1 is the right eigenvector of the Laplacian associated with its eigenvalue 1.

2.4 Distributed optimization

In this section, we include some basic convergence results regarding projected gradient descent with constant step size for minimizing an objective defined on an undirected graph. We denote by $\langle\cdot,\cdot\rangle$ the standard Euclidean inner product, by $\operatorname{grad}\phi(x)$ the gradient of the real-valued function ϕ and by $\operatorname{Hess}\phi(x)[u]$ the Hessian of the real-valued function ϕ at x evaluated at the direction u.

First, we include the following classic result [12] regarding the maximum allowed step size for projected gradient descent.

Lemma 2.4.1 (Projected gradient descent [12])**.** *Assume we are given the problem of optimizing real-valued objective ϕ over some convex compact set $\mathcal{C} \subset \mathbb{R}^n$, where ϕ is twice*

continuously differentiable with

$$\langle v, \operatorname{Hess} \phi(x)[v] \rangle \leq \mu_{\max} \langle v, v \rangle, \tag{2.25}$$

for some positive constant $\mu_{\max}$*. Then, every limit point of the sequence* $\{x^k\}$ *generated by projected gradient descent with constant step size* $\epsilon > 0$*, i.e.*

$$x^{k+1} = \operatorname{Proj}_{\mathcal{C}}\big(x^k - \epsilon \operatorname{grad} \phi(x^k)\big), \tag{2.26}$$

is a stationary point $x^\star$ *(that is* $\langle \operatorname{grad} \phi(x^\star), (x - x^\star) \rangle \geq 0$ *for all* $x \in \mathcal{C}$*) for all* $0 < \epsilon < 2/\mu_{\max}$.

Next, we consider the problem of constrained minimization of an objective defined on a graph with projected gradient descent with constant step size. Specifically, let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph and ϕ a real-valued function defined on $\mathcal{G}$ by

$$\phi(\{x_i\}_{i \in \mathcal{V}}) = \alpha \sum_{i \in \mathcal{V}} \phi_i(x_i) + \beta \sum_{\{i,j\} \in \mathcal{E}} \phi_{ij}(x_i, x_j) \tag{2.27}$$

where each vertex i maintains a vector $x_i \in \mathbb{R}^{n_i}$, α, β are nonnegative scalars, $\phi_i : \mathbb{R}^{n_i} \to \mathbb{R}$ and $\phi_{ij} : \mathbb{R}^{n_i} \times \mathbb{R}^{n_j} \to \mathbb{R}$ are twice continuously differentiable real-valued functions. We consider constrained optimization problems of the following form

$$\begin{aligned} \underset{\{x_i\}_{i \in \mathcal{V}}}{\text{minimize}} \quad & \phi(\{x_i\}_{i \in \mathcal{V}}) \\ \text{subject to} \quad & x_i \in \mathcal{C}_i, \quad i = 1, 2, \ldots, m, \end{aligned} \tag{2.28}$$

where each $\mathcal{C}_i \subset \mathbb{R}^{n_i}$ is a convex compact set. The projected gradient descent method with constant step size ϵ consists of updates of the form

$$\begin{aligned} w_i^k &= -\operatorname{grad}_{x_i} \phi(\{x_i^k\}_{i \in \mathcal{V}}), \\ x_i^{k+1} &= \operatorname{Proj}_{\mathcal{C}_i}\big(x_i^k + \epsilon w_i^k\big), \quad i = 1, 2, \ldots, m. \end{aligned} \tag{2.29}$$

Bounds on the step size ϵ that guarantee convergence to a stationary point are presented in the proposition that follows.

Proposition 2.4.2. *Assume that*

$$\langle v_i, \operatorname{Hess} \phi_i(x_i)[v_i] \rangle \leq \mu_{\max} \langle v_i, v_i \rangle, \tag{2.30}$$

$$\langle (v_i, v_j), \operatorname{Hess} \phi_{ij}(x_i, x_j)[v_i, v_j] \rangle \leq \nu_{\max} \langle (v_i, v_j), (v_i, v_j) \rangle, \tag{2.31}$$

for some positive constants $\mu_{\max}$, $\nu_{\max}$. Then, every limit point generated by the projected gradient descent method with constant step size ϵ satisfying

$$0 < \epsilon < 2/\left(\alpha \mu_{\max} + \beta \nu_{\max} d_{max}(\mathcal{G})\right), \tag{2.32}$$

is a stationary point.

The proof of the above proposition is presented in Appendix B.1 and is based on a similar proof of [136]. Note that the projected gradient descent rule (2.29) is by construction decentralized since the gradient of the objective ϕ with respect to x_i, denoted by $\operatorname{grad}_{x_i} \phi$, can be estimated using only information from the neighborhood of i.

2.5 Alternating direction method of multipliers

The Alternating direction method of multipliers (ADMM) [17] is a widely used optimization method especially suitable for large scale problems. With ADMM, one can solve optimization problems with separable objective and linear equality constraints, that is, optimization problems of the following so-called standard form:

$$\begin{aligned} \underset{x,z}{\text{minimize}} \quad & f(x) + g(z) \\ \text{subject to} \quad & Ax + Bz = c, \end{aligned} \tag{2.33}$$

where f, g are real-valued functions. For problem (2.33), the augmented Lagrangian is given by

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + (\rho/2)\|Ax + Bz - c\|_2^2, \tag{2.34}$$

where $\rho > 0$ is the penalty parameter. Then, the ADMM iterations are as follows

$$x^{k+1} := \underset{x}{\operatorname{argmin}}\, L_\rho(x, z^k, y^k), \tag{2.35}$$

$$z^{k+1} := \underset{z}{\operatorname{argmin}}\, L_\rho(x^k, z, y^k), \tag{2.36}$$

$$y^{k+1} := y^k + \rho(x^{k+1} - z^{k+1}). \tag{2.37}$$

Intuitively, the ADMM algorithm is an approximate method of multipliers. Instead of minimizing the Lagrangian jointly over x and z to compute the gradient of the dual function, the augmented Lagrangian is minimized in an alternate fashion. Although the standard form (2.33) may seem restrictive at first sight, it is in fact as general as one can wish for. Any optimization problem can be written in form (2.33) as follows. Assume that the problem at hand is minimizing a real-valued objective $f(x)$ over some set $\mathcal{C}$. Let $I_\mathcal{C}$ denotes the indicator function of the set $\mathcal{C}$, which takes the value 0 in $\mathcal{C}$ and the value $+\infty$ outside of $\mathcal{C}$. Then, we can equivalently write it in standard form as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) + I_\mathcal{C}(z) \\ \text{subject to} \quad & x - z = 0, \end{aligned} \tag{2.38}$$

and the ADMM iterations are as follows

$$x^{k+1} := \underset{x}{\operatorname{argmin}} \left\{ f(x) + (\rho/2)\|x - z^k + u^k\|_F^2 \right\}, \tag{2.39}$$

$$z_i^{k+1} := \operatorname{proj}_\mathcal{C}\left(x^{k+1} + u^k\right), \tag{2.40}$$

$$u^{k+1} := u^k + x^{k+1} - z^{k+1}, \tag{2.41}$$

where $u^k = (1/\rho)y^k$ is the scaled dual vector. For the above iterations to make sense in terms of computational tractability, the projection operation onto the set $\mathcal{C}$ has to be efficiently computable.

2.6 Stability of autonomous systems

In this section, we review some basic notions of stability. For a thorough treatment, we refer the reader to [62]. Let $B(x_0, r)$ denote the ball of radius r centered at x_0, i.e.

$$B(x_0, r) = \{x \in \mathbb{R}^n \mid \|x - x_0\|_2 < r\}. \tag{2.42}$$

Moreover, let $\text{dist}(x, M)$ denote the distance of a point x to a set M, that is

$$\text{dist}(x, M) = \inf_{x_0 \in M} \|x - x_0\|_2. \tag{2.43}$$

In addition, we need to define the following set:

$$M_\epsilon = \{x \in \mathbb{R}^n \mid \text{dist}(x, M) < \epsilon\}. \tag{2.44}$$

Now, consider the autonomous system

$$\dot{x} = f(x), \tag{2.45}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$ is a locally Lipschitz function. Without loss of generality, assume that $x = \mathbf{0}$ is an equilibrium for (2.45), that is $f(\mathbf{0}) = 0$. Then, $x = \mathbf{0}$ is *stable* if, for all $\epsilon > 0$, there exists a $\delta > 0$ such that

$$x(0) \in B(\mathbf{0}, \delta) \Rightarrow x(t) \in B(\mathbf{0}, \epsilon), \quad \forall\, t \geq 0, \tag{2.46}$$

and *unstable* if it is not stable. The origin $x = \mathbf{0}$ is *(locally) attractive* if for all $\epsilon > 0$, there exists a $\delta > 0$ and a time $T > 0$ such that

$$x(0) \in B(\mathbf{0}, \delta) \Rightarrow x(t) \in B(\mathbf{0}, \epsilon), \quad \forall\ t \geq T. \tag{2.47}$$

The origin is *asymptotically stable* if it is stable and locally attractive.

A set M is *invariant* with respect to (2.45) if

$$x(0) \in M \Rightarrow x(t) \in M, \quad \forall\ t \geq 0. \tag{2.48}$$

A closed invariant set M is *stable* if, for all $\epsilon > 0$, there exists $\delta > 0$ such that

$$x(0) \in M_\delta \Rightarrow x(t) \in M_\epsilon, \quad \forall\ t \geq 0, \tag{2.49}$$

and *(locally) attractive* if for all $\epsilon > 0$, there exists a $\delta > 0$ and a time $T > 0$ such that

$$x(0) \in M_\delta \Rightarrow x(t) \in M_\epsilon, \quad \forall\ t \geq T. \tag{2.50}$$

A closed invariant set is *asymptotically stable* if it is stable and attractive. A closed invariant set M is *unstable* if it is not stable, that is if, there is an $\epsilon > 0$, such that for all $\delta > 0$

$$x(t) \notin M_\epsilon, \quad \text{for some } x(0) \in M_\delta. \tag{2.51}$$

and some $t \geq 0$. A closed invariant set M is *uniformly unstable* if, there is an $\epsilon > 0$, such that for all $\delta > 0$ and all $x \in M$

$$x(t) \notin M_\epsilon, \quad \text{for some } x(0) \in B(x, \delta), \tag{2.52}$$

and some $t \geq 0$. Intuitively, if M is uniformly unstable, one can find an initial condition $x(0)$ arbitrarily close to any $x \in M$, such that the trajectory $x(t)$ of the system eventually exits

the set M_ϵ. A closed invariant set M is *non-attractive* if it is not attractive and *uniformly non-attractive* if the initial condition $x(0)$ can be chosen in a neighborhood of any $x \in M$.

2.7 Group theory and differential geometry

In this section, we briefly review several elementary facts from group theory and differential geometry. For a more detailed and rigorous treatment, we refer the reader to [32, 38, 104, 1, 68].

A d-dimensional *manifold* $\mathcal{M}$ can be informally defined as a set $\mathcal{M}$ that is locally homeomorphic to the Euclidean space $\mathbb{R}^d$. The tangent space $T_x\mathcal{M}$ at a point $x \in \mathcal{M}$ is the vector space consisting of all the tangents of all smooth curves in $\mathcal{M}$ passing through x. A *Riemannian manifold* is a manifold whose tangent spaces are equipped with a smoothly varying inner product, which is called a *Riemannian metric*. We use the notation $g(\xi, \zeta)$ to denote the inner product of two elements $\xi, \zeta \in T_x\mathcal{M}$ (where the point x will be clear from the context). The metric naturally induces a norm $\|\xi\| \doteq \sqrt{g(\xi, \xi)}$.

A *geodesic curve* on $\mathcal{M}$ is the generalization of a straight line (that is, a curve with zero acceleration). We denote as $\gamma_{x,\xi}(t)$ the geodesic emanating from x in the direction of $\xi \in T_x\mathcal{M}$. The *exponential map* $\exp_x : T_x\mathcal{M} \to \mathcal{M}$ is defined as $\exp_x \xi \doteq \gamma_{x,\xi}(1)$. The *logarithm map* $\log_x : \mathcal{M} \to T_x\mathcal{M}$ is the inverse of the exponential map and is generally defined only in a neighborhood of x. Where defined, we have the identity $d(x, y) = \|\log_x(y)\|$, where $d(x, y)$ is the Riemannian distance of x, y induced by the metric.

Let $F : \mathcal{M} \to \mathcal{N}$ be a smooth map between two manifolds $\mathcal{M}$ and $\mathcal{N}$. The linear mapping

$$DF(x) : T_x\mathcal{M} \to T_{F(x)}\mathcal{M},\ \xi \mapsto DF(x)[\xi],$$

is called the *differential* of F at x. For any curve $\gamma(t)$ on $\mathcal{M}$ we have

$$DF(\gamma(t))[\dot{\gamma}(t)] = \frac{d}{dt}F(\gamma(t)). \tag{2.53}$$

Furthermore, given a real-valued function $f : \mathcal{M} \to \mathbb{R}$, the Riemannian gradient $\operatorname{grad} f(x)$ of f at a point $x \in \mathcal{M}$ is the unique element of $T_x\mathcal{M}$ satisfying

$$g(\operatorname{grad} f(x), \xi) = Df(x)[\xi], \tag{2.54}$$

for all $\xi \in T_x\mathcal{M}$. The Riemannian Hessian is the self-adjoint linear map

$$\operatorname{Hess} f(x) : T_x\mathcal{M} \to T_x\mathcal{M},\ \xi \mapsto \operatorname{Hess} f(x)[\xi],$$

satisfying

$$g(\xi, \operatorname{Hess} f(x)[\xi]) = \left.\frac{d^2}{dt^2} f(\gamma_{x,\xi}(t))\right|_{t=0}, \tag{2.55}$$

for all $\xi \in T_x\mathcal{M}$.

Let $\mathcal{M}, \mathcal{N}$ be manifolds such that $\mathcal{N} \subset \mathcal{M}$. If $\mathcal{N}$ has the subspace topology inherited from $\mathcal{M}$, then $\mathcal{N}$ is called an *embedded submanifold* of $\mathcal{M}$ and $\mathcal{M}$ is termed the *emdedding space*. Note that given a Riemannian metric on $\mathcal{M}$, its restriction to $\mathcal{N}$ induces a Riemmanian metric on $\mathcal{N}$.

A group $(G, \cdot)$ is a set G along with a binary operation $\cdot : G \times G \to G$ satisfying the axioms of closure, associativity, existence of an identity element $e \in G$ and existence of inverse for each element in the group. A *Lie group* is a group that is also a manifold. If G is a group and $\mathcal{M}$ is a set, a *left action* of G on $\mathcal{M}$ is a map $G \times \mathcal{M} \to \mathcal{M}$, written as $(g, p) \mapsto g \cdot p$, satisfying $g_1 \cdot (g_2 \cdot p) = (g_1 g_2) \cdot p$, for all $g_1, g_2 \in M, p \in \mathcal{M}$ and $e \cdot p = p$ for all $p \in \mathcal{M}$. The action is said *continuous* if the corresponding map is continuous, and it is said *free* if $g \cdot p = p$ for some $p \in M$ implies that $g = e$. A group action induces an equivalence relation $\sim$ on $\mathcal{M}$: for any $x, y \in \mathcal{M}$, $x \sim y$ if $y = g \cdot x$ for some $g \in G$.

Let $\overline{\mathcal{M}}$ be a manifold equipped with an equivalence relation $\sim$. The equivalence class of a point $x \in \overline{\mathcal{M}}$ is denoted by $[x] = \{y \in \overline{\mathcal{M}} : y \sim x\}$. The *quotient space* $M = \overline{\mathcal{M}}/\sim$ is the set of all equivalence classes and $\overline{\mathcal{M}}$ is termed the *total space* or *ambient space*. The *canonical projection* is the map $\pi : \overline{\mathcal{M}} \to \mathcal{M}$ defined by $\pi(x) = [x]$. The quotient space is called a *quotient manifold* if the canonical projection is a submersion, that is the differential of π at every point is surjective. If the quotient space is a manifold and $\dim(\mathcal{M}/\sim) < \dim(\mathcal{M})$, then each equivalence class $\pi^{-1}(\pi(x))$, $x \in \mathcal{M}$, is an embedded submanifold of $\mathcal{M}$. Consider any $x \in \mathcal{M}$ and let $\overline{x} \in \pi^{-1}(x) \subseteq \overline{\mathcal{M}}$. The *vertical space* $\mathcal{V}_{\overline{x}} = T_{\overline{x}}(\pi^{-1}(x))$ at $\overline{x}$ is the tangent space to the equivalence class $\pi^{-1}(x)$. The *horizontal space* $\mathcal{H}_{\overline{x}}$ is the orthogonal complement of $\mathcal{V}_{\overline{x}}$ in $T_{\overline{x}}\overline{\mathcal{M}}$, that is,

$$\mathcal{V}_{\overline{x}} \oplus \mathcal{H}_{\overline{x}} = T_{\overline{x}}\overline{\mathcal{M}}. \tag{2.56}$$

Given any and $\xi \in T_x\mathcal{M}$, there exists exactly one *horizontal lift* $\overline{\xi}_{\overline{x}} \in \mathcal{H}_{\overline{x}}$ satisfying $D\pi(\overline{x})[\overline{\xi}] = \xi$.

In the context of this work, we will frequently use the Lie group of three dimensional rotations

$$SO(3) = \{R \in \mathbb{R}^{3\times 3} : R^T R = I, \det(R) = 1\}. \tag{2.57}$$

The tangent space at a point $R \in SO(3)$ is given by

$$T_R SO(3) = \{R\Omega : \Omega \in \mathfrak{so}(3)\}, \tag{2.58}$$

where $\mathfrak{so}(3)$ denotes the vector space of 3×3 skew-symmetric matrices. Given $R, Q \in SO(3)$ and $\xi \in T_R SO(3)$, the exponential and the logarithm maps are given by

$$\exp_R(\xi) = R \exp_I(R^T \xi), \tag{2.59}$$

$$\log_R(Q) = R \log_I(R^T Q), \tag{2.60}$$

where $\exp_I$ and $\log_I$ denote the exponential and the logarithm map at the identity which coincide with the matrix exponential and logarithm. For the group of rotations there are explicit formulas, e.g. Rodrigues' formula, for the matrix exponential and the matrix logarithm [84]. Before defining the metric of manifold of rotations, we need to introduce some notation,. The *hat* operator $\wedge : \mathbb{R}^3 \to \mathfrak{so}(3)$ is defined as

$$\widehat{u} \doteq \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}, \tag{2.61}$$

where $u = (u_1, u_2, u_3)^T$. If $u, v \in \mathbb{R}^3$ and $\times$ denotes the cross product of vectors in $\mathbb{R}^3$, then $u \times v = \widehat{u}v$. The inverse map of hat operator is the *vee* operator $\vee : \mathfrak{so}(3) \to \mathbb{R}^3$. The standard metric of $SO(3)$ at a point $R \in SO(3)$ is given by

$$g(\xi_1, \xi_2) = \frac{1}{2}\operatorname{tr}(\xi_1^T \xi_2) = \frac{1}{2}\operatorname{tr}(\Omega_1^T \Omega_2) = \omega_1^T \omega_2, \tag{2.62}$$

where $\xi_i = R\Omega_i \in T_R SO(3)$ and $\omega_1 = \Omega_1^\vee$.

For modeling the translational part of the trifocal tensor, we use Kendall's shape space [61]. Following the Kendall's notation, we define

$$\mathbb{S}_2^3 = \{X \in \mathbb{R}^{2\times 2} : \|X\|_F = 1\}, \tag{2.63}$$

as the space of triangles in 2-D. The tangent space at a point $X \in \mathbb{S}_2^3$ is

$$T_X \mathbb{S}_2^3 = \{\xi \in \mathbb{R}^{2\times 2} : \operatorname{tr}(X^T \xi) = 0\} = X^\perp, \tag{2.64}$$

and the Riemannian metric is the usual Euclidean inner product. We also introduce the space

$$\mathbb{S}_2^{3*} = \{X \in \mathbb{S}_2^3 : \operatorname{rank}(X) = 2\} \subset \mathbb{S}_2^3, \tag{2.65}$$

which is the space of non-degenerate triangles. Finally, the exponential and the logarithm maps can be computed as

$$\exp_X(\xi) = \cos(\|\xi\|)X + \frac{\sin(\|\xi\|)}{\|\xi\|}\xi, \tag{2.66}$$

$$\log_X(Y) = \frac{\arccos(\operatorname{tr}(X^TY))}{\sqrt{1-\operatorname{tr}(X^TY)^2}}(Y - X\operatorname{tr}(X^TY)), \tag{2.67}$$

for $X, Y \in \mathbb{S}_2^3$ and $\xi \in T_X\mathbb{S}_2^3$.

Chapter 3

Distributed permutation synchronization

3.1 Introduction

In this chapter, we consider the problem of multiway matching with fully observable pairwise associations. We formulate it as an instance of permutation synchronization which refers to recovering a set of permutations from noisy relative permutations. We present two novel distributed optimization algorithms for permutation synchronization. The first consists of a consensus-like algorithm that results from a convex relaxation of the permutation synchronization problem. We show that this consensus-like algorithm is fully decentralized, provably converges, does not depend on initialization and guarantees cycle consistency. The second algorithm is a distributed projected gradient descent on the set of doubly stochastic matrices that enforces cycle consistency while promoting sparsity.

The remainder of this chapter is structured as follows. A formalization of permutation synchronization is the subject of Section 3.2 followed by the proposed problem formulation in Section 3.3. The first approach is presented in Section 3.4 and the second one, along with

the combined algorithm, is presented in Section 3.5.

3.2 Problem statement

In this section, we formalize the problem of permutation synchronization. We assume there are m collections, each containing n features. For instance, consider the case of m images of a car from different viewpoints, each having n keypoints. The pairwise association between collections i and j, denoted by $\pi_{ij} \in \mathfrak{S}_n$, is defined as follows: we have that $\pi_{ij}(k) = l$ if the kth feature in collection j corresponds to the lth feature in collection i. We denote by $\widetilde{\pi}_{ij} \in \mathfrak{S}_n$ the, possibly erroneous, estimated pairwise association between collections i and j, which is the output of some pairwise matching algorithm, e.g. graph matching, and let $\widetilde{\Pi}_{ij}$ denote the corresponding matrix representation of $\widetilde{\pi}_{ij}$.

The availability of a pairwise measurement between collections i and j defines a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ as follows: the vertex set contains the m collections, that is $\mathcal{V} = \{1, 2, \ldots, m\}$, and $\{i, j\} \in \mathcal{E}$ if there is a pairwise association $\widetilde{\pi}_{ij}$ between collections i and j. We assume that there are no self-loops, that is $\{i, i\} \notin \mathcal{E}$ for any $i \in \mathcal{V}$ and the measured pairwise associations are symmetric, in the sense that $\widetilde{\pi}_{ji} = \widetilde{\pi}_{ij}^{-1}$ (resp. $\widetilde{\Pi}_{ji} = \widetilde{\Pi}_{ij}^{T}$) for all $\{i, j\} \in \mathcal{E}$. We refer to this graph $\mathcal{G}$ as the *sensor graph*.

Related to the sensor graph $\mathcal{G}$ is the *data association graph* $\mathcal{D} = (\mathcal{V}_{\mathcal{D}}, \mathcal{E}_{\mathcal{D}}, w_{\mathcal{D}})$, where $\mathcal{V}_{\mathcal{D}} = \mathcal{V} \times \{1, 2, \ldots, n\}$. There is an edge from (i, l) to (j, k) if and only if $\{i, j\} \in \mathcal{E}$ and $(\widetilde{\Pi}_{ij})_{lk} = 1$.

Before stating the problem at hand, we need a precise definition of *cycle consistency* [53].

Definition 3.2.1 (Cycle consistency)**.** *A set of pairwise associations $\{\widetilde{\pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$ is cycle consistent if for any cycle $i_0, i_1, \ldots, i_k, i_0$ we have*

$$\widetilde{\pi}_{i_0 i_1} \circ \widetilde{\pi}_{i_1 i_2} \circ \ldots \circ \widetilde{\pi}_{i_k i_0} = e, \tag{3.1}$$

where e is the identity permutation and $\circ$ denotes the function composition.

Next, we present a necessary and sufficient condition for cycle consistency, originally identified in [53].

Lemma 3.2.2 (Consistency and the universe of features)**.** *Assume that the sensor $\mathcal{G}$ is connected. Then, a set of pairwise associations $\{\widetilde{\pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$ is consistent if and only if there exist $\pi_1, \pi_2, \ldots, \pi_m \in \mathfrak{S}_n$ (resp. $\Pi_1, \Pi_2, \ldots, \Pi_m \in \mathcal{P}_n$), such that*

$$\widetilde{\pi}_{ij} = \pi_i \circ \pi_j^{-1} \qquad (\textit{resp. } \widetilde{\Pi}_{ij} = \Pi_i \Pi_j^T), \tag{3.2}$$

for all $\{i, j\} \in \mathcal{E}$.

Proof. First, assume that there exist $\pi_1, \pi_2, \ldots, \pi_m \in \mathfrak{S}_n$ such that $\widetilde{\pi}_{ij} = \pi_i \circ \pi_j^{-1}$ for all $\{i, j\} \in \mathcal{E}$. Then, for any cycle $i_0, i_1, \ldots, i_k, i_0$ we have

$$\widetilde{\pi}_{i_0 i_1} \circ \widetilde{\pi}_{i_1 i_2} \circ \ldots \circ \widetilde{\pi}_{i_k i_0} = \pi_{i_0} \circ \pi_{i_1}^{-1} \circ \pi_{i_1} \circ \pi_{i_2}^{-1} \circ \ldots \circ \pi_{i_k} \circ \pi_{i_0}^{-1} = e,$$

and thus, cycle consistency is satisfied.

Conversely, assume that cycle consistency is satisfied. Define $\pi_1 = e$ and for any vertex $i_0 \neq 1$, define $\pi_{i_0} \doteq \widetilde{\pi}_{i_0 i_1} \circ \widetilde{\pi}_{i_1 i_k} \circ \ldots \circ \widetilde{\pi}_{i_k 1}$ where $i_0, i_1, \ldots, i_k, 1$ is a path from i_0 to 1 (and vice versa). Such a path exists since $\mathcal{G}$ is assumed to be undirected and connected. Moreover, by cycle consistency, the above definition does not depend on the choice of the path. Then, for all $\{i_0, j_0\} \in \mathcal{E}$, by cycle consistency, we have

$$\widetilde{\pi}_{i_0 j_0} = \widetilde{\pi}_{i_0 i_1} \circ \ldots \circ \widetilde{\pi}_{i_k 1} \circ \widetilde{\pi}_{1 j_k} \circ \ldots \widetilde{\pi}_{j_1 j_0} = \pi_{i_0} \circ \pi_{j_0}^{-1}.$$

The proof is complete. □

Intuitively, $\pi_i \in \mathfrak{S}_n$ is a permutation from the local feature enumeration (local labels) of collection i to some global feature enumeration (global labels), termed the "universe of features" in some works [23, 156]. Next, we present the cycle consistency conditions, as stated

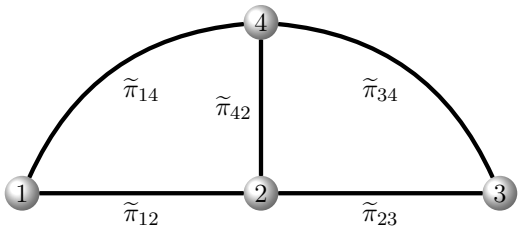


Figure 3.1: In this example, cycle consistency is satisfied if and only if $\widetilde{\pi}_{14} \circ \widetilde{\pi}_{42} = \widetilde{\pi}_{12}$, $\widetilde{\pi}_{34} \circ \widetilde{\pi}_{42} = \widetilde{\pi}_{32}$ and $\widetilde{\pi}_{14} \circ \widetilde{\pi}_{43} = \widetilde{\pi}_{12} \circ \widetilde{\pi}_{23}$.

by Huang and Guibas [53], in terms of the representations of the pairwise associations in the special case when all pairwise associations are available.

Lemma 3.2.3 (Conditions for consistency [53])**.** *Assume that the sensor $\mathcal{G}$ is fully connected, that is $\{i,j\} \in \mathcal{E}$ for all $i \neq j$. Given pairwise associations $\{\widetilde{\pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$, where each $\widetilde{\pi}_{ij} \in \mathfrak{S}_n$, define the block matrix $\widetilde{\mathbf{\Pi}}$ by $[\widetilde{\mathbf{\Pi}}]_{ij} = \widetilde{\Pi}_{ij}$ for $i \neq j$ and $[\widetilde{\mathbf{\Pi}}]_{ii} = I_n$, where $\widetilde{\Pi}_{ij}$ is the matrix representation of $\widetilde{\pi}_{ij}$. Then, the following are equivalent:*

(i) The set of pairwise associations $\{\widetilde{\pi}_{ij}\}_{(i,j)\in\mathcal{E}}$ is consistent.

(ii) $\operatorname{rank}(\widetilde{\mathbf{\Pi}}) = n$ *and $\widetilde{\mathbf{\Pi}}$ can be factorized as*

$$\widetilde{\mathbf{\Pi}} = \begin{bmatrix} \Pi_1 \\ \Pi_2 \\ \vdots \\ \Pi_m \end{bmatrix} \cdot \begin{bmatrix} \Pi_1^T & \Pi_2^T & \cdots & \Pi_m^T \end{bmatrix}. \tag{3.3}$$

for some $\Pi_1, \Pi_2, \ldots, \Pi_m \in \mathcal{P}_n$.

(iii) $\widetilde{\mathbf{\Pi}} \succeq 0$.

Next, we present the problem statement.

Definition 3.2.4 (Permutation synchronization)**.** *Given, possibly erroneous, pairwise associations $\{\widetilde{\pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$, estimate consistent pairwise associations $\{\pi_{ij}\}_{\{i,j\}\in\mathcal{E}}$ that are close to the given ones or, equivalently, find permutations $\pi_1, \pi_2, \ldots, \pi_m \in \mathfrak{S}_n$ (resp. $\Pi_1, \Pi_2, \ldots, \Pi_m \in$*

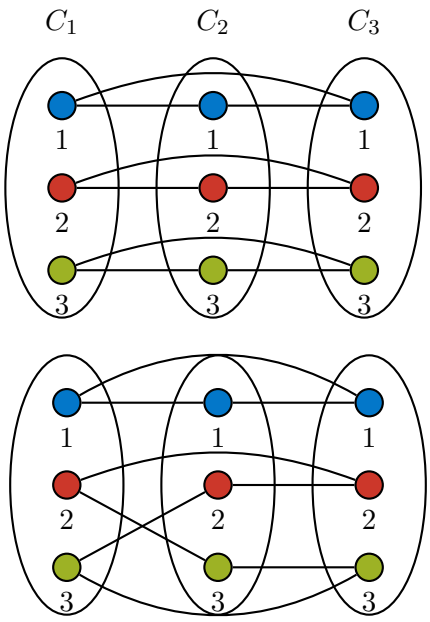


Figure 3.2: Example with $m = 3$ collections C_1, C_2, C_3 observing $n = 3$ features. Top: consistent associations. Bottom: inconsistent associations since $\pi_{12} \circ \pi_{23}(2) = 3$ but $\pi_{13}(2) = 2$ and thus, $\pi_{12} \circ \pi_{23} \neq \pi_{13}$ violating cycle consistency.

$\mathcal{P}_n$*), such that*

$$\widetilde{\pi}_{ij} \approx \pi_i \circ \pi_j^{-1}, \qquad (\textit{resp. } \widetilde{\Pi}_{ij} \approx \Pi_i \Pi_j^T), \tag{3.4}$$

for all $\{i, j\} \in \mathcal{E}$.

3.3 Proposed formulation

We cast the problem of permutation synchronization as the following combinatorial optimization problem:

$$\underset{\pi_1,\ldots,\pi_m \in \mathfrak{S}_n}{\text{minimize}} \quad \sum_{\{i,j\} \in \mathcal{E}} d(\pi_i \circ \pi_j^{-1}, \widetilde{\pi}_{ij}). \tag{3.5}$$

where $d(\cdot, \cdot)$ is the distance of symmetric group as defined as (2.18). Unfortunately, problem (3.5) is computationally intractable. For this reason, we first, derive an equivalent formulation of problem (3.5) in terms of the matrix representations $\Pi_1, \ldots, \Pi_m \in \mathcal{P}_n$ of $\pi_1, \ldots, \pi_m \in \mathfrak{S}_n$ and then, relax the domain of the equivalent formulation from permuta-

tion matrices to doubly stochastic matrices.

We consider the following cost for each edge $\{i,j\} \in \mathcal{E}$, $\phi_{ij} : \mathfrak{S}_n \times \mathfrak{S}_n \to \mathbb{R}_+$ defined as

$$\phi_{ij}(\pi_i, \pi_j) \doteq d(\pi_i \circ \pi_j^{-1}, \widetilde{\pi}_{ij}) = d(\pi_i, \widetilde{\pi}_{ij}\pi_j). \tag{3.6}$$

Since, $d(\pi_i, \widetilde{\pi}_{ij}\pi_j) = (1/2)\|\Pi_i - \widetilde{\Pi}_{ij}\Pi_j\|_F^2$, the edge costs can by naturally extended to take permutation matrices as arguments, by defining $\phi_{ij} : \mathcal{P}_n \times \mathcal{P}_n \to \mathbb{R}_+$ as follows

$$\phi_{ij}(\Pi_i, \Pi_j) \doteq \frac{1}{2}\|\Pi_i - \widetilde{\Pi}_{ij}\Pi_j\|_F^2. \tag{3.7}$$

Observe that since $\widetilde{\Pi}_{ij}$ is a permutation matrix and we have assumed $\widetilde{\Pi}_{ij}^T = \widetilde{\Pi}_{ji}$, it follows that the edge costs ϕ_{ij} are symmetric, in the sense,

$$\phi_{ij}(\Pi_i, \Pi_j) = \phi_{ji}(\Pi_j, \Pi_i). \tag{3.8}$$

Then, problem (3.5) is equivalent to the following problem:

$$\begin{aligned} \underset{\{\Pi_i\}_{i\in\mathcal{V}}}{\text{minimize}} \quad & \phi(\{\Pi_i\}_{i\in\mathcal{V}}) \doteq \sum_{\{i,j\}\in\mathcal{E}} \phi_{ij}(\Pi_i, \Pi_j) \\ \text{subject to} \quad & \Pi_i \in \mathcal{P}_n. \end{aligned} \tag{3.9}$$

The above optimization problem is still computationally intractable due to its combinatorial nature. To address this problem, we propose to relax the domain of the problem from the set of permutation matrices to its convex hull, the set of doubly stochastic matrices. This is the topic of the next section.

It can be easily that the objective ϕ remains unchanged if the matrices $\{\Pi_i\}_{i\in\mathcal{V}}$ are all right-multiplied by the same permutation. This observation suggests that the global minimizer of problem (3.9) is not unique. This is summarized in the following remark.

Remark 3.3.1 (Global minimizer is not unique)**.** *For any permutation matrix* $\Pi_0 \in \mathcal{P}_n$*,*

$$\phi(\{\Pi_i\}_{i\in\mathcal{V}}) = \phi(\{\Pi_i\Pi_0\}_{i\in\mathcal{V}}). \tag{3.10}$$

Thus, if the feasible point $\{\Pi_i\}_{i\in\mathcal{V}}$ *is globally optimal for problem* (3.9) *then,*

$$\{\Pi_i\Pi_0\}_{i\in\mathcal{V}}$$

is globally optimal as well for any permutation matrix $\Pi_0 \in \mathcal{P}_n$*.*

3.4 A first approach by convex relaxation

3.4.1 The convex relaxation

Problem (3.9) is computationally intractable due to the permutation constrains. For this reason, we relax the problem domain from the set of permutations to its convex hull, the set of doubly stochastic matrices. In this way, we obtain the following convex optimization problem:

$$\begin{aligned} \underset{\{\Pi_i\}_{i\in\mathcal{V}}}{\text{minimize}} \quad & \phi(\{\Pi_i\}_{i\in\mathcal{V}}) \\ \text{subject to} \quad & \Pi_i\mathbf{1}_n = \mathbf{1}_n, \\ & \mathbf{1}_n^T\Pi_i = \mathbf{1}_n^T, \qquad \forall\ i \in \mathcal{V} \\ & \Pi_i \geq 0, \\ & \Pi_1 = I, \end{aligned} \tag{3.11}$$

where the edge cost functions ϕ_{ij} are implicitly extended to take matrices with real entries as arguments.

Let $\Pi = [\Pi_1^T \cdots \Pi_m^T]^T \in \mathbb{R}^{mn\times n}$. With a slight abuse of notation we write $\phi(\Pi)$ as a shorthand for $\phi(\{\Pi_i\}_{i\in\mathcal{V}})$ and we view ϕ as real-valued function defined on $\mathbb{R}^{mn\times n}$. Observe

that

$$\phi(\Pi) = \frac{1}{2}\operatorname{tr}(\Pi^T \mathcal{L}(\mathcal{D})\Pi), \tag{3.12}$$

where $\mathcal{L}(\mathcal{D})$ is the Laplacian of the data association graph, that is

$$[\mathcal{L}(\mathcal{D})]_{ij} = \begin{cases} |\mathcal{N}_i|\, I_n, & \text{if } i = j, \\ -\widetilde{\Pi}_{ij}, & \text{if } i \neq j,\ \{i,j\} \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.13}$$

Moreover, let $\mathcal{C}$ denote the constraint set of problem (3.11). Then, problem (3.11) can be written more compactly as follows

$$\begin{aligned} \underset{\Pi \in \mathbb{R}^{mn \times n}}{\text{minimize}} \quad & \frac{1}{2}\operatorname{tr}(\Pi^T \mathcal{L}(\mathcal{D})\Pi) \\ \text{subject to} \quad & \Pi \in \mathcal{C}. \end{aligned} \tag{3.14}$$

3.4.2 The update rule and its limit

We propose to solve problem (3.14) by distributed projected gradient descent with a constant step size $\epsilon > 0$. The updates are given by

$$\Pi(t+1) = \operatorname{Proj}_{\mathcal{C}}((I - \epsilon \mathcal{L}(\mathcal{D}))\Pi(t)). \tag{3.15}$$

In this particular case, projected gradient descent iterations take the following form:

$$\begin{aligned} &\Pi_1(t+1) = \Pi_1(t) = I, \\ &\Pi_i(t+1) = (1 - \epsilon\,|\mathcal{N}_i|)\Pi_i(t) + \epsilon \sum_{j \in \mathcal{N}_i} \widetilde{\Pi}_{ij}\Pi_j(t), \quad i = 2, 3, \ldots, m, \end{aligned} \tag{3.16}$$

which can be written as the following discrete time linear system:

$$\Pi(t+1) = P_\epsilon(\mathcal{D}')\Pi(t), \tag{3.17}$$

where $P_\epsilon(\mathcal{D}') = I - \epsilon\mathcal{L}(\mathcal{D}')$ is the Perron matrix of the digraph $\mathcal{D}'$ which is almost identical to the data association graph $\mathcal{D}$. The digraph $\mathcal{D}'$ can be obtained from the data association graph $\mathcal{D}$ as follows. The vertex set of $\mathcal{D}'$ is the same as the vertex set of $\mathcal{D}$, that is $\mathcal{V}_{\mathcal{D}'} = \mathcal{V} \times \{1, 2, \ldots, n\}$. There is an edge from (i, l) to (j, k) if and only if $\{i, j\} \in \mathcal{E}$, $(\widetilde{\Pi}_{ij})_{lk} = 1$ and $j \neq 1$.

Next, we find the range of values for the step size ϵ that results in convergence guarantees.

Proposition 3.4.1. *For $\epsilon \in (0, 1/d_{\max}(\mathcal{G}))$, the sequence $\{\Pi(t)\}_{t=0,1,2,\ldots}$, as generated by* (3.15)*, converges to a global minimizer of problem* (3.11) *from any initialization. Furthermore, the limit $\Pi = \lim_{t\to\infty} \Pi(t)$ of the sequence $\{\Pi(t)\}_{t=0,1,2,\ldots}$ satisfies $\Pi_1 = I$ and*

$$\sum_{j\in\mathcal{N}_i} (\Pi_i - \widetilde{\Pi}_{ij}\Pi_j) = 0, \;\; i = 2, 3, \ldots, m. \tag{3.18}$$

Proof. The convergence of the sequence $\{\Pi(t)\}_{t=0,1,\ldots}$ follows directly from the fact that $\lim_{k\to\infty} P_\epsilon(\mathcal{D}')^k$ exists (see proof of Theorem 3.4.5). Convergence to a stationary point of problem (3.11) follows directly from Lemma 2.4.1 and from the fact that the maximum eigenvalue of $\mathcal{L}(\mathcal{D})$ is less or equal to $2d_{\max}(\mathcal{G})$. Since the problem at hand is convex, a stationary point is globally optimal. Finally, the limit $\Pi = \lim_{t\to\infty} \Pi(t)$ of the sequence $\{\Pi(t)\}_{t=0,1,2,\ldots}$ satisfies $\Pi = P_\epsilon(\mathcal{D}')\Pi$ which yields $\mathcal{L}(\mathcal{D}')\Pi = 0$ which is equivalent to (3.18). The proof is complete. □

Next, we show that under perfect pairwise associations, the true labels $\{\Pi_i\}_{i\in\mathcal{V}}$ are recovered (up to a global permutation).

Lemma 3.4.2. *In the absence of outliers, we have that*

$$\lim_{t\to\infty} \Pi(t) = \begin{bmatrix} \Pi_{10}^T & \Pi_{20}^T & \cdots & \Pi_{m0}^T \end{bmatrix}^T \tag{3.19}$$

for some $\Pi_{10}, \Pi_{20}, \ldots, \Pi_{m0} \in \mathcal{P}_n$ such that $\Pi_{10} = I$ and $\widetilde{\Pi}_{ij} = \Pi_{i0}\Pi_{j0}^T$ for all $\{i, j\} \in \mathcal{E}$.

Proof. In the absence of outliers, we know that there exist permutations $\Pi_{10}, \ldots, \Pi_{m0}$ such

that $\Pi_{10} = I$ and $\widetilde{\Pi}_{ij} = \Pi_{i0}\Pi_{j0}^T$ for all $\{i,j\} \in \mathcal{E}$. Define $\Pi'_i \doteq \Pi_{i0}^{-1}\Pi_i$. Then, (3.18) can be equivalently written as follows

$$\begin{aligned} \Pi'_1 &= I, \\ \sum_{j\in\mathcal{N}_i} (\Pi'_i - \Pi'_j) &= 0, \quad i = 2,3,\ldots,m, \end{aligned} \tag{3.20}$$

or equivalently

$$(\mathcal{L}(\mathcal{G}') \otimes I_n)\Pi' = 0, \quad \Pi'_1 = I, \tag{3.21}$$

where the digraph $\mathcal{G}'$ is constructed from the sensor graph $\mathcal{G}$ as follows: $\mathcal{G}'$ has the same vertex set as $\mathcal{G}$ and for each edge $\{i,j\} \in \mathcal{E}$ we add two edges (i,j) and (j,i) in the edge set of $\mathcal{G}'$. The only constraint is that vertex 1 is allowed to have only outgoing edges. It can be easily seen that if $\mathcal{G}$ is connected then the digraph $\mathcal{G}'$ has a rooted out-branching as a subgraph. In this case, by Proposition 3.8 [87], the rank of $\mathcal{L}(\mathcal{G}')$ is $m-1$ and the nullspace of $\mathcal{L}(\mathcal{G}')$ is spanned by the vector of all ones. Therefore, in this case, (3.21) implies $\Pi'_1 = \Pi'_2 = \ldots = \Pi'_m = I$. The proof is complete. □

Remark 3.4.3 (Reduction to consensus)**.** *In the absence of outliers, we know that there exist permutations* $\Pi_{10},\ldots,\Pi_{m0}$ *such that* $\Pi_{10} = I$ *and* $\widetilde{\Pi}_{ij} = \Pi_{i0}\Pi_{j0}^T$ *for all* $\{i,j\} \in \mathcal{E}$*. Define* $\Pi'_i \doteq \Pi_{i0}^{-1}\Pi_i$*. Then, the update rule* (3.16) *can be equivalently written*

$$\begin{aligned} \Pi'_1(t+1) &= \Pi'_1(t) = I, \\ \Pi'_i(t+1) &= (1 - \epsilon\,|\mathcal{N}_i|)\Pi'_i(t) + \epsilon \sum_{j\in\mathcal{N}_i} \Pi'_j(t), \quad i = 2,3,\ldots,m. \end{aligned} \tag{3.22}$$

Next, we analytically compute the limit of the sequence $\{\Pi(t)\}_{t=0,1,2,\ldots}$. First, we write the Perron matrix $P_\epsilon(\mathcal{D}') = I - \epsilon\mathcal{L}(\mathcal{D}')$ of digraph $\mathcal{D}'$ in block form as follows

$$P_\epsilon(\mathcal{D}') = \begin{bmatrix} I_n & 0 \\ P_{21} & P_{22} \end{bmatrix}, \tag{3.23}$$

where $P_{21} \in \mathbb{R}^{(m-1)n \times n}$ and $P_{22} \in \mathbb{R}^{(m-1)n \times (m-1)n}$.

Lemma 3.4.4. *The matrix P_{22} as defined in* (3.23) *satisfies*

$$\lim_{k \to \infty} P_{22}^k = 0. \tag{3.24}$$

Proof. We observe that $P_\epsilon(\mathcal{D}')$ is the state transition matrix of a Markov chain whose first n states are absorbing and, given that $\mathcal{G}$ is connected, the remaining states are transient. The desired result follows directly from the results of Section 3.8 of [33] . □

Theorem 3.4.5. *Assume that the sensor graph $\mathcal{G}$ is connected and $\epsilon \in (0, 1/d_{\max}(\mathcal{G}))$. Then, we have*

$$\lim_{k \to \infty} P_\epsilon(\mathcal{D}')^k = \begin{bmatrix} I & 0 \\ (I - P_{22})^{-1} P_{21} & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ [\mathcal{L}(\mathcal{D})]_{22}^{-1} [A(\mathcal{D})]_{21} & 0 \end{bmatrix}. \tag{3.25}$$

As a consequence, for $\Pi_1(0) = I$ and any $\Pi_2(0), \ldots, \Pi_m(0)$, we get

$$\lim_{t \to \infty} \Pi(t) = \begin{bmatrix} I \\ [\mathcal{L}(\mathcal{D})]_{22}^{-1} [A(\mathcal{D})]_{21} \end{bmatrix}. \tag{3.26}$$

Proof. By induction on k, we have for all positive integers k that

$$P_\epsilon(\mathcal{D}')^k = \begin{bmatrix} I & 0 \\ (I + P_{22} + \ldots + P_{22}^k) P_{21} & P_{22}^k \end{bmatrix} \tag{3.27}$$

Moreover, since $\lim_{k \to \infty} P_{22}^k = 0$, it follows that (see Lemma 3.10 of [33])

$$\lim_{k \to \infty} (I + P_{22} + \ldots + P_{22}^k) = (I - P_{22})^{-1}. \tag{3.28}$$

Since $P_{22} = I - \epsilon[\mathcal{L}(\mathcal{D})]_{22}$ and $P_{21} = [A(\mathcal{D})]_{21}$, it follows that $(I - P_{22})^{-1} P_{21} = [\mathcal{L}(\mathcal{D})]_{22}^{-1} [A(\mathcal{D})]_{21}$, which concludes the proof. □

Finally, after the convergence of the proposed distributed protocol to a set of doubly stochastic matrices $\{\widetilde{\Pi}_i\}_{i\in\mathcal{V}}$, we project the solution onto the set of permutation matrices by solving:

$$\begin{aligned} \underset{\Pi_i}{\text{maximize}} \quad & \langle \widetilde{\Pi}_i, \Pi_i \rangle \\ \text{subject to} \quad & \Pi_i \in \mathcal{P}_n. \end{aligned} \tag{3.29}$$

The above problem can be solved efficiently using the Hungarian algorithm [67] in $O(n^3)$ time. Naturally, we have the following lemma.

Lemma 3.4.6. *The reconstructed pairwise associations defined by* $\Pi_{ij} \doteq \Pi_i \Pi_j^T$ *for all* $\{i,j\} \in \mathcal{E}$ *are consistent.*

3.4.3 KKT conditions

At this point, we derive the optimality conditions for problem (3.11) along with several properties of problem (3.11). First of all, since problem (3.11) is a convex optimization problem, the Karush-Kuhn-Tucker (KKT) conditions [18] are necessary and sufficient for optimality. The Lagrangian of problem (3.11) is given by

$$\begin{aligned} L(\{\Pi_i, \mu_i, \nu_i, Z_i\}_{i\in\mathcal{V}}, Y_1) = \frac{1}{2}\operatorname{tr}(\Pi^T \mathcal{L}(\mathcal{D})\Pi) + \operatorname{tr}(Y_1^T(\Pi_1 - I)) \\ + \sum_{i\in\mathcal{V}} \left(\mu_i^T(\Pi_i \mathbf{1} - \mathbf{1}) + \nu_i^T(\Pi_i^T \mathbf{1} - \mathbf{1}) - \operatorname{tr}(\Pi_i^T Z_i)\right) \end{aligned} \tag{3.30}$$

where for all $i \in \mathcal{V}$, $\mu_i \in \mathbb{R}^n$ are the Lagrange multipliers associated with the constraints $\Pi_i \mathbf{1} = \mathbf{1}$, $\nu_i \in \mathbb{R}^n$ are the Lagrange multipliers associated with the constraints $\Pi_i^T \mathbf{1} = \mathbf{1}$, Y_1 is the Lagrange multiplier associated with the constraint $\Pi_1 = I$ and $Z_i \in \mathbb{R}^{n\times n}$ are the Lagrange multipliers associated with the nonnegativity constraint $\Pi_i \geq 0$. The KKT conditions consist of the condition that the gradient of the Lagrangian with respect to Π_i must vanish, the feasibilty of Π_i, the (dual) feasibilty of Z_i and complementary slackness.

The KKT conditions for problem (3.11) are summarized below:

$$\sum_{j\in\mathcal{N}_i}(\Pi_i - \widetilde{\Pi}_{ij}\Pi_j) + \mu_i\mathbf{1}^T + \mathbf{1}\nu_i^T - Z_i = 0, \tag{3.31}$$

$$\sum_{j\in\mathcal{N}_1}(\Pi_1 - \widetilde{\Pi}_{1j}\Pi_j) + Y_1 = 0, \tag{3.32}$$

$$\Pi_i\mathbf{1} = \mathbf{1}, \tag{3.33}$$

$$\Pi_i^T\mathbf{1} = \mathbf{1}, \tag{3.34}$$

$$\Pi_1 = I, \tag{3.35}$$

$$\Pi_i \geq 0, \tag{3.36}$$

$$Z_i \geq 0, \tag{3.37}$$

$$\Pi_i \odot Z_i = 0, \tag{3.38}$$

for all $i = 2, \ldots, m$.

Lemma 3.4.7. *The limit of the sequence generated by protocol* (3.15) *satisfies the KKT conditions.*

Proof. It can be easily checked that the following values for the Lagrange multipliers satisfy the KKT conditions:

$$\mu_i = \nu_i = 0, \quad Z_i = 0, \quad Y_1 = -\sum_{j\in\mathcal{N}_1}(I - \widetilde{\Pi}_{1j}\Pi_j). \tag{3.39}$$

□

3.5 A nonconvex distributed optimization approach

In this section, we present a different approach to approximately solving problem (3.9). We propose to add a regularizer that penalizes deviations from the set of permutation matrices.

Let ψ denote the real-valued function defined by

$$\psi(X) = -\frac{1}{2}\operatorname{tr}(X^T X) = -\frac{1}{2}\|X\|_F^2. \tag{3.40}$$

Next, we show that the set of local and global minimizers of ψ coincides with $\mathcal{P}_n$, the set of $n \times n$ permutation matrices.

Lemma 3.5.1. *We have that*

(i) The set of global minima of ψ over the set $\mathcal{D}_n$ of $n \times n$ doubly stochastic matrices is exactly $\mathcal{P}_n$, namely the set of $n \times n$ permutation matrices.

(ii) A doubly stochastic matrix $X \in \mathcal{D}_n$ is a local minimum of ψ if and only if $X \in \mathcal{P}_n$, or in other words, ψ does not have local minima that are not global.

Proof. Let x_i^T denote the ith row of a $n \times n$ doubly stochastic matrix X. We note that $\|x_i\|_2^2 \leq 1$ and $\|x_i\|_2^2 = 1$ if and only if x_i contains exactly on element equal to 1 and all other equal to 0. Therefore, ψ is minimized or equivalently $\|X\|_F^2$ is maximized over the set of doubly stochastic matrices when all rows of X have norm 1, that is when X is a permutation matrix. So, the first part has been proved. For the second part, we observe that ψ is a strictly concave function. It is a well-known fact from convex analysis [112], that the minima (local and global) of a strictly concave function over a bounded polyhedron must be extrema of the the polyhedron. Since the extrema of the polyhedron of doubly stochastic matrices are permutation matrices, the second part trivially follows. □

We formulate permutation synchronization as the following optimization problem over the set of $n \times n$ doubly stochastic matrices:

$$\underset{\{\Pi_i \in \mathcal{D}_n\}_{i \in \mathcal{V}}}{\text{minimize}} \quad \phi_\gamma\left(\{\Pi_i\}_{i \in \mathcal{V}}\right) \doteq -(1-\gamma) \sum_{\{i,j\} \in \mathcal{E}} \operatorname{tr}(\Pi_i^T \widetilde{\Pi}_{ij} \Pi_j) - \frac{\gamma}{2} \sum_{i \in \mathcal{V}} \operatorname{tr}(\Pi_i^T \Pi_i) \tag{3.41}$$

where $0 < \gamma < 1$. We propose to numerically solve problem (3.41) by projected gradient descent with constant step size ϵ. That is, at every time step t, each Π_i, $i \in \mathcal{V}$, is updated

as follows:

$$\operatorname{grad}_{\Pi_i} \phi_\gamma \left(\{\Pi_i(t)\}_{i\in\mathcal{V}}\right) = -(1-\gamma) \sum_{\{i,j\}\in\mathcal{E}} \widetilde{\Pi}_{ij}\Pi_j(t) - \gamma\Pi_i(t) \tag{3.42}$$

$$\Pi_i(t+1) = \operatorname{Proj}_{\mathcal{D}_n}\left(\Pi_i(t) - \epsilon \operatorname{grad}_{\Pi_i} \phi_\gamma \left(\{\Pi_i(t)\}_{i\in\mathcal{V}}\right)\right). \tag{3.43}$$

Next, we derive conditions on the step size ϵ that guarantee convergence of the proposed projected gradient descent rule to a stationary point of problem (3.41). The maximum allowed step size depends on the choice of γ and on the spectrum of the Hessian of the objective ϕ_γ. A uniform upper bound on the maximum eigenvalue of the Hessian of ϕ_γ is presented in the following lemma.

Lemma 3.5.2. *The maximum eigenvalue of the Hessian of the objective can be uniformly upper bounded by*

$$\lambda_{\max}(\operatorname{Hess}\phi_\gamma) \leq -\gamma + (1-\gamma)d_{\max}(\mathcal{G}). \tag{3.44}$$

The proof of the above lemma is a straightforward application of Gershgorin discs theorem [51]. An immediate consequence of the above lemma, is that the objective becomes concave for some values of γ. This is summarized in the following corollary.

Corollary 3.5.3. *Let*

$$\gamma_{\max} = \frac{d_{\max}(\mathcal{G})}{1 + d_{\max}(\mathcal{G})}. \tag{3.45}$$

Then, for $\gamma > \gamma_{\max}$, the objective ϕ_γ is strictly concave.

The convergence properties of the proposed method and the corresponding range of the parameters γ and ϵ are summarized in the following theorem.

Theorem 3.5.4. *For $0 < \gamma < \gamma_{\max}$. every limit point of the sequence generated by the proposed update rule asymptotically is a stationary point of problem* (3.41) *for any step size ϵ satisfying*

$$0 < \epsilon < \frac{2}{-\gamma + (1-\gamma)d_{\max}(\mathcal{G})}. \tag{3.46}$$

The overall proposed approach for permutation synchronization is presented in Algorithm 1. We get an initial solution via the convex relaxation method of the previous section. Then, we solve the regularized problem (3.41) twice. Once for $\gamma \in (0, \gamma_{\max})$ and one for $\gamma \in (\gamma_{\max}, 1)$ that will almost surely return permutations due to the concavity of the objective. Experimental results for MatchDPS are included in the next chapter along with the results of the more general approach of that chapter.

Algorithm 1 MatchDPS - Distributed Permutation Synchronization

Input: Pairwise associations $\{\widetilde{\Pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$.
Ouput: Labels $\{\Pi_i\}_{i\in\mathcal{V}}$, consistent associations $\{\Pi_{ij} \doteq \Pi_i\Pi_j^T\}_{\{i,j\}\in\mathcal{E}}$
Pick a step size $\epsilon \in (0, 1/d_{\max}(\mathcal{G}))$.
for $t = 0, 1, 2, \ldots, T$ **do**
  $\Pi_1(t+1) = I$
  $\Pi_i(t+1) = \Pi_i(t) - \epsilon \sum_{j\in\mathcal{N}_i}(\Pi_i(t) - \widetilde{\Pi}_{ij}\Pi_j(t))$, $i = 2, 3, \ldots, m$.
end for
Pick a $\gamma \in (0, \gamma_{\max})$ and a step size $\epsilon \in (0, 2/(-\gamma + (1-\gamma)d_{\max}(\mathcal{G}))$
for $t = 0, 1, 2, \ldots, T$ **do**
  $W_i(t) = (1-\gamma)\sum_{\{i,j\}\in\mathcal{E}} \widetilde{\Pi}_{ij}\Pi_j(t) + \gamma\Pi_i(t)$, $i = 1, 2, \ldots, m$.
  $\Pi_i(t+1) = \mathrm{Proj}_{\mathcal{D}_n}(\Pi_i(t) + \epsilon W_i(t))$, $i = 1, 2, \ldots, m$.
end for
Pick a $\gamma \in (\gamma_{\max}, 1)$ and a step size $\epsilon > 0$.
for $t = 0, 1, 2, \ldots, T$ **do**
  $W_i(t) = (1-\gamma)\sum_{\{i,j\}\in\mathcal{E}} \widetilde{\Pi}_{ij}\Pi_j(t) + \gamma\Pi_i(t)$, $i = 1, 2, \ldots, m$.
  $\Pi_i(t+1) = \mathrm{Proj}_{\mathcal{D}_n}(\Pi_i(t) + \epsilon W_i(t))$, $i = 1, 2, \ldots, m$.
end for

3.6 Conclusions

In this chapter, we proposed two novel and fully decentralized approaches for the problem of permutation synchronization along with a combined algorithm. The first approach consisted of a convex relaxation of the permutation synchronization problem which we solve by a consensus-like algorithm with global convergence guarantees. The second approach was a nonconvex regularized relaxation of permutation synchronization, numerically solved with projected gradient descent. We rigorously analyzed the convergence properties of the proposed numerical methods and explored the connection with consensus algorithms.

Chapter 4

Distributed consistent multiway matching

4.1 Introduction

In this chapter, we present a novel distributed optimization approach to the problem of multiway matching under partially observable pairwise associations. We show that the proposed method provably converges and produces, by construction, consistent associations. The proposed approach, albeit decentralized, demonstrates performance comparable to the state of the art centralized approaches.

The remainder of this chapter is structured as follows. A formalization of multiway matching is the subject of Section 4.2 followed by the proposed approach in Section 4.3. Finally, experimental evaluation and comparison with existing approaches are included in Section 4.4.

4.2 Problem statement

In this section, we formalize the problem of consistent multiway matching for the case of occlusions, that is, we assume that each collection of features (sensor) contains (observes) a subset of the total available features. We assume there are m collections and each collection i has n_i features and let $N = \sum_{i=1}^{m} n_i$. Let K denote the total number of distinct features, termed size of the "universe of features" [23, 156], with $n_i \leq K$ for all $i = 1, \ldots, m$. We have that $(\widetilde{X}_{ij})_{lk} = 1$ if the kth feature in collection j corresponds to the lth features in collection i. As in the previous chapter, the availability of pairwise measurement $\widetilde{X}_{ij}$ defined a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Then, assuming that $\mathcal{G}$ is connected, the set of pairwise correspondences $\{\widetilde{X}_{ij}\}_{i,j\in\mathcal{E}}$ is *(cycle) consistent* if and only if

$$\widetilde{X}_{ij} = X_i X_j^T, \quad \forall \{i, j\} \in \mathcal{E}, \tag{4.1}$$

for some partial permutation matrices $X_1, X_2, \ldots, X_m$ such that each $X_i \in \mathcal{P}_{n_i,K}$. Intuitively, $X_i \in \mathcal{P}_{n_i,K}$ is the matrix representation of a (partial) permutation map $\pi_i : [n_i] \to [K]$ from the labels of collection i to some global labels, termed the "universe of features" [23, 156]. Then, we state the problem we address in this chapter.

Problem Statement 4.2.1. *Given pairwise associations $\{\widetilde{X}_{ij}\}_{\{i,j\}\in\mathcal{E}}$, estimate consistent pairwise associations $\{X_{ij}\}_{i,j\in\mathcal{E}}$ that are close to the given ones in a decentralized fashion. Equivalently, given, possibly erroneous, pairwise associations $\{\widetilde{X}_{ij}\}_{\{i,j\}\in\mathcal{E}}$, find partial permutations $\{X_i\}_{i\in\mathcal{V}}$, such that*

$$\widetilde{X}_{ij} \approx X_i X_j^T, \quad \forall \{i, j\} \in \mathcal{E}. \tag{4.2}$$

4.3 Problem formulation

In this section, we present our proposed approach. Intuitively, we formulate consistent multiway matching as an optimization problem on the set of partial permutation matrices. Due to computational intractability, we relax the domain of the optimization problem from partial permutation matrices to partial doubly stochastic matrices and we add a regularizer to the objective to penalize deviations from the set of partial permutations. Then, we solve the relaxed problem by distributed projected gradient descent with constant step size. Finally, we provide a bound on the maximum step size for which the proposed method is guaranteed to converge to a stationary point of the relaxed problem.

Based on the problem statement of the previous section, we formulate consistent multiway matching as the following combinatorial optimization problem:

$$\begin{aligned} \underset{\{X_i\}_{i\in\mathcal{V}}}{\text{minimize}} \quad & \sum_{\{i,j\}\in\mathcal{E}} \|\widetilde{X}_{ij} - X_i X_j^T\|_F^2 \\ \text{subject to} \quad & X_i \in \mathcal{P}_{n_i,K}, \qquad \forall i \in \mathcal{V}. \end{aligned} \tag{4.3}$$

Existing works [53, 23, 156] for multiway matching attempt to find a low-rank positive definite matrix $\mathbf{X} \in \mathbb{R}^{N\times N}$ that satisfies the consistency constraints and is close to the measured one. However, these approaches are centralized and do not scale well with the number of views. In contrast to these approaches that optimize over the pairwise correspondences, we propose to solve directly for the labels $\{X_i\}_{i\in\mathcal{V}}$. The advantages of optimizing directly in the label space are twofold. First, the resulting problem can be solved in a decentralized fashion using distributed projected gradient descent, as we will shortly see. Second, the dimension of the domain of the optimization problem reduces significantly from N^2 to $NK \ll N^2$.

There are two main reasons as to why problem (4.3) is challenging. The first challenge is the combinatorial hardness due to the permutation constraints along with the nonlinearity

of the objective. This is customarily remedied by relaxing the domain of the problem from permutations to doubly stochastic. This is, indeed, the path we follow. The second challenge is that even if the domain of the problem was convex, the objective would still be nonconvex. Therefore, we expect that the proposed optimization based approach returns only local minimizers.

At this point, we introduce a regularizer that penalizes deviations from the set of partial permutation matrices. Its definition and properties are summarized in the following lemma.

Lemma 4.3.1. *Let* $\phi_i : \mathcal{D}_{n_i,K} \to \mathbb{R}$ *defined by*

$$\phi_i(X_i) = (1/4)\|I - X_i X_i^T\|_F^2. \tag{4.4}$$

Then,

(a) $\phi_i(X_i) \geq 0$ *for all* $X_i \in \mathcal{D}_{n_i,K}$,

(b) $\phi_i(X_i) = 0$ *if and only if* $X_i \in \mathcal{P}_{n_i,K}$.

The proof of Lemma 4.3.1 is fairly straightforward and therefore, omitted. Let

$$\phi_{ij}(X_i, X_j) = (1/2)\|\widetilde{X}_{ij} - X_i X_j^T\|_F^2, \tag{4.5}$$

and

$$\phi(\{X_i\}_{i\in\mathcal{V}}) = \gamma \sum_{i\in\mathcal{V}} \phi_i(X_i) + (1-\gamma) \sum_{\{i,j\}\in\mathcal{E}} \phi_{ij}(X_i, X_j), \tag{4.6}$$

for some $\gamma \in (0,1)$. Typical value of γ is $\gamma = 1/2$. Based on Lemma 4.3.1, problem (4.3) is equivalent to

$$\begin{aligned} \underset{\{X\}_{i\in\mathcal{V}}}{\text{minimize}} \quad & \phi(\{X_i\}_{i\in\mathcal{V}}) \\ \text{subject to} \quad & X_i \in \mathcal{P}_{n_i,K}, \qquad \forall i \in \mathcal{V}, \end{aligned} \tag{4.7}$$

which is in turn is relaxed to the following problem

$$\begin{aligned} \underset{X}{\text{minimize}} \quad & \phi(\{X_i\}_{i\in\mathcal{V}}) \\ \text{subject to} \quad & X_i \in \mathcal{D}_{n_i,K}, \qquad \forall i \in \mathcal{V}. \end{aligned} \tag{4.8}$$

which is of the form (2.28).

The proposed approach is summarized in Algorithm 2. The gradient of the objective with respect to each X_i, can be computed by

$$\operatorname{grad}_{X_i} \phi(\{X_i\}_{i\in\mathcal{V}}) = \gamma(X_i X_i^T - I)X_i + (1-\gamma)\sum_{j\in\mathcal{N}_i}(X_i X_j^T - \widetilde{X}_{ij})X_j. \tag{4.9}$$

At every iteration we need to project the current estimate X_i^k onto $\mathcal{D}_{n_i,K}$. For this purpose, we propose an ADMM algorithm in Appendix A. Finally, to obtain a permutation matrix, we employ the Hungarian algorithm [67].

Algorithm 2 MatchDGD

Input: Pairwise associations $\{\widetilde{X}_{ij}\}_{\{i,j\}\in\mathcal{E}}$, initial $\{X_i^0\}_{i\in\mathcal{V}}$, step size $\epsilon > 0$
Ouput: Partial permutations $\{X_i\}_{i\in\mathcal{V}}$, consistent associations $\{X_{ij} \doteq X_i X_j^T\}_{i,j\in\mathcal{V}}$
for $k \in \{0, 1, \ldots, K\}$ **do**
  $W_i^k = -\operatorname{grad}_{X_i} \phi(\{X_i^k\}_{i\in\mathcal{V}})$
  $X_i^{k+1} = \operatorname{Proj}_{\mathcal{D}_{n_i,K}}\left(X_i^k + \epsilon W_i^k\right)$
end for
Project X_i onto $\mathcal{P}_{n_i,K}$ using Hungarian algorithm.

Finally, we derive a bound on the maximum step size for the projected gradient descent part of Algorithm 2. First, we need the following two lemmata.

Lemma 4.3.2. *For any $U_i \in \mathbb{R}^{n_i\times K}$ and $X_i \in \mathcal{D}_{n_i\times K}$, we have*

$$\langle U_i, \operatorname{Hess}\phi_i(X_i)[U_i]\rangle \leq 2\langle U_i, U_i\rangle. \tag{4.10}$$

Lemma 4.3.3. *If $\widetilde{X}_{ij} \geq 0$, $\widetilde{X}_{ij}\mathbf{1} \leq \mathbf{1}$, $\widetilde{X}_{ij}^T\mathbf{1} \leq \mathbf{1}$, then for any $(U_i, U_j) \in \mathbb{R}^{n_i\times K} \times \mathbb{R}^{n_j\times K}$, we have*

$$\langle (U_i, U_j), \operatorname{Hess}\phi_{ij}(X_i, X_j))[(U_i, U_j)]\rangle \leq 3\left(\|U_i\|_F^2 + \|U_j\|_F^2\right). \tag{4.11}$$

For detailed proofs, we refer the reader to Appendix C.1 and C.2. A corollary of the two above lemmata and Proposition 2.4.2 now follows.

Corollary 4.3.4. *For any $X = [X_1^T \cdots X_m^T]^T$, each $X_i \in \mathcal{D}_{n_i,K}$, and $U \in \mathbb{R}^{N\times K}$, we have*

$$\langle U, \operatorname{Hess} \phi(X)[U]\rangle \leq (2\gamma + 3(1-\gamma)d_{max}(\mathcal{G}))\,\langle U, U\rangle. \tag{4.12}$$

Next, we state the maximum allowed step size that guarantees convergence to a stationary point of the relaxed problem (4.8).

Proposition 4.3.5. *Given a step size $\epsilon \in (0, \epsilon_{\max})$, where*

$$\epsilon_{\max} = \frac{2}{2\gamma + 3(1-\gamma)d_{max}(\mathcal{G})}, \tag{4.13}$$

the every limit point of the sequence generated by the proposed projected gradient descent method for solving problem (4.8) *is a stationary point.*

The proof of the above proposition is a straightforward application of Corollary 4.3.4 and Lemma 2.4.1. Next, we include experimental results on both synthetic and real data.

4.4 Experiments

4.4.1 Synthetic data

First, we evaluate the performance of the proposed method using synthetic data. We compare the two proposed methods, MatchDPS and MatchDGD, against the spectral method [101], the semidefinite programming approach MatchLift [23] and the rank-minimization approach MatchALS [156].

Experiment in permutation synchronization. We fix the number of features in each collection as $n = 20$. We vary the number of collections from 5 to 50 and the pairwise

matching error rate from 0.1 to 0.9. To quantify the performance of each method, we use the F-score, i.e. the harmonic mean of precision p and recall r, given by $F = \frac{2pr}{p+r}$. The output error we plot in Fig. 4.1 is then $1 - F$.

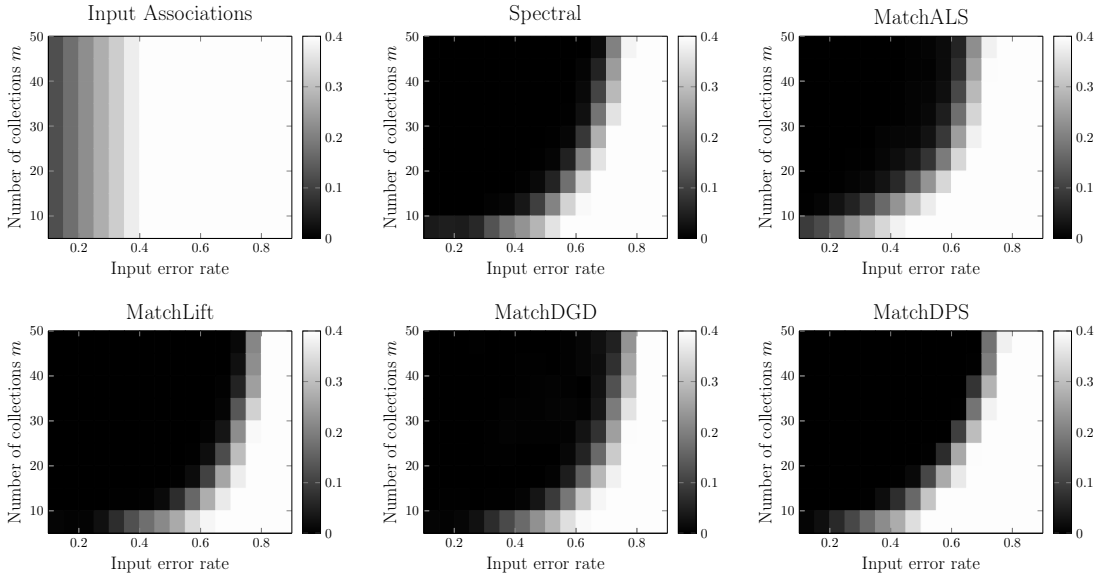


Figure 4.1: Comparison of the proposed approach with the spectral method [101], MatchLift [23] and MatchALS [156], under various input error rates and number of collections m. The darker the area the smaller the output error rate.

4.4.2 Real data

Multiimage feature matching. Next, we compare the same methods in a multiimage matching scenario. First, we use the CMU Hotel[1] and House[2] datasets, Although simple, these datasets have been used in all prior works. The House sequence contains 111 images and the Hotel sequence contains 101. We also include results on the first halfs of the two sequences. In addition, we evaluate using the Affine Covariant Regions Datasets [3] (Wall, UBC, Bikes, Leuven, Trees, Graffiti, Bark) which consist of sequences of 6 images with significant overlap but with viewpoint, scale and image quality variability. To obtain

[1] `http://vasc.ri.cmu.edu/idb/html/motion/hotel/`
[2] `http://vasc.ri.cmu.edu//idb/html/motion/house/`
[3] `http://www.robots.ox.ac.uk/~vgg/data/data-aff.html`

pairwise correspondences, we extract SIFT descriptors [81] and use SIFT matching.

Dataset	MatchDPS	MatchDGD	MatchALS	MatchLift	Spectral	Input
Hotel101	0.865	0.900	0.933	0.747	0.693	0.726
House111	0.954	0.942	0.960	0.822	0.824	0.793
Hotel51	0.961	0.950	0.906	0.929	0.843	0.852
House56	1	0.993	0.994	0.986	0.970	0.913
Wall	-	0.505	0.543	0.558	0.467	0.519
UBC	-	0.864	0.873	0.688	0.747	0.837
Bikes	-	0.859	0.862	0.677	0.706	0.836
Leuven	-	0.868	0.853	0.789	0.712	0.827
Trees	-	0.663	0.688	0.718	0.541	0.648
Graffiti	-	0.443	0.465	0.464	0.375	0.456
Bark	-	0.389	0.361	0.420	0.342	0.376

Table 4.1: Comparison of the competing methods on the CMU and on Affine Covariant Regions datasets. We report the F-score for each method.

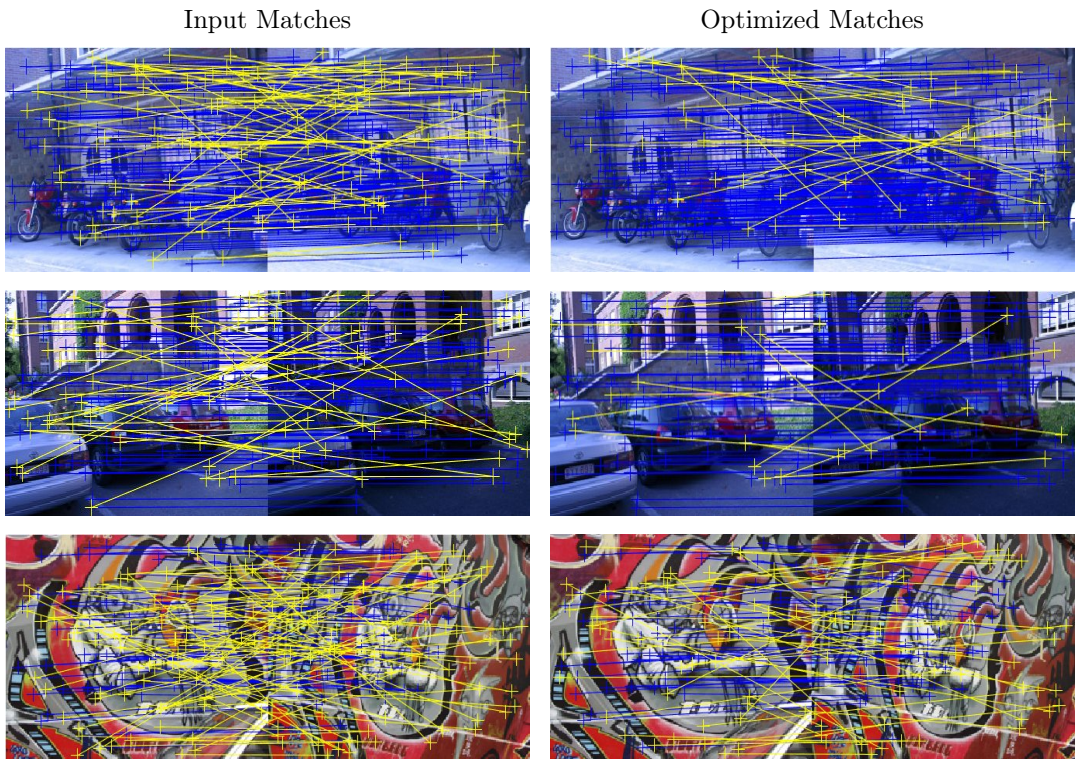


Figure 4.2: Qualitative results on the Affine Covariant Regions datasets obtained by MatchDGD. Blue and yellow lines correspond to inliers and outliers, respectively.

Intraclass semantic keypoint matching. We use the WILLOW Object Class dataset [24] which provides images of five object classes (Car, Face, Motorbike, Bottles, Duck) as well as annotated keypoints (10) corresponding to several discriminative parts for each class. Each class contains at least 40 images with different instances. Our goal is to find the correspondences of keypoints between images in the same class. The main difficulty in this dataset arises from the large variety of object appearance, which makes traditional geometric descriptors like SIFT inapplicable. To make use of high-level semantic and structural information, we use the hypercolumns [44] extracted from AlexNet [66] (pretrained on ImageNet [30]) as the keypoint descriptors, and then run graph matching [25] for each pair of images to obtain the initial pairwise correspondences.

Dataset	MatchDPS	MatchDGD	MatchALS	MatchLift	Spectral	Input
Car	0.965	0.941	0.931	0.961	0.914	0.767
Duck	0.797	0.790	0.767	0.805	0.793	0.603
Bottles	0.967	0.967	0.970	0.970	0.976	0.885
Motorbike	0.955	0.950	0.931	0.955	0.950	0.716
Face	1	1	1	1	1	0.998

Table 4.2: Comparison of the competing methods on the WILLOW Object Class datasets. We report the F-score for each method. All of the competing methods have similar performance on these datasets and significantly improve the accuracy of the input pairwise correspondences. This evaluation demonstrates the effectiveness of multiway matching in the presence of sufficient number of images in a dataset.

4.5 Conclusions

In this chapter, we proposed a distributed optimization approach to consistent multiway matching. We rigorously analyzed its convergence properties and provided experimental evidence supporting that the proposed approach has performance comparable with the state of the art centralized approaches.

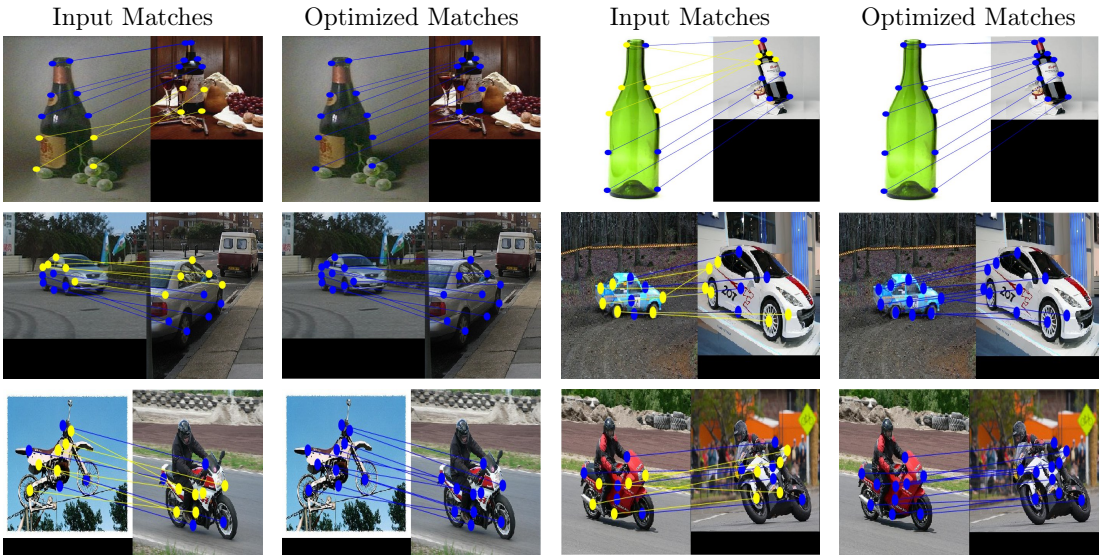


Figure 4.3: Qualitative results on the WILLOW Object Class dataset obtained by the proposed method MatchDGD. Blue and yellow lines correspond to inliers and outliers, respectively.

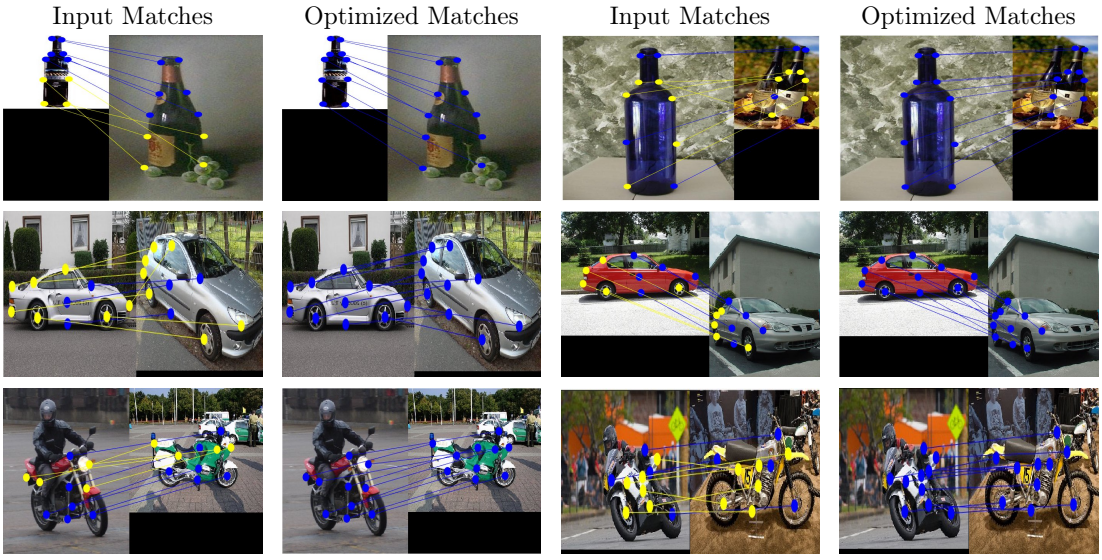


Figure 4.4: Qualitative results on the WILLOW Object Class dataset obtained by the proposed method MatchDPS. Blue and yellow lines correspond to inliers and outliers, respectively.

Chapter 5

Distributed eigenvector computation

5.1 Introduction

In this chapter, we propose a fully decentralized method for computing the k smallest (or largest) eigenvalues and eigenvectors of a matrix representing a graph using ideas from dynamical systems theory. The idea of using dynamical systems to perform matrix decompositions is not new. In his seminal paper, Brockett [19] posed the matrix diagonalization problem as a gradient flow on the orthogonal group. Our work can be viewed as a generalization of Brockett's work to the distributed case. In particular, we propose a distributed dynamical system for computing the invariant subspaces of a matrix with the sparsity pattern of a communication graph.

In contrast to the approaches based on the Orthogonal Iteration, the orthogonality constraints that must be satisfied among the eigenvectors of a symmetric matrix are asymptotically satisfied by the dynamical system herein proposed. Thus, the main computational burden of the Decentralized Orthogonal Iteration, namely, the orthonormalization step, is not present in our approach.

5.2 Problem statement

In this section, we formally define the problems we deal with in this work. In a nutshell, given a matrix that is distributed across a network, we address the problem of distributedly estimating both the $k \geq 1$ smallest eigenvalues and the associated eigenvectors. First, we give a more precise definition for the case $k = 1$ and then, for the case $k > 1$.

Problem Statement 5.2.1 (Distributed computation of a single eigenvector)**.** *We are given a network of n agents whose interactions are captured by an undirected and connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ along with a symmetric matrix $A \in \mathbb{R}^{n \times n}$ that is distributed across $\mathcal{G}$ and has eigendecomposition given by*

$$A = \sum_{i=1}^{n} \lambda_i v_i v_i^T, \tag{5.1}$$

with

$$\lambda_{\min}(A) = \lambda_1 < \lambda_2 < \cdots < \lambda_n = \lambda_{\max}(A).$$

Let $x(t) = [x_1(t) \cdots x_n(t)]^T \in \mathbb{R}^n$, where each component x_i is maintained by agent i, denote the collective estimate of the eigenvector v_1. Our goal is to design a local update rule so that all agents asymptotically estimate the smallest eigenvalue λ_1 and $x(t)$ asymptotically converges to v_1 (up to a sign flip), that is,

$$\lim_{t \to \infty} x(t) = \pm v_1. \tag{5.2}$$

Next, we give a precise definition for the problem of distributedly computing $k > 1$ smallest eigenvalues and the associated eigenvectors.

Problem Statement 5.2.2 (Distributed computation of k smallest eigenvectors)**.** *We are given a network of n agents whose interactions are captured by a connected and undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ along with a symmetric matrix $A \in \mathbb{R}^{m \times m}$ that is distributed across $\mathcal{G}$ and has eigendecomposition given by*

$$A = \sum_{i=1}^{m} \lambda_i v_i v_i^T, \tag{5.3}$$

with

$$\lambda_{\min}(A) = \lambda_1 < \lambda_2 < \cdots < \lambda_m = \lambda_{\max}(A).$$

Let $X(t) = [X_1(t)^T \cdots X_n(t)^T]^T \in \mathbb{R}^{m\times k}$*, where each* $X_i \in \mathbb{R}^{m_i\times k}$ *is maintained by agent* i*,* $m = \sum_{i=1}^n m_i$ *and* $k > 1$ *is the number of eigenvalues and associated eigenvectors we want to compute. Our goal is to design a local update rule so that all agents asymptotically estimate* $\{\lambda_i\}_{i=1}^k$ *and* $X(t)$ *asymptotically converges to the associated eigenvectors (up to a sign flip), that is,*

$$\lim_{t\to\infty} X(t) = [\pm v_1, \pm v_2, \ldots, \pm v_k]. \tag{5.4}$$

The first problem is addressed in Section 5.3 while the second problem is addressed in Section 5.4.

5.3 Single eigenvector computation

In this section, we address the problem of distributedly computing the smallest eigenvalue and the associated eigenvector of a given matrix $A \in \mathbb{R}^{n\times n}$ that is distributed across a network. Specifically, we have a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = n$ agents. We denote by $x(t) = [x_1(t) \cdots x_n(t)]^T \in \mathbb{R}^n$ the collective estimate of the eigenvector associated with the smallest eigenvalue of the given matrix at time $t \geq 0$, with each agent i maintaining only the ith component of x, namely x_i.

The main difficulty of the problem at hand is the decentralization of the computations, meaning that each agent should use information only from its neighbors. However, the unit-norm constraint on the estimated eigenvector is global, in the sense that it involves information from the entire group of agents. To achieve decentralization of the estimation process, we propose that all agents track two collective time-varying quantities, namely $1 - x(t)^T x(t)$ and $x(t)Ax(t)$, by an approach inspired by dynamic consensus [121]. We

observe that these two quantities can be written as the average of local quantities as follows:

$$1 - x(t)^T x(t) = \frac{1}{n}\sum_{i=1}^{n}(1 - nx_i(t)^2), \tag{5.5}$$

$$x(t)^T A x(t) = \frac{1}{n}\sum_{i=1}^{n} nx_i(t)a_i^T x(t), \tag{5.6}$$

where a_i^T denotes the ith row of A. Note that we have assumed that the number n of agents is known to every agent. Let $z_i(t) \in \mathbb{R}$ denote the estimate of $1 - x(t)^T x(t)$ by agent i and $w_i(t) \in \mathbb{R}$ denote the estimate of $x(t)^T A x(t)$ by agent i. Based on the aforementioned observation and inspired by dynamics consensus estimators [121], we propose the following dynamics for the two sets of estimators $\{z_i(t)\}_{i=1}^n$ and $\{w_i(t)\}_{i=1}^n$:

$$\dot{z}_i = \sum_{j\in\mathcal{N}_i}(z_j - z_i) - 2nx_i\dot{x}_i, \tag{5.7}$$

$$\dot{w}_i = \sum_{j\in\mathcal{N}_i}(w_j - w_i) + 2n(a_i^T x)\dot{x}_i, \tag{5.8}$$

with initialization given by

$$z_i(0) = 1 - nx_i(0)^2, \tag{5.9}$$

$$w_i(0) = nx_i(0)(a_i^T x), \tag{5.10}$$

for all $i = 1, 2, \ldots, n$. Intuitively, the dynamics consist of two terms: a consensus term that averages estimates by adjacent agents and a term that compensates for the time-varying nature of $x(t)$. The choice for the dynamics of $\{z_i(t)\}_{i=1}^n$ and $\{w_i(t)\}_{i=1}^n$ will become more clear in view of the following lemma which states that the collective averages of the local estimates $\{z_i(t)\}_{i=1}^n$ and $\{w_i(t)\}_{i=1}^n$ are correct for all times $t \geq 0$.

Lemma 5.3.1. *Assume that $x(t)$ exists for all $t \geq 0$. Then, the estimators $\{z_i\}_{i=1}^n$ and $\{w_i\}_{i=1}^n$ with dynamics given by eqs.* (5.7) *and* (5.8) *and initial conditions as in eqs.* (5.9)

and (5.10) *satisfy*

$$\frac{1}{n}\sum_{i=1}^{n} z_i(t) = 1 - x(t)^T x(t), \tag{5.11}$$

$$\frac{1}{n}\sum_{i=1}^{n} w_i(t) = x(t)^T A x(t), \tag{5.12}$$

for all $t \geq 0$.

The proof of the last lemma is straightforward and therefore, omitted.

It remains to determine the dynamics of $x(t)$. We propose the following dynamics for each component $x_i(t)$ of the collective estimate $x(t)$

$$\dot{x}_i = (z_i + \alpha w_i)x_i - (\beta(l_i^T w) + \alpha(1 + z_i))(a_i^T x), \tag{5.13}$$

for all $i = 1, 2, \ldots, n$, where α, β are positive scalars whose range will be determined shortly and l_i^T is the ith row of $\mathcal{L}(\mathcal{G})$. The choice of the dynamics for the collective estimate $x(t)$ of v_1 may seem counter-intuitive at first sight. However, these dynamics stem from a properly chosen quadratic potential function. More details are included in the proof of Theorem 5.3.4. Next, we include an interpretation of the proposed dynamics.

Remark 5.3.2 (Connection with the gradient flow of the Rayleigh quotient on the unit sphere)**.** *Consider the dynamics of* $x(t)$ *as in* (5.13) *with initial condition* $x(0)$ *having unit Euclidean norm. Furthermore, let* $z_1(t) = \ldots = z_n(t) = 1 - x(t)^T x(t)$ *and* $w_1(t) = \ldots = w_n(t) = x(t)^T A x(t)$ *for all* $t \geq 0$*, that is all agents have access to the values of* $1 - x(t)^T x(t)$ *and* $x(t)^T A x(t)$*. Then, the proposed dynamics of the collective estimate* $x(t)$ *reduce to the gradient flow of the Rayleigh quotient* $x^T Ax/x^T x$ *on the unit sphere of* $\mathbb{R}^n$ *[3], i.e.*

$$\dot{x} = -\alpha P_x A x \tag{5.14}$$

where $P_x = I - xx^T$.

Next, we analyze the convergence properties of the proposed dynamical system. As a first step towards this direction, we compute its set of equilibria.

Lemma 5.3.3. *Let λ_i denote the ith eigenvalue of A and v_i denote the associated unit-norm eigenvector. Moreover, let $\alpha > 0$ such that*

$$I - 2\alpha A \succ 0. \tag{5.15}$$

Then, the set of equilibria of the dynamics given in eqs. (5.7), (5.8) *and* (5.13), *which we denote by $\mathcal{S}_1$, is as follows:*

$$\mathcal{S}_1 = \{(\mathbf{0}, \mathbf{1}, \mathbf{0}), (\pm v_1, \mathbf{0}, \lambda_1 \mathbf{1}), \ldots, (\pm v_n, \mathbf{0}, \lambda_n \mathbf{1})\}. \tag{5.16}$$

Intuitively, the proposed dynamical system has $2n + 1$ isolated equilibria, $2n - 1$ of which are undesirable, namely,

$$\mathcal{S}_1^u = \{(\mathbf{0}, \mathbf{1}, \mathbf{0}), (\pm v_2, \mathbf{0}, \lambda_1 \mathbf{1}), \ldots, (\pm v_n, \mathbf{0}, \lambda_n \mathbf{1})\}, \tag{5.17}$$

and two are desirable, namely

$$\mathcal{S}_1^d = \{(\pm v_1, \mathbf{0}, \lambda_1 \mathbf{1})\}. \tag{5.18}$$

An illustration of the equilibria for the case $n = 2$ is presented in Figure 5.1.

Next, we present the first main result of this work in the following theorem regarding the convergence of the proposed dynamical system.

Theorem 5.3.4 (Almost-global convergence)**.** *Let λ_i denote the ith eigenvalue of A, with $\lambda_1 < \lambda_2 < \cdots < \lambda_n$, and v_i denote the associated unit-norm eigenvector. Then, given $\alpha > 0$ satisfying* (5.15) *and β satisfying*

$$\beta > \max\{1, (n/4)\operatorname{diam}(\mathcal{G})\}\alpha^2, \tag{5.19}$$

we have that:

(i) The protocol in eqs. (5.7)*,* (5.8) *and* (5.13) *with initial conditions* $\{z_i(0), w_i(0)\}_{i=1}^n$ *as in eqs.* (5.9) *and* (5.10) *asymptotically converges to an equilibrium point.*

(ii) All undesirable equilibria are unstable.

(iii) None of the undesirable equilibria is attractive.

Sketch of Proof: We use the function defined by

$$\phi_1(x, z, w) = \frac{1}{2} z^T z + \alpha z^T w + \frac{\beta}{2} w^T \mathcal{L}(\mathcal{G}) w + \alpha n x^T A x. \tag{5.20}$$

Intuitively, the term $z^T z$ penalizes deviations from the unit sphere, the term $w^T \mathcal{L}(\mathcal{G}) w$ penalizes deviations in the estimates of the value of the collective quantity $x^T A x$ among adjacent agents and the term $x^T A x$ is the Rayleigh quotient when x has norm one. Next, we show that $\dot{\phi}_1 \leq 0$ along the trajectories of the system and that $\dot{\phi}_1(x, z, w) = 0$ if and only if (x, z, w) is an equilibrium. Next, we show that the trajectories cannot escape to infinity while ϕ_1 is nonincreasing and we conclude that the trajectories of the system converge to an equilibrium. Finally, we prove that that all undesirable equilibria, that is all equilibria except for $(\pm v_1, \mathbf{0}, \lambda_1 \mathbf{1})$, are unstable and non-attractive. A detailed proof of Theorem 5.3.4 is included in Appendix D.2. □

Remark 5.3.5. *In practice, the presence of any numerical perturbation (e.g. as in the proof of Theorem 5.3.4), together with Theorem 5.3.4, implies that the proposed dynamical system eventually leaves any neighborhood of an unstable equilibrium, and converges to one of the two desired equilibria.*

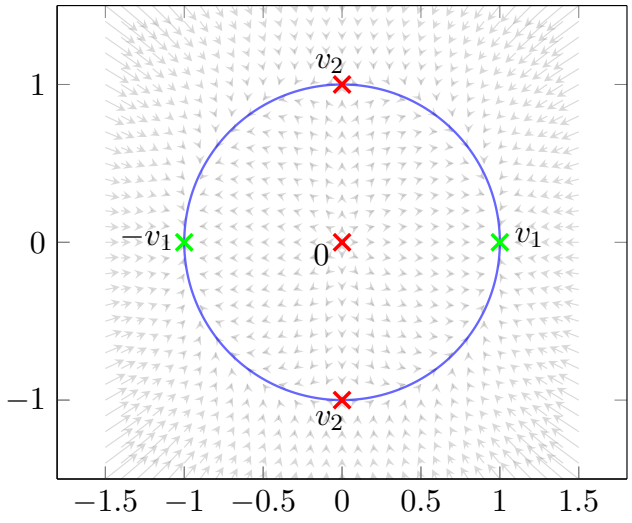


Figure 5.1: Desirable equilibria in green and undesirable equilibria in red for the toy case $n = 2$.

5.4 Mupltiple eigenvector computation

In this section, we address the problem of distributedly computing the k smallest eigenvalues and the associated eigenvectors of a given matrix $A \in \mathbb{R}^{m\times m}$ that is distributed across a network. Specifically, we have a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = n$ agents and let $X(t) = [X_1(t)^T \cdots X_n(t)^T]^T \in \mathbb{R}^{m\times k}$ be a collective quantity whose ith block-component $X_i \in \mathbb{R}^{m_i\times m}$ is maintained by agent i.

We propose that all agents track two collective time-varying quantities, namely $I_k - X(t)^T X(t)$ and $X(t)^T AX(t)$. We propose that each agent i maintains an estimate $Z_i(t) \in \mathbb{R}^{k\times k}$ of $I - X(t)^T X(t)$ and a local estimate $W_i(t) \in \mathbb{R}^{k\times k}$ of $X(t)^T AX(t)$. As before, we observe that these two global quantities can be written as the average of local quantities as follows:

$$I - X(t)^T X(t) = \frac{1}{n}\sum_{i=1}^{n}(I - nX_i(t)^T X_i(t)), \tag{5.21}$$

$$X(t)^T AX(t) = \frac{1}{n}\sum_{i=1}^{n} nX_i(t)A_i^T X(t), \tag{5.22}$$

where A_i^T denotes the ith block-row of A. Based on this observation, we propose dynamics

for $\{Z_i(t)\}_{i=1}^n$ and $\{W_i(t)\}_{i=1}^n$, see (5.24) and (5.25), consisting of two terms: a consensus term and a term that compensates for the time-varying nature of $X(t)$.

We propose the following dynamical system for solving the problem at hand:

$$\dot{X}_i = X_i(Z_i + \alpha W_i) - A_i^T X(\alpha(Z_i + I_k) + \beta L_i^T W), \tag{5.23}$$

$$\dot{Z}_i = \sum_{j \in \mathcal{N}_i} (Z_j - Z_i) - n(X_i^T \dot{X}_i + \dot{X}_i^T X_i), \tag{5.24}$$

$$\dot{W}_i = \sum_{j \in \mathcal{N}_i} (W_j - W_i) + n(\dot{X}_i^T A_i^T X + X^T A_i, \dot{X}_i), \tag{5.25}$$

where $L_i^T = l_i^T \otimes I_k$. The local estimates $\{Z_i\}_{i=1}^n$, $\{W_i\}_{i=1}^n$ are initialized as follows:

$$Z_i(0) = I_k - X_i(0)^T X_i(0), \tag{5.26}$$

$$W_i(0) = (n/2)(X_i(0)^T A_i^T X(0) + X(0)^T A_i X_i(0)), \tag{5.27}$$

for all $i = 1, 2, \ldots, n$.

Next, we analyze the convergence of the proposed dynamical system. The first result concerns the estimators $\{Z_i(t)\}_{i=1}^n$ and $\{W_i(t)\}_{i=1}^n$. Specifically, the collective averages of the local estimates $\{Z_i(t)\}_{i=1}^n$ and $\{W_i(t)\}_{i=1}^n$ are correct for all times. The proof is straightforward and therefore, omitted.

Lemma 5.4.1. *The local quantities $\{Z_i\}_{i=1}^n$ and $\{W_i\}_{i=1}^n$ with dynamics given by eqs.* (5.24) *and* (5.25) *and initial conditions as in eqs.* (5.26) *and* (5.27) *satisfy*

$$\frac{1}{n} \sum_{i=1}^n Z_i(t) = I_k - X(t)^T X(t), \tag{5.28}$$

$$\frac{1}{n} \sum_{i=1}^n W_i(t) = X(t)^T A X(t), \tag{5.29}$$

for all $t \geq 0$.

Next, we compute the equilibria of the proposed dynamical system.

Lemma 5.4.2. *Let $\alpha, \beta > 0$ satisfying eqs.* (5.15) *and* (5.19)*. Then, a triplet (X, Z, W) is an equilibrium for the dynamical given in eqs.* (5.23) *to* (5.25) *if and only if*

$$Z_i = I_k - X^T X, \quad \text{and} \quad W_i = X^T A X, \quad \forall i = 1, 2, \ldots, n, \tag{5.30}$$

and X can be written as

$$X = U_{m\times r} Q_{k\times r}^T, \tag{5.31}$$

for some $k \times r$ matrix Q satisfying $Q^T Q = I_r$ and for some $m \times r$ matrix U that satisfies $U^T U = I_r$ and

$$U(U^T A U) = AU, \tag{5.32}$$

that is $\operatorname{span}(U)$ *must be invariant subspace of A.*

The proof of the above lemma can be found in Appendix D.3.

Remark 5.4.3 (Non-isolated equilibria)**.** *In contrast to the $k = 1$ case, the equilibria for the proposed dynamical system are not isolated. For instance, if $X = U_{m\times r} Q_{k\times r}^T$ for some $k \times r$ matrix Q satisfying $Q^T Q = I_r$ and for some $m \times r$ matrix U that satisfies $U^T U = I_r$ and* (5.32)*, then XR where R is some $r \times r$ orthogonal matrix also satisfies the assumptions of Lemma 5.4.2. Since R can be chosen to be arbitrarily close to the identity matrix, its follows that there is a connected set of equilibria containing X.*

The sets of equilibria can be divided into two categories, namely desirable and undesirable. The desirable sets of equilibria correspond to

$$X = \begin{bmatrix} \pm v_{\pi(1)} & \pm v_{\pi(2)} & \ldots & \pm v_{\pi(k)} \end{bmatrix} Q \tag{5.33}$$

for some orthogonal $k \times k$ matrix Q, where π is a permutation of $[k]$ and $v_1, \ldots, v_k$ are the unit-norm eigenvectors of A associated with the k smallest eigenvalues. The undesirable sets are simply sets that are not desirable.

The second and most important result of this work is presented in the following theorem.

Theorem 5.4.4 (Almost-global convergence)**.** *Consider the dynamical system in eqs.* (5.23) *to* (5.25) *with initial conditions* $\{Z_i(0), W_i(0)\}_{i=1}^n$ *as in eqs.* (5.26) *and* (5.27) *and* α, β *satisfying eqs.* (5.15) *and* (5.19)*. Then,*

(i) The trajectories of the system asymptotically approach some set of equilibria.

(ii) All undesirable sets of equilibria are (uniformly) unstable.

(iii) All undesirable sets of equilibria are (uniformly) non-attractive.

Sketch of proof. Generalizing the approach for the case $k = 1$, we use the potential function ϕ_2 defined by

$$\phi_2(X, Z, W) = \frac{1}{2}\operatorname{tr}(Z^T Z) + \frac{\beta}{2}\operatorname{tr}(W^T(\mathcal{L}(\mathcal{G}) \otimes I_k)W) + \alpha \operatorname{tr}(Z^T W) + \alpha n \operatorname{tr}(X^T A X), \quad (5.34)$$

Intuitively, the term $\operatorname{tr}(Z^T Z)$ promotes orthogonality for the columns of X, the term $\operatorname{tr}(W^T(\mathcal{L}(\mathcal{G}) \otimes I_k)W)$ penalizes deviations in the estimates of the value of the collective quantity $X^T A X$ among adjacent agents and the term $\operatorname{tr}(X^T A X)$ is equal to generalized Rayleigh quotient $\operatorname{tr}(X^T A X (X^T X)^{-1})$ when X has orthogonal columns. Next, we show that $\dot{\phi}_2 \leq 0$ along the trajectories of the system and that the trajectories of the system cannot escape to infinity while $\dot{\phi}_2 \leq 0$. We explicitly find all points (X, Z, W) such that $\dot{\phi}_2(X, Z, W) = 0$ and show that the dynamical system at hand asymptotically approaches some set of equilibria. In addition, we show that all undesirable sets of equilibria are both unstable and non-attractive. A detailed proof of Theorem 5.4.4 is included in Appendix D.4.

Remark 5.4.5. *In practice, the presence of any numerical perturbation, e.g. as in the proof of Theorem 5.4.4, together with Theorem 5.4.4, imply that the proposed dynamical system eventually leaves any neighborhood of an undesirable set of equilibria, and asymptotically converges to a desired set of equilibria. Let* $W_i(t) = Q_i(t)\Delta_i(t)Q_i(t)^T$ *be an eigendecomposition of* $W_i(t)$ *and define* $Y_i(t) \doteq X_i(t)Q_i(t)$*. Then, up to a reordering of columns, we have*

that for all $i = 1, 2, \ldots, n$

$$\lim_{t\to\infty} [Y_1(t)^T \cdots Y_n(t)^T]^T = \begin{bmatrix} \pm v_1 & \pm v_2 & \ldots & \pm v_k \end{bmatrix}, \tag{5.35}$$

and

$$\lim_{t\to\infty} \Delta_i(t) = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_k). \tag{5.36}$$

5.5 Simulations

5.5.1 Smallest eigenvalue of an adjacency matrix

As a first experiment, we compute the minimum eigenvalue and the associated eigenvector of the adjacency matrix of the six-regular graph of Fig. 5.2. The smallest eigenvalue and the associated unit-norm eigenvector of an r-regular graph are equal to

$$\lambda_1 = -r, \qquad v_1 = \pm(1/\sqrt{n})\mathbf{1}. \tag{5.37}$$

We used $n = 20$ vertices and $r = 6$. Hence, the minimum eigenvalue λ_1 is equal to -6 and the associated eigenvector is approximately equal to $v_1 \pm 0.2236 \cdot \mathbf{1}$. All $\{x_i(0)\}_{i=1}^n$ are randomly generated by sampling a normal distribution with zero mean and standard deviation $1/n$. Results are presented in Fig. 5.3(a)-(c). In all experiments, we used the 4th order Runge-Kutta method with step-size $h = 0.01$.

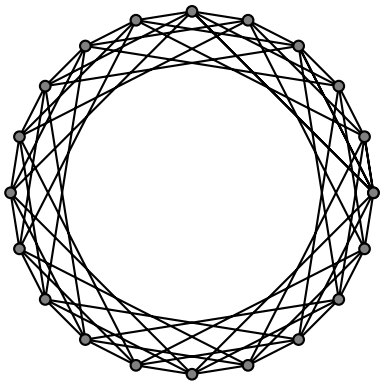


Figure 5.2: Six-regular graph used in the simulations.

5.5.2 Diagonalizing an entire graph Laplacian

As a second experiment, we estimate all eigenvalues and the associated eigenvectors of the Laplacian matrix of a network of $n = 10$ vertices whose adjacency matrix $A(\mathcal{G})$ is given by

$$A(\mathcal{G}) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{5.38}$$

Representative results are presented in Fig. 5.3(d)-(f). All eigenvalues of the Laplacian are accurately estimated as can be seen in Fig. 5.3(d). Each column of $\{X_i(0)\}_{i=1}^n$ is randomly generated as in the first example.

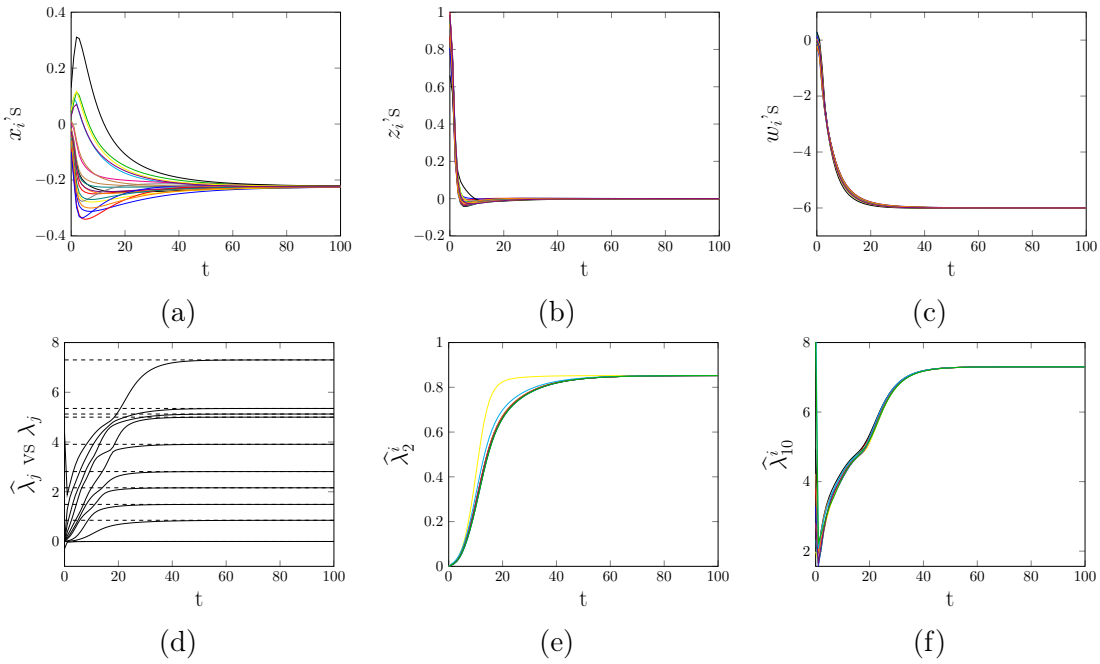


Figure 5.3: Top row: evolution of the agents' states over time for the first experiment. It can be seen that $x(t) \to -(1/\sqrt{n})\mathbf{1}$, where $1/\sqrt{n} \approx 0.2236$, $z(t) \to \mathbf{0}$ and $w(t) \to \lambda_1(A)\mathbf{1}$ where $\lambda_1(A) = -r = -6$. Bottom row: (d) mean estimates of $\lambda_i(\mathcal{L}(\mathcal{G}))$'s (solid) vs actual values (dashed) in the second experiment. (e)-(f) estimates of $\lambda_2(\mathcal{L}(\mathcal{G}))$ and $\lambda_{10}(\mathcal{L}(\mathcal{G}))$ for each agent over time. Note that $\lambda_2(\mathcal{L}(\mathcal{G})) = 0.8526$, $\lambda_{10}(\mathcal{L}(\mathcal{G})) = 7.3018$, $\widehat{\lambda}_j^i$ denotes the estimate of $\lambda_j(\mathcal{L}(\mathcal{G}))$ by agent i and $\widehat{\lambda}_j$ denotes the average estimate over the network of the same quantity.

The first 5 eigenvectors of the graph Laplacian as computed by Matlab's routine `eig` are

$$V_{1:5} = \begin{bmatrix} -0.3162 & 0.0705 & 0.2490 & 0.3669 & -0.1737 \\ -0.3162 & 0.1651 & 0.1694 & -0.5694 & 0.4429 \\ -0.3162 & 0.1426 & -0.1228 & 0.0968 & -0.2584 \\ -0.3162 & -0.9076 & -0.1143 & -0.0139 & 0.1553 \\ -0.3162 & 0.0810 & 0.0681 & -0.5431 & -0.5153 \\ -0.3162 & 0.2384 & -0.7421 & 0.2153 & 0.2057 \\ -0.3162 & 0.0705 & 0.2490 & 0.3669 & -0.1737 \\ -0.3162 & 0.1310 & -0.2548 & -0.1311 & 0.0915 \\ -0.3162 & -0.1338 & 0.0561 & 0.0161 & -0.2814 \\ -0.3162 & 0.1425 & 0.4423 & 0.1956 & 0.5072 \end{bmatrix}, \tag{5.39}$$

and the fist 5 eigenvectors of graph Laplacian as computed by the proposed decentralized method are almost identical (up to a sign flip):

$$\widehat{V}_{1:5} = \begin{bmatrix} -0.3162 & 0.0705 & 0.2490 & -0.3669 & -0.1737 \\ -0.3162 & 0.1651 & 0.1694 & 0.5694 & 0.4429 \\ -0.3162 & 0.1426 & -0.1228 & -0.0968 & -0.2584 \\ -0.3162 & -0.9075 & -0.1143 & 0.0139 & 0.1553 \\ -0.3162 & 0.0810 & 0.0681 & 0.5431 & -0.5153 \\ -0.3162 & 0.2384 & -0.7421 & -0.2153 & 0.2057 \\ -0.3162 & 0.0705 & 0.2490 & -0.3669 & -0.1737 \\ -0.3162 & 0.1310 & -0.2548 & 0.1311 & 0.0915 \\ -0.3162 & -0.1337 & 0.0562 & -0.0161 & -0.2815 \\ -0.3162 & 0.1425 & 0.4423 & -0.1956 & 0.5072 \end{bmatrix}. \tag{5.40}$$

5.6 Application in permutation synchronization

In this section, we show how to use the approach of this section to solve the spectral relaxation of the permutation synchronization problem in a decentralized fashion.

Recall from Chapter 3, that permutation synchronization can be cast as the following combinatorial optimization problem:

$$\underset{\Pi_1,\ldots,\Pi_n \in \mathfrak{S}_m}{\text{maximize}} \quad \sum_{\{i,j\}\in\mathcal{E}} \operatorname{tr}(\Pi_i^T \widetilde{\Pi}_{ij} \Pi_j) \tag{5.41}$$

Unfortunately, problem (3.5) is computationally intractable. To address the computational

intractability of problem (3.5), we, next, propose a spectral relaxation. In contrast to the spectral relaxation of [101], we do not assume that all pairwise association matrices $\widetilde{\Pi}_{ij}$ are available.

The following lemma is crucial for deriving the spectral relaxation of problem (3.5).

Lemma 5.6.1. *Let $\Delta \doteq \operatorname{diag}(\widetilde{\mathbf{\Pi}}\mathbf{1})$. Assume that the pairwise associations $\{\widetilde{\Pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$ are consistent and the sensor graph $\mathcal{G}$ is connected. Then, the matrix $\Delta^{-1}\widetilde{\mathbf{\Pi}}$ has exactly m leading eigenvalues equal to 1, where m is the size of the universe of features. Furthermore, there exist permutation matrices $\Pi_1, \Pi_2, \ldots, \Pi_n \in \mathcal{P}_m$, unique up to a global permutation, satisfying*

$$\widetilde{\mathbf{\Pi}}\Pi = \Delta\Pi, \tag{5.42}$$

where $\Pi = [\Pi_1^T \ \Pi_2^T \ \cdots \ \Pi_n^T]^T$.

For a proof of Lemma 5.6.1 we refer the reader to [7].

An approximate solution to problem (5.41), under relaxed orthonormality and nonnegativity constraints, is determined by the m leading eigenvectors of $\Delta^{-1}\widetilde{\mathbf{\Pi}}$ [7]. Since, $\Delta^{-1}\widetilde{\mathbf{\Pi}}$ is not, in general, a symmetric matrix, the approach of this chapter is not directly applicable. Nevertheless, $\Delta^{-1}\widetilde{\mathbf{\Pi}}$ is similar to the symmetric matrix $\Delta^{-1/2}\widetilde{\mathbf{\Pi}}\Delta^{-1/2}$. Let $V\Lambda V^T$ be an eigendecomposition of $\Delta^{-1/2}\widetilde{\mathbf{\Pi}}\Delta^{-1/2}$, where V is an orthogonal matrix and Λ a diagonal. Then, let

$$\Delta^{-1/2}V = \begin{bmatrix} Q_1 \\ \vdots \\ Q_n \end{bmatrix} \doteq Q, \tag{5.43}$$

for some $m \times m$ matrices $Q_1, Q_2, \ldots, Q_n$. In the noiseless case, $\operatorname{span}(\Pi)$ and $\operatorname{span}(Q)$ are equal, and thus, $\Pi = QG^{-1}$ for some invertible matrix G. Without loss of generality, we can assume that $\Pi_1 = I_m$ and thus, G must be equal to Q_1. Based on this observation, the last step of the proposed approach consists of each agent i computing the approximately

optimal $\Pi_i^\star$ by

$$\Pi_i^\star = \text{Proj}_{\mathcal{P}_m}(Q_i Q_1^{-1}), \tag{5.44}$$

where $\text{Proj}_{\mathcal{P}_m}$ denotes the projection onto set of $m \times m$ permutation matrices which can be computed using the Hungarian algorithm [67]. Finally, cycle consistent pairwise associations can be computed by

$$\Pi_{ij}^\star = \Pi_i^\star \Pi_j^{\star T}. \tag{5.45}$$

Overall, we propose the following four-step approach for solving the spectral relaxation of permutation synchronization in a decentralized fashion:

1. All agents collectively compute the m leading eigenvectors $V = [v_1 \ \cdots \ v_m]$ of $\Delta^{-1/2}\widetilde{\mathbf{\Pi}}\Delta^{-1/2}$ using the approach of Section 5.4.

2. Each agent i computes Q_i as defined in (5.43).

3. Agent 1 transmits Q_1 to the entire group. This operation takes $\text{diam}(\mathcal{G})$ time steps.

4. Each agent i computes $\Pi_i^\star$ as in (5.44) and pairs of agents compute their corresponding associations by (5.45).

Finally, we evaluate the propose spectral relaxation. We consider three graph topologies, namely, a six-regular, a ten-regular graph and a complete graph, all of which have $n = 20$ vertices. We used $m = 30$ as the number of features per collection. We vary the percentage of outliers of the initial pairwise associations $\{\widetilde{\pi}_{ij}\}_{\{i,j\}\in\mathcal{E}}$ from 0% to 90%. Results are presented in Fig. 5.4. We observe that, not surprisingly, increasing the connectivity of a graph, significantly improves the accuracy of the spectral relaxation for permutation synchronization. Furthermore, given enough noisy pairwise associations, exact recovery of the true pairwise associations is achieved for a significant percentage of outliers, namely about 60%.

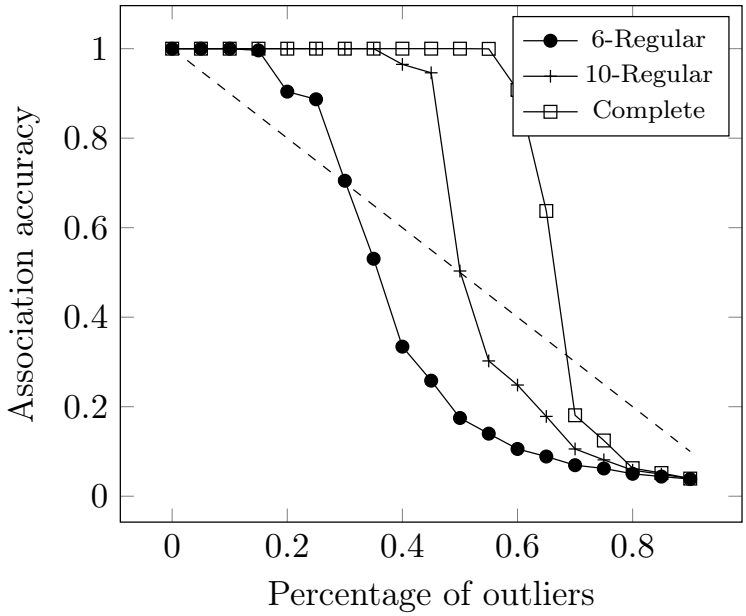


Figure 5.4: Accuracy of pairwise associations after permutation synchronization versus initial percentage of outliers for a six-regular, a ten-regular and a complete graph with 20 vertices. We observe that in the six-regular graph case, exact recovery of the true pairwise associations can be achieved for up to 20% of outliers, whereas this percentage increases to 40% for the case of the 10-regular graph and to 60% for the case of a complete graph.

5.7 Conclusions

In this chapter, we proposed a dynamical systems approach to distributedly compute any number of extreme eigenvalues and associated eigenvectors of a matrix that is distributed across a network with almost-global convergence guarantees. In contrast to approaches based on Orthogonal Iteration, orthogonality constraints are only asymptotically satisfied by the dynamical system herein proposed. Thus, the main computational burden of the Decentralized Orthogonal Iteration, namely, the orthonormalization step, is not present in our approach. In addition, we applied the proposed method to permutation synchronization, specifically to decentralize the spectral relaxation of the permutation synchronization.

Chapter 6

Distributed cooperative state estimation for mobile agents

6.1 Introduction

In this chapter, we first propose a method for fusion of two random vectors with unknown cross-correlations which is less conservative than the widely used Covariance Intersection (CI) while taking cross-correlations into account. Then, we extend our formulation for the case of a linear measurement model. Finally, we present numerical examples and simulations, in a distributed cooperative localization scenario, which demonstrate the validity of the proposed approach and that the proposed approach significantly outperforms Covariance Intersection, while taking correlations into account.

This chapter is structured as follows: in Section 6.2.1 we include definitions of consistency and related notions and we introduce the problem at hand. Our game-theoretic approach to fusing two random variables with unknown cross-correlations is the topic of Section 6.2.2 and it is generalized for arbitrary linear measurement models in Section 6.3. In Section 6.4 we include details on the implemented numerical algorithm. Numerical examples and

simulation results are presented in Sections 6.5 and 6.6 respectively.

6.2 Fusion under unknown correlations

6.2.1 Problem statement

In this subsection, we formalize the problem of fusing two estimates or a random vector whose correlations are unknown.

First, we need a precise definition of *consistency*. Let $\succeq$ denote the generalized matrix inequality of the cone of positive semidefinite matrices. Moreover, let $E[X]$ and $\mathrm{Cov}\,(X)$ denote, respectively, the expectation and covariance of a random vector X. Then, the notion of consistency is defined as follows.

Definition 6.2.1 (Consistency [58])**.** *Let z be a random vector with expectation $E[z] = \overline{z}$. An estimate $\widetilde{z}$ of $\overline{z}$ is another random vector. The associated error covariance is denoted by $\widetilde{\Sigma}_{zz} \doteq \mathrm{Cov}\,(\widetilde{z} - \overline{z})$. The pair $(\widetilde{z}, \Sigma_{zz})$ is consistent if*

$$E[\widetilde{z}] = \overline{z}, \quad \textit{and} \quad \Sigma_{zz} \succeq \widetilde{\Sigma}_{zz}. \tag{6.1}$$

Intuitively, consistency of the pair $(\widetilde{z}, \Sigma_{zz})$ is satisfied if $\widetilde{z}$ is an unbiased estimator of $\overline{z}$ and Σ_{zz} is a conservative estimate of the actual error covariance $\widetilde{\Sigma}_{zz}$.

Next, we precisely define the problem of fusing two random vector under unknown correlations.

Problem Statement 6.2.2 (Consistent fusion)**.** *Assume we are given two consistent estimates $(\widetilde{x}, \Sigma_{xx})$, $(\widetilde{y}, \Sigma_{yy})$ of $\overline{z}$, where Σ_{xx}, Σ_{yy} are known upper bounds on the true error covariances. Furthermore, we assume that the cross-covariance between $\widetilde{x}$ and $\widetilde{y}$ is unknown. The problem at hand consists of fusing the two consistent estimates $(\widetilde{x}, \Sigma_{xx})$, $(\widetilde{y}, \Sigma_{yy})$ into a*

single consistent estimate $(\widetilde{z}, \Sigma_{zz})$*, where* $\widetilde{z}$ *is of the form*

$$\widetilde{z} = W_x \widetilde{x} + W_y \widetilde{y}, \tag{6.2}$$

subject to $W_x + W_y = I$ *(in order for the mean to be preserved).*

The most widely used solution of the above problem is given by the Covariance Intersection algorithm [58]. Given upper bounds $\Sigma_{xx} \succeq \widetilde{\Sigma}_{xx}$, $\Sigma_{yy} \succeq \widetilde{\Sigma}_{yy}$ the Covariance Intersection equations read

$$\begin{aligned} \widetilde{z} &= \Sigma_{zz} \left\{ \omega \Sigma_{xx}^{-1} \widetilde{x} + (1-\omega) \Sigma_{yy}^{-1} \widetilde{y} \right\}, \\ \Sigma_{zz}^{-1} &= \omega \Sigma_{xx}^{-1} + (1-\omega) \Sigma_{yy}^{-1}, \end{aligned} \tag{6.3}$$

where $\omega \in [0, 1]$. It is not hard to show that

$$\Sigma_{zz} \left\{ \omega \Sigma_{xx}^{-1} + (1-\omega) \Sigma_{yy}^{-1} \right\} = I, \tag{6.4}$$

which implies $E[\widetilde{z}] = \overline{z}$. Moreover, it is easy to check that $(\widetilde{z}, \Sigma_{zz})$ is consistent. The above can be easily generalized for the case of more than two random variables, for partial measurements and for the linear measurement model we consider in Section 6.3. Usually, ω is chosen such that either $\mathrm{tr}(\Sigma_{zz})$ or $\log\det(\Sigma_{zz}^{-1})$ is minimized.

Next, we introduce a notion related to consistency but with relaxed requirements. Let S_+^n denote the positive semidefinite cone, that is the set of $n \times n$ positive semidefinite matrices. A function $f : S_+^n \to \mathbb{R}$ is S_+^n*-nondecreasing* [18] if

$$X \succeq Y \Rightarrow f(X) \geq f(Y) \tag{6.5}$$

for any $X, Y \in S_+^n$. An example of such a function is $f(X) = \mathrm{tr}(X)$. Now, we are ready to introduce the notion of *consistency with respect to a* S_+^n*-nondecreasing function.*

Definition 6.2.3 (f-Consistency)**.** *Let* $f : S_+^n \to \mathbb{R}$ *be a nondecreasing function (with respect*

to S_+^n) satisfying $f(0) = 0$. Let z be a random vector with expectation $E[z] = \overline{z}$ and $\widetilde{z}$ be an estimate of $\overline{z}$ with associated error covariance $\widetilde{\Sigma}_{zz}$. The pair $(\widetilde{z}, \Sigma_{zz})$ is f-consistent if $E[\widetilde{z}] = \overline{z}$ and $f(\Sigma_{zz}) \geq f(\widetilde{\Sigma}_{zz})$.

Remark 6.2.4. *Consistency implies f-consistency. However, the converse in not necessarily true.*

Based on the preceding definition of f-consistency, we next present a relaxed version of the consistent fusion problem.

Problem Statement 6.2.5 (Trace-consistent fusion)**.** *Given two consistent estimates $(\widetilde{x}, \Sigma_{xx})$, $(\widetilde{y}, \Sigma_{yy})$ of $\overline{z}$, where Σ_{xx}, Σ_{yy} are known upper bounds on the true error variances. The problem at hand consists of fusing the two consistent estimates $(\widetilde{x}, \Sigma_{xx})$, $(\widetilde{y}, \Sigma_{yy})$ in a single trace-consistent estimate $(\widetilde{z}, \Sigma_{zz})$, where $\widetilde{z}$ is a linear combination of $\widetilde{x}$ and $\widetilde{y}$ and* $\operatorname{tr}(\Sigma_{zz}) \geq \operatorname{tr}(\widetilde{\Sigma}_{zz})$.

Next, we introduce a minimax formulation for the problem of trace-consistent fusion. Relaxing the consistency constraint to the trace-consistency constraint enables us to estimate the weighting matrices W_x, W_y, in full generality, according to some optimality criterion, which is none other than the minimax of the trace of the covariance matrix or equivalently the mean-squared error of the fused estimate.

6.2.2 A minimax formulation to trace-consistent fusion

The goal of this subsection is the derivation of our minimax approach that tackles the problem of trace-consistent fusion as defined earlier.

First, we need some basic notions from game theory. A two-player game on $\mathbb{R}^m \times \mathbb{R}^n$ is defined by a pay-off function $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$. Intuitively, the first player makes a move $u \in \mathbb{R}^m$ then, the second player makes a move $v \in \mathbb{R}^m$ and receives payment from the first player equal to $f(u, v)$. The goal of the first player is to minimize its payment and the goal of the second player is to maximize the received payment. The game is *convex-concave* if

the pay-off function $f(u, v)$ is convex in u for fixed v and concave in v for fixed u. For a in-depth review of convex-concave games in the context of convex optimization, we refer the reader to [41].

Back to the problem at hand, let z be a random vector with expectation $E[z] = \overline{z}$. Assume we have two estimates $(\widetilde{x}, \Sigma_{xx})$, $(\widetilde{y}, \Sigma_{yy})$ of $\overline{z}$ where Σ_{xx}, Σ_{yy} are approximations to the true error covariances $\widetilde{\Sigma}_{xx}$, $\widetilde{\Sigma}_{yy}$. Based on the preceding discussion of Section 6.2.1, the fused estimate is of the form

$$\widetilde{z} = (I - K)\widetilde{x} + K\widetilde{y}, \tag{6.6}$$

and the associated error covariance $\widetilde{\Sigma}_{zz}$ is given by

$$\widetilde{\Sigma}_{zz} = \begin{bmatrix} I - K & K \end{bmatrix} \begin{bmatrix} \widetilde{\Sigma}_{xx} & \widetilde{\Sigma}_{xy} \\ \widetilde{\Sigma}_{xy}^T & \widetilde{\Sigma}_{yy} \end{bmatrix} \begin{bmatrix} I - K^T \\ K^T \end{bmatrix}. \tag{6.7}$$

However, $\widetilde{\Sigma}_{xx}$, $\widetilde{\Sigma}_{yy}$ are not known. Therefore, we define

$$\Sigma_{zz} \doteq \begin{bmatrix} I - K & K \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - K^T \\ K^T \end{bmatrix}. \tag{6.8}$$

However, not all values of the cross-covariance Σ_{xy} result in a positive-semidefinite covariance matrix. For this reason, we include the following Linear Matrix Inequality (LMI) constraint on Σ_{xy}:

$$\begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_{yy} \end{bmatrix} \succeq 0. \tag{6.9}$$

Next, we present a lemma that is instrumental in showing that the trace of Σ_{zz}, as a function of K and Σ_{xy}, is convex-concave.

Lemma 6.2.6. *Given $\Sigma_{xx}, \Sigma_{yy} \succ 0$, and Σ_{xy} satisfying* (6.9) *we have that*

$$\Sigma_{xx} + \Sigma_{yy} - 2\Sigma_{xy} \succeq 0. \tag{6.10}$$

The proof of Lemma 6.2.6 is included in Appendix E.1.

Next, we state and prove that $\mathrm{tr}(\Sigma_{zz})$, viewed as a function of K and Σ_{xy}, is convex-concave.

Proposition 6.2.7. *We have that, for a fixed Σ_{xy} satisfying* (6.9)*, $\mathrm{tr}(\Sigma_{zz})$ is convex in K. Moreover, for a fixed K, $\mathrm{tr}(\Sigma_{zz})$ is linear, and thus concave, in Σ_{xy} with a convex domain defined by* (6.9)*. As a result, $\mathrm{tr}(\Sigma_{zz})$ is a convex-concave function in (K, Σ_{xy}).*

Proof. Let $\widetilde{f}(K) \doteq \mathrm{tr}(\Sigma_{zz})$ for a fixed Σ_{xy} satisfying (6.9). The Hessian of $\widetilde{f}$ can be computed by

$$\begin{aligned} \mathrm{Hess}\, \widetilde{f}(K)[U, U] &= \mathrm{tr}(U(\Sigma_{xx} + \Sigma_{yy} - 2\Sigma_{xy})U^T) && (6.11) \\ &= u^T(I \otimes (\Sigma_{xx} + \Sigma_{yy} - 2\Sigma_{xy}))u, && (6.12) \end{aligned}$$

where $u = \mathrm{vec}(U)$. From Lemma 6.2.6 and from the properties of Kronecker product, it follows that the Hessian of $\widetilde{f}$ is positive semidefinite. Thus, for a fixed Σ_{xy} satisfying (6.9) $\mathrm{tr}(\Sigma_{zz})$ is convex in K. The remainder of the proof is straightforward. □

We formulate the problem of finding the weighting matrix K as a convex-concave game: the first player chooses K to minimize $\mathrm{tr}(\Sigma_{zz})$ whereas the second player chooses Σ_{xy} to maximize $\mathrm{tr}(\Sigma_{zz})$. More specifically, let $(K^\star, \Sigma_{xy}^\star)$ be the solution to the following minimax optimization problem

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & \sup_{\Sigma_{xy}} \ \mathrm{tr}(\Sigma_{zz}) \\ \text{subject to} \quad & \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_{yy} \end{bmatrix} \succeq 0. \end{aligned} \tag{6.13}$$

Then, the fused estimated and the associated error covariance are given by

$$\begin{aligned} \widetilde{z} &= (I - K^\star)\widetilde{x} + K^\star \widetilde{y}, \\ \Sigma_{zz}^\star &= \begin{bmatrix} I - K^\star & K^\star \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy}^\star \\ \Sigma_{xy}^{\star T} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - K^{\star T} \\ K^{\star T} \end{bmatrix}. \end{aligned} \tag{6.14}$$

Naturally, we have the following lemma.

Lemma 6.2.8. *If $(\widetilde{x}, \Sigma_{xx})$ and $(\widetilde{y}, \Sigma_{yy})$ are consistent, then the pair $(\widetilde{z}, \Sigma_{zz}^\star)$ given by* (6.14) *is trace-consistent.*

The proof of Lemma 6.2.8 is a special case of the proof of Lemma 6.3.2, whose proof is included in Appendix E.4.

The problem of numerically solving problem (6.13) is the topic of Section 6.4. The case under consideration in this section can be viewed as a special case of Section 6.3.

Remark 6.2.9. *The problem of maximizing* $\operatorname{tr}(\Sigma_{zz})$ *over the cross-correlation* Σ_{xy} *is equivalent to*

$$\begin{aligned} \underset{R}{\text{maximize}} \quad & 2\operatorname{tr}(F(K)^T R) \\ \text{subject to} \quad & R^T R \preceq I, \end{aligned} \tag{6.15}$$

with

$$F(K) = \Sigma_{xx}^{1/2}(K - K^T K)\Sigma_{yy}^{1/2}. \tag{6.16}$$

The optimal solution of problem (6.15) *(see Appendix E.2) is given by*

$$R^\star = UV^T, \tag{6.17}$$

where $U\Sigma V^T$ is a singular value decomposition (SVD) of $F(K)$. It follows that

$$\max_{R:R^T R \preceq I} 2\operatorname{tr}(F(K)^T R) = 2\|F(K)\|_\star, \tag{6.18}$$

where $\|\cdot\|_\star$ denotes the nuclear norm of a matrix. Problem (6.13) *can be equivalently written as an unconstrained minimization problem with respect to K with objective:*

$$f(K, \Sigma_{xy}^\star) = \operatorname{tr}(K(\Sigma_{xx} + \Sigma_{yy})K^T) - 2\operatorname{tr}(\Sigma_{xx}K) + 2\|\Sigma_{xx}^{1/2}(K - K^T K)\Sigma_{yy}^{1/2}\|_\star. \tag{6.19}$$

This observation suggests an algorithmic possibility for solving problem (6.13) *by the subgradient method as proposed by [153]. However, we have experimentally observed that the proposed Newton method of Section 6.4 converges to the optimal solution of problem* (6.13) *much faster than the subgradient method.*

6.3 Minimax linear update

In this section, we explore the problem of fusing an estimate of a random variable with another noisy partial estimate whose correlation is uknown. The problem of Section 6.2.2 can be viewed as a special case of the problem considered in this section.

We assume we have two random vectors x, y with expectations $E[x] = \overline{x}$ and $E[y] = \overline{y}$. We have some estimates $\widetilde{x}$ and $\widetilde{y}$ of $\overline{x}$ and $\overline{y}$ respectively with associated error covariances $\widetilde{\Sigma}_{xx}$ and $\widetilde{\Sigma}_{yy}$. As before, we assume that the true error covariances are only approximately known. Let Σ_{xx} and Σ_{yy} denote these approximate values. We assume we have a linear measurement model of the form:

$$z = C\overline{x} + D\overline{y} + \eta, \tag{6.20}$$

where η is a zero-mean noise process with covariance Σ_η. We assume that the measurement noise process η is independent of the estimates $\widetilde{x}$ and $\widetilde{y}$.

As in the classic Kalman filter derivation, we propose an update step of the form

$$\widetilde{x}^+ = \widetilde{x} + K(z - \widetilde{z}), \tag{6.21}$$

where $\widetilde{z} \doteq C\widetilde{x} + D\widetilde{y}$. The error of the update is given by

$$\widetilde{x}^+ - \overline{x} = (I - KC)(\widetilde{x} - \overline{x}) - KD(\widetilde{y} - \overline{y}) + K\eta, \tag{6.22}$$

and the associated error covariance, defined by $\widetilde{\Sigma}^+_{xx} \doteq \mathrm{Cov}\,(\widetilde{x}^+ - \overline{x})$, is given by

$$\widetilde{\Sigma}^+_{xx} = \begin{bmatrix} I - KC & -KD \end{bmatrix} \begin{bmatrix} \widetilde{\Sigma}_{xx} & \widetilde{\Sigma}_{xy} \\ \widetilde{\Sigma}^T_{xy} & \widetilde{\Sigma}_{yy} \end{bmatrix} \begin{bmatrix} I - C^T K^T \\ -D^T K^T \end{bmatrix} + K\Sigma_\eta K^T. \tag{6.23}$$

However, we assume that the true error covariances $\widetilde{\Sigma}_{xx}$ and $\widetilde{\Sigma}_{yy}$ are not known, in general. Therefore, we need to define

$$\Sigma^+_{xx} \doteq \begin{bmatrix} I - KC & -KD \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma^T_{xy} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - C^T K^T \\ -D^T K^T \end{bmatrix} + K\Sigma_\eta K^T, \tag{6.24}$$

where Σ_{xy} should satisfy (6.9) in order to be a valid cross-covariance.

Proposition 6.3.1. *For a fixed Σ_{xy} satisfying* (6.9)*, $\mathrm{tr}(\Sigma^+_{xx})$ is convex in K. For a fixed K, $\mathrm{tr}(\Sigma^+_{xx})$ is linear, and thus concave, in Σ_{xy} with a convex domain defined by* (6.9)*. As a result, $\mathrm{tr}(\Sigma^+_{xx})$ is a convex-concave function of (K, Σ_{xy}).*

By rewriting (6.9) using Schur complement, the minimax formulation we propose is as follows:

$$\begin{aligned} &\underset{K}{\text{minimize}} \quad \sup_{\Sigma_{xy}} \quad \mathrm{tr}(\Sigma^+_{xx}) \\ &\text{subject to} \qquad \Sigma_{yy}^{-1/2}\Sigma^T_{xy}\Sigma^{-1}_{xx}\Sigma_{xy}\Sigma_{yy}^{-1/2} - I \preceq 0. \end{aligned} \tag{6.25}$$

Let $(K^\star, \Sigma^\star_{xy})$ be the optimal solution of problem (6.25). Then, the fusion estimate $\widetilde{x}^+$ and

the associated error covariance $\Sigma_{xx}^{+\star}$ are given by:

$$\begin{aligned}\widetilde{x}^{+} &= (I - K^{\star}C)\widetilde{x} - K^{\star}D\widetilde{y} \\ \Sigma_{xx}^{+\star} &= \begin{bmatrix} I - K^{\star}C & -K^{\star}D \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy}^{\star} \\ \Sigma_{xy}^{\star T} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - C^{T}K^{\star T} \\ -D^{T}K^{\star T} \end{bmatrix} + K\Sigma_{\eta}K^{T}.\end{aligned} \tag{6.26}$$

Naturally, we have the following lemma whose proof is included in Appendix E.4.

Lemma 6.3.2. *If $(\widetilde{x}, \Sigma_{xx})$ and $(\widetilde{y}, \Sigma_{yy})$ are consistent, then the pair $(\widetilde{x}^{+}, \Sigma_{xx}^{+\star})$ given by* (6.26) *is trace-consistent.*

Remark 6.3.3. *Observe that for $C = I$, $D = -I$ and $\Sigma_{\eta} = 0$, the problem at hand reduces to the problem of the previous section.*

6.4 Numerical solution with interior-point methods

In this section, we describe the numerical method we use to solve problem (6.25). We use the barrier method with infeasible start Newton method [18] (see Algorithm 3). Intuitively, a sequence of unconstrained minimization problems is solved, using the last point iteration is the starting point for the next iteration.

It is more convenient to derive the Netwon method equations for $X \doteq K^{T}$ instead of K. Let $f(X, \Sigma_{xy}) \doteq \operatorname{tr}(\Sigma_{xx}^{+})$, where Σ_{xx}^{+} as defined in (6.24). Define for $t > 0$, the cost function $f_t(X, Q)$ by

$$f_t(X, Q) = tf(X, Q) + \log\det(-f_1(Q)), \tag{6.27}$$

where $f_1(Q)$ is defined by

$$f_1(Q) = \Sigma_{yy}^{-1/2} Q^{T} \Sigma_{xx}^{-1} Q \Sigma_{yy}^{-1/2} - I. \tag{6.28}$$

Intuitively, $\frac{1}{t} f_t$ approaches f as $t \to \infty$. Note that $f_t(X, Q)$ is still convex-concave for $t > 0$.

The barrier method consists of solving a sequence of unconstrained minimax problems with objective $f_t(X, Q)$ with a gradually increasing parameter t.

We use the infeasible start Newton method [18], outlined in Algorithm 3, to find the optimal solution of the unconstrained problem:

$$\underset{X}{\text{minimize}} \ \underset{Q}{\text{maximize}} \ f_t(X, Q). \tag{6.29}$$

The optimality conditions for the unconstrained minimax problem with objective $f_t(X, Q)$, for a fixed $t > 0$, are simply

$$\text{grad}_X \, f_t(X^\star, Q^\star) = 0, \quad \text{grad}_Q \, f_t(X^\star, Q^\star) = 0, \tag{6.30}$$

where explicit expressions for the gradients $\text{grad}_X \, f_t$ and $\text{grad}_Q \, f_t$ are included in Appendix E.5 along with the linear equations for computing the Newton step $(\Delta X_{nt}, \Delta Q_{nt})$ of the infeasible start Newton method. Intuitively, at each step, the directions $\Delta X_{nt}, \Delta Q_{nt}$ are the solutions of the first order approximation

$$0 = R(X, Q) + DR(X, Q)[\Delta X_{nt}, \Delta Q_{nt}], \tag{6.31}$$

where $R(X, Q) = \text{grad} \, f_t(X, Q)$ is the optimality residual. Then, a backtracking line search is performed on the norm of the residual along the Newton step direction.

Finally, the structure of the problem allows us to easily identify a strictly feasible initial point (X^0, Q^0) where $Q^0 = 0$ and X^0 satisfies

$$(C\Sigma_{xx}C^T + D\Sigma_{yy}D^T + \Sigma_\eta)X^0 = C\Sigma_{xx}. \tag{6.32}$$

For details on the convergence of the infeasible start Newton method and the barrier method for convex-concave games, we refer the reader to [41, 18].

Algorithm 3 Barrier method with infeasible start Newton method.

given: starting points X^0 as in (6.32) and $Q^0 = 0$, tolerance $\epsilon, \delta > 0$, $t = t^0 > 0$, $\mu > 1$, $\alpha \in (0, 1/2)$, $\beta \in (0, 1)$.
Repeat
  Repeat
    1. Compute residual $R(X, Q) = \operatorname{grad} f_t(X, Q)$ as in (E.18).
    2. Compute Newton steps $(\Delta X_{nt}, \Delta Q_{nt})$.
    3. Backtracking line search on $\|R\|_F$.
       $s = 1$.
       $(X_s, Q_s) = (X, Q) + s(\Delta X_{nt}, \Delta Q_{nt})$.
       While $\|R(X_s, Q_s)\|_F > (1 - \alpha s)\|R(X, Q)\|_F$
          $s = \beta s$.
          $(X_s, Q_s) = (X, Q) + s(\Delta X_{nt}, \Delta Q_{nt})$.
       EndWhile
    4. Update: $(X, Q) = (X, Q) + s(\Delta X_{nt}, \Delta Q_{nt})$.
  until $\|R(X, Q)\|_F \leq \delta$
  Increase t by $t = \mu t$.
until $1/t < \epsilon$

6.5 Numerical examples

In this section, we present two numerical examples which shed light on the differences between the Covariance Intersection (CI) and the proposed Robust Fusion (RF). First, consider the example of fusing two random variables with means and covariances

$$\widetilde{x} = \widetilde{y} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ \Sigma_{xx} = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \ \Sigma_{yy} = \begin{bmatrix} 3 & 0 \\ 0 & 7 \end{bmatrix}. \tag{6.33}$$

Let $(\widetilde{z}_{CI}, \Sigma_{CI})$ and $(\widetilde{z}_{RF}, \Sigma_{RF})$ be the fused estimates and the corresponding error covariances obtained from Covariance Intersection and Robust Fusion. We have that

$$\widetilde{z}_{CI} = \widetilde{z}_{RF} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ \Sigma_{CI} = \begin{bmatrix} 3.79 & 0 \\ 0 & 5.79 \end{bmatrix}, \ \Sigma_{RF} = \begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix}, \tag{6.34}$$

Although $\operatorname{tr}(\Sigma_{CI})$ is less than each of $\operatorname{tr}(\Sigma_{xx})$ and $\operatorname{tr}(\Sigma_{yy})$, we see that the produced upper bound on the error covariance is very conservative

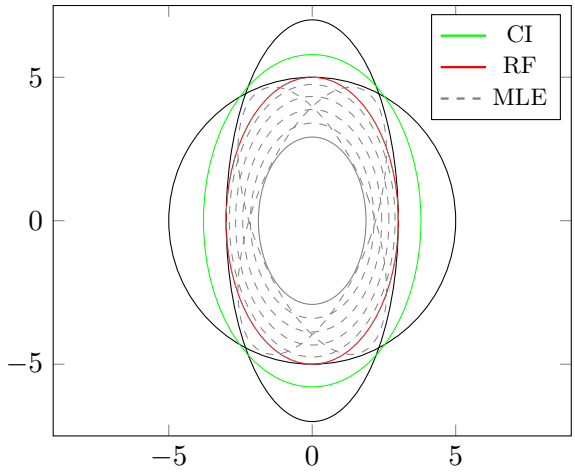


Figure 6.1: Confidence ellipses: given a covariance matrix Σ we draw the set $\{x : x^T\Sigma^{-1}x = 1\}$. Initial confidence ellipse (black), Maximum likelihood Estimate (MLE) confidence ellipse (gray dashed) for various values of correlation, CI confidence ellipse (green) and RF confidence ellipse (red). The confidence ellipses obtained from MLE lie in the intersection of the two ellipsoids $\{x : x^T\Sigma_{xx}^{-1}x \leq 1\}$ and $\{x : x^T\Sigma_{yy}^{-1}x \leq 1\}$. The RF confidence ellipse lies in the intersection of the two ellipsoids as well. When correlation increases, the trace of the covariance of MLE approaches the trace of Σ_{RF}.

In the second example, we consider the case of partial measurements. More specifically, using notation of Section 6.3, let

$$\widetilde{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ \Sigma_{xx} = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \tag{6.35}$$

and $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $z = \widetilde{z} = 0$, $\Sigma_{yy} = 1$, $D = 1$ and $\Sigma_\eta = 0$. Both Covariance Intersection and Robust Fusion yield $\widetilde{z}^+ = 0$ but

$$\Sigma_{CI} = \begin{bmatrix} 3 & 0 \\ 0 & 6 \end{bmatrix}, \ \Sigma_{RF} = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix}. \tag{6.36}$$

We observe that despite having a measurement of only the first coordinate, the error variance of the second coordinate increased. The reason for this phenomenon is that the CI updates the current estimate and the associated error covariance along a predefined direction only.

Although $\mathrm{tr}(\Sigma_{CI}) < \mathrm{tr}(\Sigma_{xx})$, the bound on the true error covariance estimated by Covariance Intersection is very conservative.

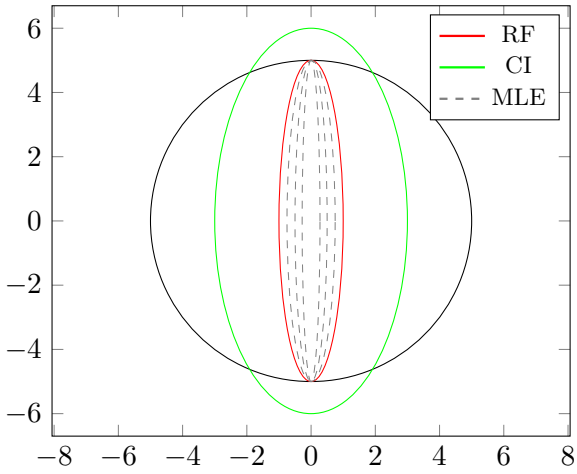


Figure 6.2: Illustration of the second numerical example of fusion under unknown correlations. Initial confidence ellipse (black), Maximum likelihood Estimate (MLE) confidence ellipse (gray dashed) for various values of correlation, CI confidence ellipse (green) and RF confidence ellipse (red).

6.6 Application in decentralized cooperative localization

Finally, we consider the application of the proposed method in distributed cooperative localization using relative position measurements. Given a group of mobile agents able to sense each other, we propose that each agent maintains only its own state estimate and thus, the cross-covariance between any two agents state estimates are not known. Despite this, the approach of Section 6.3 can be employed to update the state estimates of the agents from noisy pairwise measurements. This choice significantly simplifies the proposed cooperative localization protocol and ensures its scalability to large groups of mobile agents.

We experiment with a group of $n = 4$ mobile agents on the plane with a communication network topology as depicted in Fig. 6.3. If there is an edge from i to j, then agent i transmits its current state estimate and the corresponding error covariance estimate to agent j which, upon receipt, takes a measurement of the relative position and updates its own state estimate

and associated error covariance estimate.

We assume that each agent i can be modeled as a unicycle, i.e. its dynamics are given by

$$\dot{x}_i = v_i \cos\theta_i, \tag{6.37}$$

$$\dot{y}_i = v_i \sin\theta_i, \tag{6.38}$$

$$\dot{\theta}_i = \omega_i, \tag{6.39}$$

where $(x_i, y_i) \in \mathbb{R}^2$ are the coordinates of the position of agent i, $\theta_i \in \mathbb{R}$ denotes the orientation of agent i and v_i and ω_i denote the corresponding velocity and angular velocity of agent i. We assume that the inputs, namely v_i and ω_i contain linear additive normally distributed noise. The discrete time analog of the dynamics given in (6.37), (6.38) and (6.39), with time step T, are approximately given by

$$x_i(t+1) = x_i(t) + Tv_i(t)\cos\theta_i(t), \tag{6.40}$$

$$y_i(t+1) = y_i(t) + Tv_i(t)\sin\theta_i(t), \tag{6.41}$$

$$\theta_i(t+1) = \theta_i(t) + T\omega_i(t). \tag{6.42}$$

We assume that the first agent is equipped with global position system (GPS), that is we have a measurement of the form

$$\mathbf{y}_1(t) = \begin{bmatrix} x_1(t) \\ y_1(t) \end{bmatrix} + \eta_1(t), \tag{6.43}$$

where $\eta_1(t) \sim \mathcal{N}(0, R_1(t))$. Agent 1 performs a standard Kalman Filter update step for its own state estimate after a GPS measurement.

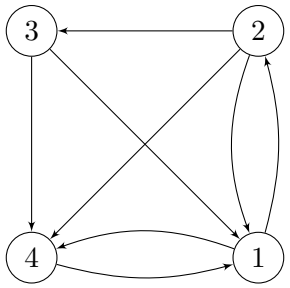


Figure 6.3: Communication network topology.

For each edge (i, j) we have a pairwise measurement of the form

$$\mathbf{y}_{ij}(t) = \begin{bmatrix} x_i(t) - x_j(t) \\ y_i(t) - y_j(t) \end{bmatrix} + \eta_{ij}(t), \tag{6.44}$$

where $\eta_{ij}(t) \sim \mathcal{N}(0, R_{ij}(t))$. Each agent can update its state estimate by either ignoring cross-correlations (Naive Fusion), by Covariance Intersection or by the proposed method of Section 6.3, which we call Robust Fusion (RF).

The individual prediction step is the same as the Kalman Filter (KF) prediction step, that is

$$\widetilde{\mathbf{x}}_i(t+1|t) = \mathbf{f}(\widetilde{\mathbf{x}}_i(t|t), u_i(t)) \tag{6.45}$$

$$\Sigma_i(t+1|t) = A_i(t)\Sigma_i(t|t)A_i(t)^T + B_i(t)Q_i(t)B_i(t)^T, \tag{6.46}$$

where $\widetilde{\mathbf{x}}_i(t+1|t)$ denotes the estimate of agent i for its state at time $t+1$ having received measurements up to time t, Σ_i is the associated error covariance, $u_i(t) = (v_i(t), \omega_i(t))$, $\mathbf{f}$ denotes the discrete-time dynamical model as defined in (6.40), (6.41) and (6.42), and

$$A_i(t) = \begin{bmatrix} 1 & 0 & -Tv_i(t)\sin\widetilde{\theta}_i(t|t) \\ 0 & 1 & Tv_i(t)\cos\widetilde{\theta}_i(t|t) \\ 0 & 0 & 1 \end{bmatrix}, \; B_i(t) = \begin{bmatrix} T\cos\widetilde{\theta}_i(t|t) & 0 \\ T\sin\widetilde{\theta}_i(t|t) & 0 \\ 0 & T \end{bmatrix}. \tag{6.47}$$

We evaluate three estimators, two decentralized and one centralized:

- Robust Fusion (RF),
- Covariance Intersection (CI) and
- Centralized Kalman Filter (CKF).

The Centralized Kalman Filter (CKF) is simply a standard Kalman Filter containing all agent states. It serves as a measure of how close the decentralized estimators are to the optimal centralized estimator. We used the following values for the noise parameters: $Q_i = 10^{-2} I_2$ for all agents, $R_1 = I_2$ and $R_{ij} = 10^{-2} I_2$ for all pairwise measurements. The Robust Fusion based estimator significantly outperforms the Covariance Intersection based estimator and achieves performance comparable with the Centralized Kalman Filter without tracking the cross-covariances between state estimates of different agents.

Agent #	CKF	RF	CI
1	0.0483 ± 0.0367 m	0.0441 ± 0.0380 m	0.0476 ± 0.0372 m
2	0.0793 ± 0.0286 m	0.0597 ± 0.0401 m	0.1131 ± 0.0637 m
3	0.0524 ± 0.0350 m	0.0673 ± 0.0472 m	0.1349 ± 0.0713 m
4	0.0588 ± 0.0357 m	0.0527 ± 0.0494 m	0.1263 ± 0.0705 m

Table 6.1: Steady state position errors of all three compared methods for the first experiment.

6.7 Conclusions

In this chapter, we addressed the problem of fusing two random vectors with unknown cross-correlations by proposing a novel minimax approach. We extended our formulation to linear measurement models and proposed an efficient method for solving the resulting minimax optimization problem. As an application, we considered the problem of decentralized cooperative localization for a group of mobile agents. The proposed estimator takes cross-correlations into account while being less conservative than the widely used Covariance Intersection. As a result, it produces more accurate estimates as numerical examples and

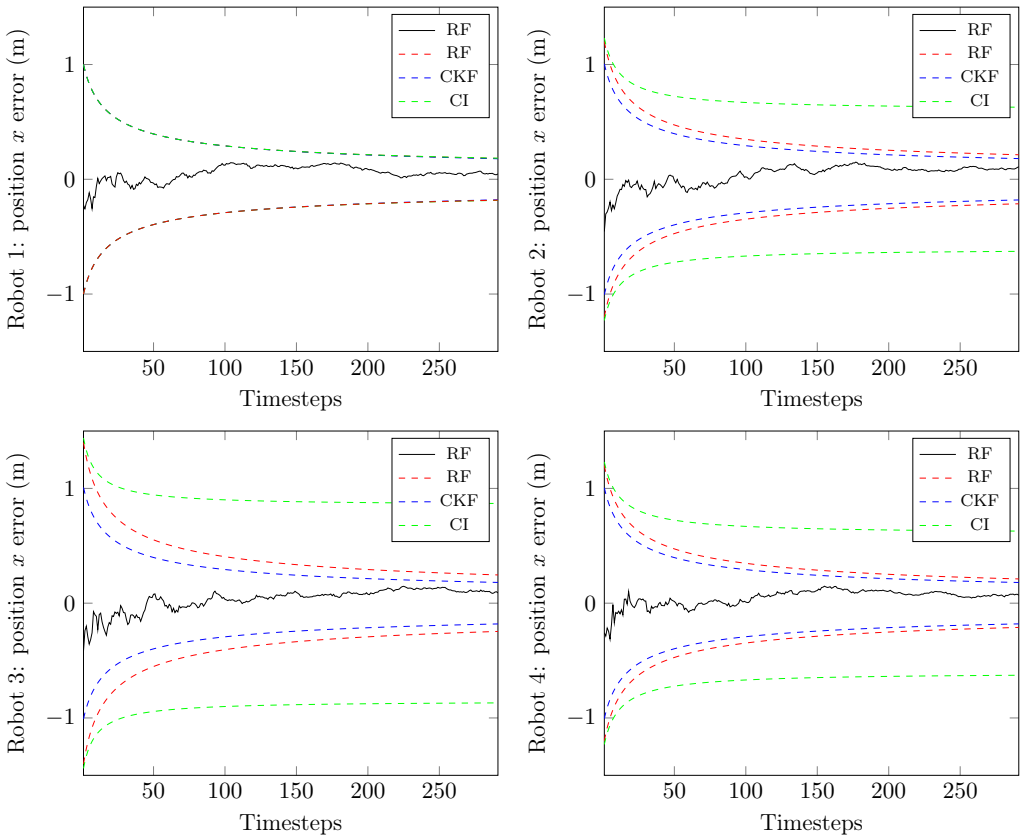


Figure 6.4: Error of the proposed method for the x coordinate of positions of the 4 agents and corresponding 3σ intervals for the proposed method, the centralized Kalman filter and the Covariance Intersection. It can be seen that the proposed method produces accurate estimates while being significantly less conservative than Covariance Intersection.

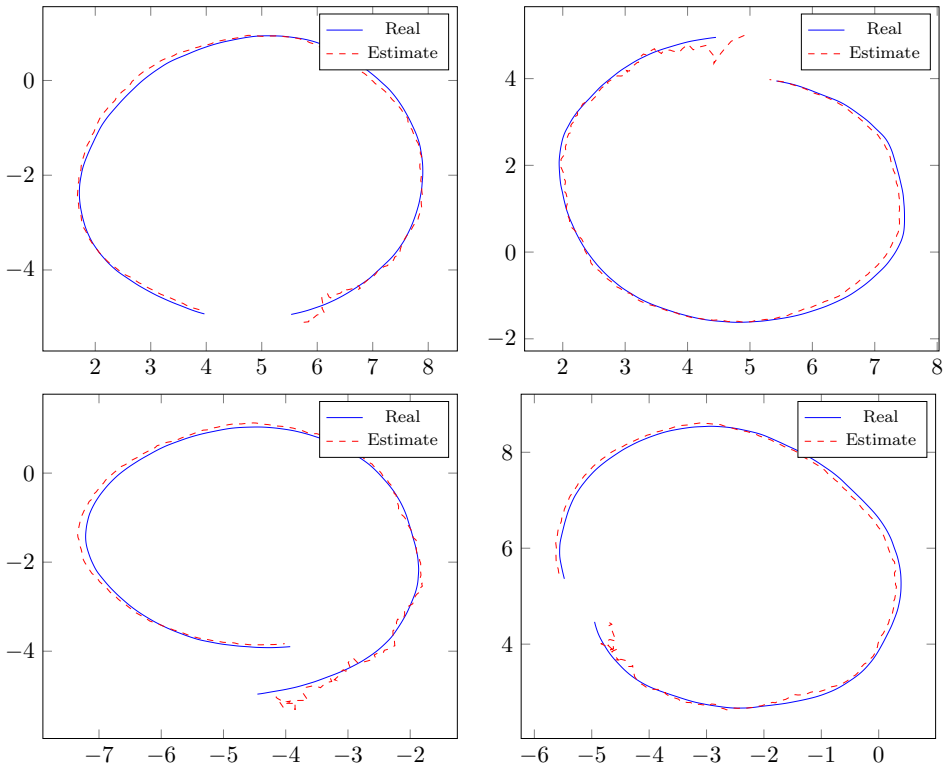


Figure 6.5: Actual and estimated trajectories for all agents produced by the proposed method.

simulations clearly demonstrated.

Chapter 7

Distributed rotation localization from bearing measurements

7.1 Introduction

The goal of this chapter is to provide distributed estimation algorithms for the 3-D bearing-only rotation localization problem. As in [106], we do not assume knowledge of any relative rotation between nodes; however, with respect to the state of the art we make the following contributions:

- We consider a large class of graph topologies that also allow the presence of bearing measurements with respect to external auxiliary points (e.g., feature points matched across cameras at different agents);
- We prove stronger localizability results than those provided in [106] (e.g., we show that the three-nodes fully-connected network is uniquely localizable);
- We compare and contrast our results with the epipolar geometry theory from computer vision;

- We provide a distributed algorithm, based on distributed Riemannian gradient descent, that can work on any localizable network.

This chapter is structured as follows: Section 7.2 includes notation and preliminaries, and Section 7.3 contains the problem statement. Sufficient localizability conditions are given in Section 7.4. The proposed distributed optimization algorithm for solving the problem at hand is the objective of Section 7.5. A spectral relaxation is discussed in Section 7.6. Finally, experimental results are presented in Section 7.7.

7.2 Notation and preliminaries

7.2.1 Graph theory

We first introduce definitions that allow us to model the types of problems for which we can show our localizability result. We assume that the network of agents can be modeled as a graph $G = (V \cup V', E \cup E')$ where $V = \{1, \dots, N\}$ represents the set of frames (agents) to localize, V' is a set of auxiliary nodes (e.g., 3-D points with unknown location), $E \subset V \times V$ represents the set of *frame-to-frame* measurements such that $(i, j) \in \mathcal{E}$ implies that node $i \in \mathcal{V}$ can sense and communicate with node $j \in V$, and $E' \subset V \times V'$ represents the set of *frame-to-point* measurements between agents that need to be localized and auxiliary nodes. We denote by $\mathcal{N}_i$ the set of frame-to-frame neighbors of agent i, that is

$$\mathcal{N}_i = \{j \in V : (i, j) \in E\}, \tag{7.1}$$

We assume that each agent $i \in V$ can freely exchange data with any of the neighbors $\mathcal{N}_i$ (auxiliary points in V' are not expected to exchange data, unless they are also agents in V).

Furthermore, we make the assumption that the graph G satisfies the following definition.

Definition 7.2.1. *A graph $G = (V \cup V', E \cup E')$ is said to be* plated *if both of the following*

conditions hold:

- *The set of edges E is symmetric, that is, $(i,j) \in \mathcal{E} \implies (j,i) \in E$;*
- *For any edge $(i,j) \in \mathcal{E}$, there exist at least one $k \in V'$ such that the two edges $(i,k),(j,k)$ are in E'.*

Note that this definition includes the important particular case where $V \cap V' \neq \emptyset$, and $E \cap E' \neq \emptyset$; in this case, nodes in $V \cap V'$ correspond to agents that need to be localized, but that can also serve as an auxiliary node for other edges.

This assumption can be somewhat relaxed by simply assuming that the graph G contains a plated graph as a subgraph (although our algorithms, as presented here, might not make use of all available measurements in this case).

Definition 7.2.1 is at the core of our sufficiency results, as it partitions the vertices in G into triples.

Definition 7.2.2. *The* set of triples *of a plated graph G is defined as $T_G = \{(i,j,k) \in V \times V \times V' : j \in \mathcal{N}_i, (i,k),(j,k) \in E'\}$.*

Each triple is ordered (i.e., the order in which i, j, and k appear is important). Every agent can become a aware of the triples it belongs to through simple distributed mechanisms (e.g., by exchanging information about possible k with every neighbor $j \in \mathcal{N}_i$, assuming i is always the first index for its triplets). For convenience, we define the set $\mathcal{N}_{ij}$ of frame-to-point neighbors of two frame-to-frame neighbors i,j as

$$\mathcal{N}_{ij} = \{k : (i,j,k) \in T_G\}. \tag{7.2}$$

7.2.2 Poses, points and vectors

Let $\mathcal{W}$ represent an arbitrary global reference frame. For each agent $i \in \mathcal{V}$, we associate a reference frame $\mathcal{B}_i$, and we define the pose $(R_i, T_i) \in SE(3)$ to be the rigid body transfor-

mation from $\mathcal{B}_i$ to $\mathcal{W}$; $R_i \in SO(3)$ is a rotation, while $T_i \in \mathbb{R}^3$ is a translation. Note that T_i can be also interpreted as the coordinates of agent i in the world frame.

We denote as ${}^w x$, and ${}^i x$ the coordinates of a point x expressed, respectively, in the world and i-th local frame. With this notation, a point $x \in \mathbb{R}^3$ transforms under the pose (R_i, T_i) according to

$$ {}^w x = R_i {}^i x + T_i. \tag{7.3}$$

In this work, it will be necessary to distinguish between points and vectors. We define a vector to be the difference of two points. The coordinates for a vector in a given reference frame can be obtained by taking the difference of the coordinates of the points in that frame; it then follows from (7.3) that vectors transform across reference frames according to

$$ {}^w v = R_i {}^i v, \tag{7.4}$$

that is, they are not affected by the translation component.

7.3 Problem statement

Let t_{ij} denote the normalized vector between T_j and T_i, that is,

$$ {}^w t_{ij} = \frac{T_j - T_i}{\|T_j - T_i\|_2}. \tag{7.5}$$

We assume that, for every $(i, j) \in \mathcal{E} \cup E'$, agent i measures the bearing ${}^i t_{ij}$ of agent j in its own reference frame, which, according to (7.4), is given by:

$$ {}^i t_{ij} = R_i^T {}^w t_{ij}. \tag{7.6}$$

Using this relation and the definition (7.5), we obtain the following relation between the bearings associated to the two edge directions between the same pair of nodes i, j:

$$R_i {}^i t_{ij} = -R_j {}^j t_{ji}. \tag{7.7}$$

Throughout the rest of this chapter, we make the standing assumption that the locations of the nodes in $V \cup V'$ are in general position. More precisely, we assume that for every triplet $(i, j, k) \in T_G$, the locations T_i, T_j, T_k do not belong to a common line in $\mathbb{R}^3$. Thanks to this assumption, each triplet $(i, j, k) \in T_G$ uniquely defines a plane; let n_{ijk} denote a normal vector to this plane.

Note that t_{ij}, t_{ji}, t_{ik}, t_{jk} all belong to the plane of triplet $(i, j, k) \in T_G$. Using the properties of the cross product in $\mathbb{R}^3$ (denoted as $\times$), the normal n_{ijk} can be separately computed in the i-th and j-th local frame as:

$$ {}^i n_{ijk} = R_i^T {}^w n_{ijk} = ({}^i t_{ij} \times {}^i t_{ik}) / \| {}^i t_{ij} \times {}^i t_{ik} \|_2, \tag{7.8}$$

$$ {}^j n_{ijk} = R_j^T {}^w n_{ijk} = ({}^j t_{jk} \times {}^j t_{ji}) / \| {}^j t_{jk} \times {}^j t_{ji} \|_2, \tag{7.9}$$

$$ {}^w n_{ijk} = ({}^w t_{ij} \times {}^w t_{ik}) / \| {}^w t_{ij} \times {}^w t_{ik} \|_2, \tag{7.10}$$

see also Fig. 7.1 for a graphical representation. It is important to notice that the ordering of the cross products in (7.8) and (7.9) is important, as they guarantee that ${}^i n_{ijk}$ and ${}^j n_{ijk}$ are the local expression of the same world-frame vector ${}^w n_{ijk}$ (i.e., they do *not* correspond to opposite orientations of the triangle). Moreover, (7.8) and (7.9) can be readily and independently computed at each one of the nodes i and j, without any knowledge of the absolute or relative poses of the agents.

The goal of this work is to solve the following rotation localization problem [106]:

Problem 1. *Given a set of nodes $G = (V, E)$, and a set of local bearings $\{{}^i t_{ij}\}_{(i,j)\in\mathcal{E}}$, find matrices $\{R_i\}_{i\in\mathcal{V}}$ that represent the rotation from the local reference frames $\{\mathcal{B}_i\}$ to a*

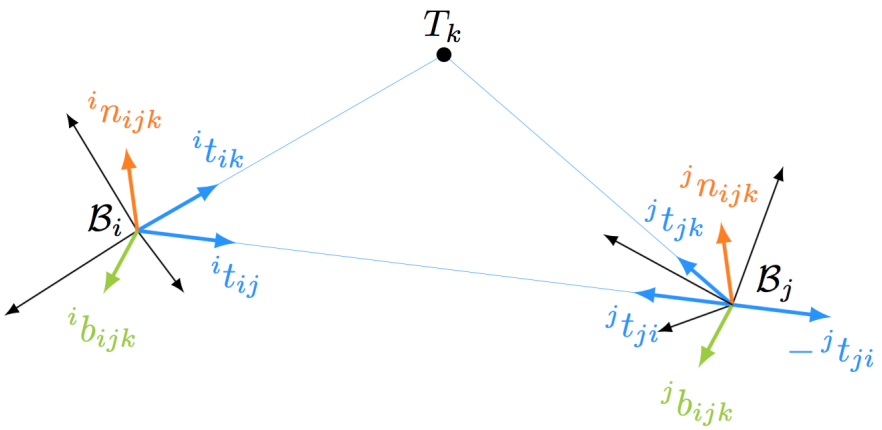


Figure 7.1: Illustration of the measurements available to the nodes in a triplet. Black arrows: local reference frames at nodes i and j; cyan arrows: available bearing measurements; orange arrows: normals obtained from the bearing measurements.

common global reference frame $\mathcal{W}$.

Note that the conditions for *translation localizability*, and the closely related concept of *bearing rigidity*, are well understood [132]. However, these conditions are not the same as those for rotation localizability.

7.4 Sufficient localizability conditions

The commonly accepted technical definition for rotation localizability of a network is the following [135, 106].

Definition 7.4.1. *A network is said to be* rotation localizable *if, given the rotation* $R_{\hat{\imath}}$ *for a node* $\hat{\imath} \in V$*, every rotation* R_i*,* $i \in \mathcal{V}$ *is uniquely determined by the available measurements.*

The requirement on $R_{\hat{\imath}}$ is necessary to fix the ambiguity in the choice of a global reference frame.

We give the following main theoretical result for our setting (3-D bearing only measurements):

Theorem 7.4.2. *The plated graph* $G = (V \cup V', E \cup E')$ *is rotation localizable if the subgraph*

$G_V = (V, E)$ *contains a spanning tree.*

The proof of Theorem 7.4.2 is based on the following lemma, which considers the localizability of two nodes in a triple.

Lemma 7.4.3. *Let (i, j, k) be a triple in T_G. Then the rotation $R_{ij} = R_i^T R_j$ can be uniquely determined from the measurements $\{{}^i t_{ij}, {}^j t_{ji}, {}^i t_{ik}, {}^j t_{jk}\}$.*

Proof. The proof of the lemma is constructive (i.e., it directly provides a way to compute R_{ij}). Define the *binormal* to the plane for triple $(i, j, k) \in T_G$ as

$$b_{ijk} = n_{ijk} \times t_{ij}; \tag{7.11}$$

due to the properties of the cross product, b_{ijk} is orthogonal to both n_{ijk} and t_{ij}. Since n_{ijk} and t_{ij} have both unit norm, the vector b_{ijk} has unit norm as well. As a consequence, the tree vectors $(t_{ij}, n_{ijk}, b_{ijk})$ define orthonormal axes.

The binormal b_{ijk} can be independently computed in the reference frames $\mathcal{B}_i$ and $\mathcal{B}_j$ as:

$${}^i b_{ijk} = {}^i n_{ijk} \times {}^i t_{ij}, \tag{7.12}$$

$${}^j b_{ijk} = {}^j n_{ijk} \times -{}^j t_{ji}. \tag{7.13}$$

Again by using the rule for transforming vectors (7.4), we have

$${}^i b_{ijk} = R_i^T R_j {}^j b_{ijk}. \tag{7.14}$$

Let the two matrices $Q_i, Q_j \in \mathbb{R}^{3\times 3}$ be defined as

$$Q_i = \begin{bmatrix} {}^i t_{ij} & {}^i n_{ijk} & {}^i b_{ijk} \end{bmatrix}, \tag{7.15}$$

$$Q_j = \begin{bmatrix} -{}^j t_{ji} & {}^j n_{ijk} & {}^j b_{ijk} \end{bmatrix}; \tag{7.16}$$

these matrices contain the coordinates of the triple of axes $(t_{ij}, n_{ijk}, b_{ijk})$ in the reference frames of the two nodes.

Using Q_i, Q_j, we can combine (7.7), (7.8), (7.9) and (7.14) into the matrix equation $Q_i = R_{ij}Q_j$, from which we can uniquely compute R_{ij} as:

$$R_{ij} = Q_i Q_j^T. \tag{7.17}$$

This expression depends exclusively on measurements available at nodes i and j (in other words, the specific world reference frame used for the derivation does not matter), thus proving the claim. □

Remark 7.4.4. *The matrices Q_i and Q_j have an intuitive interpretation; let $\mathcal{W}_{ij}$ be a reference frame co-centered with $\mathcal{B}_i$ and having the x, y, and z axes aligned with t_{ij} (which is the same as $-t_{ji}$), n_{ijk}, and b_{ijk}, respectively (see Figure 7.1); then Q_i (respectively, Q_j) represents the rotation from $\mathcal{W}_{ij}$ to B_i (respectively, B_j). Additionally, by construction and the properties of the cross product, in $\mathcal{W}_{ij}$, T_j will be located along the x axis, while T_k will have zero y component, and negative z component.*

Given the above, the proof of the theorem is quite simple.

Proof. (Theorem 7.4.2) Let $G_{st} \subset G_V$ be the spanning tree of the plated graph G. Due to Definition 7.2.1, G_{st} is undirected. After fixing $R_{\hat{\imath}}$, we can use Lemma 7.4.3, and in particular (7.17), to uniquely determine the rotation R_j of each neighbor $j : (\hat{\imath}, j) \in E$. This process can be inductively repeated until all rotations R_i, $i \in \mathcal{V}$ are determined. □

7.4.1 Relation with epipolar geometry

This section explains the relations between the results of this work (in particular Lemma 7.4.3) and those in traditional computer vision.

We start by reviewing the following fundamental result in standard computer vision geometry[82, 80]:

Proposition 7.4.5. *Given at least five image correspondences from two views (nodes) i and j, it is possible to estimate, up to sign, the* essential matrix

$$E_{ij} = {}^{i}\hat{t}_{ij} R_{ij} \in \mathbb{R}^{3\times 3}. \tag{7.18}$$

Note that we used the notation $\hat{v} \in \mathbb{R}^{3\times 3}$ to indicate the skew-symmetric matrix defined such that $\hat{v}w = v \times w$ for any $w \in \mathbb{R}^3$. In computer vision, the normalized translation ${}^{i}t_{ij}$ is also known as the *epipole*. In the terminology used in this work, the assumption in Proposition 7.4.5 translates into having at least five auxiliary vertices $k_l \in V'$, $l \in \{1, \dots, 5\}$, such that $(i, k_l), (j, k_l) \in E'$, but without necessarily assuming $(i, j) \in E$.

The essential matrix E_{ij} can be estimated from the image correspondences using standard algorithms [82, 45]. Once the essential matrix is obtained, it can be decomposed according to the following:

Proposition 7.4.6. *For any essential matrix E there exist four valid decompositions, in the sense that there are four pairs of rotations and normalized translations $\{(R_l, t_l)\}$, $l \in 1, \dots, 4$ such that $E = \pm\hat{t}_l R_l$. Of these, only one satisfies the* chirality *constraints, i.e., only one implies that the 3-D reconstructions from the image correspondences are in front of both cameras.*

The four pairs are also known as the *twisted pair ambiguity*, and, with respect to the "true" decomposition, differ by rotations of 180° and inversion of the translation [82, 133], see Figure 7.2 for a pictorial example.

Given these basic results from computer vision, it might be somewhat surprising that, in our setting, we can uniquely estimate the relative rotation between two poses (Lemma 7.4.3), without having to estimate the essential matrix, or considering the twisted pair ambiguity and the cheirality constraint. However, the key difference in our setting is that the *epipole*

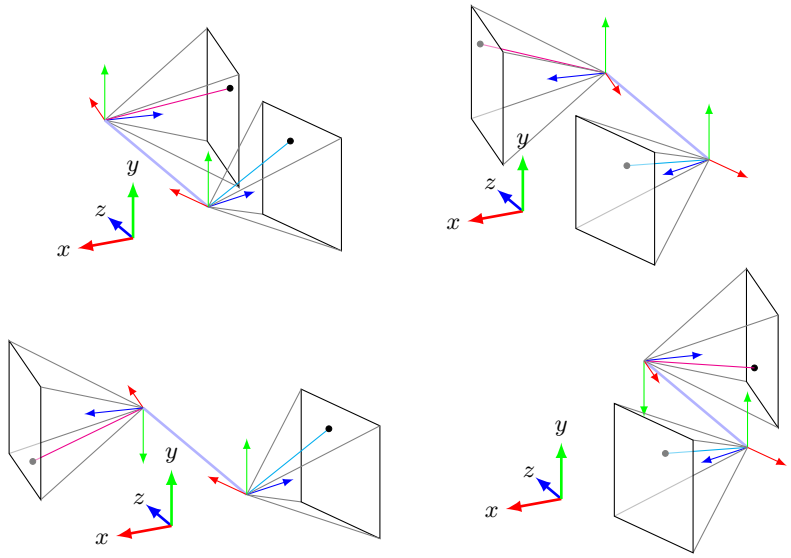


Figure 7.2: Illustration of the twisted pair ambiguity (reproduced from [133]).

${}^{i}t_{ij}$ (direction of the translation) is assumed to be directly observed (i.e., $(i,j) \in E$). This not only fixes the unknown translation, but also fixes two degrees of freedom for the rotation; as a result, a single image correspondence (i.e., auxiliary point) instead of five is sufficient for a pair of agents.

Moreover, in our case the twisted pair ambiguity is solved by two facts. First, the observation of the epipole in $\mathcal{B}_i$ fixes its sign, thus excluding two out of four solutions. Intuitively, in our case, the two remaining solutions correspond to the fact that, geometrically, one could select an opposite normal at $\mathcal{B}_j$; however, this is prevented by the observation of the epipole in $\mathcal{B}_j$, and the ordering of the agents (with the cheirality constraint equivalent to the fact that T_k must have negative z component in $\mathcal{W}_{ij}$, see Remark 7.4.4).

subsectionRelation with [106]

In Lemma 14 of [106], it is stated that for a network graph G_3 with three nodes and a complete sensing graph, there are two solutions that are compatible with the bearing measurements. However, since this graph G_3 is plated by Definition 7.2.1, according to Theorem 7.4.2 there is only one solution (up to a choice of a global reference frame). The main difference between the two results is that Lemma 14 in [106] only uses the bearing

constraints given by (7.19), and not the ones from the normals in (7.20). The secondary difference is that our most basic result (Lemma 7.4.3) involves three nodes but only four measurements (instead of six). Finally, the result from [106] makes some generic assumptions on axes of rotations derived from the bearing measurements; these assumptions, however, are not formally verified (in fact, all the bearings are essentially coplanar, so not generic), with the results that the two solution could actually coincide (in fact, preliminary numerical simulations suggest that this is the case). A more in-depth analysis of our results with respect to [106] is left as future work.

7.5 Distributed localization algorithm

In this section, we restate the constraints contained in each triple of nodes in T_G to derive a global cost φ that measures the consistency of a given localization with these constraints, and we provide a distributed gradient descent algorithm on $SO(3)^N$ that recovers a localization by minimizing this cost.

7.5.1 Constraints and cost function

To reiterate, (7.7), (7.8), (7.9) and (7.14), give the following constraints on the absolute orientations $R_i, R_j \in SO(3)$:

$$R_i{}^i t_{ij} = -R_j{}^j t_{ji}, \tag{7.19}$$

$$R_i{}^i n_{ijk} = R_j{}^j n_{ijk}, \tag{7.20}$$

$$R_i{}^i b_{ijk} = R_j{}^j b_{ijk}. \tag{7.21}$$

For each edge $(i, j) \in \mathcal{E}$, we can encode the constraints (7.19), (7.20) and (7.21) into a cost that penalizes their squared Euclidean norm. Since ${}^i t_{ij}$, ${}^j t_{ji}$, ${}^i n_{ijk}$, ${}^j n_{ijk}$, ${}^i b_{ijk}$, ${}^j b_{ijk}$ have

unit norm, and R_i, R_j are orthogonal matrices, we can simplify this cost into the following:

$$\varphi_{ij}(R_i, R_j) = \operatorname{tr}(R_i M_{ij} R_j^T), \tag{7.22}$$

where $M_{ij} \in \mathbb{R}^{3\times 3}$ is defined by

$$M_{ij} = -\frac{1}{|\mathcal{N}_{ij}|} \sum_{k\in\mathcal{N}_{ij}} (-w_t{}^i t_{ij}{}^j t_{ji}^T + w_n{}^i n_{ijk}{}^j n_{ijk}^T + w_b{}^i b_{ijk}{}^j b_{ijk}^T), \tag{7.23}$$

where $w_t, w_n, w_b \geq 0$ are scalar weights.

Remark 7.5.1 (Symmetry of pairwise objective)**.** *Observe that since $M_{ji} = M_{ij}^T$, the pairwise cost φ_{ij} is symmetric in the sense that*

$$\varphi_{ji}(R_j, R_i) = \varphi_{ij}(R_i, R_j). \tag{7.24}$$

We finally propose to recover the unknown rotations $\{R_i\}_{i=1}^N \in SO(3)^N$ by minimizing the objective

$$\varphi\big(\{R_i\}_{i=1}^N\big) = \sum_{\{i,j\}\in E} \varphi_{ij}(R_i, R_j), \tag{7.25}$$

with distributed gradient descent on $SO(3)^N$ with constant step-size. Note that we used the unordered set notation $\{i, j\} \in E$ instead of the ordered set notation $(i, j) \in E$ in order to emphasize that each pairwise objective in the total cost is counted only once, since pairwise objective are symmetric (see Remark 7.5.1).

Remark 7.5.2 (Connection with rotation synchronization)**.** *Rotation synchronization refers to the problem of estimating absolute orientations $R_1, \ldots, R_N \in SO(3)$ from noisy relative measurements $\{\widetilde{R}_{ij}\}_{\{i,j\}\in E}$. When the cost function is chosen to be the squared Frobenius norm, we obtain the following optimization problem*

$$\min_{R_1,\ldots,R_N\in SO(3)} \quad \sum_{\{i,j\}\in E} \|\widetilde{R}_{ij} - R_i^T R_j\|_F^2. \tag{7.26}$$

Problem (7.26) *is equivalent, since* $R_1, \ldots, R_N$ *are orthogonal matrices, to the following problem*

$$\min_{R_1,\ldots,R_N \in SO(3)} \quad -\sum_{\{i,j\}\in E} \operatorname{tr}(R_i \widetilde{R}_{ij} R_j^T). \tag{7.27}$$

Note that when $w_t = w_n = w_b = 1$, $M_{ij} = \dfrac{1}{|\mathcal{N}_{ij}|}\sum_{k\in\mathcal{N}_{ij}} R_{ij}$ *where* R_{ij} *is the relative rotation defined in* (7.17)*. Moreover, if for each pair of agents* $(i,j) \in E$*, we use only one auxiliary node* k*, then, the minimization of the objective* $\varphi(\{R_i\}_{i=1}^N)$ *over* $SO(3)^N$ *is identical to Problem* (7.27)*.*

7.5.2 Algorithm

In this subsection, we present the numerical algorithm for minimizing the objective function $\varphi\big(\{R_i\}_{i=1}^N\big)$. First, we compute the Riemannian gradient of the objective function. Then, we present the iterations of Riemannian gradient descent with constant step-size for this particular problem, and finally, we discuss the choice of step-size that guarantees convergence to a critical point.

The Riemannian gradient of $\varphi_{ij}(R_i, R_j)$ with respect to R_i, denoted by $\operatorname{grad}_{R_i}\varphi_{ij}(R_i,R_j)$, can be computed as

$$\operatorname{grad}_{R_i}\varphi_{ij}(R_i,R_j) = 2R_i \operatorname{skew}(R_i^T R_j M_{ij}^T), \tag{7.28}$$

where skew$(\cdot)$ denotes the skew-symmetric part of a matrix. For details on gradient computation of a real-valued function defined on a Riemannian manifold we refer the reader to [135, 2]. The gradient of the objective function with respect to R_i can be computed as:

$$\operatorname{grad}_{R_i}\varphi(\{R_i\}_{i=1}^N) = 2R_i \sum_{j\in\mathcal{N}_i} \operatorname{skew}(R_i^T R_j M_{ij}^T). \tag{7.29}$$

The Riemannian gradient descent method [135] with constant step-size ϵ at iteration t has

the general form:

$$W_i(t) = -\operatorname{grad}_{R_i} \varphi(\{R_j(t)\}_{j=1}^N), \tag{7.30}$$

$$R_i(t+1) = \exp_{R_i(t)}(\epsilon W_i(t)), \tag{7.31}$$

for all $i = 1, 2, \ldots, N$. In the specific case under consideration, equation (7.29) implies that the gradient descent at iteration t is as follows:

$$R_i(t+1) = R_i(t) \exp\big(-2\epsilon \sum_{j \in \mathcal{N}_i} \operatorname{skew}(R_i(t)^T R_j(t) M_{ij}^T)\big), \tag{7.32}$$

for all $i = 1, 2, \ldots, N$.

Choice of step-size. At this point, we discuss the choice of the step-size ϵ that guarantees convergence of the proposed algorithm. First, we need the following lemma from [131].

Lemma 7.5.3 (Adapted from Tron et al. [131])**.** *Assume that for all $\{R_i\}_{i=1}^N \in SO(3)^N$, the maximum eigenvalue, denoted by $\lambda_{\max}$, of the Hessian of φ, as defined in* (7.25)*, can be upper-bounded by some positive constant L. Then, for step-size $\epsilon \in (0, 2/L)$, every limit point of the sequence generated by* (7.32) *is a critical point for the problem of minimizing φ over $SO(3)^N$.*

We derive a value for the uniform upper-bound L in the following lemma.

Lemma 7.5.4. *Let $d_{\max}$ denote the maximum degree of graph G, that is, $d_{\max} = \max_{i \in V} |\mathcal{N}_i|$, and let $w_{\max} = \max\{w_t, w_n, w_b\}$. Then, for all $\{R_i\}_{i=1}^N \in SO(3)$, the maximum eigenvalue of the Hessian can be upper-bounded as follows:*

$$\lambda_{\max}(\operatorname{Hess} \varphi(\{R_i\}_{i=1}^N)) \leq 4 w_{\max} d_{\max}. \tag{7.33}$$

A detailed proof of the above lemma is included in Appendix F.1 and it is based on the definition of the Riemannian Hessian and on Gershgorin circle theorem. A corollary of the

two previous lemmata now follows.

Corollary 7.5.5. *For any step-size ϵ satisfying*

$$0 < \epsilon < \frac{1}{2w_{\max}d_{\max}}, \tag{7.34}$$

every limit point of the gradient descent scheme (7.32) *is a critical point of $\varphi(\{R_i\}_{i=1}^N)$.*

Remark 7.5.6. *Note that all the quantities in the bound in* (7.34) *can be computed in a distributed way using max-consensus algorithms [26] in a finite number of steps (equal to the diameter of the graph).*

7.6 Initialization by spectral relaxation

In this section, we propose a spectral relaxation for the problem at hand which in turn, can be solved in a decentralized fashion by the method of Chapter 5. First, we reformulate the problem at hand as a quadratic optimization problem over $SO(3)^N$ by including only one auxiliary node k for every pair $(i,j) \in E$. The solution of the reformulated problem, in the noiseless case, is given by the 3 leading eigenvectors of a properly defined symmetric matrix.

First, consider a pair of agents $(i,j) \in E$ and an auxiliary node k perceived by both agents i and j. Moreover, let Q_i and Q_j as in (7.15) and (7.16) respectively. Then, constraints (7.7), (7.8), (7.9) and (7.14) imply that R_i and R_j must satisfy:

$$R_j Q_j = R_i Q_i. \tag{7.35}$$

Based on these constraints only and under the choice of squared Frobenius norm penalty, we formulate rotation localization as the following optimization problem on $SO(3)^N$:

$$\underset{R_1,\ldots,R_N \in SO(3)}{\text{minimize}} \quad \sum_{\{i,j\}\in E} \|R_j Q_j - R_i Q_i\|_F^2. \tag{7.36}$$

Consider the block symmetric matrix $\mathbf{B} \in \mathbb{R}^{3N \times 3N}$ defined by

$$\mathbf{B}_{ij} = \begin{cases} I_3, & i = j \\ Q_i Q_j^T, & (i,j) \in E \\ \mathbf{0}_{3\times 3}, & (i,j) \notin E \end{cases} \tag{7.37}$$

Then, since $R_1, R_2, \ldots, R_N$ are orthogonal matrices, Problem (7.36) is equivalent to the following quadratic optimization problem over $SO(3)^N$:

$$\underset{R_1,\ldots,R_N \in SO(3)}{\text{maximize}} \quad \sum_{\{i,j\}\in E} \operatorname{tr}(R_i \mathbf{B}_{ij}^T R_j^T). \tag{7.38}$$

Next, we present the spectral relaxation of Problem (7.38). The following lemma sheds light on how to obtain a spectral relaxation of Problem (7.38).

Lemma 7.6.1 (Adapted from [7])**.** *Let Δ be the degree matrix of the plated graph G defined as the diagonal matrix with entries $|\mathcal{N}_1|, |\mathcal{N}_2|, \ldots, |\mathcal{N}_N|$. Then, in the noiseless case, the matrix $(I+\Delta)^{-1}\mathbf{B}$ has exactly 3 leading eigenvalues equal to 1. Furthermore, there exist rotations $R_1, R_2, \ldots, R_N \in SO(3)$, unique up to a global rotation, satisfying*

$$(I+\Delta)^{-1}\mathbf{B}\mathbf{R} = \mathbf{R}, \tag{7.39}$$

where $\mathbf{R} = \begin{bmatrix} R_1 & R_2 & \cdots & R_N \end{bmatrix}^T \in \mathbb{R}^{3N\times 3}$.

For a proof of Lemma 7.6.1 we refer the reader to [7]. Based on the preceding lemma, under relaxed orthogonality and determinant constraints, an approximately optimal solution to Problem (7.38) is given by the 3 leading eigenvectors of $(I+\Delta)^{-1}\mathbf{B}$. Since, $(I+\Delta)^{-1}\mathbf{B}$ is not, in general, a symmetric matrix, the approach of [71] is not directly applicable. However, $(I+\Delta)^{-1}\mathbf{B}$ is similar to the symmetric matrix $(I+\Delta)^{-1/2}\mathbf{B}(I+\Delta)^{-1/2}$. Let $V\Lambda V^T$ be an eigendecomposition of $(I+\Delta)^{-1/2}\mathbf{B}(I+\Delta)^{-1/2}$, where V is an orthogonal matrix and Λ a

diagonal. Then, let

$$(I+\Delta)^{-1/2}V = \begin{bmatrix} U_1 \\ \vdots \\ U_N \end{bmatrix} \doteq \mathbf{U}, \tag{7.40}$$

for some 3×3 matrices $U_1, U_2, \ldots, U_n$. In the noiseless case, span($\mathbf{R}$) and span($\mathbf{U}$) are equal, and thus, $\mathbf{R} = \mathbf{U}G^{-1}$ for some invertible matrix G. Without loss of generality, we can assume that $R_1 = I_3$ and thus, G must be equal to U_1. Based on this observation, the last step of the proposed approach consists of each agent i computing the approximately optimal $R_i^\star$ by

$$R_i^\star = \text{Proj}_{SO(3)}(U_i U_1^{-1})^T, \tag{7.41}$$

where $\text{Proj}_{SO(3)}$ denotes the projection onto set of 3×3 rotation matrices, which can be computed using Singular Value Decomposition (SVD) (see Appendix F.2).

Overall, we propose the following four-step approach for solving the spectral relaxation:

1. All agents collectively compute the 3 leading eigenvectors $V = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}$ of $(I + \Delta)^{-1/2}\mathbf{B}(I+\Delta)^{-1/2}$ using the decentralized approach of Chapter 5.

2. Each agent i computes U_i as defined in (7.40).

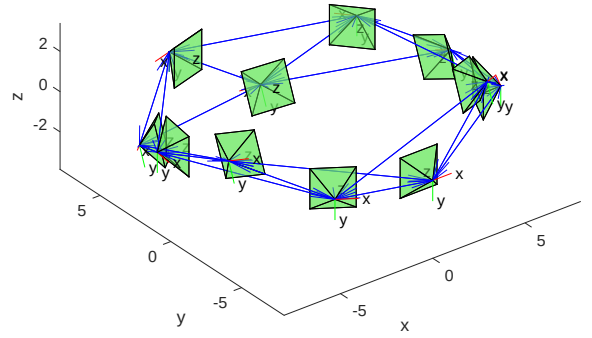
3. Agent 1 transmits U_1 to the entire group.

4. Each agent i computes $R_i^\star$ as in (7.41).

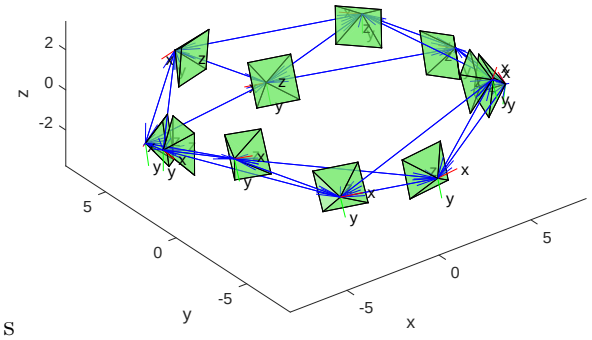
7.7 Simulations

7.7.1 Qualitative results

In this section we will present some synthetic experiments to illustrate the behavior of the proposed algorithm.



(a)



(b)

Figure 7.3: Top: actual synthetic camera network. Bottom: estimated synthetic camera network.

As a first experiment, we generate a non-planar 4-regular camera network with 11 cameras approximately located on a circle (see Fig. 7.3). Given the actual bearings ${}^{i}t_{ij}$, we add noise by

$$ {}^{i}\tilde{t}_{ij} = \exp_{{}^{i}t_{ij}}(\theta_{ij} u_{ij}), \tag{7.42} $$

where $\exp_x$ denotes the exponential map of the unit sphere at the unit norm vector x (see, e.g., [74]). We generate θ_{ij} from a normal distribution with zero mean and variance σ_θ^2. The direction u_{ij} is a uniform direction on the tangent space of ${}^{i}t_{ij}$. In the first experiment, we use $\sigma_\theta = 5$ degrees. Results are presented in Fig. 7.3. The estimated camera orientations are very close to the ground-truth.

σ_θ	Error (standard deviation)
0°	0.0188° (0.0021°)
2°	1.2861° (0.1585°)
5°	3.1733° (0.4015°)
10°	6.1272° (0.7373°)
20°	11.8944° (1.4780°)
45°	34.5653° (6.7782°)

Table 7.1: Average orientation error and corresponding standard deviation in degrees for several values of the noise parameter σ_θ.

7.7.2 Quantitative results

As a second experiment, we generate Erdos-Renyi graphs with 25 nodes and probability of each edge being present $p = 0.5$. We vary the noise parameter σ_θ in the range $\{0, 2, 5, 10, 20, 45\}$. We repeat the experiment 1000 times for each level of noise. For each trial, we collect the angle (in degrees) between the estimated and ground-truth rotations and we report the average orientation error over the nodes. The average and standard deviation of the orientation errors after the optimization are reported in Table 7.1. The corresponding histograms of orientation errors are presented in Fig. 7.4.

7.8 Conclusions

We considered the problem of estimating the orientations of a set of agents with respect to a global reference frame, using only local bearing measurements. We identified sufficient conditions for localizability, and proposed a distributed optimization approach to estimate the unknown orientations, without any prior information.

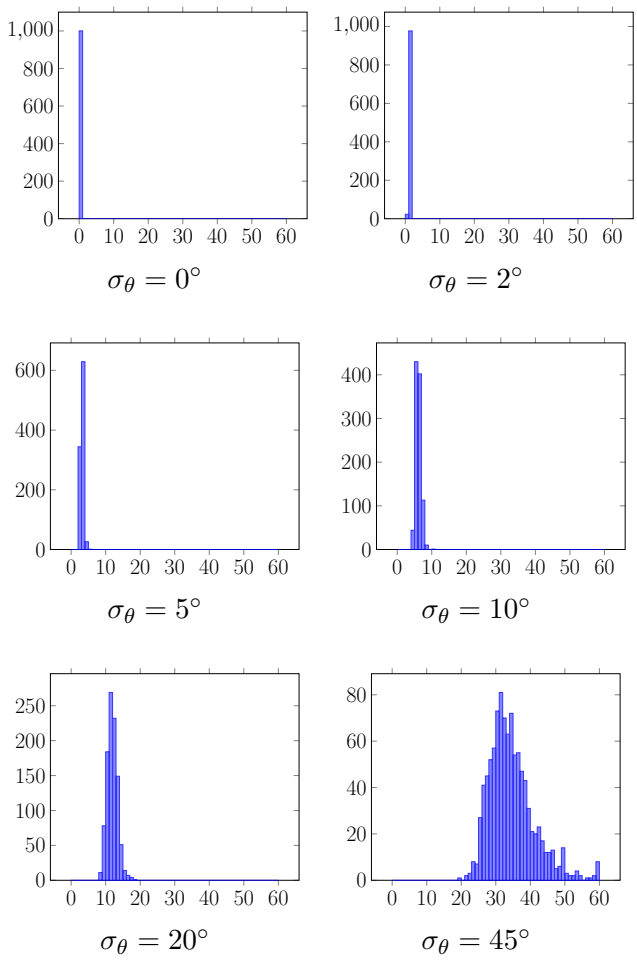


Figure 7.4: Histograms of average orientation errors (in degrees) for various values of the noise parameter σ_θ.

Chapter 8

The trifocal tensor and applications

8.1 Introduction

In this chapter, we propose a parametrization of the trifocal tensor for calibrated cameras with non-colinear pinholes based on a quotient Riemannian manifold. This parametrization is almost symmetric (we use a preferred camera only for the translations), and is derived from a particular choice of the global reference frame. We show how it can be used for refining estimates of the tensor from image data through state of the art techniques for optimization on manifolds [1]. In addition, the Riemannian structure provides a notion of distance between trifocal tensors. We show that this distance can be computed efficiently, and that it produces meaningful results in a sample Structure from Motion problem.

8.2 Derivation of the trifocal tensor

In this section, we review the derivation of the trifocal tensor that relates lines seen in three views. This derivation generalizes the one from standard textbooks [46] by not assuming that one of the camera frames coincides with the global reference frame.

Let $g_i = (R_i, T_i) \in SE(3)$ be the pose of the i-th camera such that the camera center in the global reference frame is simply given by the translation T_i. Assuming that the cameras are calibrated, the corresponding projection matrices are given by $P_i = \begin{bmatrix} R_i^T & -R_i^T T_i \end{bmatrix} \in \mathbb{R}^{3\times 4}$. Now, let $\{l_i\}_{i=1}^3$ be a set of images of three lines intersecting in 3-D. The intersection of the pre-images of the lines, that is, the three planes with normals $n_i = P_i^T l_i$, $i \in \{1, 2, 3\}$ is not empty. Then, we have that the matrix $N = \begin{bmatrix} n_1 & n_2 & n_3 \end{bmatrix} \in \mathbb{R}^{4\times 3}$ is rank-deficient. Hence, also the following matrix is rank-deficient:

$$N' = \begin{bmatrix} R_1^T & 0 \\ T_1^T & 1 \end{bmatrix} N = \begin{bmatrix} l_1 & R_1^T R_2 l_2 & R_1^T R_3 l_3 \\ 0 & T_{12}^T R_2 l_2 & T_{13}^T R_3 l_3 \end{bmatrix}, \tag{8.1}$$

where $T_{ij} = T_i - T_j$. Hence, there must be coefficients α and β such that

$$\begin{bmatrix} l_1 \\ 0 \end{bmatrix} = \begin{bmatrix} R_1^T R_2 l_2 & R_1^T R_3 l_3 \\ T_{12}^T R_2 l_2 & T_{13}^T R_3 l_3 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}. \tag{8.2}$$

From the last row, we can choose $\alpha = -T_{13}^T R_3 l_3$ and $\beta = T_{12}^T R_2 l_2$. In this way, we get

$$l_1 = l_2^T R_2^T T_{12} R_1^T R_3 l_3 - l_3^T R_3^T T_{13} R_1^T R_2 l_2. \tag{8.3}$$

We define the *canonical tensor centered on camera* 1 as

$$\mathcal{T}_i = R_2^T T_{12} e_i^T R_1^T R_3 - R_2^T R_1 e_i T_{13}^T R_3, \tag{8.4}$$

for $i \in \{1, 2, 3\}$, where e_i denotes the i-th standard basis vector in $\mathbb{R}^3$. Then, equation (8.3) becomes

$$(l_1)_i = l_2^T \mathcal{T}_i l_3, \tag{8.5}$$

where $(l_1)_i$ stands for the i-th component of vector l_1.

The canonical tensors centered on the other two cameras can be obtain by a cyclic permu-

tation of indices, that is

$$\mathcal{T}_i^{(k)} = R_{k+1}^T T_{k,k+1} e_i^T R_k^T R_{k+2} - R_{k+1}^T R_k e_i T_{k,k+2}^T R_{k+2}, \tag{8.6}$$

for $i, k = 1, 2, 3$, where indices $k, k+1, k+2$ are intended modulo 3. From now on, we will omit the superscript (1) and we will always refer to the canonical tensor centered on the first camera unless explicitly stated.

8.3 The normalized trifocal space

In this section, we define the normalized canonical decomposition of the trifocal tensor. Under the assumption of non-colinear cameras, we choose a global reference frame such that the z-axis is aligned with the normal of the plane on which the three cameras lie. Then, we define the normalized trifocal space and parametrize it with a quotient manifold.

8.3.1 The normalized canonical decomposition

First of all, we define the canonical decomposition for a trifocal tensor in the following proposition.

Proposition 8.3.1. *Any trifocal tensor admits the canonical decomposition*

$$\mathcal{T}_i = R_2^T T_{12} e_i^T R_1^T R_3 - R_2^T R_1 e_i T_{13}^T R_3, \tag{8.7}$$

where $(T_{12})_3 = (T_{13})_3 = 0$, $e_3^T (T_{12} \times T_{13}) > 0$ *and* $\|T_{12}\|_2^2 + \|T_{13}\|_2^2 = 1$.

Proof. Since T_{12}, T_{13} are invariant to global translations and since $\|R_0 T_{12}\|_2^2 + \|R_0 T_{13}\|_2^2 = \|T_{12}\|_2^2 + \|T_{13}\|_2^2$ for any $R_0 \in SO(3)$, it follows that the global scale and the global reference frame can be chosen independently. Under the assumption that the three camera centers do not coincide, we can always choose a global scale such that $\|T_{12}\|_2^2 + \|T_{13}\|_2^2 = 1$. Then,

given a tensor in the form (8.4), pick any $R_0 \in SO(3)$ that aligns the z-axis with the vector $T_{12} \times T_{13}$. Then one can verify $(R_0 T_{12})_3 = (R_0 T_{13})_3 = 0$. In conclusion, the tensor can be written as

$$\mathcal{T}_i = (R_0 R_2)^T (R_0 T_{12}) e_i^T (R_0 R_1)^T (R_0 R_3) - (R_0 R_2)^T (R_0 R_1) e_i (R_0 T_{13})^T (R_0 R_3), \tag{8.8}$$

which is in the form (8.7). □

An example of a rotation $R_0 \in SO(3)$ that aligns the z-axis with the vector $T_{12} \times T_{13}$ is as follows:

$$R_0 = \exp_I(\theta_0 \widehat{u_0} / \|u_0\|_2), \tag{8.9}$$

$$u_0 = (T_{12} \times T_{13}) \times e_3, \tag{8.10}$$

$$\theta_0 = \arccos(e_3^T (T_{12} \times T_{13}) / \|T_{12} \times T_{13}\|_2). \tag{8.11}$$

Note that the choice of R_0 in the proof is not unique: if R_0 is a rotation that satisfies the requirements, then any rotation $R_z R_0$, where R_z denotes a rotation around z-axis, will also satisfy the requirements.

Intuitively, the change of world coordinates corresponds to aligning the z-axis with the normal to the plane defined by the three cameras (which is given by $T_{12} \times T_{13}$). This plane is then parallel to the xy-plane, thus the third components of the translations become zero. For any two vectors $T_{12}, T_{13} \in \mathbb{R}^3$ such that $(T_{12})_3 = (T_{13})_3 = 0$ and $\|T_{12}\|_2^2 + \|T_{13}\|_2^2 = 1$, we will write $(T_{12}, T_{13}) \in \mathbb{S}_2^3$. If the camera centers are not colinear, then we have $(T_{12}, T_{13}) \in \mathbb{S}_2^{3*}$.

Definition 8.3.2. *We define the normalized trifocal space $\mathcal{M}_{\mathcal{T}}$ as the image of the mapping $\mathcal{T} : SO(3)^3 \times \mathbb{S}_2^3 \to \mathbb{R}^{3\times3\times3}$ defined by*

$$(R_1, R_2, R_3, (T_{12}, T_{13})) \mapsto [\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3] \tag{8.12}$$

with $\mathcal{T}_i$ as defined in (8.7)*. Since this mapping is surjective by Proposition 8.3.1, the space*

$\mathcal{M}_{\mathcal{T}}$ corresponds to the space of all trifocal tensors.

8.3.2 Ambiguities of the canonical form

The purpose of this section is to describe the ambiguities of the previously derived canonical form. In the proof of Proposition 8.3.1, we saw that the mapping from $SO(3)^3 \times \mathbb{S}_2^3$ to $\mathbb{R}^{3\times3\times3}$ as defined in (8.12) is not injective. We now state the conditions under which two configurations yield the same canonical trifocal tensor. Let $X_a, X_b \in SO(3)^3 \times \mathbb{S}_2^3$. We define the equivalence relation " $\sim$ " on $SO(3)^3 \times \mathbb{S}_2^3$ as

$$X_a \sim X_b \quad \text{iff} \quad \mathcal{T}_a = \mathcal{T}_b. \tag{8.13}$$

Then, we have the following proposition regarding the equivalence class of a point $X \in SO(3)^3 \times \mathbb{S}_2^{3*}$.

Proposition 8.3.3. *Define the groups*

$$H_z = \{(R_z(\theta), R_z(\theta), R_z(\theta), R_z(\theta)) : \theta \in (-\pi, \pi]\}, \tag{8.14}$$

$$H_{z\pi} = \{(I_3, I_3, I_3, I_3), (I_3, I_3, I_3, R_z(\pi)), \} \tag{8.15}$$

acting on the left on $SO(3)^3 \times \mathbb{S}_2^{3*}$ by componentwise multiplication. Then, given a point $X \in SO(3)^3 \times \mathbb{S}_2^{3*}$, its equivalence class with respect to " $\sim$ " is given by

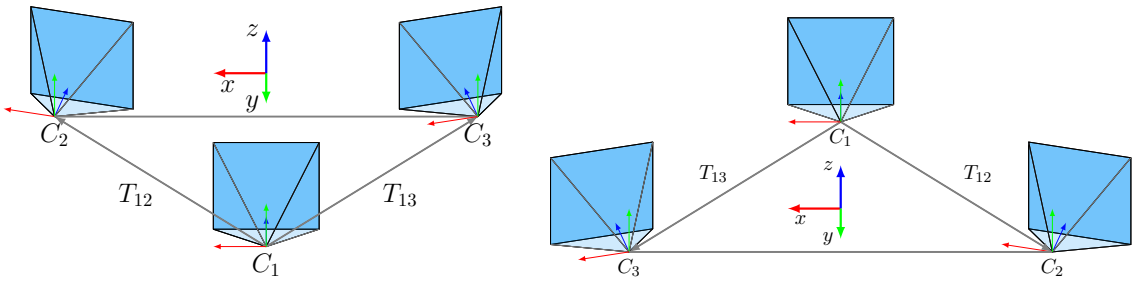
$$[X] = \{S_z S_{z\pi} X : S_z \in H_z, S_{z\pi} \in H_{z\pi}\}. \tag{8.16}$$

The above result is in accordance with the mirror image ambiguity according to which, without using the cheirality constraint, the translational parts of the trifocal tensor can be estimated only up to a sign [145]. This ambiguity corresponds to the action of $H_{z\pi}$ and it is intrinsic to the tensor estimation process. As a result, $[X]$ in (8.16) has two components. We will use $S_z = (S_{z1}, S_{z2}, S_{z3}, S_{z4})$ and $S_{z\pi} = (S_{z\pi1}, S_{z\pi2}, S_{z\pi3}, S_{z\pi4})$ to denote points in

H_z and $H_{z\pi}$ respectively. Based on Proposition 8.3.3, we propose to parametrize the space $\mathcal{M}_{\mathcal{T}}$ with the quotient space

$$\mathcal{M}_{\mathcal{T}} = (SO(3)^3 \times \mathbb{S}_2^3)/(H_z \times H_{z\pi}). \tag{8.17}$$

Remark 8.3.4. *Proposition 8.3.3 does not hold for colinear configurations, that is, for points X in the complement of $(SO(3)^3 \times \mathbb{S}_2^{3*})$ in $(SO(3)^3 \times \mathbb{S}_2^3)$. This is because, for these points, the equivalence class $[X]$ contains additional elements given by the rotation around the colinearity axis. Nonetheless, the quotient space in* (8.17) *covers all valid tensors $\mathcal{T}$. The only difficulty is that, for and only for colinear tensors, distinct points in $X_a, X_b \in \mathcal{T}$ might yield the same tensor. However, these points constitute a set of measure zero and, in practice, as we will see, this does not prevent* (8.17) *(and its signed version, which we introduce later) from being a useful parametrization.*



(a) $S_{z\pi} = (I, I, I, I)$ (b) $S_{z\pi} = (I, I, I, R_{z\pi})$

Figure 8.1: Ambiguities of the canonical form of the trifocal tensor.

8.4 The signed trifocal manifold parametrization

In this section, we use the cheirality constraint to fix the mirror image ambiguity. Intuitively, this corresponds to selecting one of the two components of each equivalence class. Then, we show that the resulting space is a Riemannian quotient manifold. Finally, we introduce geodesics, the exponential map and an efficient algorithm for computing the logarithm map.

8.4.1 Three view depth estimation

Let $X_w \in \mathbb{R}^3$ denote the coordinates of a point p in the world reference frame, $x_1, x_2, x_3 \in \mathbb{R}^3$ the normalized coordinates of point p in each of the three views and $\lambda_1, \lambda_2, \lambda_3 > 0$ the corresponding depths. Then, $X_w = \lambda_i R_i x_i + T_i$ for $i \in \{1, 2, 3\}$. By substituting the first equation into the other two, and by taking into account H_z and $H_{z\pi}$, we have

$$\lambda_2 x_2 = \lambda_1 R_2^T R_1 x_1 + R_2^T S_{z\pi 4} T_{12}, \tag{8.18}$$

$$\lambda_3 x_3 = \lambda_1 R_3^T R_1 x_1 + R_3^T S_{z\pi 4} T_{13}, \tag{8.19}$$

where the action of H_z cancels out. Then, the following proposition follows naturally.

Proposition 8.4.1. *There is only one choice of $S_{z\pi}$ for which $\lambda_1, \lambda_2, \lambda_3 > 0$.*

Proof. We know that there always exists a choice of $S_{z\pi} \in H_{z\pi}$ (the true one) such that all depths are positive. Denote this solution by $(\lambda_1, \lambda_2, \lambda_3)$. Then, the depths $(-\lambda_1, -\lambda_2, -\lambda_3)$ satisfy (8.18) and (8.19) if we now choose $S'_{z\pi} = R_z(\pi) S_{z\pi}$ (that is, the other element in $H_{z\pi}$). Thus, only one choice corresponds to positive depths. □

8.4.2 The signed trifocal manifold parametrization

In view of Proposition 8.4.1, given a point $X \in \mathcal{M}_{\mathcal{T}}$ we can always pick two of the four components of $[X]$ (the ones corresponding to the positive depths). Thus, if $\overline{\mathcal{M}}_{\mathcal{T}} = SO(3)^3 \times \mathbb{S}_2^3$, we define the signed trifocal parametrization as:

$$\mathcal{M}_{\Upsilon} = (SO(3)^3 \times \mathbb{S}_2^3)/H_z = \overline{\mathcal{M}}_{\mathcal{T}}/H_z. \tag{8.20}$$

This space admits a smooth manifold structure, as shown next.

Proposition 8.4.2. *The canonical projection $\pi : \overline{\mathcal{M}}_{\mathcal{T}} \to \mathcal{M}_{\Upsilon}$ is a smooth submersion and $\mathcal{M}_{\Upsilon}$ is a manifold of dimension 11.*

Proof. Since H_z is a compact Lie group and the action is continuous, it follows that the action is proper. Moreover, $S_{z1}R_1 = R_1$ implies $S_{z1} = I_3$. As a result, for any $X \in SO(3)^3 \times \mathbb{S}^3_2$, we have $S_z X = X$ implies that S_z is the identity element of the group. Thus, the action is also free. Finally, the action is trivially smooth and we conclude $\mathcal{M}_{\mathcal{T}}$ is a manifold of dimension:

$$\dim \mathcal{M}_{\mathcal{T}} = \dim\big(SO(3)^3 \times \mathbb{S}^3_2\big) - \dim\big(H_z\big) = 11. \tag{8.21}$$

□

Proposition 8.4.2 implies that the tangent space at a point $X \in SO(3)^3 \times \mathbb{S}^3_2$ admits the decomposition into vertical and horizontal spaces

$$T_X\overline{\mathcal{M}}_{\mathcal{T}} = \mathcal{V}_X \oplus \mathcal{H}_X. \tag{8.22}$$

We can give a closed form expression for the vertical space:

Proposition 8.4.3. *The vertical space at a point $X = (R_1, R_2, R_3, T) \in \overline{\mathcal{M}}_{\mathcal{T}}$ is given by*

$$\mathcal{V}_X = \{\lambda \widehat{e_z} \cdot (R_1, R_2, R_3, T) : \lambda \in \mathbb{R}\}, \tag{8.23}$$

where $\cdot$ denotes componentwise multiplication and $e_z = (0, 0, 1)^T$.

Proof. Let $X \in \overline{\mathcal{M}}_{\mathcal{T}}$ and note that since H_z is one dimensional, also $\mathcal{V}_X$ is one dimensional. Let $\gamma(t) = R(t) \cdot X$ be a curve in the equivalence class of X with $R(t) \in H_z$ for all t and $R(0) = I$. Then, $\dot{\gamma}(0) = \widehat{e}_z \cdot X = \widehat{e}_z X$. It follows that $\mathcal{V}_X$ is spanned by the vector $\widehat{e}_z X$. □

At this point, we will endow $T_X\overline{\mathcal{M}}_{\mathcal{T}}$ with a Riemannian metric which is necessary for defining the orthogonal projection of a vector onto the vertical and horizontal spaces, and of course for defining a metric on the signed trifocal manifold. Let any $X = (R_1, R_2, R_3, T) \in \overline{\mathcal{M}}_{\mathcal{T}}$.

A Riemannian metric $\overline{g}$ for $T_X\overline{\mathcal{M}}_{\mathcal{T}}$ can be naturally defined as

$$\overline{g}(\xi,\zeta) = \frac{1}{2}\sum_{i=1}^{3}\mathrm{tr}(\xi_i^T\zeta_i) + \mathrm{tr}(\xi_4^T\zeta_4), \tag{8.24}$$

where $\xi = (\xi_1,\xi_2,\xi_3,\xi_4)$, $\zeta = (\zeta_1,\zeta_2,\zeta_3,\zeta_4)$ with $\xi_i,\zeta_i \in T_{R_i}SO(3)$ for $i \in \{1,2,3\}$ and $\xi_4,\zeta_4 \in T_T\mathbb{S}_2^3$. Now, the orthogonal projection of a tangent vector $\xi \in T_X\overline{\mathcal{M}}_{\mathcal{T}}$ onto the vertical space $\mathcal{V}_X$ is given by

$$P_X^v\xi = \widehat{e}_z X\frac{\overline{g}(\xi,\widehat{e}_z X)}{\overline{g}(\widehat{e}_z X,\widehat{e}_z X)} = \frac{1}{4}\overline{g}(\xi,\widehat{e}_z X)\widehat{e}_z X, \tag{8.25}$$

and the corresponding orthogonal projection of a tangent vector $\xi \in T_X\overline{\mathcal{M}}_{\mathcal{T}}$ onto the horizontal $\mathcal{H}_X$ is simply given by

$$P_X^h\xi = \xi - P_X^v\xi. \tag{8.26}$$

Next, we will endow $\mathcal{M}_{\mathcal{T}}$ with a Riemannian metric. We will need the following proposition relating the horizontal lifts of the same tangent vector of the quotient space at two distinct points in the same equivalence class.

Proposition 8.4.4. *Let $X \in \overline{\mathcal{M}}_{\mathcal{T}}$ and $\xi \in T_{[X]}\mathcal{M}_{\mathcal{T}}$. Then*

$$\overline{\xi}_{RX} = R\overline{\xi}_X, \tag{8.27}$$

for all $R \in H_z$, where $\overline{\xi}_X$ denotes the horizontal lift of a tangent vector ξ at X.

We then arrive at the desired result.

Proposition 8.4.5. *The signed trifocal manifold $\mathcal{M}_{\mathcal{T}}$ admits a structure of a Riemannian quotient manifold with the Riemannian metric*

$$g_{[X]}(\xi,\zeta) \doteq \overline{g}_X(\overline{\xi}_X,\overline{\zeta}_X). \tag{8.28}$$

Proof. Let $\overline{g}_X$ be the Riemennian metric of $\overline{\mathcal{M}}_{\mathcal{T}}$ at $X \in \overline{\mathcal{M}}_{\mathcal{T}}$ defined in (8.24). Then, we

have

$$\overline{g}_{RX}(\overline{\xi}_{RX}, \overline{\zeta}_{RX}) = \overline{g}_{RX}(R\overline{\xi}_X, R\overline{\zeta}_X) \tag{8.29}$$

$$= \frac{1}{2}\sum_{i=1}^{3} \mathrm{tr}((R\xi_{X,i})^T(R\zeta_{X,i})) + \mathrm{tr}((R\xi_{X,4})^T(R\zeta_{X,4})) \tag{8.30}$$

$$= \frac{1}{2}\sum_{i=1}^{3} \mathrm{tr}(\xi_{X,i}^T\zeta_{X,i}) + \mathrm{tr}(\xi_{X,4}^T\zeta_{X,4}) \tag{8.31}$$

$$= \overline{g}_X(\overline{\xi}_X, \overline{\zeta}_X). \tag{8.32}$$

The metric (8.28) does not depend on the choice of the representative of each equivalence class and thus, it is a well-defined Riemannian metric. □

8.4.3 Geodesics and the exponential map

In this section, we show how to obtain geodesics for $\mathcal{M}_\Upsilon$ from geodesics in the ambient space $\overline{\mathcal{M}}_\mathcal{T}$ with horizontal tangent. The idea has been repeatedly used in [34] to obtain geodesics for the Stiefel and Grassmann manifold from geodesics of the orthogonal group and in [137] to obtain geodesics of the Essential manifold from geodesics of $SO(3)^2$. Since the projection $\pi_{\mathcal{M}_\Upsilon} : \overline{\mathcal{M}}_\mathcal{T} \to \mathcal{M}_\Upsilon$ is a Riemannian submersion, i.e. a submersion that preserves the metric, we have the following proposition [99]:

Proposition 8.4.6. *Let $\gamma(t)$ be a geodesic on $\overline{\mathcal{M}}_\mathcal{T}$ such that $\dot{\gamma}(t) \in \mathcal{H}_{\gamma(t)}$ for all t. Then, $\pi_{\mathcal{M}_\Upsilon}(\gamma(t)) = [\gamma(t)]$ is a geodesic on $\mathcal{M}_\Upsilon$.*

Moreover, we have the following proposition for geodesics with horizontal initial tangents.

Proposition 8.4.7. *Let $\gamma_{X,\xi}(t)$ be a geodesic on $\overline{\mathcal{M}}_\mathcal{T}$ emanating from $X = \gamma_{X,\xi}(0)$ with initial velocity $\xi = \dot{\gamma}_{X,\xi}(0)$. If $\xi = \dot{\gamma}_{X,\xi}(0) \in \mathcal{H}_X$, then $\dot{\gamma}_{X,\xi}(t) \in \mathcal{H}_{\gamma_{X,\xi}(t)}$ for all t.*

The above result combined with Proposition 8.4.6 shows that if $\gamma(t)$ is a geodesic on $\overline{\mathcal{M}}_\mathcal{T}$ with $\dot{\gamma}(0) \in \mathcal{H}_{\gamma(0)}$, i.e. initial tangent belonging to the horizontal space, then $[\gamma(t)]$ is a geodesic in the quotient space $\mathcal{M}_\Upsilon$. Thus, the exponential map $\exp : T_{[X_a]}\mathcal{M}_\Upsilon \to \mathcal{M}_\Upsilon$ is

defined as $[X_b] = \exp_{[X_a]}(\xi)$ and can be computed by $X_b = \exp_{X_a}(\overline{\xi}_{X_a})$, where $\overline{\xi}_{X_a}$ is the horizontal lift of ξ at X_a and $\exp_{X_a}$ is the exponential map of $\overline{\mathcal{M}}_{\mathcal{T}} = SO(3)^3 \times \mathbb{S}_2^3$.

8.4.4 The logarithm map and Riemannian distance

In this section, we will determine the logarithm map for the signed trifocal manifold from its ambient space, and describe an efficient algorithm for computing it. The Riemannian distance is then given by the norm of the logarithm map. Intuitively, given two points in $\overline{\mathcal{M}}_{\mathcal{T}}$, we will move the second point to another representative of its equivalence class for which the squared Riemannian distance of $\overline{\mathcal{M}}_{\mathcal{T}}$ is minimized. This change of representative will yield a horizontal vector as we will show in Proposition 8.4.8.

Let $X_a, X_b \in \overline{\mathcal{M}}_{\mathcal{T}}$ and $R_z(t)$ denote a rotation around z axis of angle t. Moreover, let

$$\theta_i(t) = \arccos\left((\mathrm{tr}(R_{ai}^T R_z(t) R_{bi}) - 1)/2\right), \tag{8.33}$$

for $i \in \{1, 2, 3\}$ and

$$\theta_4(t) = \arccos(\mathrm{tr}(T_a^T R_z(t) T_b)), \tag{8.34}$$

which are the geodesic distances in $SO(3)$ and in $\mathbb{S}_2^3$. The main proposition for the logarithm map of the signed trifocal manifold follows.

Proposition 8.4.8. *Define the cost function*

$$f(t) = \sum_{i=1}^{4} f_i(t) = \sum_{i=1}^{4} \frac{1}{2}\theta_i^2(t) = \frac{1}{2}\sum_{i=1}^{3} d^2(R_{ai}, R_z(t)R_{bi}) + \frac{1}{2}d^2(T_a, R_z(t)T_b). \tag{8.35}$$

Moreover, let $t_{opt} = \mathrm{argmin}_t\, f(t)$. *Then, the logarithm* $\log_{X_a}(R_z(t_{opt})X_b)$ *is a horizontal vector in* $\mathcal{H}_{X_a}\overline{\mathcal{M}}_{\mathcal{T}}$.

Proof. By differentiating the functions f_i we obtain

$$\dot{f}_i(t) = -\frac{\theta_i(t)}{2\sin\theta_i(t)}\operatorname{tr}(R_{ai}^T\widehat{e_z}R_z(t)R_{bi}) = \frac{1}{2}\operatorname{tr}\left((\widehat{e_z}R_{ai})^T\log_{R_{ai}}(R_z(t)R_{bi})\right), \tag{8.36}$$

for $i \in \{1,2,3\}$ and for $i = 4$, we have,

$$\dot{f}_4(t) = -\frac{\theta_4(t)}{\sin\theta_4(t)}\operatorname{tr}(T_a^T\hat{e}_zR_z(t)T_b) = \operatorname{tr}\left((\widehat{e_z}T_a)^T\log_{T_a}(R_z(t)T_b)\right). \tag{8.37}$$

Comparing the condition $\dot{f}(t_{\text{opt}}) = \sum_{i=1}^4 \dot{f}_i(t_{\text{opt}}) = 0$ with the basis of $\mathcal{V}_{X_a}$ in Proposition 8.4.3 and the definition of $\mathcal{H}_{X_a}$, we deduce that the logarithm $\log_{X_a}\left(R_z(t_{\text{opt}})X_b\right)$ must be an horizontal vector. □

At this point, we will describe an algorithm for computing the logarithm map. Although global optimization is generally hard, we can exploit the special structure of f to efficiently compute its global minimizer t_{opt}. First, the cost function f is continuous and 2π-periodic, since $R_z(t+2\pi) = R_z(t)$, but it is not everywhere smooth. For the first three terms in $f(t)$, the derivative $\dot{f}_i(t)$, $i \in \{1,2,3\}$, is not defined when $\cos\theta_i(t) = -1$. This correspond to discontinuity points t_{di}, which can be computed in closed form (see [137, Proposition 5.6]). It can also be shown [137] that $f_i(t)$, $i \in \{1,2,3\}$ is convex between discontinuity points.

It remains to analyze the behavior of $f_4(t)$. We have two distinct cases:

Case 1: $T_a = R_z(t_0)T_b$ for some $t_0 \in \mathbb{R}$. In this special case $f_4(t)$ is simply given by $f_4(t) = \frac{1}{2}(\arccos(\cos(t-t_0)))^2$. The derivative $\dot{f}_4(t)$ is not defined for $t = t_0 + (2k+1)\pi$, $k \in \mathbb{Z}$ and $\ddot{f}_4(t) = 1$ when defined. So, in this case f_4 is piecewise convex and thus, f is also piecewise convex. The four points of discontinuity of $\dot{f}$ can be computed in closed form and thus, projected Newton method [11] can be applied to each of four resulting intervals in a way similar to [137]. Then, the global optimum of $f(t)$ can be computed as the minimum of the four local minima.

Case 2: $T_a \neq R_z(t_0)T_b$ for all $t_0 \in \mathbb{R}$. This is the more general case. By differentiating $\dot{f}_4(t)$

we can obtain a closed form expression for $\ddot{f}_4$. Unfortunately, $\ddot{f}_4$ can take negative values and thus, $f_4(t)$ is not convex. However, each period can be divided into two intervals, one at which $f_4(t)$ is convex (thus easy to optimize) and one at which $f_4(t)$ is concave (for which we use a branch-and-bound search). First, we need to identify these intervals. Let

$$c_{14} = (T_b T_a^T)_{1,1} + (T_b T_a^T)_{2,2}, \tag{8.38}$$

$$c_{24} = (T_b T_a^T)_{1,2} - (T_b T_a^T)_{2,1}. \tag{8.39}$$

Then, $\dot{f}_4(t) = 0$ for $t = \arctan(c_{24}/c_{14})$. It can be immediately seen that $\dot{f}_4(t) = 0$ has two solutions: one corresponding to the minimum over a period and one corresponding to the maximum. Let t_{max} and t_{min} denote these two values. Since $\ddot{f}_4(t)$ is continuous, it follows that $\ddot{f}_4(t_{min}) > 0$ and $\ddot{f}_4(t_{max}) < 0$. As a consequence, f_4 is convex in an interval (t_{c1}, t_{c2}) around t_{min}, and concave in an interval $(t_{c2}, t_{c1} + 2\pi)$ around t_{max}. The values t_{c1} and t_{c2} can be computed from t_{max} and t_{min} using the bisection method for $\ddot{f}_4(t) = 0$. For the interval (t_{c1}, t_{c2}) we have that $f(t)$ is continuous, convex with up to three discontinuity points of the first derivative. Thus, a projected Newton method as in [137] can be again applied to each of the subintervals. For the interval $(t_{c2}, t_{c1} + 2\pi)$, $f(t)$ is generally neither convex nor concave, and we implement a branch-and-bound search. Since we already have a good initial guess from the interval where $f(t)$ is convex, most of the subintervals are quickly rejected. Moreover, a lower bound for f in an interval $[a, b]$, on which f_4 is concave, can be efficiently estimated by minimizing the following convex underestimate of $f(t)$ using the Newton method

$$f_l(t) = \sum_{i=1}^{3} f_i(t) + \frac{f_4(b) - f_4(a)}{b - a}(t - a) + f_4(a). \tag{8.40}$$

This underestimate is simply the sum of the three piecewise convex function f_1, f_2, f_3 with a linear underestimate of f_4.

In conclusion, using the above described method, the computation of the logarithm $\log_{[X_a]}[X_b]$ between two equivalence classes $[X_a], [X_b] \in \mathcal{M}_\Upsilon$ can be efficiently carried out. The distance

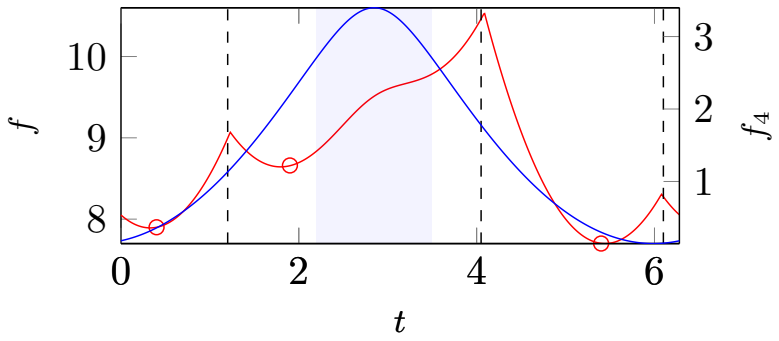


Figure 8.2: An instance of the cost $f(t)$ (in red) and $f_4(t)$ (in blue) for one period. Black dashed lines correspond to the three discontinuity points t_{di}. Shaded region corresponds to the interval on which f_4 is concave and the red circles to the local minimizers of $f(t)$.

can then be computed as the norm of the logarithm, that is,

$$d([X_a],[X_b]) = \|\log_{[X_a]}[X_b]\|. \tag{8.41}$$

8.5 Optimization on the trifocal manifold

In this section, we describe how to minimize a cost function that takes as input a trifocal tensor. For example, assume we have n point-line-line correspondences $x_{1p} \leftrightarrow l_{2p} \leftrightarrow l_{3p}$, $p = 1, 2, \ldots, n$. Such a cost function is the sum of squared algebraic errors which is given by

$$f_a(\mathcal{T}) = \sum_{p=1}^{n}\sum_{i=1}^{3}\left((x_{1p})_i l_{2p}^T \mathcal{T}_i l_{3p}\right)^2. \tag{8.42}$$

Another example is the Sampson error [46, 83], i.e. the first order approximation to the geometric error:

$$f_s(\mathcal{T}) = \sum_{p=1}^{n}\sum_{i=1}^{3}\frac{\left((x_{1p})_i l_{2p}^T \mathcal{T}_i\ l_{3p}\right)^2}{J_p(\mathcal{T})J_p(\mathcal{T})^T,} \tag{8.43}$$

where $J_p(\mathcal{T})$ is the Jacobian of the expression $(x_{1p})_i l_{2p}^T \mathcal{T}_i l_{3p}$ with respect to x_{1p}, l_{2p} and l_{3p} (note that this is a row vector).

More generally, given a real-valued function $f : \mathbb{R}^{3\times3\times3} \to \mathbb{R}$, let $f_{\overline{\mathcal{M}}_\mathcal{T}} : \overline{\mathcal{M}}_\mathcal{T} \to \mathbb{R}$ defined by

$$f_{\overline{\mathcal{M}}_\mathcal{T}} = f \circ \mathcal{T}, \tag{8.44}$$

where $\mathcal{T}$ is the map given in Definition 8.3.2. Moreover, let $X_a, X_b \in \overline{\mathcal{M}}_\mathcal{T}$ such that $X_a \sim X_b$. Since $f_{\overline{\mathcal{M}}_\mathcal{T}}$ is constant in each equivalence class, it induces a unique function $f_{\mathcal{M}_\Upsilon}$ on $\mathcal{M}_\Upsilon$ such that

$$f_{\overline{\mathcal{M}}_\mathcal{T}} = f_{\mathcal{M}_\Upsilon} \circ \pi. \tag{8.45}$$

In order to combine the parametrization of $\mathcal{M}_\Upsilon$ given by the exponential map with the trust-region methods described in [1] we need to compute $\operatorname{grad} f_{\mathcal{M}_\Upsilon}([X])$ and $\operatorname{Hess} f_{\mathcal{M}_\Upsilon}([X])[\xi]$. Tangent vectors to quotient manifolds are represented in a computer program by their horizontal lifts. In detail, $\operatorname{grad} f_{\mathcal{M}_\Upsilon}([X])$ is represented by its horizontal lift $\operatorname{grad} f_{\overline{\mathcal{M}}_\mathcal{T}}(X)$ at X. Note that $\operatorname{grad} f_{\overline{\mathcal{M}}_\mathcal{T}}(X)$ is guaranteed to be an horizontal vector. Thus, we just have to compute the (Riemannian) gradient of $f_{\overline{\mathcal{M}}_\mathcal{T}}$. In the rest of this section, we show how to obtain the expression for the Riemannian gradient from its Euclidean counterpart and an expression of the Riemannian Hessian from the Euclidean gradient and the Euclidean Hessian.

Let $X(t)$ be a geodesic curve of the form $X(t) = (R_1(t), R_2(t), R_3(t), T(t))$. Let $\mathcal{T}(t) \doteq \mathcal{T}(X(t))$. Now, consider the function

$$f_{\overline{\mathcal{M}}_\mathcal{T}}(X(t)) = f(\mathcal{T}(t)).$$

At $t = 0$ we have

$$\overline{g}\big(\dot{X}, \operatorname{grad} f_{\overline{\mathcal{M}}_\mathcal{T}}(X)\big) = \langle \dot{\mathcal{T}}, \operatorname{grad} f(\mathcal{T})\rangle, \tag{8.46}$$

$$\overline{g}\big(\dot{X}, \operatorname{Hess} f_{\overline{\mathcal{M}}_\mathcal{T}}(X)[\dot{X}]\big) = \langle \ddot{\mathcal{T}}, \operatorname{grad} f(\mathcal{T})\rangle + \langle \dot{\mathcal{T}}, \operatorname{Hess} f(\mathcal{T})[\dot{\mathcal{T}}]\rangle, \tag{8.47}$$

where $\langle \cdot, \cdot \rangle$ denotes the usual Euclidean inner product.

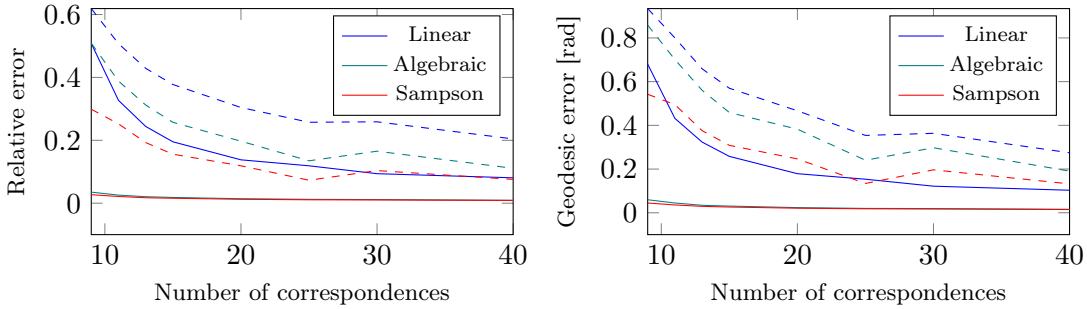


Figure 8.3: Relative (top) and geodesic (in $SO(3)^3 \times \mathbb{S}^5$, bottom) mean (dashed) and median (solid) errors before and after non-linear minimization of either the algebraic or Sampson cost.

To alleviate the notation, let $G = \operatorname{grad} f(\mathcal{T}) \in \mathbb{R}^{3\times3\times3}$ and let G_i be i-th slice of G for $i = 1, 2, 3$. We have that

$$\operatorname{grad} f_{\overline{\mathcal{M}}_\mathcal{T}}(X) = (\xi_1, \xi_2, \xi_3, \xi_4),$$

where

$$\xi_1 = 2\sum_{i=1}^{3} P_{R_1}\left(\left(R_3 G_i^T R_2^T T_{12} - R_2 G_i R_3^T T_{13}\right) e_i^T\right) \tag{8.48}$$

$$\xi_2 = 2\sum_{i=1}^{3} P_{R_2}\left(R_2 \mathcal{T}_i G_i^T\right) \tag{8.49}$$

$$\xi_3 = 2\sum_{i=1}^{3} P_{R_3}\left(R_3 \mathcal{T}_i^T G_i\right) \tag{8.50}$$

$$\xi_4 = \sum_{i=1}^{3} P_T\left(R_2 G_i R_3^T R_1 e_i, -R_3 G_i^T R_2^T R_1 e_i\right) \tag{8.51}$$

where $P_R\xi = R\operatorname{skew}(R^T\xi)$, skew denotes the skew-symmetric part of a matrix, and $P_X\xi = \xi - X\operatorname{tr}(X^T\xi)$ denotes the orthogonal projection of a vector $\zeta \in \mathbb{R}^{3\times2}$ onto the tangent space $T_T\mathbb{S}_2^3$. For a detailed derivation, we refer the reader to Appendix G.4. The Riemannian Hessian can be computed using (8.47), but the explicit expression is rather involved and therefore, omitted.

We evaluate our implementation on the fountain-P11 dataset from [123] which includes the ground-truth camera poses. We extract SIFT features [141] to obtain point correspondences across different image triplets. We keep only image triplets with more than 50 point corre-

spondences. To obtain an initial estimate of the trifocal tensor, we use the linear algorithm described in [145]. For optimization on the signed trifocal manifold, we use the trust-region solver of [14]. As error metric, we use the relative error between the estimated tensor and the ground-truth in the Frobenius norm sense. We also use the geodesic distance of $SO(3)^3 \times \mathbb{S}^5$ since the trifocal tensor can be parametrized by three camera orientations and two relative translations of unit total length. To compare with the ground-truth, we align the estimated rotations and relative translations using orthogonal Procrustes analysis. We vary the number of point correspondences from 9 to 40. For each image triplet and each number of correspondences, we repeat our experiment 50 times by randomly selecting a different set of correspondences each time.

The results are presented in Figure 8.3. We compare the solution of the linear algorithm [145] with the solutions obtained by minimizing the algebraic (8.42) and the Sampson (8.43) errors on the trifocal manifold. The proposed optimization significantly outperforms the linear algorithm and produces very accurate estimates even with only few correspondences.

8.6 Pose averaging and the Weiszfeld algorithm

The Weiszfeld algorithm has been traditionally used for computing the l_1-mean (geometric median) of a set of points in $\mathbb{R}^n$. Recently, Hartley *et al.*[47] proposed the use of the Weiszfeld algorithm for the purpose of rotation averaging under the l_1-norm. Instead of using RANSAC for outlier rejection, they obtain multiple estimates of the relative rotations from the corresponding essential matrices, and average them using the Weiszfeld algorithm on $SO(3)$. This idea has been extended in [137] for averaging essential matrices.

A generic form of the Weiszfeld algorithm for an arbitrary Riemannian manifold $\mathcal{M}$ is presented in Algorithm 4. The new iterate $x(t+1)$ is obtained by taking the exponential map of a weighted average of directions on the tangent space of the current iterate. The weights are inversely proportional to the geodesic distance between the current iterate and

each sample. Intuitively, points away from the current estimate have little impact on the update and thus, the algorithm is robust to outliers.

In this work, we use the Weiszfeld algorithm to average estimates of trifocal tensors seen as points on the signed trifocal manifold $\mathcal{M}_{\mathcal{T}}$. As in [137], the initial estimate is chosen as the midpoint of the two points having the lower cost. The sample trifocal tensors were obtained by the linear algorithm described in [145]. Unfortunately, this method does not perform well for a small number of point correspondences, resulting in noisy samples. In our experiments we observed that the algorithm converged in 10 to 15 iterations. We compare our approach with RANSAC and with the Weizsfeld algorithm on the manifold $SO(3)^2 \times \mathbb{S}^5$, i.e. a manifold parametrization of the trifocal tensor with $R_1 = I_3$. The purpose of this experiment is to show the advantage of the quotient versus non-quotient parametrization when using distances between trifocal tensors. We vary the number of samples from 10 to 50. The results are shown in Figure 8.4. Although it is hard to beat RANSAC, the Weizsfeld algorithm can be used to obtain a sufficiently good initial estimate of the trifocal tensor without the need of tuning a threshold like RANSAC. Also, the Weiszfeld algorithm performs much better on the quotient manifold, as anticipated.

8.7 Conclusions

In this chapter, we investigated a novel parametrization of the trifocal tensor for calibrated cameras with non-colinear pinholes obtained from a quotient Riemannian manifold. We incorporated techniques for optimization on manifolds and pose averaging in our approach

Algorithm 4 Weiszfeld averaging

1: **Input:** Points $x_1, x_2, \ldots, x_n \in \mathcal{M}$, initial estimate $x(0)$
2: **for** $t = 0, \ldots, N$ **do**
3: $\quad w_i(t) = d(x(t), x_i)^{-1}$
4: $\quad x(t+1) = \exp_x \left(\dfrac{\sum_i w_i(t) \log_x(x_i)}{\sum_i w_i(t)} \right)$
5: **end for**

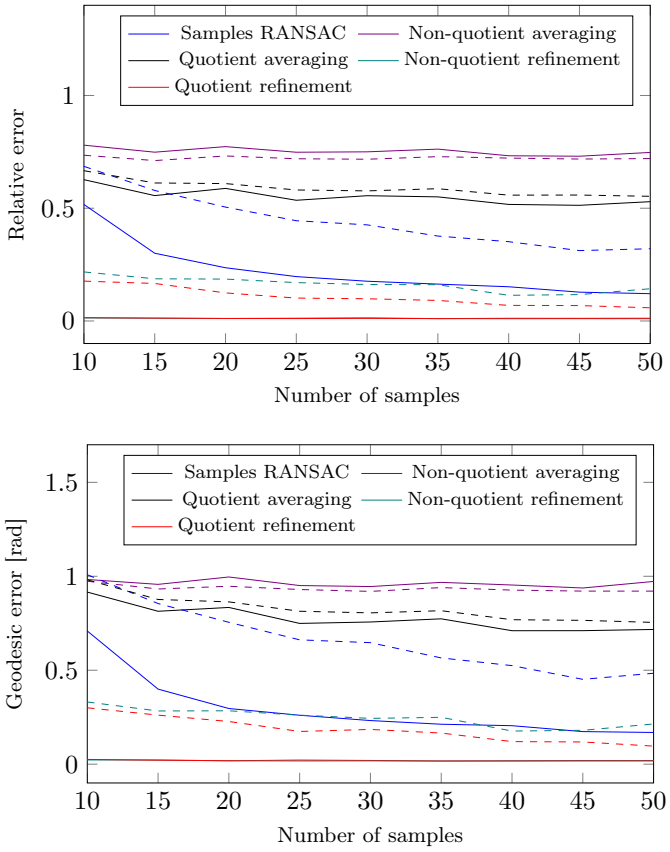


Figure 8.4: Relative (top) and geodesic (bottom) mean (dashed) and median (solid) errors for the Weiszfeld (quotient and non-quotient parametrizations) and RANSAC algoritms, without and with non-linear minimization.

and we showed that the resulting distance is meaningful.

Chapter 9

Conclusions and future vistas

This work aimed at developing collaborative algorithms for groups of visual sensors. More specifically, we aimed at solving two main problems: collaborative data association and cooperative localization. We focused on optimization algorithms that are decentralized.

First, we proposed a distributed optimization method for solving the permutation synchronization problem and a distributed optimization approach for the general case of multiway matching. We rigorously analyzed the convergence properties of the proposed algorithms and provided experimental evidence supporting that the proposed approaches, albeit decentralized, have performance comparable with the state of the art centralized approaches. A potential future direction consists of enforcing cycle consistency in neural networks, a direction which has recently gained attention [154, 55, 105]. Another potential future direction consists of applying the proposed methods for data association in a multi-robot semantic Simultaneous Localization and Mapping (SLAM) setting.

Secondly, we proposed a distributed dynamical systems approach for the problem of distributedly estimating any number of smallest eigenvalues and the associated eigenvectors of a weighted graph with global convergence guarantees. We demonstrated the validity of our approach through rigorous theoretical analysis and experimental evaluation and we applied

the proposed method to the problem of permutation synchronization. A potential future direction consists of applying the proposed method to solve spectral clustering problems [92] in a decentralized multi-agent setting.

Thirdly, we addressed the problem of decentralized cooperative localization for a group of mobile agents. We proposed a minimax formulation to deal with missing cross-correlations. The proposed estimator takes cross-correlations into account while being less conservative than the widely used Covariance Intersection. We demonstrated the superiority of the proposed method compared to Covariance Intersection with numerical examples and simulations within the specific application of decentralized localization. A potential future direction, recently suggested in [153], consists of dealing with measurements involving more than two agents.

Fourthly, we considered the problem of estimating the orientations of a set of agents with respect to a global reference frame, using only local bearing measurements. We identified sufficient conditions for localizability, and proposed a distributed optimization approach to estimate the unknown orientations, without any prior information. One of the goals of this work was to propose an algorithm that is based on a minimum number of constraints. We performed some preliminary tests using the bearings alone but the problem becomes ill-posed (the Hessian of the objective loses rank); hence the inclusion of the normals is necessary. Normals and binormals are generated by algebraically combining the real measurements that we have available; hence, using all equal weights might not be the statistically most efficient solution. A potential future direction consists of a deeper investigation of how the constraints involving triple plane normals and binormals should be included in a statistically optimal manner.

Finally, we proposed a novel parametrization of the trifocal tensor obtained from a quotient Riemannian manifold. The contributions of the last chapter were twofold. First, we incorporated the state of the art techniques for optimization on manifolds in our parametrization and second, we obtained a meaningful Riemannian distance between two trifocal tensors.

One potential future direction could be the application of clustering algorithms on Riemannian manifolds, such as mean shift [125], to the proposed manifold parametrization for the purpose of robust estimation of the trifocal tensor from noisy samples. Another potential application of the proposed parametrization could be in the problem of distributed image-based localization of a camera sensor network [139, 138] using the obtained Riemannian distance as an objective. In contrast to the essential matrix, the trifocal tensor provides a relative scale between the two relative translations between three visual sensors. This additional piece of information may improve the localization accuracy compared to utilizing only pairwise information via the essential matrix.

Overall, we believe that the work presented herein could represent the basis for future research in a number of topics in the disciplines of multi-agent systems, robotics and computer vision.

Appendix A

Closest partial doubly stochastic matrix

In this section, we consider the problem of finding the closest partial doubly stochastic matrix to a given matrix $X_0 \in \mathbb{R}^{m\times n}$ with $m \leq n$, i.e.

$$\begin{array}{ll} \underset{X\in\mathbb{R}^{m\times n}}{\text{minimize}} & (1/2)\|X - X_0\|_F^2 \\ \text{subject to} & X\mathbf{1} = \mathbf{1},\ X^T\mathbf{1} \leq \mathbf{1},\ X \geq 0. \end{array} \tag{A.1}$$

Problem (A.1) is equivalent to the following problem:

$$\begin{array}{ll} \underset{X,s}{\text{minimize}} & (1/2)\|X - X_0\|_F^2 \\ \text{subject to} & X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} + s = \mathbf{1}, \\ & X \geq 0, s \geq 0. \end{array} \tag{A.2}$$

Let

$$f(X, s) = (1/2)\|X - X_0\|_F^2, \tag{A.3}$$

with domain

$$\mathrm{dom} f = \{(X, s) \in \mathbb{R}^{m\times n} \times \mathbb{R}^n : X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} + s = \mathbf{1}\}. \tag{A.4}$$

Then, problem (A.2) is equivalent to the following problem in ADMM standard form:

$$\begin{array}{ll} \underset{(X,s)\in\mathrm{dom} f,(Z,t)\in\mathbb{R}^{m\times n}\times\mathbb{R}^n}{\text{minimize}} & f(X, s) + I_{\geq 0}(Z, t) \\ \text{subject to} & X - Z = 0,\ s - t = 0, \end{array} \tag{A.5}$$

where $I_{\geq 0}(Z, t) = 0$ if $Z \geq 0$ and $t \geq 0$ and $+\infty$ otherwise. Then, the ADMM iterations for problem (A.5) are as follows:

$$(X^{k+1}, s^{k+1}) = \underset{(X,s)\in\mathrm{dom} f}{\mathrm{argmin}} \left\{ \frac{1}{2}\|X - X_0\|_F^2 + \frac{\rho}{2}\|X - Z^k + U^k\|_F^2 + \frac{\rho}{2}\|s - t^k + w^k\|_F^2 \right\}, \tag{A.6}$$

$$Z^{k+1} := (X^{k+1} + U^k)_+, \tag{A.7}$$

$$t^{k+1} := (s^{k+1} + w^k)_+, \tag{A.8}$$

$$U^{k+1} := U^k + X^{k+1} - Z^{k+1}, \tag{A.9}$$

$$w^{k+1} := w^k + s^{k+1} - t^{k+1}. \tag{A.10}$$

Finally, we will solve the first subproblem of the ADMM iterations which is an equality constrained convex quadratic optimization problem. The Lagrangian for the first subproblem is given by

$$L(X, s, \mu, \nu) = \frac{1}{2}\|X - X_0\|_F^2 + \frac{\rho}{2}\|X - Z^k + U^k\|_F^2 + \frac{\rho}{2}\|s - t^k + w^k\|_F^2 \tag{A.11}$$

$$+ \mu^T(\mathbf{1} - X\mathbf{1}) + \nu^T(\mathbf{1} - X^T\mathbf{1} - s). \tag{A.12}$$

The KKT conditions are given by

$$X = (1+\rho)^{-1}(X_0 + \rho(Z^k - U^k) + \mu \mathbf{1}^T + \mathbf{1}\nu^T), \tag{A.13}$$

$$s = \frac{1}{\rho}\nu + t^k - w^k, \tag{A.14}$$

$$X\mathbf{1} = \mathbf{1}, \tag{A.15}$$

$$X^T\mathbf{1} + s = \mathbf{1}. \tag{A.16}$$

First, we eliminate X and s and obtain the following system of linear equation for the multipliers μ, ν:

$$\begin{aligned} n\mu + \mathbf{1}\mathbf{1}^T\nu &= \Big((1+\rho)I - (X_0 + \rho(Z^k - U^k)\Big)\,\mathbf{1} \\ \mathbf{1}\mathbf{1}^T\mu + (m + 1 + \rho^{-1})\nu &= (1+\rho)(\mathbf{1} - t^k + w^k) - (X_0 + \rho(Z^k - U^k))^T\mathbf{1}. \end{aligned} \tag{A.17}$$

The matrix

$$\begin{bmatrix} nI & \mathbf{1}\mathbf{1}^T \\ \mathbf{1}\mathbf{1}^T & (m+1+\rho^{-1})I \end{bmatrix}$$

is always invertible and its inverse can be precomputed. The overall algorithm is summarized below. We chose $\rho = 1$, $\epsilon^{\text{abs}} = 10^{-4}$. Its complexity is $O(mn)$ flops per iteration.

Algorithm 5 Closest partial doubly stochastic matrix.

Input: $X_0 \in \mathbb{R}^{m\times n}$ with $m \leq n$.

Ouput: X: closest partial doubly stochastic matrix to X_0.

repeat

Compute μ^{k+1} and ν^{k+1} by solving (A.17)

$X^{k+1} := (1+\rho)^{-1}\left(X_0 + \rho(Z^k - U^k) + \mu^{k+1}\mathbf{1}^T + \mathbf{1}(\nu^{k+1})^T\right)$

$s^{k+1} := \rho^{-1}\nu^{k+1} + t^k - w^k$

$Z^{k+1} := (X^{k+1} + U^k)_+$

$t^{k+1} := (s^{k+1} + w^k)_+$

$U^{k+1} := U^k + X^{k+1} - Z^{k+1}$

$w^{k+1} := w^k + s^{k+1} - t^{k+1}$

$R^{k+1} := [(X^{k+1} - Z^{k+1})^T, s^{k+1} - t^{k+1}]$

$S^{k+1} := -\rho[(Z^{k+1} - Z^k)^T, t^{k+1} - t^k]$

$k := k + 1$

until $\|R^{k+1}\|_F \leq \sqrt{mn}\epsilon^{\text{abs}}$ and $\|S^{k+1}\|_F \leq \sqrt{mn}\epsilon^{\text{abs}}$

Appendix B

Proofs for Chapter 2

B.1 Proof of Proposition 2.4.2

The following proof is a generalization of the proof of [136] for the case of a function defined not only on the edges of a graph but on the vertices as well. Let $x = [x_1^T \ \cdots \ x_m^T]^T$ and $v = [v_1^T \ \cdots \ v_m^T]^T$. Let $\widetilde{\phi}(t) \doteq \phi(x + tv)$. Then,

$$\frac{d^2}{dt^2}\widetilde{\phi}(t)\bigg|_{t=0} = \alpha \sum_{i\in\mathcal{V}} \frac{d^2}{dt^2}\widetilde{\phi}_i(t)\bigg|_{t=0} + \beta \sum_{\{i,j\}\in\mathcal{E}} \frac{d^2}{dt^2}\widetilde{\phi}_{ij}(t)\bigg|_{t=0} \tag{B.1}$$

$$\leq \alpha\mu_{\max} \sum_{i\in\mathcal{V}} \|v_i\|_2^2 + \beta\nu_{\max} \sum_{\{i,j\}\in\mathcal{E}} (\|v_i\|_2^2 + \|v_j\|_2^2) \tag{B.2}$$

$$= \alpha\mu_{\max}\|v\|_2^2 + \beta\nu_{\max} \sum_{i\in\mathcal{V}} |\mathcal{N}_i| \, \|v_i\|_2^2 \tag{B.3}$$

$$\leq (\alpha\mu_{\max} + \beta\nu_{\max} d_{\max}(\mathcal{G})) \, \|v\|_2^2, \tag{B.4}$$

and the proof is complete.

Appendix C

Proofs for Chapter 4

C.1 Proof of Lemma 4.3.2

First, we need the following lemma

Lemma C.1.1. *If $X \in \mathcal{D}_{m,n}$ then $X^T X \preceq I$.*

Proof. First, observe that $X^T X \mathbf{1} = X^T \mathbf{1} \leq \mathbf{1}$. By Gershgorin discs theorem [51], we have that every eigenvalue λ of $X^T X$ must lie within at least one the Gershgorin discs $D((X^T X)_{ii}, R_i)$ where $R_i = \sum_{j \neq i} (X^T X)_{ij} \leq 1 - (X^T X)_{ii}$. Thus, no eigenvalue of $X^T X$ can be larger than 1. □

The Hessian of ϕ_i is computed as follows

$$\operatorname{Hess} \phi_i(X_i)[U_i] = U_i(X_i^T X_i - I) + X_i(X_i^T U_i + U_i^T X_i). \tag{C.1}$$

Let x_i^T denote the ith row of X_i and u_i^T the ith row of U_i. We have that

$$\langle U_i, \operatorname{Hess}\phi_i(X_i)[U_i]\rangle = -\langle U_i, U_i\rangle + \langle U_i, U_i X_i^T X_i\rangle + \langle U_i, X_i X_i^T U_i\rangle + \langle U_i, X_i U_i^T X_i\rangle \quad \text{(C.2)}$$

$$\leq 2\langle U_i, U_i\rangle, \quad \text{(C.3)}$$

since

$$\langle U_i, X_i^T X_i U_i\rangle \leq \langle U_i, U_i\rangle, \quad \text{(C.4)}$$

$$\langle U_i, U_i X_i^T X_i\rangle = \sum_{i=1}^{m} u_i^T X_i^T X_i u_i \leq \sum_{i=1}^{m} u_i^T u_i = \langle U_i, U_i\rangle, \quad \text{(C.5)}$$

$$\langle U_i, X_i U_i^T X_i\rangle = \sum_{i,j=1}^{m} u_i^T x_j x_i^T u_j \leq \sum_{i,j=1}^{m} (u_i^T x_j)^2 = \sum_{i=1}^{m} u_i^T X_i^T X_i u_i \leq \langle U_i, U_i\rangle. \quad \text{(C.6)}$$

C.2 Proof of Lemma 4.3.3

The gradient of ϕ_{ij} with respect to X_i and the gradient of ϕ_{ij} with respect to X_j are given by

$$\operatorname{grad}_{X_i}\phi_{ij}(X_i, X_j) = (X_i X_j^T - \widetilde{X}_{ij})X_j, \quad \text{(C.7)}$$

$$\operatorname{grad}_{X_j}\phi_{ij}(X_i, X_j) = (X_i X_j^T - \widetilde{X}_{ij})^T X_i. \quad \text{(C.8)}$$

The corresponding differentials can be computed as follows

$$D\operatorname{grad}_{X_i}\phi_{ij}(X_i, X_j)[(U_i, U_j)] = (X_i X_j^T - \widetilde{X}_{ij})U_j + (U_i X_j^T + X_i U_j^T)X_j, \quad \text{(C.9)}$$

$$D\operatorname{grad}_{X_j}\phi_{ij}(X_i, X_j)[(U_i, U_j)] = (X_i X_j^T - \widetilde{X}_{ij})^T U_i + (U_i X_j^T + X_i U_j^T)^T X_i. \quad \text{(C.10)}$$

The corresponding Hessian is given by

$$\begin{aligned}\langle (U_i, U_j), \operatorname{Hess} \phi_{ij}(X_i, X_j))[(U_i, U_j)]\rangle = \langle U_i, U_i X_j^T X_j\rangle + \langle U_j, U_j X_i^T X_i\rangle + \langle U_i, X_i^T U_j^T X_j\rangle \\ + \langle U_j, X_j U_i^T X_i\rangle + 2\langle U_i, (X_i X_j^T - \widetilde{X}_{ij}) U_j\rangle\end{aligned} \tag{C.11}$$

As in C.1, we have $\langle U_i, U_i X_j^T X_j\rangle \leq \|U_i\|_F^2$ and $\langle U_j, U_j X_i^T X_i\rangle \leq \|U_j\|_F^2$. Moreover, by Gershgorin discs theorem [51] and by the assumptions on $\widetilde{X}_{ij}$, we have that the matrix

$$\begin{bmatrix} 0 & X_i X_j^T - \widetilde{X}_{ij} \\ (X_i X_j^T - \widetilde{X}_{ij})^T & 0 \end{bmatrix}$$

cannot have an eigenvalue strictly greater than 1 and thus,

$$\langle U_i, (X_i X_j^T - \widetilde{X}_{ij}) U_j\rangle + \langle U_j, (X_i X_j^T - \widetilde{X}_{ij})^T U_i\rangle \leq \|U_i\|_F^2 + \|U_j\|_F^2. \tag{C.12}$$

Similarly, it can be seen that the matrix

$$\begin{bmatrix} 0 & X_i^T \otimes X_j \\ (X_i^T \otimes X_j)^T & 0 \end{bmatrix}$$

cannot have an eigenvalue strictly greater than 1 as well, and thus,

$$\langle U_i, X_i^T U_j^T X_j\rangle + \langle U_j, X_j U_i^T X_i\rangle \leq \|U_i\|_F^2 + \|U_j\|_F^2. \tag{C.13}$$

The desired result trivially follows.

Appendix D

Proofs for Chapter 5

D.1 Proof of Lemma 5.3.3

First, we observe that any equilibrium (x, w, z) must satisfy $z = z_0\mathbf{1}$ for some $z_0 \in \mathbb{R}$ and $w = w_0\mathbf{1}$ for some $w_0 \in \mathbb{R}$ since $\mathcal{G}$ is assumed to be connected. By Lemma 5.3.1, it follows that $z_0 = 1 - x^T x$ and $x_0 = x^T Ax$. Based on this observation and by (5.13), we obtain the following equation for x:

$$(1 - x^T x)x - \alpha(2 - x^T x)Ax + \alpha(x^T Ax)x = 0. \tag{D.1}$$

Obviously, $x = \mathbf{0}$ satisfies (D.1). If $x \neq \mathbf{0}$, (D.1) implies

$$(1 - x^T x)x^T(I - 2\alpha A)x = 0. \tag{D.2}$$

For such that $I - 2\alpha A \succ 0$, it follows that

$$x^T x = 1, \qquad \text{and} \qquad (x^T Ax)x = Ax, \tag{D.3}$$

which shows that x is a unit-norm eigenvector of A.

D.2 Proof of Theorem 5.3.4

First, we need the following lemma.

Lemma D.2.1. *Let $\alpha, \beta > 0$ and define the matrix $P_{\alpha,\beta}$ by*

$$P_{\alpha,\beta} \doteq \begin{bmatrix} \mathcal{L}(\mathcal{G}) & \alpha\mathcal{L}(\mathcal{G}) \\ \alpha\mathcal{L}(\mathcal{G}) & \beta\mathcal{L}(\mathcal{G})^2 \end{bmatrix}, \tag{D.4}$$

where $\alpha, \beta > 0$ satisfy $\beta\lambda_2(\mathcal{L}(\mathcal{G})) > \alpha^2$. Then, $P_{\alpha,\beta} \succeq 0$, $\operatorname{rank}(P_{\alpha,\beta}) = 2n - 2$ and

$$\operatorname{null}(P_{\alpha,\beta}) = \operatorname{span}\left\{ \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix} \right\}. \tag{D.5}$$

Proof. By Schur complement, we have

$$\begin{bmatrix} \mathcal{L}(\mathcal{G}) & \alpha\mathcal{L}(\mathcal{G}) \\ \alpha\mathcal{L}(\mathcal{G}) & \beta\mathcal{L}(\mathcal{G})^2 \end{bmatrix} \succeq 0, \tag{D.6}$$

if and only if $\beta\mathcal{L}(\mathcal{G})^2 - \alpha^2\mathcal{L}(\mathcal{G})\mathcal{L}(\mathcal{G})^\dagger\mathcal{L}(\mathcal{G}) \succeq 0$. Thus, the bound $\beta\lambda_2(\mathcal{L}(\mathcal{G})) > \alpha^2$ follows. Since $\mathcal{L}(\mathcal{G})\mathbf{1} = \mathbf{0}$, it follows that $[\mathbf{1}^T\mathbf{0}^T]^T$ and $[\mathbf{0}^T\mathbf{1}^T]^T$ are in the nullspace of $P_{\alpha,\beta}$. Moreover, we will show that the span of these two vectors is exactly the nullspace of $P_{\alpha,\beta}$. Let $v_1 = (1/\sqrt{n})\mathbf{1}$ and $v_2, \ldots, v_n \perp \mathbf{1}$ be the eigenvectors of the Laplacian matrix $\mathcal{L}(\mathcal{G})$. Any vector $x \in \mathbb{R}^{2n}$ can be written as $x = \sum_{i=1}^n \begin{bmatrix} c_{i1}v_i \\ c_{i2}v_i \end{bmatrix}$ for some scalars $\{c_{i1}, c_{i2}\}_{i=1}^n$. We have that

$$x^T P_{\alpha,\beta} x = \sum_{i=1}^n \lambda_i \begin{bmatrix} c_{i1} \\ c_{i2} \end{bmatrix}^T \begin{bmatrix} 1 & \alpha \\ \alpha & \beta\lambda_i \end{bmatrix} \begin{bmatrix} c_{i1} \\ c_{i2} \end{bmatrix}. \tag{D.7}$$

If any of $\{c_{i1}, c_{i2}\}_{i=2}^n$ is not 0, then $x^T P_{\alpha,\beta} x > 0$ and thus, $x \notin \operatorname{null}(P_{\alpha,\beta})$. □

Lemma D.2.2. *The function ϕ_1 is nonincreasing along the trajectories of the system.*

Proof. We compute the time derivative of the function $\phi_1(x, z, w)$ as defined in (5.20) along the trajectories of our dynamical system as follows:

$$\dot{\phi}_1 = -\begin{bmatrix} z \\ w \end{bmatrix}^T \begin{bmatrix} \mathcal{L}(\mathcal{G}) & \alpha\mathcal{L}(\mathcal{G}) \\ \alpha\mathcal{L}(\mathcal{G}) & \beta\mathcal{L}(\mathcal{G})^2 \end{bmatrix} \begin{bmatrix} z \\ w \end{bmatrix} - 2n\|\dot{x}\|_2^2 \leq 0, \tag{D.8}$$

since $P_{\alpha,\beta} \succeq 0$ for α, β satisfying eqs. (5.15) and (5.19). □

Next, we find all (x, z, w) for which $\dot{\phi}_1(x, z, w) = 0$.

Lemma D.2.3. *Given α, β satisfying* (5.15) *and* (5.19)*, we have that $\dot{\phi}_1(x, z, w) = 0$ if and only if (x, z, w) is an equilibrium.*

Proof. We have that $\dot{\phi}_1(x, z, w) = 0$ implies

$$\begin{bmatrix} z \\ w \end{bmatrix}^T P_{\alpha,\beta} \begin{bmatrix} z \\ w \end{bmatrix} = 0, \tag{D.9}$$

which, based on Lemma D.2.1, implies that $z = (1 - x^T x)\mathbf{1}$ and $w = x^T Ax\mathbf{1}$. Finally, it is easy to see that the set of points (x, z, w) such that $\dot{\phi}_1(x, z, w)$ coincides with the set of equilibria of the dynamical system at hand. □

Next, we prove a lemma which intuitively says that the trajectories of the system cannot escape to infinity while the value of the potential is nonincreasing.

Lemma D.2.4. *We have that*

$$\|(x(t), z(t), w(t))\|_2 \to \infty \Rightarrow \phi_1(x(t), z(t), w(t)) \to \infty. \tag{D.10}$$

Moreover, for all initial conditions $x(0) \in \mathbb{R}^n$ we have

$$\phi_1(x(t), z(t), w(t)) > -\infty, \quad \forall t \geq 0. \tag{D.11}$$

Proof. First, we prove the second part. We proceed by contradiction. Assume that there is a trajectory $(x(t), z(t), w(t))$ such that $\phi_1(x(t), z(t), w(t)) \to -\infty$. Let $\widetilde{z} \doteq z - \overline{z}\mathbf{1}$ and $\widetilde{w} \doteq w - \overline{w}\mathbf{1}$ where $\overline{z}$ denotes the average value of the components of vector z. We have that $\phi_1 = \overline{\phi}_1 + \widetilde{\phi}_1$, where

$$\overline{\phi}_1 = \frac{n}{2}\overline{z}^2 + \alpha n(\overline{z}+1)\overline{w} = \frac{n}{2}((1-\|x\|^2)^2 + 2\alpha n(2-\|x\|^2)x^T A x), \tag{D.12}$$

$$\widetilde{\phi}_1 = \frac{1}{2}\begin{bmatrix}\widetilde{z}\\ \widetilde{w}\end{bmatrix}^T \begin{bmatrix} I & \alpha I \\ \alpha I & \beta\mathcal{L}(\mathcal{G})\end{bmatrix}\begin{bmatrix}\widetilde{z}\\ \widetilde{w}\end{bmatrix}. \tag{D.13}$$

Observe that $\widetilde{\phi}_1 \geq 0$ since

$$\begin{bmatrix}\widetilde{z}\\ \widetilde{w}\end{bmatrix}^T \begin{bmatrix} I & \alpha I \\ \alpha I & \beta\mathcal{L}(\mathcal{G})\end{bmatrix}\begin{bmatrix}\widetilde{z}\\ \widetilde{w}\end{bmatrix} = \begin{bmatrix}\widetilde{z}\\ \widetilde{w}\end{bmatrix}^T \begin{bmatrix} I & \alpha I \\ \alpha I & \beta(\mathcal{L}(\mathcal{G}) + \frac{1}{n}\mathbf{1}\mathbf{1}^T)\end{bmatrix}\begin{bmatrix}\widetilde{z}\\ \widetilde{w}\end{bmatrix}, \tag{D.14}$$

and by Schur complement,

$$\begin{bmatrix} I & \alpha I \\ \alpha I & \beta(\mathcal{L}(\mathcal{G}) + \frac{1}{n}\mathbf{1}\mathbf{1}^T)\end{bmatrix} \succ 0 \quad \text{iff} \quad \beta(\mathcal{L}(\mathcal{G}) + \frac{1}{n}\mathbf{1}\mathbf{1}^T) \succ \alpha^2 I, \tag{D.15}$$

which is the case here. Therefore, it suffices to show that $\overline{\phi}_1 > -\infty$. For $\overline{\phi}_1 \to -\infty$, we must have $\|x\|_2 \to \infty$. But as $\|x\|_2 \to \infty$,

$$\overline{\phi}_1 / \|x\|_2^4 \to \frac{n}{2}\frac{x^T(I - 2\alpha A)x}{x^T x} > 0, \tag{D.16}$$

and thus, $\|x\|_2 \to \infty$ implies $\overline{\phi}_1 \to +\infty$ since $I - 2\alpha A \succ 0$.

Regarding the first part, observe that $\|(\widetilde{z}, \widetilde{w})\|_2 \to \infty$ implies $\widetilde{\phi}_1 \to \infty$. Combining this observation with (D.16), we deduce that $\|(x(t), z(t), w(t))\|_2 \to \infty$ implies $\phi_1(x(t), z(t), w(t)) \to \infty$. □

Lemmata D.2.2, D.2.3 and D.2.4, imply part (i) of Theorem 5.3.4. Next, we prove parts (ii)

and (iii). Let $\lambda_1 < \lambda_2 < \cdots < \lambda_n$ be the eigenvalues of A and $v_1, v_2, \ldots, v_n$ the associated unit-norm eigenvectors. We consider the case

$$(x_0, z_0, w_0) = (v_k, \mathbf{0}, \lambda_k \mathbf{1}), \tag{D.17}$$

for some $k \in \{2, \ldots, n\}$. To show that x_0 is unstable, we use Chetaev's theorem ([62] page 125). We consider the infinitesimal perturbation of (x_0, z_0, w_0) as follows

$$x = x_0 + dx = v_k \pm \epsilon v_1, \tag{D.18}$$

$$z = z_0 - dx \odot dx = -\epsilon^2 n v_1 \odot v_1, \tag{D.19}$$

$$w = w_0 + dx \odot (Adx) = \lambda_k \mathbf{1} + \epsilon^2 n \lambda_1 v_1 \odot v_1, \tag{D.20}$$

where ϵ is an arbitrarily small positive constant. Then,

$$\phi_1(x, z, w) = \phi_1(x_0, z_0, w_0) - \epsilon^2 \alpha n (\lambda_k - \lambda_1) + O(\epsilon^4). \tag{D.21}$$

Since $\lambda_k - \lambda_1 > 0$, (D.21) implies that for sufficiently small $\epsilon > 0$, we have

$$\phi_1(x, z, w) < \phi_1(x_0, z_0, w_0). \tag{D.22}$$

The remainder of the proof is an application of Chetaev's theorem. Specifically, we consider the Lyapunov function

$$V_1(x, z, w) = \phi_1(x_0, z_0, w_0) - \phi_1(x, z, w). \tag{D.23}$$

Then, $V_1(x_0, z_0, w_0) = 0$ and by (D.22), $V_1(x, z, w) > 0$ for some (x, z, w) arbitrarily close to (x_0, z_0, w_0). We define the set

$$\Omega_{x_0,r} = \{(x, z, w) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \mid \|(x, z, w) - (x_0, z_0, w_0)\|_2 < r, V_1(x, z, w) > 0\}, \tag{D.24}$$

which is nonempty and $\dot{V}_1(x,z,w) = -\dot{\phi}_1(x,z,w) > 0$ for all $(x,z,w) \in \Omega_{x_0,r}$ for sufficiently small $r > 0$. Then, all the conditions of Chetaev's theorem are satisfied. and we conclude that (x_0, z_0, w_0) is unstable. Intuitively, Chetaev's theorem tells us that the trajectories of the system will exit a ball of radius r centered at (x_0, z_0, w_0) when started in $\Omega_{x_0,r}$. Let (x_r, z_r, w_r) be the point of exit on this ball and $t_r > 0$ be the time of exit. Then, since ϕ_1 is nonincreasing along the trajectories of the system, $(x(t), z(t), w(t)) \in \{(x,z,w) \mid \phi_1(x,z,w) \leq \phi_1(x_r, z_r, w_r)\}$ for all $t \geq t_r$, which does not include a ball of radius $\epsilon > 0$ centered at (x_0, z_0, w_0) for sufficiently small $\epsilon > 0$. Thus, (x_0, z_0, w_0) is not attractive either.

It remains to consider the case

$$(x_0, z_0, w_0) = (\mathbf{0}, \mathbf{1}, \mathbf{0}). \tag{D.25}$$

We consider the infinitesimal perturbation of (x_0, z_0, w_0) as follows

$$x = x_0 + dx = \pm\epsilon v_1, \tag{D.26}$$

$$z = z_0 - dx \odot dx = \mathbf{1} - \epsilon^2 n v_1 \odot v_1, \tag{D.27}$$

$$w = w_0 + dx \odot (Adx) = \epsilon^2 n \lambda_1 v_1 \odot v_1, \tag{D.28}$$

where ϵ is an arbitrarily small positive constant. Then,

$$\phi_1(x,z,w) = \phi_1(x_0, z_0, w_0) - \epsilon^2 n(1 - 2\alpha\lambda_1) + O(\epsilon^4). \tag{D.29}$$

Since $I - 2\alpha A \succ 0$, (D.29) implies that for sufficiently small $\epsilon > 0$, we have

$$\phi_1(x,z,w) < \phi_1(x_0, z_0, w_0). \tag{D.30}$$

The remainder of the proof is exactly analogous.

D.3 Proof of Lemma 5.4.2

As in the proof of Lemma D.2.3, it is easy to see that given that $\mathcal{G}$ is connected, $(\dot{X}, \dot{Z}, \dot{W}) = 0$ implies $Z_i = I - X^T X$ and $W_i = X^T A X$ for all $i = 1, 2, \ldots, n$. Then, we must have

$$X(I - X^T X) + \alpha X X^T A X - \alpha A X (2I - X^T X) = 0. \tag{D.31}$$

By left-multiplying with X^T and after some manipulations, we obtain

$$(I - X^T X) X^T B_\alpha X + X^T B_\alpha X (I - X^T X) = 0, \tag{D.32}$$

where $B_\alpha \doteq I - 2\alpha A \succ 0$. Let $X = U \Sigma Q^T$ be an SVD of X. Let $q_1, \ldots, q_k \in \mathbb{R}^k$ be the columns of Q. Multiplying (D.32) by q_i^T on the left by q_i on the right we obtain

$$\sigma_i^2 (\sigma_i^2 - 1)(U^T B_\alpha U)_{ii} = 0. \tag{D.33}$$

Since $U^T B_\alpha U \succ 0$, it follows that $(U^T B_\alpha U)_{ii} > 0$ and thus, σ_i is either 0 or 1. Finally, (5.32) can be easily obtained from (D.31) by observing that $X = X X^T X$.

D.4 Proof of Theorem 5.4.4

The time derivative of the potential function ϕ_2 as defined in (5.34) is given by:

$$\dot{\phi}_2 = -\operatorname{tr}\left(\begin{bmatrix} Z \\ W \end{bmatrix}^T (P_{\alpha,\beta} \otimes I_k) \begin{bmatrix} Z \\ W \end{bmatrix} \right) - 2n \sum_{i=1}^{n} \|\dot{X}_i\|_F^2 \leq 0, \tag{D.34}$$

where $P_{\alpha,\beta}$ as defined in (D.4). Thus, ϕ_2 is nonincreasing along the trajectories of the system at hand. Then, similarly to the $k = 1$ case, $\dot{\phi}_2(X, Z, W) = 0$ if and only if (X, Z, W) is an

equilibrium. In addition, we have that

$$\|(X(t), Z(t), W(t))\|_F \to \infty \Rightarrow \phi_2(X(t), Z(t), W(t)) \to \infty, \tag{D.35}$$

and for all initial conditions $\{X_i(0)\}_{i=1}^n$ and for all $t \geq 0$, we have that

$$\phi_2(X(t), Z(t), W(t)) > -\infty.$$

We conclude that the trajectories of the system asymptotically approach some set of equilibria. So, part (i) has been proved.

Now, we prove parts (ii) and (iii). First, we consider the case where $\dot{\phi}_2(X_0, Z_0, W_0) = 0$ and X has at least one singular value equal to 0, that is,

$$X_0 = UQ^T = \sum_{i=1}^{r} u_i q_i^T, \quad r < k. \tag{D.36}$$

Let $dX = \pm\epsilon u_{r+1} q_{r+1}^T$, where $u_{r+1} \in \mathbb{R}^m$ is a unit-norm vector perpendicular to $\text{span}(\{u_i\}_{i=1}^r)$ and $q_{r+1} \in \mathbb{R}^k$ is a unit-norm vector perpendicular to $\text{span}(\{q_i\}_{i=1}^r)$. We infinitesimally perturb (X_0, Z_0, W_0) as follows:

$$X = X_0 + dX = X \pm \epsilon u_{r+1} q_{r+1}^T, \tag{D.37}$$

$$Z_i = Z_{0i} - n dX_i^T dX_i, \tag{D.38}$$

$$Z_i = W_{0i} + n dX_i^T A_i^T dX_i. \tag{D.39}$$

Then, we have that

$$\phi_2(X, Z, W) - \phi_2(X_0, Z_0, W_0) = -\epsilon^2 n u_{r+1}^T (I - 2\alpha A) u_{r+1} + O(\epsilon^4). \tag{D.40}$$

Since $I - 2\alpha A \succ 0$, we get that for sufficiently small $\epsilon > 0$,

$$\phi_2(X, Z, W) < \phi_2(X_0, Z_0, W_0). \tag{D.41}$$

Let Θ_{X_0} denote the closed connected set of equilibria that contains (X_0, Z_0, W_0). Observe that Θ_{X_0} is a closed invariant set for our system and its definition depends only on X_0. We consider the function

$$V_2(X, Z, W) = \phi_2(X_0, Z_0, W_0) - \phi_2(X, Z, W). \tag{D.42}$$

Then, $V_2(X_0, Z_0, W_0) = 0$ and by (D.22), $V_2(X, Z, W) > 0$ for some (X, Z, W) arbitrarily close to (X_0, Z_0, W_0). We define the set

$$\Omega_{X_0,r} = \{(X, Z, W) \mid \text{dist}((X, Z, W), \Theta_{X_0}) < r, V_2(X, Z, W) > 0\}, \tag{D.43}$$

which is nonempty and $\dot{V}_2(X, Z, W) > 0$ for all $(X, Z, W) \in \Omega_{0,r}$ for sufficiently small $r > 0$. With a reasoning exactly as in the proof of Chetaev's theorem ([62] page 125), the trajectories $(X(t), Z(t), W(t))$ that start in $\Omega_{X_0,r}$ must eventually leave $\Omega_{X_0,r}$ through the set

$$\{(X, Z, W) \mid \text{dist}((X, Z, W), \Theta_{X_0}) = r, V_2(X, Z, W) > 0\}. \tag{D.44}$$

We conclude that Θ_{X_0} is a uniformly unstable set of equilibria, which roughly means from arbitrarily close to any point of Θ_{X_0}, the trajectories of the system escape the set $\{(X, Z, W) \mid \text{dist}((X, Z, W), \Theta_{X_0}) < r\}$ for some sufficiently small $r > 0$. Let (X_r, Z_r, W_r) be the point of exit and $t_r > 0$ be the time of exit. Then, since ϕ_2 is nonincreasing along the trajectories of the system, it follows that

$$(X(t), Z(t), W(t)) \in \{(X, Z, W) \mid \phi_2(X, Z, W) \leq \phi_2(X_r, Z_r, W_r)\}, \tag{D.45}$$

for all $t \geq t_r$, which does not include the set

$$\{(X, Z, W) \mid \text{dist}((X, Z, W), \Theta_{X_0}) < \epsilon\}, \tag{D.46}$$

for sufficiently small $\epsilon > 0$. Thus, Θ_{X_0} is (uniformly) non-attractive.

Next, we consider the case $X_0^T X_0 = I$. We have that (D.31) implies, in this case, that

$$AX_0 = X_0 X_0^T A X_0, \tag{D.47}$$

which shows that span(X_0) is an invariant subspace of A Since X_0 has orthogonal columns, it follows that $X_0 = UQ$ for some arbitrary $k \times k$ orthogonal matrix Q, and U has as columns some unit-norm eigenvectors of A. Consider the case when some column of U is not equal to an eigenvector of A associated with one of the smallest k eigenvalues. Without loss of generality, assume that this column is the first and let $l \in \{1, 2, \ldots, k\}$ such that v_l is not a column of U. Let

$$dX = \pm\epsilon \begin{bmatrix} v_l & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} Q, \tag{D.48}$$

for some small positive scalar ϵ. We infinitesimally perturb (X_0, Z_0, W_0) as follows:

$$X = X_0 + dX, \tag{D.49}$$

$$Z_i = Z_{0i} - n dX_i^T dX_i, \tag{D.50}$$

$$Z_i = W_{0i} + n dX_i^T A_i^T dX_i. \tag{D.51}$$

Then, we have that

$$\phi_2(X, Z, W) - \phi_2(X_0, Z_0, W_0) = -\epsilon^2 \alpha n(\lambda_k - \lambda_1) + O(\epsilon^4). \tag{D.52}$$

Thus, for sufficiently small $\epsilon > 0$,

$$\phi_2(X, Z, W) < \phi_2(X_0, Z_0, W_0). \tag{D.53}$$

The remainder of the proof is as in the previous case.

Appendix E

Proofs for Chapter 6

E.1 Proof of Lemma 6.2.6

Equation (6.9) implies that for any vector x, we have

$$\begin{bmatrix} x \\ -x \end{bmatrix}^T \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} x \\ -x \end{bmatrix} \geq 0 \tag{E.1}$$

or equivalently

$$x^T(\Sigma_{xx} + \Sigma_{yy} - 2\Sigma_{xy})x \geq 0. \tag{E.2}$$

The desired result trivially follows.

E.2 Problems with linear objective and spectral norm constraints

In this section, we consider convex optimization problems of the form:

$$\begin{array}{ll} \underset{X\in\mathbb{R}^{n\times n}}{\text{maximize}} & 2\,\mathrm{tr}(C^T X) \\ \text{subject to} & X^T X \preceq I. \end{array} \tag{E.3}$$

Let Z be the dual variable associated with the inequality $X^T X - I \preceq 0$. The Lagrangian for problem (E.3) is

$$L(X, Z) = -2\,\mathrm{tr}(C^T X) + \mathrm{tr}(Z(X^T X - I)). \tag{E.4}$$

The Karush-Kuhn-Tucker (KKT) conditions for problem (E.3) are:

$$X^T X - I \preceq 0, \tag{E.5}$$

$$Z \succeq 0, \tag{E.6}$$

$$Z(X^T X - I) = 0, \tag{E.7}$$

$$XZ = C. \tag{E.8}$$

For a primal-dual optimal pair (X, Z), we have that $ZX^T X = Z$ and $XZ = C$ imply

$$Z^2 = C^T C. \tag{E.9}$$

Now, let $C = U\Sigma V^T$ be an SVD for C. Equation (E.9) can be equivalently rewritten as

$$Z^2 = V\Sigma^2 V^T. \tag{E.10}$$

It can be easily seen that $(X^\star, Z^\star) = (UV^T, V\Sigma V^T)$ is a primal-dual optimal pair for (E.3)

and the optimal value of (E.3) is $p^\star = 2\|C\|_\star$.

E.3 Proof of Proposition 6.3.1

Let $\widetilde{f}(K) \doteq \operatorname{tr}(\Sigma_{xx}^+)$ for a fixed Σ_{xy} satisfying (6.9). The Hessian of $\widetilde{f}$ can be computed by

$$\operatorname{Hess} \widetilde{f}(K)[U,U] = \operatorname{tr}(USU^T) = u^T(I \otimes S)u, \tag{E.11}$$

where $u = \operatorname{vec}(U)$, and

$$S = \begin{bmatrix} C & D \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} C^T \\ D^T \end{bmatrix} + \Sigma_\eta \succeq 0. \tag{E.12}$$

It follows that the Hessian of $\widetilde{f}$ is positive semidefinite. Thus, for a fixed Σ_{xy} satisfying (6.9) $\operatorname{tr}(\Sigma_{xx}^+)$ is convex in K. The remainder of the proof is straightforward.

E.4 Proof of Lemma 6.3.2

First, it is easy to see that if $E[\widetilde{x}] = \overline{x}$, $E[\widetilde{y}] = \overline{y}$ and $E[\eta] = 0$ then $E[\widetilde{x}^+] = E[\widetilde{x}] = \overline{x}$. Now, one has to show that if $\Sigma_{xx} \succeq \widetilde{\Sigma}_{xx}$ and $\Sigma_{yy} \succeq \widetilde{\Sigma}_{yy}$ then $\operatorname{tr}(\Sigma_{xx}^{+\star}) \geq \operatorname{tr}(\widetilde{\Sigma}_{xx})$. We have that

$$\Sigma_{xx}^{+\star} - \widetilde{\Sigma}_{xx} \succeq -K^\star D(\Sigma_{xy}^{\star T} - \widetilde{\Sigma}_{xy}^T)(I - C^T K^{\star T}) - (I - K^\star C)(\Sigma_{xy}^\star - \widetilde{\Sigma}_{xy})D^T K^{\star T}, \tag{E.13}$$

since $\Sigma_{xx} \succeq \widetilde{\Sigma}_{xx}$, $\Sigma_{yy} \succeq \widetilde{\Sigma}_{yy}$ and $\Sigma_\eta \succ 0$. Since trace is S_+^n-nondecreasing, we get

$$\operatorname{tr}(\Sigma_{xx}^{+\star} - \widetilde{\Sigma}_{xx}) \geq -2 \operatorname{tr}\big((I - K^\star C)(\Sigma_{xy}^\star - \widetilde{\Sigma}_{xy})D^T K^{\star T}\big) \geq 0, \tag{E.14}$$

since $\Sigma_{xy}^\star$ is optimal over all Σ_{xy} satisfying (6.9). Verifying that $\widetilde{\Sigma}_{xy}$ satisfies (6.9) is straightforward. The proof is complete.

E.5 Computing the Newton step

First of all, the differential of $f_1(Q)$ at the direction of ΔQ is given by

$$Df_1(Q)[\Delta Q] = \Sigma_{yy}^{-1/2} \left(\Delta Q^T \Sigma_{xx}^{-1} Q + Q^T \Sigma_{xx}^{-1} \Delta Q \right) \Sigma_{yy}^{-1/2}. \tag{E.15}$$

For small ΔX, we have the first order approximation [18]:

$$\log\det(X + \Delta X) \approx \log\det(X) + \mathrm{tr}(X^{-1}\Delta X) \tag{E.16}$$

and thus, using the chain rule, we obtain

$$\mathrm{grad}_Q \log\det(-f_1(Q)) = 2\Sigma_{xx}^{-1} Q \Sigma_{yy}^{-1/2} f_1(Q)^{-1} \Sigma_{yy}^{-1/2}. \tag{E.17}$$

The gradient of f with respect to X first, and then, with respect to Q can be computed as follows:

$$\begin{aligned} \mathrm{grad}_X f(X,Q) &= 2 \left(\begin{bmatrix} C & D \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & Q \\ Q^T & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} C^T \\ D^T \end{bmatrix} + \Sigma_\eta \right) X - 2(C\Sigma_{xx} + DQ^T), \\ \mathrm{grad}_Q f(X,Q) &= 2(C^T X X^T D - X^T D). \end{aligned} \tag{E.18}$$

Let $g_1(Q) \doteq \Sigma_{yy}^{-1/2} f_1(Q)^{-1} \Sigma_{yy}^{-1/2}$. Using $(X + \Delta X)^{-1} \approx X^{-1} - X^{-1}\Delta X X^{-1}$ for small ΔX and the chain rule, we obtain the following system of linear equations for the Newton step $(\Delta X_{nt}, \Delta Q_{nt})$:

$$\begin{aligned} &\left(\begin{bmatrix} C & D \end{bmatrix} \begin{bmatrix} \Sigma_{xx} & Q \\ Q^T & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} C^T \\ D^T \end{bmatrix} + \Sigma_\eta \right) \Delta X_{nt} + (C\Delta Q_{nt} D^T X - D\Delta Q_{nt}^T (I - C^T X)) \\ &\qquad = -\frac{1}{2t} \mathrm{grad}_X f_t(X,Q), \end{aligned} \tag{E.19}$$

and

$$
\begin{aligned}
&2t(C^T\Delta X_{nt}X^TD - (I - C^TX)\Delta X_{nt}^TD) \\
&-2\Sigma_{xx}^{-1}Qg_1(Q)\big(\ \Delta Q_{nt}^T\Sigma_{xx}^{-1}Q + Q^T\Sigma_{xx}^{-1}\Delta Q_{nt}\big)g_1(Q) + 2\Sigma_{xx}^{-1}\Delta Q_{nt}g_1(Q) = -\operatorname{grad}_Q f_t(X,Q).
\end{aligned}
\tag{E.20}
$$

Appendix F

Proofs for Chapter 7

F.1 Proof of Lemma 7.5.4

First, we compute an upper bound on the maximum eigenvalue of the Hessian of the objective. The Hessian of the pairwise potential φ_{ij} can be computed as follows. Let $(R_i(t), R_j(t))$ be a geodesic of $SO(3) \times SO(3)$. By differentiating once with respect to time t, we obtain

$$\dot{\varphi}_{ij}(t) = \mathrm{tr}(\dot{R}_i(t) M_{ij} R_j(t)^T) + \mathrm{tr}(R_i(t) M_{ij} \dot{R}_j(t)^T). \tag{F.1}$$

Differentiating once more with respect to t, we obtain

$$\begin{aligned}\ddot{\varphi}_{ij}(t) = {} & \mathrm{tr}(\ddot{R}_i(t) M_{ij} R_j(t)^T) + \mathrm{tr}(R_i(t) M_{ij} \ddot{R}_j(t)^T) \\ & + \mathrm{tr}(\dot{R}_i(t)^T \dot{R}_j(t) M_{ji}) + \mathrm{tr}(\dot{R}_j(t)^T \dot{R}_i(t) M_{ij}).\end{aligned}$$

Let $P_{ij} \doteq R_i M_{ij} R_j^T$. At $t = 0$, we have that

$$\ddot{\varphi}_{ij}(0) = \begin{bmatrix} \mathrm{vec}\big(\dot{R}_i(0)\big) \\ \mathrm{vec}\big(\dot{R}_j(0)\big) \end{bmatrix}^T \underbrace{\begin{bmatrix} H_{ii} & H_{ij} \\ H_{ji} & H_{jj} \end{bmatrix}}_{\mathbf{H}_{ij}} \begin{bmatrix} \mathrm{vec}\big(\dot{R}_i(0)\big) \\ \mathrm{vec}\big(\dot{R}_j(0)\big) \end{bmatrix} \tag{F.2}$$

where $H_{ii} = H_{jj} = -\operatorname{sym}(P_{ij})$ and $H_{ij} = H_{ji}^T = P_{ij}$. Using the fact that vectors in (7.23) form orthonormal bases, each summand in (7.23) can be interpreted as a Singular Value Decomposition (SVD) with singular values w_t, w_c, w_b. Combining this with the Gershgorin circle theorem, the maximum eigenvalue of $\mathbf{H}_{ij}$, denoted by $\lambda_{\max}(\mathbf{H}_{ij})$ cannot exceed $2w_{\max}$. Therefore,

$$\ddot{\varphi}_{ij}(0) \leq 4w_{\max}(\|\dot{R}_i(0)\|^2 + \|\dot{R}_j(0)\|^2). \tag{F.3}$$

Now, let $\mathbf{R}(t) = (R_1(t), \ldots, R_N(t))$ be a geodesic on $SO(3)^N$ and let $\varphi(t) \doteq \varphi(\{R_i(t)\}_{i=1}^N)$. At $t = 0$, we have

$$\ddot{\varphi}(0) = \sum_{\{i,j\}\in E} \ddot{\varphi}_{ij}(0) \leq \sum_{\{i,j\}\in E} 4w_{\max}\big(\|\dot{R}_i(0)\|^2 + \|\dot{R}_j(0)\|^2\big) \leq 4w_{\max} d_{\max} \|\dot{\mathbf{R}}(0)\|^2, \tag{F.4}$$

which shows that the maximum eigenvalue of the Hessian of φ cannot exceed $4d_{\max}w_{\max}$ in any point of its domain.

F.2 Closest rotation matrix

In this section, we consider the problem of estimating the closest rotation matrix to a given 3×3 matrix R_0 in the squared Frobenius norm sense, that is

$$\underset{R\in SO(3)}{\text{minimize}} \quad \|R_0 - R\|_F^2. \tag{F.5}$$

Problem (F.5) is equivalent to minimizing $-\operatorname{tr}(R_0^T R)$ over $SO(3)$. Let $f : SO(3) \to \mathbb{R}$ defined by $f(R) = -\operatorname{tr}(R_0^T R)$. The Riemannian gradient of $f(R)$ can be computed by $\operatorname{grad} f(R) = -2R\operatorname{skew}(R^T R_0)$. All critical points of f must satisfy $\operatorname{skew}(R^T R_0) = 0$ or equivalently

$$R^T R_0 = R_0^T R. \tag{F.6}$$

Let $R_0 = U\Sigma V^T$ be an SVD of R_0. Then, condition (F.6) can be equivalently written as

$$\Sigma = U^T R V \Sigma U^T R V. \tag{F.7}$$

Without loss of generality, assume that R_0 has distinct singular values $\sigma_1 > \sigma_2 > \sigma_3 \geq 0$. In this case, it can be easily seen that (F.7) is satisfied if and only if $R = UJV^T$ for some diagonal J that is a square root of the identity, that is $J^2 = I$. If $\det(UV^T) = 1$, then the smallest possible value of f, equal to $-(\sigma_1 + \sigma_2 + \sigma_3)$, out of the possible critical points is achieved for $R = UV^T$. If $\det(UV^T) = -1$, the smallest possible value of f out of the possible critical points is $-(\sigma_1+\sigma_2-\sigma_3)$ and is attained at $R = U\operatorname{diag}(1, 1, -1)V^T$. Overall, if $R_0 = U\Sigma V^T$ is an SVD for R_0, then the optimal solution of (F.5) is given by

$$R^\star = U\operatorname{diag}(1, 1, \det(UV^T))V^T. \tag{F.8}$$

Appendix G

Proofs for Chapter 8

G.1 Proof of Proposition 8.3.3

Let $(R_1, R_2, R_3, (T_{12}, T_{13})) \in SO(3)^3 \times \mathbb{S}_2^{3*}$. We will determine when

$$(R_1, R_2, R_3, (T_{12}, T_{13})) \sim (S_1 R_1, S_2 R_2, S_3 R_3, (U_{12}, U_{13})), \tag{G.1}$$

for $S_1, S_2, S_3 \in SO(3)$ and $(U_{12}, U_{13}) \in \mathbb{S}_2^{3*}$. By definition of the equivalence relation " $\sim$ ", we have that (G.1) holds if and only if

$$\begin{gathered} R_2^T T_{12} e_i^T R_1^T R_3 - R_2^T R_1 e_i T_{13}^T R_3 = \\ s R_2^T S_2^T U_{12} e_i^T R_1^T S_1^T S_3 R_3 - s R_2^T S_2^T S_1 R_1 e_i U_{13}^T S_3 R_3, \end{gathered} \tag{G.2}$$

where $s \in \{-1, +1\}$. The sign s corresponds to the fact that if $\mathcal{T}$ is a valid trifocal tensor then $-\mathcal{T}$ is a valid trifocal tensor too.

According to Theorem 2 of [145], given a trifocal tensor $\mathcal{T}$, there is only one choice of

$R, S \in SO(3)$ and $T, U \in \mathbb{R}^3$ such that $\mathcal{T}$ can be written as

$$\mathcal{T}_i = TS_i^T - R_i U, \tag{G.3}$$

for $i \in \{1, 2, 3\}$, where R_i and S_i denote the i-th column of R and S respectively. However, since $-\mathcal{T}$ is also a valid trifocal tensor, it follows that $-U$ and $-T$ are also valid solutions. Thus, R, S can be uniquely determined and U, T can be determined up to a (common) sign flip. Using this theorem, we get $S_1 = S_2 = S_3 = S$ for some $S \in SO(3)$ and

$$(U_{12}, U_{13}) = \pm(ST_{12}, ST_{13}). \tag{G.4}$$

Now, since T_{12}, T_{13} are assumed to be non-colinear, they form a basis for the $z = 0$ plane. If we write the standard basis vectors e_1 and e_2 as a linear combination of T_{12} and T_{13}, we get

$$(Se_1)_3 = (Se_2)_3 = 0. \tag{G.5}$$

Furthermore, $e_3^T(U_{12} \times U_{13}) > 0$ and $e_3^T(T_{12} \times T_{13}) > 0$ imply $e_3^T S e_3 > 0$. We conclude that S should be of the form

$$S = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ -\sin(\theta) & -\cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} = R_z(\theta). \tag{G.6}$$

We conclude that

$$(R_1, R_2, R_3, T_{12}, T_{13}) \sim (Q_1, Q_2, Q_3, U_{12}, U_{13}), \tag{G.7}$$

implies

$$Q_i = R_z(\theta)R_i, \quad U = R_z(\theta)T, \tag{G.8}$$

$$Q_i = R_z(\theta)R_i, \quad U = R_z(\theta)R_z(\pi)T, \tag{G.9}$$

for $i \in \{1, 2, 3\}$, where $U = (U_{12}, U_{13})$ and $T = (T_{12}, T_{13})$. It can be trivially verified that the above condition is also sufficient.

G.2 Proof of Proposition 8.4.4

The proof that follows is based on the proof of [1] (page 44) for the real projective space. Let $f : \mathcal{M}_\top \to \mathbb{R}$ be an arbitrary smooth function and define $\overline{f} = f \circ \pi : \overline{\mathcal{M}}_\mathcal{T} \to \mathbb{R}$. Consider the function $g : \overline{\mathcal{M}}_\mathcal{T} \to \overline{\mathcal{M}}_\mathcal{T}$ defined by $g(X) = RX$, where $R \in H_z \times H_{x\pi}$ is arbitrary but constant. Clearly, $\overline{f}(g(X)) = \overline{f}(X)$ for all X. By taking the differential of both sides we obtain

$$D\overline{f}(g(X))[Dg(X)[\overline{\xi}_X]] = D\overline{f}(X)[\overline{\xi}_X], \tag{G.10}$$

where $\overline{\xi}_X \in T_X\overline{\mathcal{M}}_\mathcal{T}$ is the horizontal lift at X of a tangent vector $\xi \in T_{[X]}\mathcal{M}_\top$. By the definition of horizontal lifts, we have

$$D\overline{f}(X)[\overline{\xi}_X] = Df(\pi(X))[\xi]. \tag{G.11}$$

Combing equations (G.10) and (G.11) with the fact $Dg(X)[\xi] = R\xi$, we obtain

$$D\overline{f}(RX)[R\overline{\xi}_X] = D\overline{f}(X)[\overline{\xi}_X] = Df(\pi(RX))[\xi], \tag{G.12}$$

since $\pi(RX) = \pi(X)$. This, since it is true for any smooth function f, implies that $D\pi(RX)[R\overline{\xi}_X] = \xi$.

Finally, it remains to show that $R\overline{\xi}_X$ is an element of $\mathcal{H}_{RX}$ given that $\overline{\xi}_X \in \mathcal{H}_X$. First, we

will show that $R\bar{\xi}_X \in T_{RX}\overline{\mathcal{M}}_{\mathcal{T}}$. Let $X = (R_1, R_2, R_3, T)$. Since $\bar{\xi}_X$ is tangent vector of $\overline{\mathcal{M}}_{\mathcal{T}}$, it is of the form

$$\bar{\xi}_X = (R_1\Omega_1, R_2\Omega_2, R_3\Omega_3, \zeta_4), \tag{G.13}$$

for some $\Omega_i \in \mathfrak{so}(3)$ and $\zeta_4 \in T^{\perp}$. Moreover, $R\bar{\xi}_X$ is of the form

$$R\bar{\xi}_X = (RR_1\Omega_1, RR_2\Omega_2, RR_3\Omega_3, R\zeta_4), \tag{G.14}$$

which clearly belongs to the tangent space $T_{RX}\overline{\mathcal{M}}_{\mathcal{T}}$. Finally, $R\bar{\xi}_X$ is a horizontal vector since

$$\overline{g}(R\bar{\xi}_X, \widehat{e_z}RX) = \overline{g}(\bar{\xi}_X, R^T\widehat{e_z}RX) = \overline{g}(\bar{\xi}_X, \widehat{e_z}X) = 0, \tag{G.15}$$

where we used that $R^T\widehat{e_z}R = \widehat{e_z}$ for any $R \in H_z \times H_{x\pi}$. Since $D\pi(RX)[R\bar{\xi}_X] = \xi$ and $R\bar{\xi}_X$ is an element of $\mathcal{H}_{RX}$, we conclude that $R\bar{\xi}_X$ is the horizontal lift of ξ at RX, i.e.

$$\bar{\xi}_{RX} = R\bar{\xi}_X. \tag{G.16}$$

G.3 Proof of Proposition 8.4.7

Let $X = (R_1, R_2, R_3, T) \in \overline{\mathcal{M}}_{\mathcal{T}}$ and corresponding tangent vector $\xi = (\xi_1, \xi_2, \xi_3, \xi_4) \in T_X\overline{\mathcal{M}}_{\mathcal{T}}$. Furthermore, assume that $\xi = \dot{\gamma}_{X,\xi}(0) \in \mathcal{H}_X$. It follows that $\xi_i \perp \widehat{e_z}R_i$ for $i = 1, 2, 3$ and $\xi_4 \perp \widehat{e_z}T$.

For the rotational components, recall that $\gamma_{R_i,\xi_i}(t) = R_i \exp_I(tR_i^T\xi_i)$ and $\dot{\gamma}_{R_i,\xi_i}(t) = \xi_i \exp_I(tR_i^T\xi_i)$ where $\exp_I$ denotes the usual matrix exponential. Thus, we get

$$\operatorname{tr}\left(\dot{\gamma}_{R_i,\xi_i}(t)^T(\widehat{e_z}\gamma_{R_i,\xi_i}(t))\right) = \operatorname{tr}\left(\xi_i^T\widehat{e_z}R_i\right) = 0. \tag{G.17}$$

Similarly, for the component $T \in \mathbb{S}^3_2$, recall that

$$\gamma_{T,\xi_4}(t) = T\cos(\|\xi_4\|t) + \frac{\xi_4}{\|\xi_4\|}\sin(\|\xi_4\|t), \tag{G.18}$$

$$\dot{\gamma}_{T,\xi_4}(t) = -T\|\xi_4\|\sin(\|\xi_4\|t) + \xi_4\cos(\|\xi_4\|t). \tag{G.19}$$

It can be easily seen that

$$\mathrm{tr}(\dot{\gamma}_{T,\xi_4}(t))^T \widehat{e}_z \gamma_{T,\xi_4}(t)) = 0, \tag{G.20}$$

since $T \perp \widehat{e}_z T$, $\xi_4 \perp \widehat{e}_z\xi_4$ and $\xi_4 \perp \widehat{e}_z T$. As a conclusion, $\dot{\gamma}_{X,\xi}(t)$ is perpendicular to the vertical space $\mathcal{V}_{\gamma_{X,\xi}(t)}$ for all t. Thus, $\dot{\gamma}_{X,\xi}(t) \in \mathcal{H}_{\gamma_{X,\xi}(t)}$ for all t, as desired.

G.4 Derivation of Riemannian gradient

Let $X(t)$ be a geodesic curve of the form $X(t) = (R_1(t), R_2(t), R_3(t), T(t))$. Let $\mathcal{T}(t) \doteq \mathcal{T}(X(t))$. The tangent of $\mathcal{T}(t)$ is given by

$$\dot{\mathcal{T}}_i = R_2^T(T_{12}e_i^T\dot{R}_1^T - \dot{R}_1 e_i T_{13}^T)R_3 + \dot{R}_2^T R_2 \mathcal{T}_i + \mathcal{T}_i R_3^T \dot{R}_3 + R_2^T(\dot{T}_{12}e_i^T R_1 - R_1 e_i \dot{T}_{13}^T)R_3, \tag{G.21}$$

for $i = 1, 2, 3$. Now, consider the function $f_{\overline{\mathcal{M}}_\mathcal{T}}(X(t)) = f(\mathcal{T}(t))$. At $t = 0$, we have

$$\overline{g}\big(\dot{X}, \mathrm{grad}\, f_{\overline{\mathcal{M}}_\mathcal{T}}(X)\big) = \langle \dot{\mathcal{T}}, \mathrm{grad}\, f(\mathcal{T})\rangle, \tag{G.22}$$

where $\langle\cdot,\cdot\rangle$ denotes the usual Euclidean inner product. To alleviate the notation, let $G = \mathrm{grad}\, f(\mathcal{T}) \in \mathbb{R}^{3\times3\times3}$ and let G_i be i-th slice of G for $i = 1, 2, 3$. Then $\langle\dot{\mathcal{T}}, G\rangle =$

$\sum_{i=1}^{3} \operatorname{tr}\big(G_i^T \dot{\mathcal{T}}_i\big)$ and

$$\operatorname{tr}\big(G_i^T \dot{\mathcal{T}}_i\big) = \operatorname{tr}\big(\dot{R}_1^T (R_3 G_i^T R_2^T T_{12} e_i^T - R_2 G_i R_3^T T_{13} e_i^T)\big) + \operatorname{tr}\big(\dot{R}_2^T R_2 \mathcal{T}_i G_i^T\big) + \operatorname{tr}\big(\dot{R}_3^T R_3 \mathcal{T}_i^T G_i\big) \tag{G.23}$$

$$+ \operatorname{tr}\big(\dot{T}_{12}^T R_2 G_i R_3^T R_1 e_i\big) - \operatorname{tr}\big(\dot{T}_{13}^T R_3 G_i^T R_2^T R_1 e_i\big). \tag{G.24}$$

Since the manifold $\overline{\mathcal{M}}_{\mathcal{T}}$ is a submanifold of a Euclidean space, it follows that any Euclidean vector $\zeta \in T_X \overline{\mathcal{M}}_{\mathcal{T}}$ can be uniquely orthogonally decomposed as $\zeta = \zeta' + \zeta''$ with $\zeta' \in T_X \overline{\mathcal{M}}_{\mathcal{T}}$ and $\zeta'' \in (T_X \overline{\mathcal{M}}_{\mathcal{T}})^{\perp}$. Hence $\langle \xi, \zeta \rangle = \langle \xi, \zeta' \rangle$ for any $\xi \in T_X \overline{\mathcal{M}}_{\mathcal{T}}$. Using this fact, the provided formulas easily follows.

Bibliography

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008.

[2] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008.

[3] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[4] Rosario Aragues, Guodong Shi, Dimos V Dimarogonas, C Sagues, and Karl Henrik Johansson. Distributed algebraic connectivity estimation for adaptive event-triggered consensus. In *American Control Conference (ACC), 2012*, pages 32–37. IEEE, 2012.

[5] Rosario Aragues, Guodong Shi, Dimos V Dimarogonas, Carlos Sagüés, Karl Henrik Johansson, and Youcef Mezouar. Distributed algebraic connectivity estimation for undirected graphs with upper and lower bounds. *Automatica*, 50(12):3253–3259, 2014.

[6] Pablo O Arambel, Constantino Rago, and Raman K Mehra. Covariance intersection algorithm for distributed spacecraft state estimation. In *Proceedings of the IEEE American Control Conference*, volume 6, pages 4398–4403, 2001.

[7] Mica Arie-Nachimson, Shahar Z Kovalsky, Ira Kemelmacher-Shlizerman, Amit Singer, and Ronen Basri. Global motion estimation from point matches. In *International Con-*

ference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), pages 81–88. IEEE, 2012.

[8] Søren Asmussen. *Applied probability and queues*, volume 51. Springer Science & Business Media, 2008.

[9] Alexander Bahr, Matthew R Walter, and John J Leonard. Consistent cooperative localization. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3415–3422. IEEE, 2009.

[10] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *Advances in neural information processing systems*, pages 831–837, 2001.

[11] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, 1999.

[12] Dimitri P Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.

[13] Dimitri P Bertsekas and John N Tsitsiklis. *Parallel and distributed computation: numerical methods*, volume 23. Prentice hall Englewood Cliffs, NJ, 1989.

[14] Nicolas Boumal, Bamdev Mishra, P.-A. Absil, and Rodolphe Sepulchre. Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15:1455–1459, 2014.

[15] Sean L Bowman, Nikolay Atanasov, Kostas Daniilidis, and George J Pappas. Probabilistic data association for semantic slam. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1722–1729. IEEE, 2017.

[16] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE/ACM Transactions on Networking (TON)*, 14(SI):2508–2530, 2006.

[17] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Dis-

tributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[18] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[19] Roger W Brockett. Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems. *Linear Algebra and its applications*, 146:79–91, 1991.

[20] Nikos Canterakis. A minimal set of constraints for the trifocal tensor. In *European Conference on Computer Vision*, pages 84–99, 2000.

[21] Luis C Carrillo-Arce, Esha D Nerurkar, José L Gordillo, and Stergios I Roumeliotis. Decentralized multi-robot cooperative localization using covariance intersection. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1412–1417, 2013.

[22] Lingji Chen, P. O. Arambel, and R. K. Mehra. Estimation under unknown correlation: covariance intersection revisited. *IEEE Transactions on Automatic Control*, 47(11):1879–1882, 2002.

[23] Yuxin Chen, Leonidas Guibas, and Qixing Huang. Near-optimal joint object matching via convex relaxation. In *International Conference on Machine Learning*, 2014.

[24] Minsu Cho, Karteek Alahari, and Jean Ponce. Learning graphs to match. In *IEEE Interational Conference on Computer Vision*, 2013.

[25] Minsu Cho, Jungmin Lee, and Kyoung Lee. Reweighted random walks for graph matching. *Computer Vision–ECCV 2010*, pages 492–505, 2010.

[26] J. Cortés. Distributed algorithms for reaching consensus on general functions. *Automatica*, 44(3):726–737, 2008.

[27] Jan Čurn, Dan Marinescu, Niall O'Hara, and Vinny Cahill. Data incest in cooperative

localisation with the common past-invariant ensemble kalman filter. In *Information Fusion (FUSION), 2013 16th International Conference on*, pages 68–76. IEEE, 2013.

[28] Nair Maria Maia De Abreu. Old and new results on algebraic connectivity of graphs. *Linear algebra and its applications*, 423(1):53–73, 2007.

[29] Maria Carmela De Gennaro and Ali Jadbabaie. Decentralized control of connectivity for multi-agent systems. In *IEEE Conference on Decision and Control*, pages 3628–3633. IEEE, 2006.

[30] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[31] D. Devarajan and R. Radke. Calibrating distributed camera networks using belief propagation. *EURASIP Journal of Applied Signal Processing*, pages 221–221, 2007.

[32] M.P. do Carmo. *Riemannian Geometry*. Mathematics (Boston, Mass.). Birkhäuser, 1992.

[33] Robert Dobrow. *Introduction to Stochastic Processes with R*. Wiley, 2016.

[34] Alan Edelman, Tomás A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353.

[35] Tolga Eren, W Whiteley, A. S. Morse, P. N. Belhumeur, and Brian D. O. Anderson. Sensor and Network Topologies of Formations with Direction, Bearing and Angle Information between Agents. *Decision and Control . . .*, (December):3064–3069, 2003.

[36] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous robots*, 8(3):325–344, 2000.

[37] Mauro Franceschelli, Andrea Gasparri, Alessandro Giua, and Carla Seatzu. Decen-

tralized laplacian eigenvalues estimation for networked multi-agent systems. In *IEEE Conference on Decision and Control*, pages 2717–2722. IEEE, 2009.

[38] Jean Gallier and Jocelyn Quaintance. Notes on differential geometry and lie groups a second course.

[39] Yongxin Gao, X Rong Li, and Enbin Song. Robust linear estimation fusion with allowable unknown cross-covariance. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(9):1314–1325, 2016.

[40] C. Geyer and K. Daniilidis. Mirrors in motion: Epipolar geometry and motion estimation. In *International Conference on Computer Vision*, pages 766–773, 2003.

[41] A Ghosh and S Boyd. Minimax and convex-concave games. *lecture notes for course EE392:'Optimization Projects" Stanford Univ., Stanford, CA*, 2003.

[42] Chris Godsil and Gordon F Royle. *Algebraic graph theory*, volume 207. Springer Science & Business Media, 2013.

[43] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[44] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 447–456, 2015.

[45] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[46] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[47] Richard Hartley, Khurrum Aftab, and Jochen Trumpf. L1 rotation averaging using the weiszfeld algorithm. In *Computer Vision and Pattern Recognition (CVPR), 2011*

IEEE Conference on, pages 3041–3048. IEEE, 2011.

[48] Richard I. Hartley. Projective reconstruction from line correspondences. In *In Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 903–907, 1994.

[49] Richard I. Hartley. Lines and points in three views and the trifocal tensor. *Int. J. Comput. Vision*, 22(2):125–140, March 1997.

[50] Uwe Helmke, Knut Hüper, Pei Yean Lee, and John Moore. Essential matrix estimation using gauss-newton iterations on a manifold. *International Journal of Computer Vision*, 74(2):117–136, 2007.

[51] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[52] Nan Hu, Qixing Huang, Boris Thibert, UG Alpes, and Leonidas Guibas. Distributable consistent multi-object matching.

[53] Qi-Xing Huang and Leonidas Guibas. Consistent shape maps via semidefinite programming. *Computer Graphics Forum*, 32(5):177–186, 2013.

[54] Qi-Xing Huang, Guo-Xin Zhang, Lin Gao, Shi-Min Hu, Adrian Butscher, and Leonidas Guibas. An optimization approach for extracting and encoding consistent maps in a shape collection. *ACM Transactions on Graphics*, 31(6):167, 2012.

[55] Xiangru Huang, Zhenxiao Liang, Xiaowei Zhou, Yao Xie, Leonidas Guibas, and Qixing Huang. Learning transformation synchronization. *arXiv preprint arXiv:1901.09458*, 2019.

[56] Ali Jadbabaie, Jie Lin, and A Stephen Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[57] Simon J Julier and Jeffrey K Uhlmann. New extension of the kalman filter to non-

linear systems. In *AeroSense'97*, pages 182–193. International Society for Optics and Photonics, 1997.

[58] Simon J Julier and Jeffrey K Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the IEEE American Control Conference*, volume 4, pages 2369–2373, 1997.

[59] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[60] David Kempe and Frank McSherry. A decentralized algorithm for spectral analysis. *Journal of Computer and System Sciences*, 74(1):70–83, 2008.

[61] David G. Kendall. Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces. *Bulletin of the London Mathematical Society*, 16:81–121, 1984.

[62] Hassan K Khalil. *Noninear Systems*. Prentice-Hall, 1996.

[63] Solmaz S Kia, Stephen Rounds, and Sonia Martinez. Cooperative localization for mobile agents: a recursive decentralized algorithm based on kalman-filter decoupling. *IEEE Control Systems*, 36(2):86–101, 2016.

[64] Vladimir G Kim, Wilmot Li, Niloy J Mitra, Stephen DiVerdi, and Thomas A Funkhouser. Exploring collections of 3d models using fuzzy correspondences. *ACM Transactions on Graphics*, 31(4):54, 2012.

[65] J. Knuth and P. Barooah. Distributed collaborative 3D pose estimation of robots from heterogeneous relative measurements: an optimization on manifold approach. *Robotica*, pages 1–29, 2013.

[66] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012.

[67] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[68] John M. Lee. Introduction to smooth manifolds. 2000.

[69] S. Leonardos, C. Allen-Blanchette, and J. Gallier. The exponential map for the group of similarity transformations and applications to motion interpolation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 377–382, 2015.

[70] S. Leonardos and K. Daniilidis. A distributed optimization approach to consistent multiway matching. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 89–96, 2018.

[71] S. Leonardos, V. Preciado, and K. Daniilidis. A dynamical systems approach to distributed eigenvector computation. In *IEEE International Conference on Decision and Control*, 2017.

[72] Spyridon Leonardos and Kostas Daniilidis. A game-theoretic approach to robust fusion and kalman filtering under unknown correlations. In *American Control Conference (ACC), 2017*, pages 2568–2573. IEEE, 2017.

[73] Spyridon Leonardos, Roberto Tron, and Kostas Daniilidis. A metric parametrization for trifocal tensors with non-colinear pinholes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[74] Spyridon Leonardos, Xiaowei Zhou, and Kostas Daniilidis. Articulated motion estimation from a monocular image sequence using spherical tangent bundles. In *IEEE International Conference on Robotics and Automation*, pages 587–593, 2016.

[75] Spyridon Leonardos, Xiaowei Zhou, and Kostas Daniilidis. Distributed consistent data association. *arXiv preprint arXiv:1609.07015*, 2016.

[76] Spyridon Leonardos, Xiaowei Zhou, and Kostas Daniilidis. Distributed consistent data

association via permutation synchronization. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 2645–2652. IEEE, 2017.

[77] Keith YK Leung, Timothy D Barfoot, and Hugh HT Liu. Decentralized localization of sparsely-communicating robot networks: A centralized-equivalent approach. *IEEE Transactions on Robotics*, 26(1):62–77, 2010.

[78] Chaoyong Li and Zhihua Qu. Distributed estimation of algebraic connectivity of directed networks. *Systems & Control Letters*, 62(6):517–524, 2013.

[79] Hao Li and Fawzi Nashashibi. Cooperative multi-vehicle localization using split covariance intersection filter. *IEEE Intelligent transportation systems magazine*, 5(2):33–44, 2013.

[80] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[81] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[82] Yi Ma. *An invitation to 3-D vision: from images to geometric models*. Springer, 2004.

[83] Yi Ma, Jana Kosecká, and Shankar Sastry. Optimization criteria and geometric algorithms for motion and structure estimation. *International Journal of Computer Vision*, 44(3):219–249, 2001.

[84] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[85] Eleonora Maset, Federica Arrigoni, and Andrea Fusiello. Practical and efficient multiview matching. In *Proceedings of IEEE International Conference on Computer Vision*, volume 2, page 5, 2017.

[86] N. Matni and M. B. Horowitz. A convex approach to consensus on SO(n). In *Allerton*

Conference on Communication, Control, and Computing, pages 959–966, 2014.

[87] Mehran Mesbahi and Magnus Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[88] Bojan Mohar, Y Alavi, G Chartrand, and OR Oellermann. The laplacian spectrum of graphs. *Graph theory, combinatorics, and applications*, 2(871-898):12, 1991.

[89] Eduardo Montijano, Juan I Montijano, and Carlos Sagues. Adaptive consensus and algebraic connectivity estimation in sensor networks with chebyshev polynomials. In *IEEE Conference on Decision and Control*, pages 4296–4301. IEEE, 2011.

[90] Esha D Nerurkar and Stergios I Roumeliotis. Power-slam: A linear-complexity, consistent algorithm for slam. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 636–643, 2007.

[91] Esha D Nerurkar, Stergios I Roumeliotis, and Agostino Martinelli. Distributed maximum a posteriori estimation for multi-robot cooperative localization. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 1402–1409. IEEE, 2009.

[92] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

[93] Andy Nguyen, Mirela Ben-Chen, Katarzyna Welnicka, Yinyu Ye, and Leonidas Guibas. An optimization approach to improving collections of shape maps. *Computer Graphics Forum*, 30(5):1481–1491, 2011.

[94] B. Noack, J. Sijs, and U. D. Hanebeck. Inverse covariance intersection: New insights and properties. In *2017 20th International Conference on Information Fusion (Fusion)*, 2017.

[95] Benjamin Noack, Joris Sijs, Marc Reinhardt, and Uwe D. Hanebeck. Decentralized data fusion with inverse covariance intersection. *Automatica*, 79:35 – 41, 2017.

[96] Klas Nordberg. The key to three-view geometry. *Int. J. Comput. Vision*, 94(3):282–294, September 2011.

[97] K.-K. Oh and H.-S. Ahn. Formation control and network localization via orientation alignment. *IEEE Transactions on Automatic Control*, 59(2):540–545, 2014.

[98] Reza Olfati-Saber, J Alex Fax, and Richard M Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[99] Barrett O'Neill. The fundamental equations of a submersion. *The Michigan Mathematical Journal*, 13(4):459–469, 12 1966.

[100] Boris N Oreshkin, Mark J Coates, and Michael G Rabbat. Optimization and analysis of distributed averaging with short node memory. *IEEE Transactions on Signal Processing*, 58(5):2850–2865, 2010.

[101] Deepti Pachauri, Risi Kondor, and Vikas Singh. Solving the multi-way matching problem by permutation synchronization. In *Advances in Neural Information Processing Systems*, pages 1860–1868, 2013.

[102] Théodore Papadopoulo and Olivier Faugeras. A new characterization of the trifocal tensor. In *European Conference on Computer Vision*, pages 109–123, 1998.

[103] Liam Paull, Mae Seto, and John J Leonard. Decentralized cooperative trajectory estimation for autonomous underwater vehicles. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 184–191. IEEE, 2014.

[104] Peter Petersen. *Riemannian geometry*, volume 171. Springer, 2006.

[105] Stephen Phillips and Kostas Daniilidis. All graphs lead to rome: Learning geometric

and cycle-consistent representations with graph convolutional networks. *arXiv preprint arXiv:1901.02078*, 2019.

[106] G. Piovan, I. Shames, B. Fidan, F. Bullo, and B. Anderson. On frame and orientation localization for relative sensing networks. *Automatica*, 49(1):206–213, 2013.

[107] Jean Ponce and Martial Hebert. Trinocular geometry revisited. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[108] Jean Ponce, Kenton McHenry, Théodore Papadopoulo, Monique Teillaud, and Bill Triggs. On the absolute quadratic complex and its application to autocalibration. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 780–787. IEEE, 2005.

[109] Victor M Preciado, Michael M Zavlanos, Ali Jadbabaie, and George J Pappas. Distributed control of the laplacian spectral moments of a network. In *American Control Conference (ACC), 2010*, pages 4462–4467. IEEE, 2010.

[110] M. Reinhardt, B. Noack, P. O. Arambel, and U. D. Hanebeck. Minimum covariance bounds for the fusion under unknown correlations. *IEEE Signal Processing Letters*, 22(9):1210–1214, 2015.

[111] C. Ressl. A minimal set of constraints and a minimal parameterization for the trifocal tensor. In *International Archives of Photogrammetry and Remote Sensing*, pages 277–282, Volume XXXIV / 3A, 2002. talk: Symposium der ISPRS-Comm. III, Graz; 2002-09-09 – 2002-09-13.

[112] Ralph Tyrell Rockafellar. *Convex analysis*. Princeton university press, 2015.

[113] Stergios I Roumeliotis and George A Bekey. Distributed multirobot localization. *IEEE Transactions on Robotics and Automation*, 18(5):781–795, 2002.

[114] Lorenzo Sabattini, Nikhil Chopra, and Cristian Secchi. On decentralized connectivity

maintenance for mobile robotic systems. In *IEEE Conference on Decision and Control*, pages 988–993. IEEE, 2011.

[115] A. Shashua. Algebraic functions for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:779–789, 1994.

[116] B. Shirmohammadi and C.J. Taylor. Self localizing smart camera networks. *ACM Transactions on Sensor Networks*, 2010.

[117] Joris Sijs and Mircea Lazar. Empirical case-studies of state fusion via ellipsoidal intersection. In *FUSION*, pages 1–8, 2011.

[118] Joris Sijs, Mircea Lazar, and PPJVD Bosch. State fusion with unknown correlation: Ellipsoidal intersection. In *Proceedings of the 2010 American Control Conference*, pages 3992–3997. IEEE, 2010.

[119] Stefano Soatto, Ruggero Frezza, and Pietro Perona. Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, 41(3):393–413, 1996.

[120] Harold Wayne Sorenson. *Kalman filtering: theory and application*. IEEE, 1985.

[121] Demetri P Spanos, Reza Olfati-Saber, and Richard M Murray. Dynamic consensus on mobile networks. In *IFAC World Congress*, pages 1–6, 2005.

[122] Minas E. Spetsakis and John Aloimonos. Structure from motion using line correspondences. *International Journal of Computer Vision*, 4(3):171–183, 1990.

[123] Christoph Strecha, Wolfgang von Hansen, Luc Van Gool, Pascal Fua, and Ulrich Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[124] Raghav Subbarao, Yakup Genc, and Peter Meer. Robust unambiguous parametrization of the essential manifold. In *IEEE Conference on Computer Vision and Pattern*

Recognition, 2008.

[125] Raghav Subbarao and Peter Meer. Nonlinear mean shift over riemannian manifolds. *International Journal of Computer Vision*, 84(1):1, 2009.

[126] Johan Thunberg, Johan Markdahl, and Jorge GonÃğalves. Dynamic controllers for column synchronization of rotation matrices: A qr-factorization approach. *Automatica*, 93:20 – 25, 2018.

[127] Johan Thunberg, Wenjun Song, Eduardo Montijano, Yiguang Hong, and Xiaoming Hu. Distributed attitude synchronization control of multi-agent systems with switching topologies. *Automatica*, 50(3):832 – 840, 2014.

[128] P.H.S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. *Image and Vision Computing*, 15:591–605, 1997.

[129] Nikolas Trawny, Stergios I Roumeliotis, and Georgios B Giannakis. Cooperative multi-robot localization under communication constraints. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4394–4400. IEEE, 2009.

[130] R. Tron, B. Afsari, and R. Vidal. Intrinsic consensus on $SO(3)$ with almost-global convergence. In *IEEE International Conference on Decision and Control*, pages 2052–2058, 2012.

[131] R. Tron, B. Afsari, and R. Vidal. Riemannian consensus for manifolds with bounded curvature. *IEEE Transactions on Automatic Control*, 58(4):921–934, 2013.

[132] R. Tron, L. Carlone, F. Dellaert, and K. Daniilidis. Rigid components identification and rigidity enforcement in bearing-only localization using the graph cycle basis. In *IEEE American Control Conference*, 2015.

[133] R. Tron and K. Daniilidis. The space of essential matrices as a riemannian quotient manifold. *SIAM Journal on Imaging Sciences*, 2017.

[134] R. Tron, J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar. A distributed optimization framework for localization and formation control with applications to vision-based measurements. *IEEE Control Systems Magazine*, 2016.

[135] R. Tron and R. Vidal. Distributed 3-D localization of camera sensor networks from 2-D image measurements. *IEEE Transactions on Automatic Control*, 2014.

[136] Roberto Tron, Bijan Afsari, and René Vidal. Riemannian consensus for manifolds with bounded curvature. *IEEE Transactions on Automatic Control*, 58(4):921–934, 2013.

[137] Roberto Tron and Kostas Daniilidis. On the quotient representation for the essential manifold. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1574–1581, 2014.

[138] Roberto Tron, Justin Thomas, Giuseppe Loianno, Kostas Daniilidis, and Vijay Kumar. A distributed optimization framework for localization and formation control: applications to vision-based measurements. *IEEE Control Systems*, 36(4):22–44, 2016.

[139] Roberto Tron and René Vidal. Distributed 3-d localization of camera sensor networks from 2-d image measurements. *IEEE Transactions on Automatic Control*, 59(12):3325–3340, 2014.

[140] Roberto Tron, Xiaowei Zhou, Carlos Esteves, and Kostas Daniilidis. Fast multi-image matching via density-based clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4057–4066, 2017.

[141] A. Vedaldi and B. Fulkerson. VLFeat - an open and portable library of computer vision algorithms. In *ACM International Conference on Multimedia*, 2010.

[142] Andrea Vedaldi and Stefano Soatto. Quick shift and kernel methods for mode seeking. *Computer vision–ECCV 2008*, pages 705–718, 2008.

[143] René Vidal, Yi Ma, Shawn Hsu, and Shankar Sastry. Optimal motion estimation from multiview normalized epipolar constraint. In *ICCV*, pages 34–41, 2001.

[144] Thumeera R Wanasinghe, George KI Mann, and Raymond G Gosine. Decentralized cooperative localization for heterogeneous multi-robot system using split covariance intersection filter. In *Computer and Robot Vision (CRV), 2014 Canadian Conference on*, pages 167–174. IEEE, 2014.

[145] Juyang Weng, Thomas S. Huang, and Narendra Ahuja. Motion and structure from line correspondences; closed-form solution, uniqueness, and optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3):318–336, 1992.

[146] Xun Xu and Shahriar Negahdaripour. Application of extended covariance intersection principle for mosaic-based optical positioning and navigation of underwater vehicles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 2759–2766, 2001.

[147] Junchi Yan, Minsu Cho, Hongyuan Zha, Xiaokang Yang, and Stephen M Chu. Multi-graph matching via affinity optimization with graduated consistency regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(6):1228–1242, 2016.

[148] Junchi Yan, Jun Wang, Hongyuan Zha, Xiaokang Yang, and Stephen Chu. Consistency-driven alternating optimization for multigraph matching: A unified approach. *IEEE Transactions on Image Processing*, 24(3):994–1009, 2015.

[149] Peng Yang, Randy A Freeman, Geoffrey J Gordon, Kevin M Lynch, Siddhartha S Srinivasa, and Rahul Sukthankar. Decentralized estimation and control of graph connectivity for mobile sensor networks. *Automatica*, 46(2):390–396, 2010.

[150] Christopher Zach, Manfred Klopschitz, and Manfred Pollefeys. Disambiguating visual relations using loop constraints. In *IEEE Conference on Computer Vision and Pattern*

Recognition, 2010.

[151] Michael M Zavlanos, Magnus B Egerstedt, and George J Pappas. Graph-theoretic connectivity control of mobile robot networks. *Proceedings of the IEEE*, 99(9):1525–1540, 2011.

[152] Michael M Zavlanos, Ali Jadbabaie, and George J Pappas. Flocking while preserving network connectivity. In *IEEE Conference on Decision and Control*, pages 2919–2924. IEEE, 2007.

[153] Xiaohai Zhang. A fast numerical method for the optimal data fusion in the presence of unknown correlations. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 377–383. IEEE, 2018.

[154] Zaiwei Zhang, Zhenxiao Liang, Lemeng Wu, Xiaowei Zhou, and Qixing Huang. Path-invariant map networks. *arXiv preprint arXiv:1812.11647*, 2018.

[155] Shiyu Zhao and Daniel Zelazo. Localizability and distributed protocols for bearing-based network localization in arbitrary dimensions. *Automatica*, 69:334 – 341, 2016.

[156] Xiaowei Zhou, Menglong Zhu, and Kostas Daniilidis. Multi-image matching via fast alternating minimization. In *IEEE International Conference on Computer Vision*, pages 4032–4040, 2015.