2016-08

A distributed optimization framework for localization and formation control: applications to vision-based measurements

A Distributed Optimization Framework for Localization and Formation Control with Applications to Vision-Based Measurements

Roberto Tron, Justin Thomas, Giuseppe Loianno, Kostas Daniilidis, and Vijay Kumar

Multi-agent systems have been a major area of research for the last fifteen years. This interest has been motivated by tasks that can be executed more rapidly in a collaborative manner, or that are nearly impossible to carry out otherwise. In order to be effective, the agents need to have the notion of a common goal shared by the entire network (for instance, a desired formation), and individual control laws to realize it. The common goal is typically *centralized*, in the sense that involves the state of all the agents at the same time. On the other hand, it is often desirable to have individual control laws that are *distributed*, in the sense that the desired action depends only on the measurements and states available at the node and at a small number of neighbors. This is an attractive quality because it implies an overall system which is modular and intrinsically more robust to communication delays and node failures.

Regarding the measurements available at each agent, a popular choice is to use simple inexpensive sensors such as Inertial Measurements Units (IMUs) and cameras. This applies not only in distributed control and estimation applications (which are of central interest in this article and for which the relevant literature is reviewed below), but also in other domains,

such as Simultaneous Localization and Mapping and control of Micro Aerial Vehicles (see, for instance, [25], [48] and references within). This common combination of sensors is rich enough to gather interesting and useful information, but it comes with a set of peculiarities that need to be taken into account during the design of the distributed algorithms. For instance, IMU measurements provide reasonably accurate information on instantaneous rotational velocities and linear accelerations, but the integration of these into longer-term absolute positions is prone to the accumulation of errors. On the other hand, cameras can provide accurate direction (bearing) information, but the estimation of distances is typically noisier and it is not possible without a known structure in the environment.

This article considers two central problems in multi-agent systems: mutual localization (estimating the pose of each static agent with respect to a common reference frame) and formation control (maneuvering the agents to achieve a specified set of relative positions or directions). Both problems involve two aspects: a geometric one given by the geometry of the poses (rotation and/or translation) of the agents, and a graph-theoretic one where vertices in a graph represent agents and edges are associated to measurements or other pairwise quantities. The present work focuses on vision-based settings where bearing measurements (i.e., measures of relative direction without distance) have special importance. Commonalities, differences, and synergies between the estimation and control problems are highlighted.

After giving a general overview of the state-of-the-art from the literature, this work concentrates on a particular set of theoretical and practical tools that can be applied to both the mutual localization and formation control problems. In particular, the treatment below shows:

1) How the notions of shape decomposition and rigidity characterize the well-posedness of

the problems.

2) How to encode the desired solutions (localization and formation configuration) into the global minimizers of network-wide objective functions, and how these are related to the notion of rigidity.
3) How to obtain algorithms to minimize the network-wide costs that are distributed, in the sense that each node only requires communication in a local neighborhood.
4) How to obtain estimates of the convergence basin of the algorithms that, in some cases, lead to global convergence results (under the assumption of ideal measurements). This can be achieved despite the fact that the costs might be non-convex.
5) How the measurements can be obtained on aerial vehicles from vision and IMU sensors. This requires strategies for identifying and tracking neighboring agents in the images obtained from the onboard camera.

These theoretical and practical tools are validated through simulations and experimental results. In this regard, the results cover both the location estimation and formation control problems, but with a heavier emphasis on the latter.

Overall, the goal of this article is to introduce the reader to a set of theoretical and practical ideas that can be used to build and understand state-of-the-art, vision-based, distributed localization and formation control systems. The authors hope that these ideas will inspire other researchers to tackle the many problems that remain open in these areas.

CHALLENGES IN MUTUAL LOCALIZATION AND FORMATION CONTROL

Three main challenges are common to both the mutual localization and formation control problems. The first one is that, in its most general forms, these problems involve optimization or control over rotations. These belong to a manifold (see side panel "The geometry of rotations") which is non-linear and has a compact topology. This makes the design and analysis of algorithms operating on this space significantly more challenging. For instance, the compact topology of the manifold implies that the only globally convex continuous functions in this space are trivial constant functions. As such, any optimization-based approach then cannot rely on convexity to exclude the presence of local minimizers and show global convergence.

The second challenge is that, as in all distributed algorithms, each node has only *local* information about the state of the entire network (corresponding to itself and its neighboring nodes). In spite of this, the algorithms need to achieve a common *global* objective (such as reaching a predefined formation). This requires mechanisms to coordinate distant agents that cannot interact directly, and that avoid sub-optimal solutions due to the myopic knowledge of the agents. For instance, for optimization-based approaches, collaboration between distant agents is enforced by the presence of a common cost function, but this cost must be free of local minima in order to avoid sub-optimal solutions.

The third and final challenge is given by the use of vision-based sensors. In general, these sensors give *projective* measurements that do not contain distance information. As a consequence, it is possible to obtain only bearing (that is, direction) information between two agents. In some

cases, these measurements can be augmented with distance information, for instance by using an additional depth sensor, or visible structures with known dimensions. However, the noise in such distance measurements is typically higher than in the corresponding bearings, and can have a very different distribution. Mutual localization and formation control algorithms need to be able to deal with these peculiarities, allowing the optional incorporation of distance information with different weights.

These challenges are addressed by the methods illustrated below, leading to distributed algorithms for mutual localization and formation control with guarantees of convergence to a globally optimal solution either without or with distance measurements.

REVIEW OF THE STATE OF THE ART

This section provides a concise review of existing work in mutual localization and formation control. These problems have been of interest for a long time in various communities such as sensor networks, control systems, robotics, and computer vision. As such, the scope of the literature on the various instances that can be formulated for these problems is extensive. The review below gives only a brief glance to the main ideas that have emerged, and focuses more on the papers that consider these problems in a distributed setting and with vision-based, bearing-only measurements.

Mutual localization

The term mutual localization refers to the problem where each node needs to find its own (static) position in a reference frame common to the entire network (this is equivalent to finding relative poses between neighboring agents that are globally consistent [67]). This is a different problem than the one of collaboratively localizing a (possibly moving) target using an already mutually localized network (see [56] for an example). The mutual localization problem is considered by each community under a different light with different tools and priorities.

In the automatic control and sensor network communities, the problem has been considered in different settings (planar versus tridimensional, centralized versus distributed), with different types of measurements (distances, angles of arrival, bearing measurements, coordinate transformations; see [45, 61] and references therein), with presence of *anchors* (that is, nodes with a known position, [2], [45]) or markers [4], or with other special assumptions (such as moving objects [30]) and in the presence of noise [60]. The common theme here is to find solutions that are computationally light-weight, that can be implemented in embedded devices, and that are (in some cases) robust to noise and communication loss.

Among these, for the case of vision-based measurements, Devarajan and Radke [19] propose a method for combining relative poses between cameras (obtained from images), using Belief Propagation to obtain a distributed algorithm. However, this work does not fully consider the non-Euclidean structure of the space of poses, which imposes constraints on measurements along a cycle in the graph. These constraints are instead at the basis of the work of [58], which, however, only considers the rotation part for the pose of each agent and provides an algorithm

which is only partially distributed (the nodes need to communicate through entire cycles, as opposed to only their neighbors).

In computer vision, a traditional problem related to mutual localization is *Structure-from-Motion* (SfM) [39]. Given a set of images, the goal of traditional SfM is to estimate not only the poses of the cameras corresponding to images (as considered in this article) but also a 3-D point cloud reconstruction of the scene. The joint estimation of the two leads to an optimization procedure commonly known as *Bundle Adjustment* (BA). This procedure requires a careful initialization in order to avoid undesired local minima. A way to do this is to first solve the mutual localization problem, that is, to remove the estimation of the 3-D structure from the problem and consider constraints on the poses alone. In the computer vision community, the focus for this subproblem is to obtain solutions that are centralized and that can scale well with the number of images. A subset of works is based on the idea of finding a globally optimal solution after approximating the group structure of the space of rotations. For instance, Govindu [34], Martinec and Pajdla [51], Arie-Nachimson et al. [1], Fredriksson and Olsson [28] propose linear and quadratic relaxations, while Crandall et al. [16] use a discretized version of the problem. Alternative solutions respect the structure of the space of rotations, but only consider local optimization updates, as done by Govindu [35] and Hartley et al. [40]. In all cases, these algorithms either consider only a centralized setting or do not provide any guarantee that the solution found is globally optimal.

In the robotics community, a traditional problem related to both mutual localization and SfM is Simultaneous Localization And Mapping (SLAM). Similarly to SfM, the goal here is to estimate both the motion of a robot and the 3-D structure of the environment. However, in this

case, it is usually assumed that the robot can acquire inertial measurements (from an Inertial Measurement Unit, or IMU) in addition to the images. Again, the joint optimization of motion and structure is prone to converge to sub-optimal solutions. This has led to the *graph-based SLAM*, which drops the estimation of the map and focuses on the poses alone. This approach was originated by Lu and Milios [49], and it has seen numerous contributions, with [47, 42, 36] being the most popular solutions. Other works build on these by considering on-line updates [38, 20], multi-scale solvers [37, 29], large problem sizes [46], or robustness [57, 10]. As in the computer vision community, the main theme in these works is to use local optimization techniques while exploiting the specific structure of the problem to speed-up computations. Again, these works only provide centralized solutions and do not consider the scale ambiguities intrinsic to pure-vision measurements.

With respect to the work illustrated above, the approach presented later in this article gives a completely distributed algorithm, provides guarantees of convergence to a globally optimal solution under ideal conditions, and explicitly considers the challenges of vision-based measurements. As such, this approach follows the same priorities of the automatic control and sensor network communities. In principle, the same algorithms could be also used for SfM and SLAM applications, but other centralized solutions are likely to be more efficient for these settings.

Formation control

The study of the formation control problem has a long history, starting from early papers such as those from Wang [73], Chen and Luh [11], Balch and Arkin [3], Desai et al. [17].

Existing approaches can be broadly classified by:

- The model used for the agents: single integrators, as done in the majority of works, or more general linear and non-linear models [65], such as those derived from mechanical systems [64, 13]. Single integrators lead to simpler control laws and analysis, while other models can be more realistic but pose additional challenges.
- The technique used for obtaining convergence results: Lyapunov's method, as done in the majority of works, or methods based on passivity [41, 12], contraction theory [13], and Hamiltonian bond-graphs [64]. In passivity-based methods, the basic principle is to design local controllers that are passive (intuitively, that do not produce a power gain from the inputs to the outputs); their interconnection can then been shown to be stable under mild conditions on the network topology (for instance, the graph must be symmetric or balanced). Contraction theory methods are instead based on the idea of showing that any two trajectories of the overall system asymptotically converge to each other; this leads to a convergence analysis based on eigenvalues of matrices that combine the local dynamics with the structure of the network (represented by the graph *Laplacian*). The main advantage is that this analysis can be used to show *exponential* stability, with the eigenvalues of the matrices providing bounds on the convergence speed. Finally, Hamiltonian bond-graphs characterize each node in terms of potential and kinetic energies, and consider their interconnection in terms of how these energies are transfered. This formalism is naturally suited to interconnections of electromechanical systems, and local asymptotic convergence guarantees can be obtained by showing that the total energy of the system (potential plus kinetic) decreases along the trajectories of the system.
- The strength of the theoretical results: asymptotic stability, as shown in most of the methods

based on a Lyapunov analysis, in contrast to exponential stability and robustness to external disturbances (as in methods based on contraction theory).

- The type of information on the agents' translations required to compute the control law. This can be the full state with respect to an absolute reference frame, the full relative state between pairs of agents, or only a partial relative state knowledge (such as distance or bearing-only measurements). See the review from Oh et al. [55] for more details on this aspect.
- The use of the relative rotations between agents in the control law (or equivalently, the rotation of each agent with respect to a common reference frame). In general, this information is needed to compare actual measurements with the desired ones that specify the formation. The majority of existing works assume (either implicitly or explicitly) that this information is known. Other works, such as those by Franchi et al. [27], Montijano et al. [52], Oh and Ahn [54] explicitly incorporate the estimation of the relative rotations with the formation control task (and analyze the convergence jointly), or do not require this information at all, as in those by Bishop et al. [6], Zhao et al. [79].

In general, methods based on a Lyapunov analysis tend to be more ad-hoc and oriented toward agents with simple dynamics (simple integrators) and simple interconnections; on the other hand, methods based on other techniques (such as passivity, contraction theory and bond-graphs) can obtain relatively stronger results (agents with higher-order dynamics, directed and time-varying interconnections), but they are harder to adapt to situations where the relative state of the agents is not fully known (which is the case when only bearing or distances are available). By harder, here it is meant that these techniques require the use of additional estimators or other related sub-systems.

Since this article focuses on vision-based applications, the most interesting approaches are those that use relative bearings to specify the desired formation. Focusing on this category alone, the first distinction that can be made is in how the bearing information is actually used in the control law. Approaches such as those by Bishop et al. [5, 7, 8] and Zhao and Zelazo [77] require, during the control operations, also the distances between agents in addition to the relative directions (that is equivalent to say that they require the full relative positions, thus imposing restrictions on their application). Other approaches, such as those by Zhao and Zelazo [78], Franchi et al. [27, 26], and Stacey and Mahony [63] require only one or no distance measurements. This is achieved by either directly specifying a control law that does not require them (as in [78]), or by substituting the unknown distances with quantities estimated from triplets of nodes [27], distributed estimators [26], or on-line local estimators plus information on the agents' velocity [63]. Yet another approach is to use only the internal angles between pairs of bearings measured at the same agent (which are compared to the internal angles expected at the desired formation). This has the advantage that the agents do not need to know their relative orientations (as mentioned above). However, current existing works from Bishop et al. [6] and Zhao et al. [79] are limited to either triangular or 2-D formations with graphs containing a single cycle, respectively.

A different way to classify bearing-based approaches is by considering whether they allow or require leader agents (these are agents that are independently controlled and that do not follow the same formation control law as the others). Intuitively speaking, from the point of view of analysis, the presence of leaders facilitates the derivation of convergence results, because they fix some of the translation and scale ambiguities intrinsic in the formation control problem. Most of the existing works do not consider the presence of leaders, and the behavior of such

algorithms in their presence is not known. The approaches from Franchi et al. [27, 26] instead *require* the presence of two agents leaders. The only work explicitly considering leaders without requiring them is [77]. When proving stability in the presence of leaders, all published works assume them to be stationary.

Regarding the model for the agents, almost all the existing works on bearing-only formation control use simple first-order 2-D or 3-D integrators, possibly augmented with a 2-D rotation (for those approaches jointly considering rotation localization and formation control). The only exception is [63] which, by using the bond-graph approach, uses second-order mechanical systems.

Regarding the convergence guarantees provided, most of the works mentioned above show global asymptotic convergence (since the overall system is time-invariant, this implies global uniform asymptotic stability). The only exceptions are the work from Stacey and Mahony [63], which only provides *local* convergence guarantees, and the works from Zhao et al. [77, 78, 79], which instead show exponential convergence.

In all of the articles above, the measurements are all assumed to be ideal (without noise and without range or field-of-view restrictions), and the measurement graph is assumed to be fixed. Regarding the measurement graph, only the works from Franchi and Giordano [26], Zhao et al. [79], Bishop et al. [6] make specific assumptions on the graph topology. All the other works allow any arbitrary graph, subject to the constraint that the problem must be well posed (that is, the desired bearings and distances must be sufficient to specify the desired formation, as captured by the notion of *rigidity* reviewed later in this article).

With respect to the literature above, the approach to formation control considered in the present work uses the same assumptions as most of the existing solutions (simple integrator model for the agents, known rotation localization, ideal measurements), while providing more flexibility. The approach can be applied to bearing-only measurements, without the use of additional estimators, but it allows the incorporation of any number of optional range measurements; it can handle leaderless or leader-based formations and, in both cases, it uses a Lyapunov approach to show global asymptotic convergence guarantees without restrictions on the graph topology (again, with the caveat that the problem must be rigid in order to be well-posed). This framework is also fairly general (it can be applied to various instances of localization problems in addition to formation control problems), it does not require additional estimators to complement bearing-only measurements (as required by bond-graph approaches) and can be applied in situations not covered by other techniques (it does not require differential stability, as in contraction-based approaches, and it does not explicitly restrict the local controllers to be passive). On the downside, the considered framework is not directly applicable to higher-order models for the agents or time-varying graphs (at least not in its current form).

In addition to simulations, the presented approach is validated with experiments on three aerial robots equipped with onboard processing for the vision-based feedback. Note that most existing works test the respective proposed algorithms only through simulations.

NOTATION AND MEASUREMENT MODEL

General Notation

As customary, $SO(d)$ and $SE(d)$ denote the space of rotations and rigid body motions in a d-dimensional space (see also side panel "The geometry of rotations"). In practical situations, either $d = 2$ or $d = 3$. The notation f' is used for the derivative of a function $f : \mathbb{R} \to \mathbb{R}$ with respect to its only argument. Given a function $\varphi : \mathbb{R}^d \to \mathbb{R}$, its gradient with respect to the variable x is denoted as $\operatorname{grad}_x \varphi$. The gradient can be defined as a vector which, given any smooth curve $x(t)$, satisfies

$$\frac{\mathrm{d}\varphi\big(x(t)\big)}{\mathrm{d}t} = \operatorname{grad}_x \varphi\big(x(t)\big)^{\mathrm{T}} \dot{x}(t), \tag{1}$$

where $\dot{x}(t)$ is the tangent to the curve. This definition is valid also for functions on non-Euclidean spaces (see side panel "The geometry of rotations"), and it reduces to the simpler one as a vector of derivatives for functions on $\mathbb{R}^d$. A *critical point* of a function φ is a point where $\operatorname{grad} \varphi = 0$ or where the gradient does not exist (because φ has a discontinuous derivative).

The graphical model

Throughout the text, the team of agents is modeled as a graph $G = (V, E)$, where the set of vertices $V = \{1, \dots, N\}$ represents the N agents and the edges $E \subseteq V \times V$ represent the pairs of agents $(i, j) \in E$ that share measurements and that can communicate with each other. For simplicity, the graph is assumed to be symmetric (that is, if $(i, j) \in E$ then also $(j, i) \in E$). As a convention, non-bold letters with subscript i or ij refer to quantities referring to a node or

an edge (for instance, x_i), and the same letters in bold without subscript refer to the collection of the same quantities across all the nodes or edges (for instance, $\boldsymbol{x} = \{x_i\}_{i \in V}$). As a consequence, in all the expressions below, one could substitute each bold letter with the corresponding set of quantities without any change in meaning.

The measurement model

It is assumed that each node i is associated to a pair $g_i = (R_i, T_i)$, where $R_i \in SO(d)$ represents the rotation from a common absolute reference frame to the local reference, and $T_i \in \mathbb{R}^d$ represents the location of the local reference frame in the absolute reference frame. In other words, given the coordinates X_a of a point in the absolute reference frame, the local coordinates of the same point are given by $X_{b,i} = R_i(X_a - T_i)$. The local reference frames at nodes i and j can be related through the relative rotations R_{ij},

$$R_{ij} = R_i R_j^{\mathrm{T}}, \tag{2}$$

the relative bearings (translation direction expressed in the reference frame of node i) t_{ij},

$$t_{ij} = \frac{R_i(T_j - T_i)}{\|T_j - T_i\|}, \tag{3}$$

and the relative distances λ_{ij}

$$\lambda_{ij} = \|T_j - T_i\|. \tag{4}$$

In general, in the estimation and control applications considered later, it is assumed that the nodes can always measure their relative rotations R_{ij} and bearings using vision and IMU sensors (see also the section LOCAL PROCESSING AND EXPERIMENTAL TESTBED). The case where some or all of the distances λ_{ij} can be measured is also considered.

From a control systems point of view, the set of poses $\boldsymbol{x} = \{(R_i, T_i)\}_{i \in V}$ can be considered the *state* of the system, while the relative quantities $\boldsymbol{z} = \{(R_{ij}, t_{ij})\}_{(i,j) \in E}$ or $\boldsymbol{z} = \{(R_{ij}, t_{ij}, \lambda_{ij})\}_{(i,j) \in E}$ correspond to the *output* of the same system. A pair of a graph and a configuration for the nodes $(G, \boldsymbol{x})$ is called a *framework*.

MUTUAL LOCALIZATION, FORMATION CONTROL, AND RIGIDITY

This section introduces more formally the mutual localization and formation control problems, and highlights their mutual relation. Then, it introduces the notion of rigidity, which characterizes when these problems are well-posed, and what ambiguities are present.

Relation Between the Estimation and Control Problems

In the mutual localization problem, the nodes are static and acquire the input measurements $\tilde{\boldsymbol{z}} = \{(\tilde{R}_{ij}, \tilde{t}_{ij})\}_{(i,j) \in E}$ or $\tilde{\boldsymbol{z}} = \{(\tilde{R}_{ij}, \tilde{t}_{ij}, \tilde{\lambda}_{ij})\}_{(i,j) \in E}$. The nodes maintain an estimate of the state $\boldsymbol{x}$, and can compute from it the corresponding set of expected measurements $\boldsymbol{z}$. The goal is then to find the estimated state $\boldsymbol{x}$ such that $\boldsymbol{z}$ matches the given measurements $\tilde{\boldsymbol{z}}$ as close as possible. In general, the designer of the system is free to choose different ways in which $\boldsymbol{x}$ can be updated to achieve the task, subject to the constraint that $\boldsymbol{x}$ must always respect the geometry of the problem (e.g., rotations must remain rotations).

In the formation control problem, the state $\boldsymbol{x}$ represents the physical state of the agents, and is generally unknown to the agents themselves. The measurements $\tilde{\boldsymbol{z}}$ correspond to the observation taken from a desired formation, and the measurements $\boldsymbol{z}$ correspond to actual

Symbol	Description
d	Ambient dimension ($d = 2$ or $d = 3$)
f'	Derivative of a univariate function
$\operatorname{grad}_x \varphi$	Gradient of the function φ with respect to x
N	Number of agents
$G = (V, E)$	Graph with vertices V representing the agents and the edges E representing pairs of agents that can exchange relative measurements
R_i	Rotation from from an absolute reference frame to the local reference frame of agent i
T_i	Location of the center of the local reference frame of agent i in the absolute reference frame
$R_{ij}, \tilde{R}_{ij}$ $t_{ij}, \tilde{t}_{ij}$ $\lambda_{ij}, \tilde{\lambda}_{ij}$	Relative rotations, bearings, and distances between nodes i and j that are either estimated (no tilde, mutual localization), desired (no tilde, formation control) or actually measured (with tilde)
$g, \boldsymbol{s}, \boldsymbol{\sigma}, \alpha$	Decomposition of the state into a common rigid transformation, shape, normalized shape, and scale (see Figure 1)
$\boldsymbol{x}$	Aggregate state for estimation/control tasks (rotations alone, translations alone, or translations and distances)
$\boldsymbol{z}, \tilde{\boldsymbol{z}}$	Aggregate measurements for estimation/control tasks (rotations alone, bearings alone, or bearings with distances)
φ	Global cost function used for estimation/control
φ_{ij}	Pairwise cost function used for estimation/control
f_R, f_T, f_C, f_D	Reshaping functions used in the pairwise cost for estimation/control

TABLE I

LIST OF SYMBOLS USED IN THE ARTICLE

observations taken from the onboard sensors. Similar to before, the goal is then to drive the states $\boldsymbol{x}$ such that $\boldsymbol{z}$ matches $\tilde{\boldsymbol{z}}$ as close as possible. However, the updates of $\boldsymbol{x}$ must satisfy the local dynamics of the agents.

As summarized in Table II, the two problems are almost identical from a modeling point of view. The practical implication of this is that one can use similar tools to study and solve the two problems, as discussed in the next few sections.

State decomposition and rigidity

In both the estimation and control problems, the agents need to rely exclusively on the information contained in the relative measurements. As such, if different states $\boldsymbol{x}$ lead to the same measurements, then it is impossible to distinguish them: this fact is captured by the notion of *rigidity*. Before giving a rigorous characterization of this concept, it is necessary to consider a decomposition of the state $\boldsymbol{x}$ into different pose, shape and scale elements. Ideas for this kind of decomposition were pioneered by Kendalll [44] and Bookstein [9], and were first used in the formation control context by [18]. While it is not used in this work, it is worth mentioning that it is possible to explicitly parametrize the result of this decomposition using Jacobi coordinates (see, for instance, [75]). The decomposition starts by dividing $\boldsymbol{x}$ into a pair $(\boldsymbol{s}, g)$, where the *shape* $\boldsymbol{s}$ represents the relative location of the agents up to some global rigid transformation $g = (R, T) \in SE(d)$, which acts in parallel on each pose g_i in $\boldsymbol{x}$. To be precise, the shape represents the equivalence class obtained by applying all the possible rigid transformations to a representative configuration $\boldsymbol{x}$ expressed in the global reference frame. The decomposition $\boldsymbol{x} = (\boldsymbol{s}, g)$ can then be seen as first selecting a class $\boldsymbol{s}$ and then picking a specific member of

	Mutual Localization	Formation Control
$\boldsymbol{x}$	Estimate of the states, maintained at each node	Physical state of the agents, unknown to the nodes
$\boldsymbol{z}$	Measurements computed from the states	Measurements from the sensors
$\tilde{\boldsymbol{z}}$	Measurements from the sensors	Measurements at the desired formation
Goal	Drive $\boldsymbol{x}$ so that $\boldsymbol{z} = \tilde{\boldsymbol{z}}$	Drive $\boldsymbol{x}$ so that $\boldsymbol{z} = \tilde{\boldsymbol{z}}$
Restrictions on updates of $\boldsymbol{x}$	None (except respecting the geometry of the problem)	Local agents' dynamics

TABLE II

RELATION BETWEEN LOCALIZATION AND FORMATION CONTROL. THE TABLE CONTAINS THE MEANING OF THE STATES $\boldsymbol{x}$ AND OF THE MEASUREMENTS $\boldsymbol{z}$, $\tilde{\boldsymbol{z}}$ IN EACH PROBLEM, TOGETHER WITH GOALS AND RESTRICTIONS. WHILE THERE ARE SIGNIFICANT DIFFERENCES, THE GOALS ARE SIMILAR.

this class with g. Furthermore, the shape $\boldsymbol{s}$ can be decomposed into a pair $(\boldsymbol{\sigma}, \alpha)$ where $\boldsymbol{\sigma}$ is the *normalized shape* and $\alpha \in \mathbb{R}$, $\alpha > 0$ is a scale. Again, $\boldsymbol{\sigma}$ represents the equivalence class of all possible scalings of a given representative shape $\boldsymbol{s}$. This decomposition is summarized in Figure 1.

With this decomposition, two configurations $\boldsymbol{x}$, $\boldsymbol{x}'$ are said to be:

- *Equivalent* if they produce the same measurements; that is, $\boldsymbol{z}(\boldsymbol{x}) = \boldsymbol{z}(\boldsymbol{x}')$.
- *Identical* if they have the same configuration, $\boldsymbol{x} = \boldsymbol{x}'$.
- *Congruent* if they have the same shape $\boldsymbol{s}$ (that is, both $\boldsymbol{\sigma}$ and α agree).
- *Similar* if they have the same normalized shape $\boldsymbol{\sigma}$.

The relation between equivalent, identical, congruent and similar states is very important and not trivial, as illustrated by the Venn diagram in Figure 2. Congruent states are always equivalent (with or without distance measurements) and similar states are always equivalent if $\boldsymbol{z}$ does not contain distances. This means that, given the measurements $\boldsymbol{z}$ alone, it is not possible to recover either the global rotation or the global translation of the agents (the g component of the decomposition). On the other hand, equivalent states are not always similar or congruent due

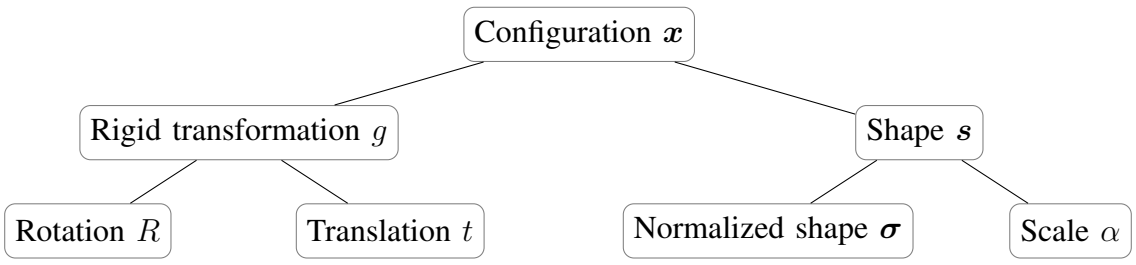


Figure 1. Decomposition of the state $\boldsymbol{x}$.

to the possible ambiguities that are intrinsic to the problem. If $\boldsymbol{z}$ does not include any distance estimate λ_{ij}, a framework is said to be *rigid* if all frameworks equivalent to it are also similar. This means that it is possible to reconstruct (more formally, *observe*) only the normalized shape $\boldsymbol{\sigma}$ (and not the scale α) from the measurements alone. If $\boldsymbol{z}$ contain at least one distance estimate λ_{ij}, a framework is said to be *rigid* if all frameworks equivalent to it are also congruent. This means that it is possible to observe the shape $\boldsymbol{s}$ (which includes the scale α).

In practice, one can check whether a formation is rigid by checking the rank of the *rigidity matrix* (see [5, 59] for a definition and details). For generic states $\boldsymbol{x}$, one can also give combinatorial conditions which depend only on the graph G (see [67, 21, 22]).

Note that the notion of rigidity considered in this context is only affected by the translations of the nodes. This is because of the assumption that the graph G is connected, and that the relative rotations $\{R_{ij}\}_{(i,j)\in E}$ are included in $\boldsymbol{z}$. Hence, once the global rotation R is fixed, all the local rotations R_i can be fixed too. If $\boldsymbol{z}$ did not include the relative rotations, then the notion of rigidity would have to be extended beyond congruency and similarity. Some work in this direction has appeared in [58, 74], but a full characterization of rigidity in the 3-dimensional case is still an open problem. Finally, it is worth mentioning that if the framework is not rigid, one can either identify a partitioning into rigid sub-frameworks and decompose the problem or add additional measurements (that is, edges) to make the problem rigid (see [71] for details).

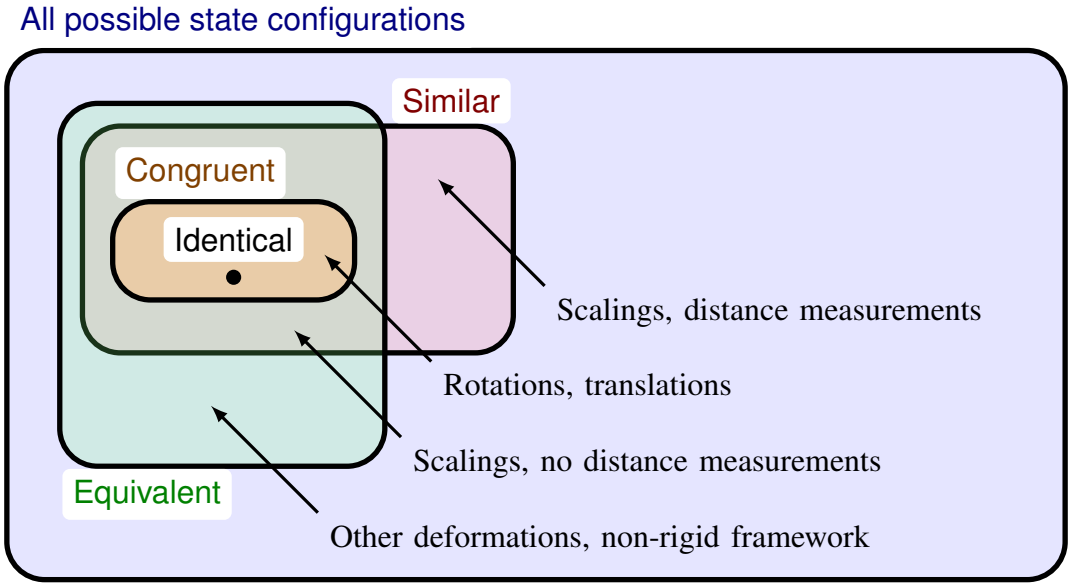


Figure 2. Venn diagram illustrating the relation between identical, congruent, similar and equivalent states. Among all possible state configurations, some can be equivalent, and some can be similar. Congruent configurations are both similar and equivalent. Identical configurations reduce to a single point. The arrows give details on what kind of configurations lie in the various set differences: configurations that are similar but not equivalent arise when the measurements contain at least one distance and the configuration is scaled; configurations that are equivalent but not similar are given in a non-rigid framework by deformations that are not rotations, translations or scaling but maintain the same measurements; configurations that are similar and equivalent but not congruent arise when the measurements contain no distances and the configuration is scaled.

A DISTRIBUTED GRADIENT-DESCENT APPROACH

This section considers an approach for formulating actual distributed algorithms for solving the localization and formation control problems. This approach is based on the minimization of a Lyapunov function defined on the graph G. The treatment starts with a general formulation based on gradient descent, leaving specializations to different problems for the next section. In order to help the reader grasp the general framework, the discussion is complemented by referring to a simple concrete problem: finding the location (that is, translations) $\{T_i\}_{i\in V}$ of the agents with respect to a common reference frame given measurements of their relative bearings (without the relative distances) $\{\tilde{t}_{ij}\}_{i\in V}$ and assuming that their rotational reference frames $\{R_i\}_{i\in V}$ are fixed and known. This example is considered again in more detail in the next section. Additional details that are beyond the scope of the present work can be found in [67].

Let $x_i \in \mathbb{R}^n$ be a state associated with node $i \in V$, and y_{ij} an optional state associated with edge $(i,j) \in E$. In the localization example, $x_i = T_i$ denotes the unknown translation of agent i, and $y_{ij} = \lambda_{ij}$ denotes the unknown distance between agents i and j (again, the rotations R_i are assumed to be fixed and known). For ease of understanding, the treatment below considers states in an Euclidean space, but the treatment can be extended to variables lying on differential manifolds, such as rotations. The notation $\boldsymbol{x} = \{x_i\}_{i\in V}$ and $\boldsymbol{y} = \{y_{ij}\}_{(i,j)\in E}$ refer to the aggregate states of the entire network.

For the sake of analysis and as in all the existing work in the area, it is assumed that measurements taken by the nodes are without noise. However, as shown in the experimental

validation, the same framework can be applied in the presence of noise and unmodeled disturbances.

Global cost, global minima, and rigidity

Consider a Lyapunov candidate φ defined as follows:

$$\varphi(\boldsymbol{x}, \boldsymbol{y}) = \sum_{(i,j)\in E} \varphi_{ij}(x_i, x_j, y_{ij}), \tag{5}$$

where the pairwise functions φ_{ij}, $(i, j) \in E$ encode the measurements and their relations with the states. The following assumptions are used:

1) The pairwise terms are non-negative, that is, $\varphi_{ij} \geq 0$.
2) It holds that $\varphi(\boldsymbol{x}_0, \boldsymbol{y}_0) = 0$ for some $(\boldsymbol{x}_0, \boldsymbol{y}_0)$ if and only if $\boldsymbol{x}_0$ is equivalent to the desired localization or formation.

Intuitively, the functions $\{\varphi_{ij}\}_{(i,j)\in E}$ represent the discrepancy between the current states and those expected or desired from the measurements. For instance, in the running localization example, one can choose

$$\varphi_{ij}(T_i, T_j, \lambda_{ij}) = \|T_j - T_i - \lambda_{ij} R_i^{\mathrm{T}} \tilde{t}_{ij}\|^2. \tag{6}$$

The two assumptions above imply that $\varphi(\boldsymbol{x}_0, \boldsymbol{y}_0) = 0$ if and only if $\varphi_{ij}(x_{0i}, x_{0j}, y_{0ij}) = 0$ for all $(i, j) \in E$. In turn, this means that the measurements are assumed to be without noise (as already stated), and that the global minimizers (that is, points where $\varphi = 0$) represent only and all the equivalent configurations that are consistent with such measurements. Hence, the cost function φ has multiple global minimizers, and the minimization procedure could find, in

general, any one of them. However, if the problem is rigid, the set of global minimizers (that is, the set of equivalent configurations, which is also the set where $\varphi = 0$) is in exact correspondence with the set of states that are either similar or congruent to the desired ones. As a consequence, given the pairwise costs for a rigid problem, solving the task at hand (in the running example, finding the locations $\{T_i\}_{i \in V}$ that agree with all the measurements $\{\tilde{t}_{ij}\}_{(i,j) \in E}$) is equivalent to driving the states to a global minimizer of the cost φ.

Regarding the specific choice of the pairwise functions $\{\varphi_{ij}\}_{(i,j) \in E}$, it is usually straightforward to come up with a basic form for each one of them, given the specific application (some notion of discrepancy or error between actual and desired measurements is sufficient). At the same time, there is some freedom in the specific form that can be used. For instance, in (6), one could have taken any monotonically increasing function of the norm, instead of just the square. This freedom can be used to obtain better convergence properties (as discussed later) or to make the approach more robust to noise and spurious measurements by using robust cost functions [76].

For ease of treatment, in the present work it is assumed that the functions φ_{ij} are twice differentiable on their domain of interest. However, similar results can be obtained when the terms φ_{ij} are only continuous, provided that the discontinuities in the derivatives correspond to local maxima [67] or are isolated points [68].

Gradient descent and distributed control law

Let $\boldsymbol{x}(0), \boldsymbol{y}(0)$ contain the initial states of the agents (in the localization example and without any prior knowledge, these could be completely random). This section considers the

problem of how to update the states in order to obtain a trajectory $\boldsymbol{x}(t), \boldsymbol{y}(t)$ which converges toward a (at least local) minimizer of φ. The easiest choice is to set the time-derivative of the states to be equal to the negative gradient of (5). This choice leads to

$$\dot{x}_i = -k \sum_{j:(i,j)\in E} \Big(\mathrm{grad}_{x_i}\, \varphi_{ij}(x_i, x_j, y_{ij}) + \mathrm{grad}_{x_i}\, \varphi_{ji}(x_j, x_i, y_{ji}) \Big), \tag{7}$$

$$\dot{y}_{ij} = -k\, \mathrm{grad}_{y_{ij}}\, \varphi_{ij}(x_i, x_j, y_{ij}), \tag{8}$$

where $k > 0$ is a scalar gain common to all the nodes. Note that the gradient in (7) contains two terms because x_i appears, for each edge $(i, j) \in E$, as an argument for both φ_{ij} and φ_{ji}. On the other hand, (8) contains only one term because y_{ij} appears only once in φ_{ij}; as a consequence y_{ij} and y_{ji} could potentially converge to different values if the terms φ_{ij} and φ_{ji} are not consistent with each other (due to noise).

The updates given by (7)–(8) are naturally distributed in the sense that, in order to update its state, node i only needs to communicate and obtain the states of its neighbors $j : (i, j) \in E$ in the graph G, and updating the state for each edge (i, j) only needs the states at the two endpoints. Moreover, one can also include local constraints by using local projections of the updates (passing from simple gradient descent to *projected* gradient descent). In the localization example, this projection can be used to enforce the constraint that all the estimated distances $\boldsymbol{y} = \{\lambda_{ij}\}_{(i,j)\in E}$ must be strictly positive (see the next section for details).

Convergence basin

In general, the gradient updates (7)–(8) drive the states to a critical point of the function φ. This section shows how a few conditions on the terms φ_{ij} are sufficient to ensure that all the

critical points of φ are also global minimizers.

The analysis is based on radial lines of the form $\tilde{x}_i(s) = x_{0i} + sv_i$ and $\tilde{y}_{ij}(s) = y_{0ij} + sv_{ij}$ (and their corresponding aggregate versions $\tilde{\boldsymbol{x}}(s)$, $\tilde{\boldsymbol{y}}(s)$), where $s \geq 0$ is the line parameter, $(\boldsymbol{x}_0, \boldsymbol{y}_0)$ represents an arbitrary global minimizer of φ and $(v_i, v_{ij}) \neq 0$ are arbitrary, non-zero directions. The pairwise and total costs evaluated along these radial lines are defined as $\tilde{\varphi}_{ij} = \varphi_{ij}\big(\tilde{x}_i, \tilde{x}_j, \tilde{y}_{ij}\big)$, $\tilde{\varphi} = \varphi\big(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}\big) = \sum_{(i,j)\in E} \tilde{\varphi}_{ij}$.

The following is the main global convergence result of the theoretical framework.

Theorem 1. *Assume that the functions $\{\varphi_{ij}\}_{(i,j)\in E}$ are differentiable everywhere, and that*

$$\frac{\mathrm{d}\tilde{\varphi}_{ij}}{\mathrm{d}s} \geq 0 \text{ for all } s \geq 0, (i,j) \in E, \tag{9}$$

with equality if and only if $(\tilde{x}_i(s), \tilde{x}_{ij}(s), \tilde{y}_{ij}(s))$ is a global minimizer of φ_{ij}. Then the trajectories defined by (7)–(8) *converge to the set of global minimizers of φ from any initial condition.*

Proof. The main idea is to use (9) to show that the only critical points of φ (that is, points where $\mathrm{grad}_{\boldsymbol{x},\boldsymbol{y}}\, \varphi = 0$) are global minimizers. If this can be shown, then standard arguments from optimization theory show that gradient descent converges to this set of global minimizers (that is, points where $\varphi = 0$).

Given arbitrary directions v_i, v_{ij}, the definition of gradient can be used to "probe" if any point along the radial lines $\tilde{\boldsymbol{x}}(s)$ and $\tilde{\boldsymbol{y}}(s)$ is a critical point:

$$\frac{\mathrm{d}\tilde{\varphi}}{\mathrm{d}s} = \mathrm{grad}_{\boldsymbol{x}}\, \varphi^{\mathrm{T}} \frac{\mathrm{d}\tilde{\boldsymbol{x}}}{\mathrm{d}s} + \mathrm{grad}_{\boldsymbol{y}}\, \varphi^{\mathrm{T}} \frac{\mathrm{d}\tilde{\boldsymbol{y}}}{\mathrm{d}s}. \tag{10}$$

Since the tangents $\frac{\mathrm{d}\tilde{\boldsymbol{x}}}{\mathrm{d}s}$, $\frac{\mathrm{d}\tilde{\boldsymbol{y}}}{\mathrm{d}s}$ are nothing but the line directions, and since, by assumption, they are

non-zero, it holds that

$$\frac{\mathrm{d}\tilde{\varphi}}{\mathrm{d}s} \neq 0 \implies \operatorname{grad}_{\boldsymbol{x},\boldsymbol{y}} \varphi \neq 0. \tag{11}$$

At the same time, the fact that the operator $\frac{\mathrm{d}}{\mathrm{d}s}$ is linear implies that

$$\frac{\mathrm{d}\tilde{\varphi}}{\mathrm{d}s} = \sum_{(i,j)\in E} \frac{\mathrm{d}\tilde{\varphi}_{ij}}{\mathrm{d}s} \geq 0, \tag{12}$$

with equality if and only if $\tilde{\varphi}$ is identically zero (that is, the radial lines traverse only global minimizers). Now, by considering all the possible line directions $(\{v_i\}_{i\in V},\ \{v_{ij}\}_{(i,j)\in E})$, it is possible to sweep the entire state space $(\boldsymbol{x}, \boldsymbol{y})$. Combining (11) with (12), it follows that a particular point $(\boldsymbol{x}_1, \boldsymbol{y}_1)$ is a critical point if and only if $\varphi(\boldsymbol{x}_1, \boldsymbol{y}_1) = 0$, that is, $(\boldsymbol{x}_1, \boldsymbol{y}_1)$ is also a global minimizer. This concludes the proof. □

Similar proofs can be found, for instance, in [70, Theorem 5] and [66, Theorem 3.2.5]. A schematic illustration of the main idea of the theorem is presented in Figure 3.

Intuitively, the significance of this result is that if the functions $\{\varphi_{ij}\}_{(i,j)\in E}$ can be chosen (using the freedom mentioned in the previous section) such that condition (9) is satisfied, then this *pairwise* condition is enough to show that, *globally*, the function φ has only global minimizers and no other critical point. The result can be alternatively visualized by noticing that (9) implies that the level sets of φ are star-shaped around any of the global minimizers. Hence, φ does not necessarily need to be convex or quasi-convex (that is, have convex level sets).

Notice that the claim in Theorem 1 holds for any topology of the graph. In fact, this result is true even when the problem is non-rigid (that is, it is not well-posed). As mentioned before, the notion of rigidity is instead necessary to show the exact correspondence between the sets of global minimizers of φ and of valid solutions for the problem (when the measurements

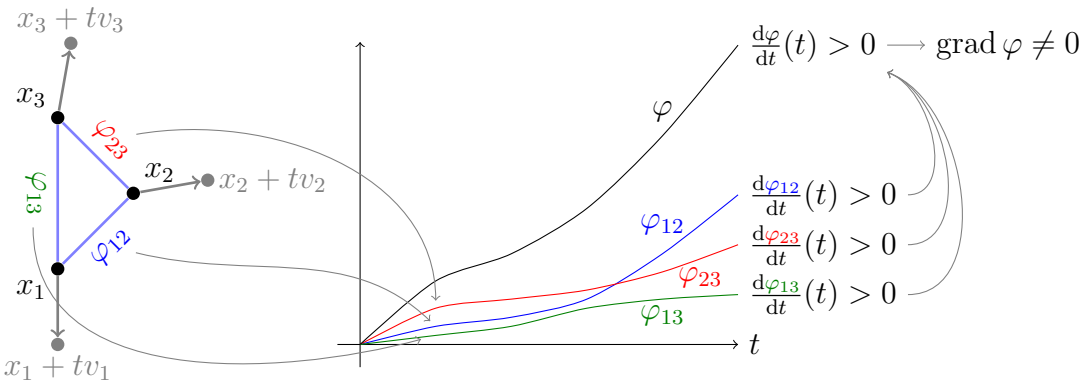


Figure 3. A schematic illustration of the proof of Theorem 1. Left: a network with three agents moving from their desired position $\{x_1, x_2, x_3\}$ towards three arbitrary directions v_1, v_2, v_3 (the variables $\boldsymbol{y}$ are not used). Right: the functions $\varphi_{12}, \varphi_{23}, \varphi_{13}$ corresponding to the three edges are monotonically increasing (for simplicity, the corresponding terms φ_{21}, φ_{32}, and φ_{31} are omitted from the picture). Hence, also its sum φ is monotonically increasing. This implies that the gradient along any point on the curve $\boldsymbol{x} + t\boldsymbol{v}$ in the state space, except the origin, cannot be zero.

are without noise).

Generally speaking, finding functions $\{\varphi_{ij}\}_{(i,j)\in E}$ satisfying the monotonicity assumption (9) is the most difficult step in applying this framework to a new problem. However, as shown in the next section, for the case of mutual localization and formation control, the resulting restrictions are mild.

It is possible to adapt the analysis above to derive local convergence results by restricting the radial lines to a subset $\mathcal{X}$ of the state space. However, in this case, one needs also to show that the trajectories do not leave the set $\mathcal{X}$.

DISTRIBUTED ALGORITHMS FOR ESTIMATION AND CONTROL

This section is devoted to show how the general framework from the previous section can be used in concrete estimation and control problems.

Rotation Estimation

Consider the problem of estimating the absolute rotations $\{R_i\}_{i\in V}$ of the nodes using the relative rotations $\{R_{ij}\}_{(i,j)\in E}$. As mentioned in the review of prior work, the knowledge of R_i at each node $i \in V$ is a prerequisite in many of the state-of-the-art formation control methods (including the one presented later).

In the general framework from the previous section, each state x_i corresponds to the rotation R_i, while the states $\boldsymbol{y}$ are not used. Based on the relation between states and outputs in (2), the pairwise terms are defined as

$$\varphi_{ij}(R_i, R_j) = f_R\big(d_{SO(3)}(R_i R_j^{\mathrm{T}}, \tilde{R}_{ij})\big), \tag{13}$$

where $d_{SO(3)}$ is the Riemannian distance in the space of rotations (see "The geometry of rotations") and $f_R : \mathbb{R} \to \mathbb{R}$ is a *reshaping function* which is monotonically increasing and quadratic near zero. The gradient can be computed by using the chain rule and the logarithm map; note that the continuous gradient updates (7) can be discretized in a practical implementation using the exponential map (see again "The geometry of rotations" and [67]).

In general, the cost (13) is not differentiable everywhere due to its non-trivial topology (the logarithm in the gradient might not be defined). Also, it is not convex (not even locally), due to the curvature of the space. However, assuming ideal measurements, one can show that

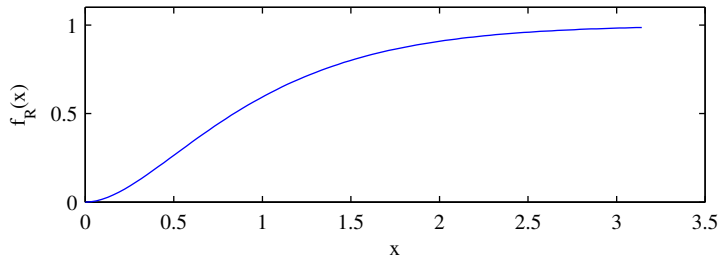


Figure 4. A plot of the function f_R with $b = 2$.

(13) satisfies the monotonicity condition (9) (for any choice of monotonically increasing f_R) when the rotations $\{R_i\}_{i\in V}$ are restricted to the set

$$\mathcal{X} = \big\{\{R_i\}_{i\in V} \in SO(d)^N : d_{SO(3)}(R_i R_j^{\mathrm{T}}, \tilde{R}_{ij}) < \frac{\pi}{2}\big\}. \tag{14}$$

Thus, one can perform the analysis detailed in the previous section and show that the set $\mathcal{X}$ contains only global minimizers. With some additional work [69], one can also show that by choosing

$$f_R(x) = 1 - (1 + bx)\exp(-bx), \tag{15}$$

with the parameter b sufficiently high, all the critical points of φ outside of $\mathcal{X}$ are either saddle points or local maxima (this particular result requires additional theoretical tools, and the interested reader is referred to [69] and [67] for details). In either case, these points are unstable for the gradient descent updates defined in (7). Hence, the estimation strategy considered here has almost global convergence guarantees, at least in the ideal case.

It is interesting to notice that the shape of the function f_R (shown in Figure 4) weights relatively less measurements that have large deviations (which are also referred to as *outliers*). Hence, this choice not only gives global convergence guarantees, but it also gives more robust estimations.

Bearing-Based Estimation

For this section, it is assumed that the rotations $\{R_i\}_{i\in V}$ are known (or have been obtained by using the previous algorithm). The goal is then to find the absolute translations $\{T_i\}_{i\in V}$ from the measured relative bearings $\{\tilde{t}_{ij}\}_{(i,j)\in E}$. It is assumed that the distances $\{\lambda_{ij}\}_{(i,j)\in E}$ are not measured, but estimated together with the translation. A detailed treatment of the material covered in this section can be found in [67] and [66], with the caveat that the convention used for expressing the translations of the nodes has been changed for ease of exposition. The resulting algorithms, however, are equivalent.

For the problem just stated, the states x_i and y_{ij} in the general formulation correspond to T_i, $i \in V$ and λ_{ij}, $(i,j) \in E$, respectively. Based on the definition of relative bearings in (3), the pairwise terms are defined as

$$\varphi_{ij}(T_i, T_j, \lambda_{ij}) = f_T\big(\|T_j - T_i - \lambda_{ij} R_i^{\mathrm{T}} \tilde{t}_{ij}\|\big), \tag{16}$$

where $f_T : \mathbb{R} \to \mathbb{R}$ is another reshaping function which is monotonically increasing and quadratic near zero. As before, one can check that, with ideal measurements, the pairwise cost (16) satisfies the monotonicity condition (9) for any choice of monotonically increasing f_T.

The fact that the distances $\{\lambda_{ij}\}_{(i,j)\in E}$ (and, in fact, the global scale α in the decomposition mentioned before) are included in the estimation introduces a problem. The trivial solution with $T_i = 0$, $i \in V$, and $\lambda_{ij} = 0$, $(i,j) \in E$ is always a global minimizer of φ. In order to avoid this, the global scale α is canonically fixed by introducing the constraints

$$\lambda_{ij} \geq 1 \text{ for all } (i,j) \in E. \tag{17}$$

Note that, with these constraints, the estimated global scale does not generally correspond to

the true one (which is unobservable). However, all the relative scales between different pairs of edges are correct. The constraints (17) can be taken into account by projecting the gradient on them: it can be shown that this is equivalent to setting $\dot{\lambda}_{ij} = 0$ whenever $\lambda_{ij} = 1$, $(i, j) \in E$.

Using the analysis from the general formulation, in the ideal case of noiseless measurements, and with the correct rotations $\{R_i\}_{i \in V}$, it is possible to show that this algorithm converges to the global minimizer from any initial condition. In other words, the algorithm has almost global asymptotic convergence [67, 66]. This is remarkable because, for a general choice of a monotonically increasing function f_T, the cost φ is non-convex.

Bearing-Based Control

As in the previous section, it is assumed that the rotations $\{R_i\}_{i \in V}$ are known. However, in this case, the goal is to drive the absolute translations $\{T_i\}_{i \in V}$ so that the measured directions $\{\tilde{t}_{ij}\}_{(i,j) \in E}$ are the same as the desired ones $\{t_{ij}\}_{(i,j) \in E}$. A complete treatment of the material covered in this section can be found in [68]. A simple first-order integrator model is assumed for the agents, so that their position can be directly controlled through $\dot{T}_i$. Contrarily to what was done in the previous section, the unknown distances $\{\lambda_{ij}\}_{(i,j) \in E}$ are not included for estimation. Hence, in the general formulation, $x_i = T_i$, while the states $\boldsymbol{y}$ are not used. The pairwise cost is defined as

$$\varphi_{ij}(T_i, T_j) = \lambda_{ij} f_C(c_{ij}), \tag{18}$$

$$c_{ij} = \cos\big(\angle(t_{ij}, \tilde{t}_{ij})\big) \tag{19}$$

where $f_C : [-1, 1] \to \mathbb{R}$ is a monotonically decreasing reshaping function (recall that $\cos\big(\angle(t_{ij}, \tilde{t}_{ij})\big) = 1$ when t_{ij} and $\tilde{t}_{ij}$ coincide). Notably, the cost (18) depends on the unknown

distances λ_{ij}. However, the gradient becomes

$$\mathrm{grad}_{T_i}\, \varphi_{ij}(T_i, T_j) = -f_C(c_{ij})\tilde{t}_{ij} - f'_C(c_{ij})(I - \tilde{t}_{ij}\tilde{t}_{ij}^{\mathrm{T}})t_{ij}, \tag{20}$$

which only depends on the available measurements (because λ_{ij} cancels out).

If the function f_C satisfies

$$f_C(c_{ij}) + (1 - c_{ij})f'_C(c_{ij}) \leq 0, \tag{21}$$

and the measurements are without noise, then (18) satisfies the monotonicity condition (9) (see [68] for a proof). Using the analysis from the general formulation, one can then deduce that, for rigid formations, the gradient-based control law given in (7) asymptotically drives the states to a configuration similar to the desired one from any initial condition (again, the scale α cannot be observed, and so it cannot be controlled). An example of a function satisfying (21) is $f(c) = 1-c$.

Bearing and Distance Estimation and Control

Assume that some of the nodes are able to measure some of the distances $\{\lambda_{ij}\}_{(i,j)\in E'}$ for some edges $E' \subset E$. Then, the cost functions from the previous two sections can be "upgraded" by adding terms taking into account these measurements. For the localization problem, one can perform the substitution

$$\varphi_{ij}(T_i, T_j, \lambda_{ij}) \leftarrow \varphi_{ij}(T_i, T_j, \lambda_{ij}) + f_D(\lambda_{ij} - \tilde{\lambda}_{ij}), \tag{22}$$

and remove the constraints on λ_{ij}. For the formation control problem, one can instead set

$$\varphi_{ij}(T_i, T_j, \lambda_{ij}) \leftarrow \varphi_{ij}(T_i, T_j, \lambda_{ij}) + f_D(\lambda_{ij}c_{ij} - \tilde{\lambda}_{ij}). \tag{23}$$

In both cases, in order to satisfy the monotonicity condition (9), $f_D : \mathbb{R} \to \mathbb{R}$ must be a differentiable reshaping function such that $\mathrm{sign}\big(f'_D(x)\big) = \mathrm{sign}(x)$ (such as $f_D(x) = x^2$).

The expressions for the gradients need to be modified accordingly. Note that the function f_D can be chosen to change the relative weights of the bearing and distance measurements, thus accomodating the different noise characteristics of the two that appear in practical settings (as mentioned in the section CHALLENGES IN MUTUAL LOCALIZATION AND FORMATION CONTROL).

Using the analysis from the general formulation, one can show that, in both cases, the monotonicity condition (9) and the global convergence properties are maintained.

Discussion

The pairwise costs for the estimation and control problems above are summarized in Table III. As the reader might have already noticed, although there are similarities between the estimation and control problems, the cost functions used are significantly different. The estimation problem relies on the estimation of the unknown distances λ_{ij}, while the formation control does not. In principle, one could use either formulation for both problems. However, one needs to consider that an incorrect initialization of the estimates of the scales might create large transitory effects on the other states, which is not desirable in a formation control setting, where the trajectories of the system correspond to real physical movements of the agents.

Another consideration is that all of the convergence results given above are valid only for ideal noiseless measurements. For the noisy case, one can still expect to have a large basin of attraction for the global minimizers, but a rigorous characterization is still an open problem.

Application	Pairwise cost φ_{ij}	Convergence (noiseless measurements)	Remarks
Rotation estimation	$f_R\big(d_{SO(3)}(R_iR_j^{\mathrm{T}}, \tilde{R}_{ij})\big)$	Local in general, almost global with (15)	States are non-Euclidean
Bearing-only localization	$f_T\big(\|T_j - T_i - \lambda_{ij}R_i^{\mathrm{T}}t_{ij}\|\big)$	Global	Constraints $\lambda_{ij} \geq 1$
Bearing-only formation	$\lambda_{ij}f_C\Big(\cos\big(\angle(t_{ij}, \tilde{t}_{ij})\big)\Big)$	Global	No scale estimates, f_C decreasing
Bearing+distance localization	$\varphi_{ij} + f_D(\lambda_{ij} - \tilde{\lambda}_{ij})$	Global	$\mathrm{sign}\big(f_D'(x)\big) = \mathrm{sign}(x)$
Bearing+distance formation	$\varphi_{ij} + f_D(\lambda_{ij}c_{ij} - \tilde{\lambda}_{ij})$	Global	$\mathrm{sign}\big(f_D'(x)\big) = \mathrm{sign}(x)$

TABLE III

SUMMARY OF THE COST FUNCTIONS FOR EACH APPLICATION. ALTHOUGH THERE ARE SIMILARITIES BETWEEN BEARING-ONLY ESTIMATION AND CONTROL PROBLEMS, THE COST FUNCTIONS USED CAN BE SIGNIFICANTLY DIFFERENT. IN PARTICULAR, IT IS POSSIBLE TO ESTIMATE ALL UNKNOWN DISTANCES (UP TO A SCALE), AS IN THE MUTUAL LOCALIZATION PROBLEM, OR COMPLETELY ELIMINATE THEM, AS IN THE FORMATION CONTROL PROBLEM. IN BOTH CASES, WHEN DISTANCES ARE AVAILABLE AS MEASUREMENTS, ONE CAN ADD TERMS TO THE COST AND OBTAIN SIMILAR DISTRIBUTED ALGORITHMS.

LOCAL PROCESSING AND EXPERIMENTAL TESTBED

This section explains the processing that is required onboard the agents to obtain the input measurements for the methods above. This section also includes a description of the experimental setup for the formation control experiments.

Additional notation

Let $P = [X_C \ Y_C \ Z_C \ 1]^{\mathrm{T}}$ and $p = [u \ v \ 1]^{\mathrm{T}}$ be the homogeneous coordinates of a 3-D point in the camera's frame and its projection on the camera's image plane, respectively. The two can be related by the pinhole camera model [50]:

$$\mu p = K \Pi P, \tag{24}$$

where μ is the depth of the point in the camera's frame, K is an invertible matrix that transforms metric coordinates into pixel coordinates (and is commonly known in the computer vision literature as the *calibration matrix*), $\Pi = [I \ 0] \in \mathbb{R}^{3\times 4}$ is the *standard projector* matrix and $I \in \mathbb{R}^{3\times 3}$ is the identity matrix. Using appropriate calibration techniques, the matrix K can be assumed to be known. Therefore, without loss of generality, it is assumed that $K = I$ in the following (if this is not the case, it is sufficient to apply K^{-1} to each image point p).

Rotation and Bearing Estimation Through Image Features

One way to compute the relative rotation R_{ij} and bearing t_{ij} from neighboring agents i and j, $(i, j) \in E$, is to use feature points extracted from the images. Specifically, it is assumed that each agent can extract from an image of the environment N_i feature points $\{p_i^{(k_i)}\}$, where

$k_i \in \{1, \ldots, N_i\}$ indicates the index of the point in the image.

It is also assumed that, for each edge $(i, j) \in E$, one can establish the correspondences between image points $p_i^{(k_i)}$ and $p_j^{(k_j)}$ in images i and j. Then, using (24) and the rigid body transformation between the two cameras, these points can be related by

$$\mu_i^{(k_i)} p_i^{(k_i)} = \mu_j^{(k_j)} R_{ij} p_j^{(k_j)} + T_{ij}. \tag{25}$$

Multiplying this relation on the left by ${p_i^{(k_i)}}^{\mathrm{T}} \hat{T}_{ij}$ (where $\hat{v}$ denotes the matrix representation of the cross product such that $\hat{v}w = v \times w$ for all $v, w \in \mathbb{R}$) leads to the *epipolar constraint*:

$${p_i^{(k_i)}}^{\mathrm{T}} \hat{T}_{ij} R_{ij} p_j^{(k_j)} = 0. \tag{26}$$

Given five or more point correspondences, equation (26) can be used to estimate the *essential matrix* $E_{ij} = \hat{T}_{ij} R_{ij}$ from which, in turn, one can extract R_{ij} and $t_{ij} = T_{ij}/\|T_{ij}\|$ [50]. It is custom to use RANSAC (see Side Panel on RANSAC) to robustly fit the essential matrix E_{ij} in the presence of wrong correspondences between the points in the two images.

Bearing Estimation Through Direct Observation

While estimating bearings using image features is possible, it is also appealing to estimate the bearings directly through observation of the other vehicles. For instance, this approach would be necessary if there are not enough features in the surroundings (as in a textureless hallway). To give the reader an idea of how this can be done in practice and to support the experiments presented later, this section describes an approach where a colored circular target is mounted on each agent, and the relative bearings are obtained by extracting and measuring this target in the images of other agents.

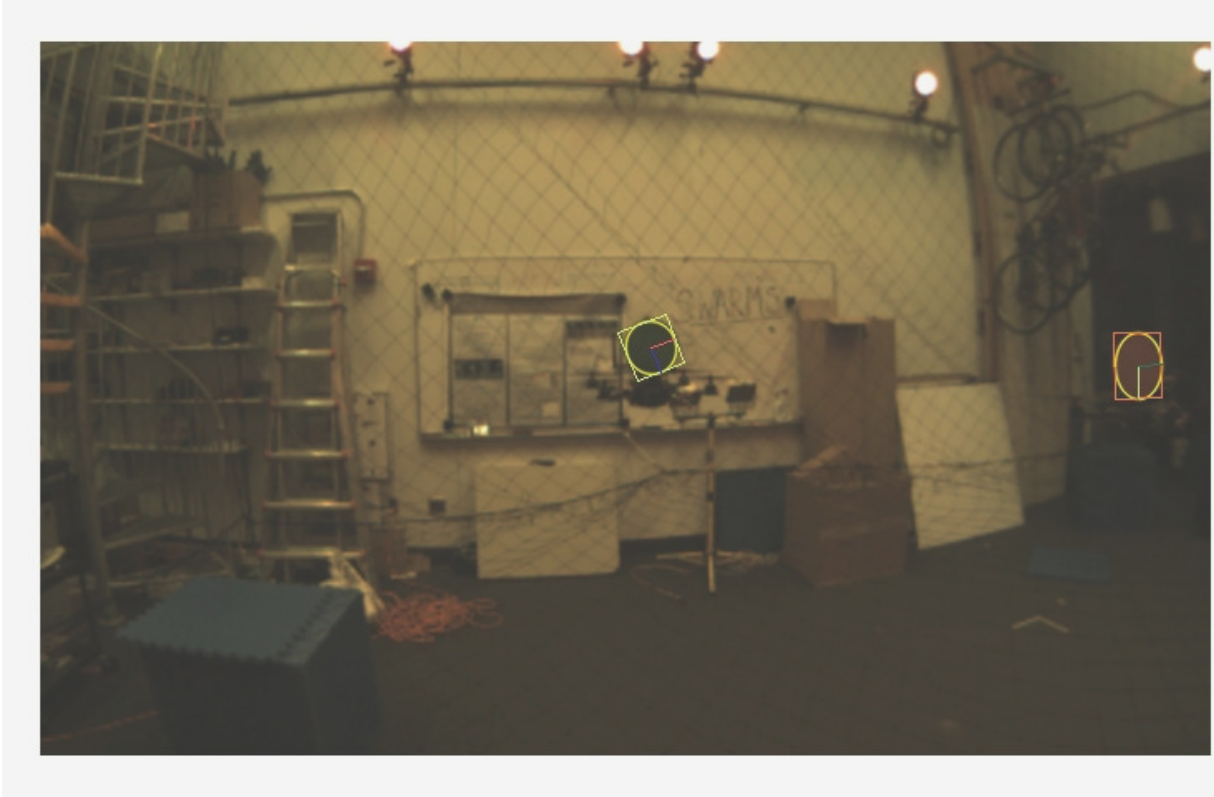


Figure 5. Ellipse detections observed from one of the vehicles.

Each vehicle is configured to visually detect the colored circular identifiers on the other two robots as shown in Fig. 5. First, a color image is thresholded for each color of interest. Then, an ellipse is fitted to the contour points in the image plane by solving a constrained minimum least squares optimization problem [24]. The bearing and scale measurements can then be obtained using the fitted model (see Side Panel: "Projective geometry of circular targets"). For outlier rejection, one can employ an adaptive 5-point RANSAC algorithm [23] since the minimum number of samples to determine the ellipse's parameters is five. A point $p_{i_{1:2}}$ is chosen to be an inlier based on its corresponding fitting residual.

The procedure is stopped when the number of inliers is above a predefined threshold. In

particular, the number of iterations N is chosen in an adaptive way and recomputed each time a new set of inliers is found according to (35) (see Side Panel on RANSAC). The number of points per sample (that is, the number of points needed to estimate the parameters of an ellipse) is $n = 5$. The requested probability of success is set as $p = 0.99$, and ϵ is recomputed after each iteration as the average between the number of inliers and the total number of points detected as ellipse contour after thresholding. To accomodate the real-time control constraints, each ellipse is detected using a separate thread [15] in order to exploit different processor cores. As shown in the next section, the Inertial Measurement Unit (IMU) on each robot is used to rotate the bearings to frames with common z-axes, which are then used in the formation control strategy discussed previously.

Rotation Estimation from Bearings and IMU

Given the bearing measurements obtained with the direct method explained above, one can use this information together with IMU measurements to establish a common rotational reference frame. This is a necessary precondition for the specific formation control approach considered in this article.

The rotation localization in this setting follows two steps. First, the gravity vector estimate from the IMU is used to reduce the task to a 2-D rotation (yaw) problem. Then, the relation between bearings at neighboring agents is used to find the relative (yaw) rotations, between the agents, from which a localization can be obtained.

IMU Attitude Compensation

Ideally, one could estimate the rotation between the local and world frames at each agent using feedback from the Inertial Measurement Unit (IMU).

However, the robot's estimate of its yaw, the rotation about $[0\ 0\ 1]^{\mathrm{T}}$ in the world frame, cannot be trusted since it depends on the orientation of the robot at initialization and is not directly observable using an IMU (without a magnetometer), rendering the estimate susceptible to the accumulation of errors. Conversely, the estimate of the other two degrees of freedom (pitch and roll) can be assumed to be reliable, thanks to observations of the gravity direction. Thus, the local bearing measurements can be transformed into a plane-leveled frame where the third axis is parallel with the gravity vector. Using this procedure, it is now only necessary to obtain the relation between the yaw angles at each node.

Rotation (Yaw) Localization

The estimates for the relative yaw between the neighboring nodes can be determined directly from the bearing measurements by using communication. This is illustrated in the sketch in Figure 6. The only requirement is that, for each pair of robots for which the relative yaw is desired, the robots must see each other and cannot be placed directly above each other. In the context of the formation control problem, it is sufficient that this condition is satisfied for all the edges in a spanning tree of the formation control graph G.

Once the relative yaws are determined, the complete pairwise rotation measurements can be combined into a full rotation localization estimate. In the setup used for real experiments,

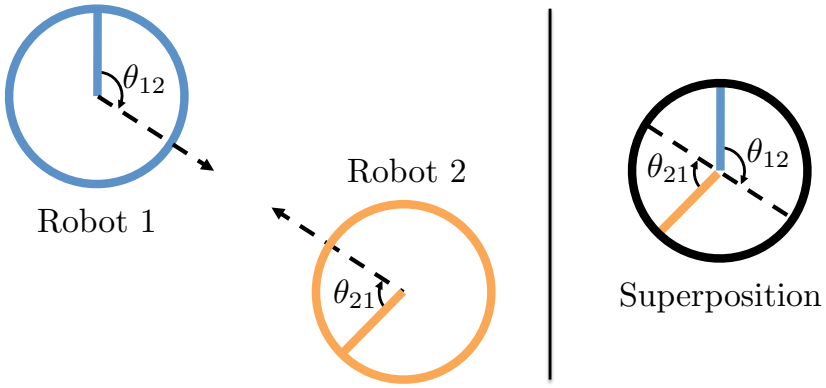


Figure 6. Two robots determine their relative yaw based on corresponding bearings. Once the bearings are mapped to a frame similar to the inertial frame, the projection onto the level plane can be used to determine the robots' relative yaw. In the case pictured here, the rotation of Robot 2 can be determined in the frame of Robot 1 to be $\psi_{21} = \theta_{12} + \pi - \theta_{21}$.

there are only three robots. Hence, one can select a leader node, canonically fix its reference frame to the identity, and then propagate the relative yaw measurements to the two neighbors. For larger formations, it is preferable to implement the full distributed rotation localization algorithm explained in the previous section.

Scale Consensus for Formation Control

From the notion of rigidity, it is known that in order to control the scale of a formation, it is necessary to obtain distance measurements. Unfortunately, the ellipse detection illustrated in the Bearing Estimation Through Direct Observation subsection might provide inaccurate estimates, thus introducing outliers. In the specific case of the experiments presented later in this work, there are three robots in a fully connected graph, which provides two bearing and two range measurements per robot. In the experiments, this implicit redundancy is used to filter out the

outliers at a central base station, which uses a consensus algorithm to determine reliable estimates for the scale of the formation.

One robot is designated as a leader node. Each robot can compute the interior angle formed by observation of the other two robots, which defines the formation's three interior angles. Then, only one side length is required to determine the scale, but the robots have six different measurements of various side lengths. Each robot can share the observed scale and bearing measurements of the other robots (since each vehicle has a unique color marker). One side length (called the "base") is chosen for comparison and the non-base lengths are used with the bearings to determine the length of the base side. Finally, a RANSAC algorithm [23] (see side panel) is used to estimate the base side length, which establishes the scale for the entire formation.

This approach is ad-hoc for the experiments below. For larger networks, it is preferable to leverage the tools presented in previous sections to exploit the redundancy in the measurements in a distributed manner.

Robots and Ground Station Configuration

The experiments in the next section use a team of three Ascending Technologies Hummingbird quadrotors [32], each equipped with an ODROID-XU computer [53] and a forward-facing color camera [33] with a 125° field of view as displayed in Figure 7. The image is processed onboard the robot, and the bearings and range are computed accordingly. A ground station is used for the yaw rotation estimation, scale consensus, and to provide an interface for the user. For real-life scenarios, the estimation could be distributed, running on each of the

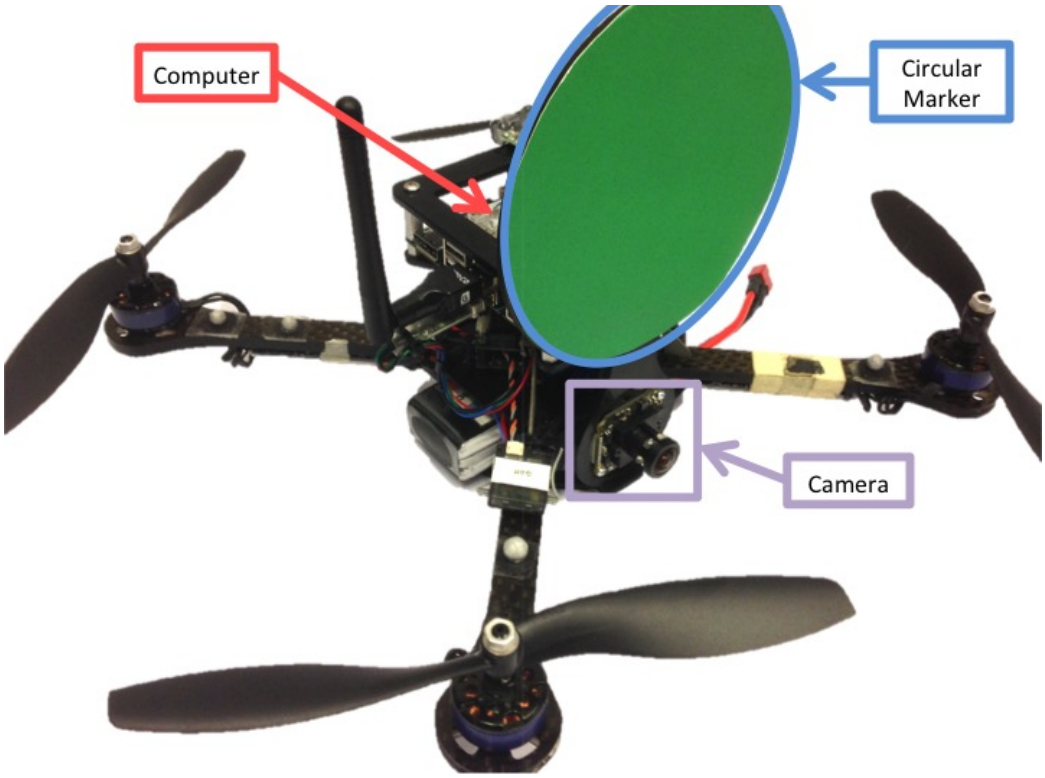


Figure 7. A quadrotor equipped for formation flight. The robot has a computer, camera, and a uniquely colored marker onboard. All vision processing occurs on the onboard computer.

robots, while the ground station would only be used to communicate high-level commands (such as a change in the desired formation) from an external user. A block diagram of the entire system is presented in Figure 8.

Simulation and Experimental Results

This section presents the simulations and experimental results for the mutual localization and formation control approaches considered in this work.

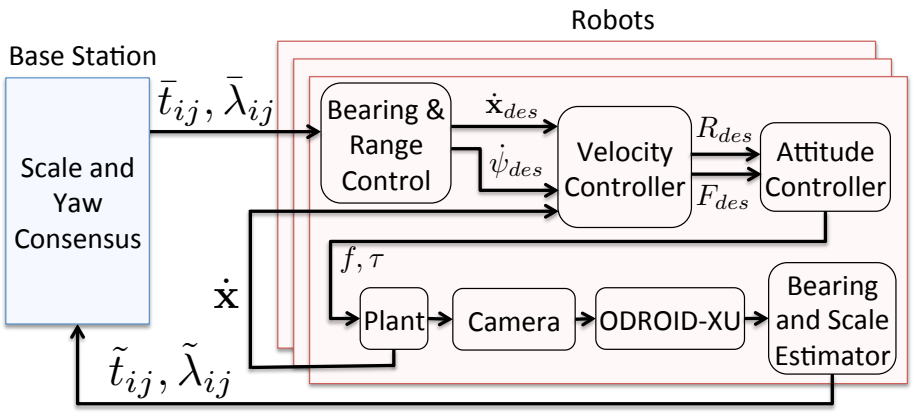


Figure 8. A block diagram of the experimental setup. Each robot computes a desired velocity from the control law, and a desired yaw rate from a simple proportional controller that keeps neighbors in the field of view. A velocity controller computes a desired force, $F_{des} \in \mathbb{R}^3$, and a desired orientation, R, which encodes the yaw. Then, the attitude controller computes a net force, $f \in \mathbb{R}$, and desired moments about the body frame primary axes, $\tau \in \mathbb{R}^3$.

Mutual Localization

The first experiment presents an application of the mutual localization algorithm on real images, using a dataset of 14 images of a building on the Johns Hopkins University campus (Figure 9). Since, in this case, the ground-truth poses are not available, the experiment uses a state-of-the-art *Structure from Motion* system (*Bundler* [62]) to estimate the pose of the camera for each picture. This system extracts features from the images (yellow crosses in Figure 9) corresponding to 3-D points in the scene that can be reliably matched among different images. Then, it finds both the position of these 3-D points and the poses of the cameras that are the most consistent with the image coordinates of the features. This approach is essentially centralized in the sense that all the data is available and updated at a central location. Moreover, if the same

3-D point is visible in more than just two images, this induces constraints on more than two poses. The same system is used on every possible *pair* of images in order to obtain estimates of the relative pose between cameras that have significantly overlapping fields of view. This provides the input to the distributed algorithm. Note that this is the only input: the algorithm does not use 3-D points, image coordinates of the features, or constraints among more than two cameras. For this experiment, the optimization is performed first over the rotations alone, then over the translations and scales with the rotations fixed, and, finally, over all the variables by running the distributed gradient descent algorithm over the sum of the two costs ((13) plus (16)).

Figure 10 visually compares the results of the described distributed algorithm with the ground-truth obtained from the centralized algorithm (see [67] for additional, quantitative results). Considering that the distributed algorithm uses significantly less information, the two localization estimates are reasonably similar. This is especially evident in the estimate of the rotations, which have only a few degrees of error with respect to the ground truth (all errors are less than eight degrees, with a median error of around three degrees). Visually, this can be seen by comparing the poses of the pyramids for each green-red pair in Figure 10.

However, there are a few inaccuracies in the estimation of the translations (see the relatively long orange lines in Figure 10). This is due to the fact that the set of estimated relative bearings included a significant number of outliers (for instance, one translation was estimated with almost 180 degrees error). These outliers could not be rejected, despite the fact that a robust cost function was used.

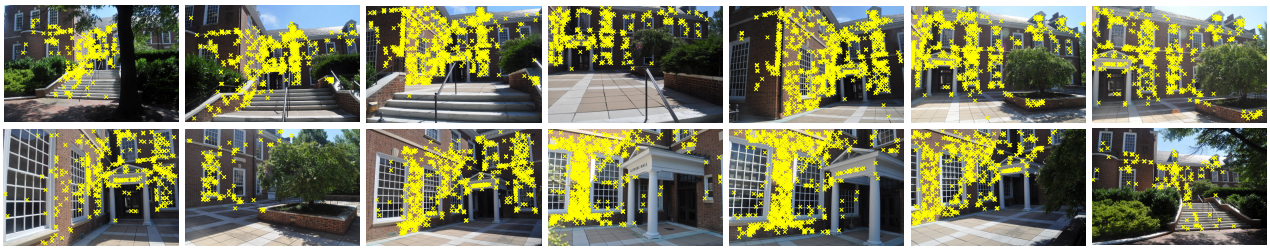


Figure 9. Images used for the localization experiments. The yellow crosses denote the location of the features used by Bundler to compute the relative poses (which are used as input to the distributed algorithm) as well as the absolute poses (which are used as ground truth).

Formation Control

This section presents simulations and experiments on formation control. The simulator is used to demonstrate the effectiveness and scalability of the control law considered in this article, while the experiments are used to demonstrate feasibility under realistic conditions with noisy measurements and a moving leader.

The results span three different simulations, each becoming increasingly more complex. The first one demonstrates a 2-D bearing-only formation, the second a 2-D bearing+distance formation, and the third a 3-D bearing+distance formation. In the bearing+distance formations, only one range measurement is used, and is represented as a thicker line in the plots.

Each agent is modeled as a single integrator that can observe either 2 or 3 neighbors in the 2-D cases, and between 3 and 6 neighbors in the 3-D case. In all examples, the agents' initial positions are assigned to be random, and the centroid is translated so that it is at the same location of the centroid of the desired formation, allowing for easier comparison.

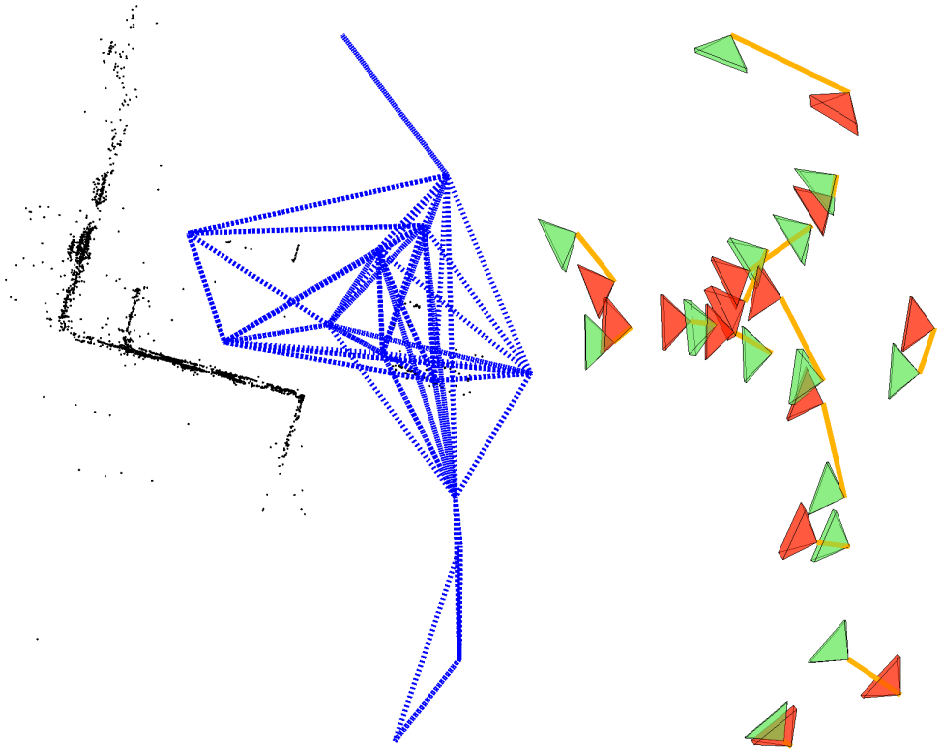


Figure 10. Localization results with the centralized and distributed algorithms. Black dots: 3-D points from *Bundler*. Green (bright) pyramids: camera poses from *Bundler*. Red (dark) pyramids: camera poses from the distributed algorithm. Blue dashed lines: vision graph (with an offset to aide visualization). Orange lines between pyramids: ground-truth to estimation correspondence.

In Figure 11a, a formation of seven agents is achieved despite a random initialization of the agents. In this case, the agents are simulated in a 2-D environment, and they use only bearing measurements. In Figure 11b, the agents are additionally provided with one range measurement for all the agents, and in Figure 11c, a 3-D case is presented where 11 agents have some bearing measurements each and one range measurement for the entire formation. In all cases, the bearing angle errors decrease, and, in the cases with a range measurement, the distance errors also decrease. This happens for both 2-D and 3-D formations, as expected from the theoretical analysis. Interestingly, the plots show that the distance errors remain bounded, and the rate of

convergence appears to be exponential; these facts have not yet been rigorously proved in the current theoretical framework, but they suggest promising future research directions.

The experimental validation of the control law is presented next. These results are based on the setup described in the LOCAL PROCESSING AND EXPERIMENTAL TESTBED section and shown in Figure 12. The experiments cover three tasks:

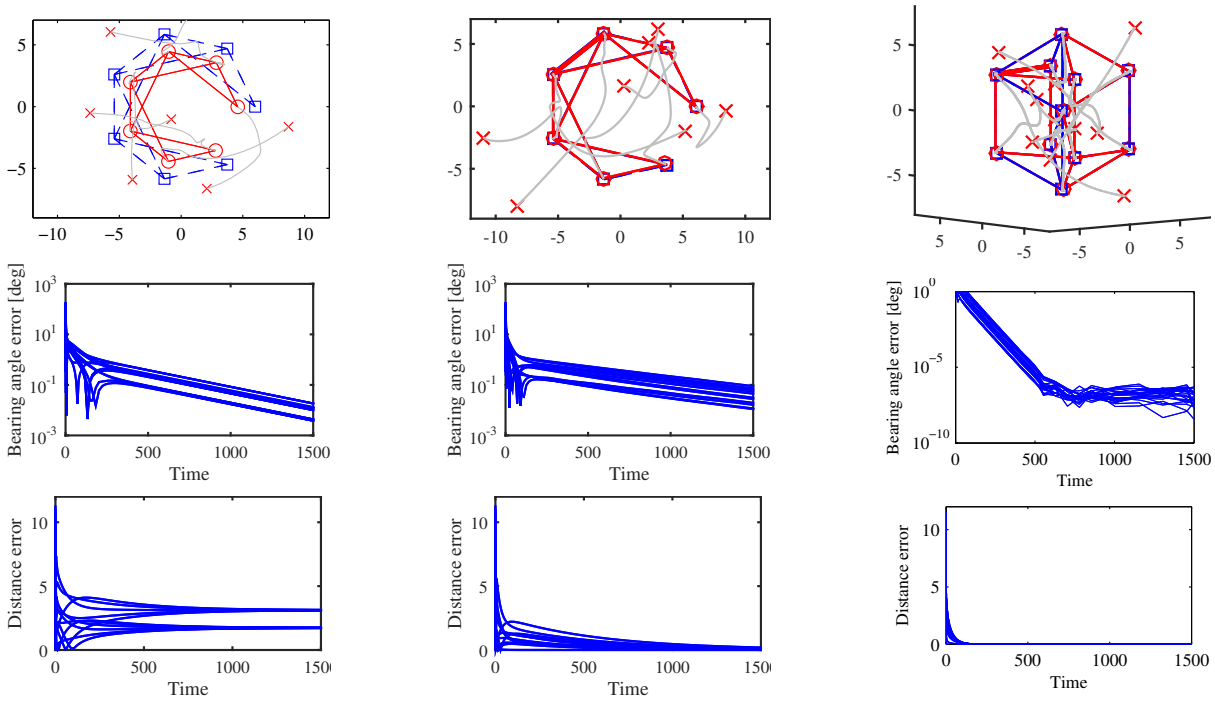
- Switch to geometrically similar formation of a different scale
- Change the shape of the formation
- Gross 3D motion of the formation

In order to specify the formation's global position and orientation, it is necessary to directly control one robot, designated as the leader. This robot is identified by the pink circular pattern in Figure 12. It is position-controlled using an external motion capture system [72] for the first two experiments and velocity controlled in the last experiment.

The other robots in the formation are controlled using vision and rely on an external motion capture system only for velocity feedback in the body frame. The vision algorithm (executed at 15 Hz), the velocity controller, and the position controller run onboard each of the vehicles (see the block diagram in Figure 8).

In the first experiment, the robots initially form an equilateral triangle parallel to the ground. During the trial, the scale of the desired formation is changed to have side lengths 0.2 meters greater than the initial configuration (see the left column of Figure 13).

The second experiment uses the same initial configuration as the first one, but it consists in changing the formation to an isosceles triangle with the leader at the connection of the equal-



(a) Pure bearing formation, 2-D

(b) Bearing+distance formation, 2-D (thick lines indicate the edge with the single distance measurement)

(c) Bearing+distance formation, 3-D (thick lines indicate the edge with the single distance measurement)

Figure 11. Simulation results with a leaderless formation and random initialization. Top: Cartesian view of the simulation (blue dashed lines and squares: desired formation; red crosses: initial configuration; red lines and circles: final formation; gray lines: agent trajectories). Middle: angular error between measured and desired bearings (log scale). Bottom: absolute difference between actual and desired distances (these correspond to all the edges in E, and thus are not all used in the control law).

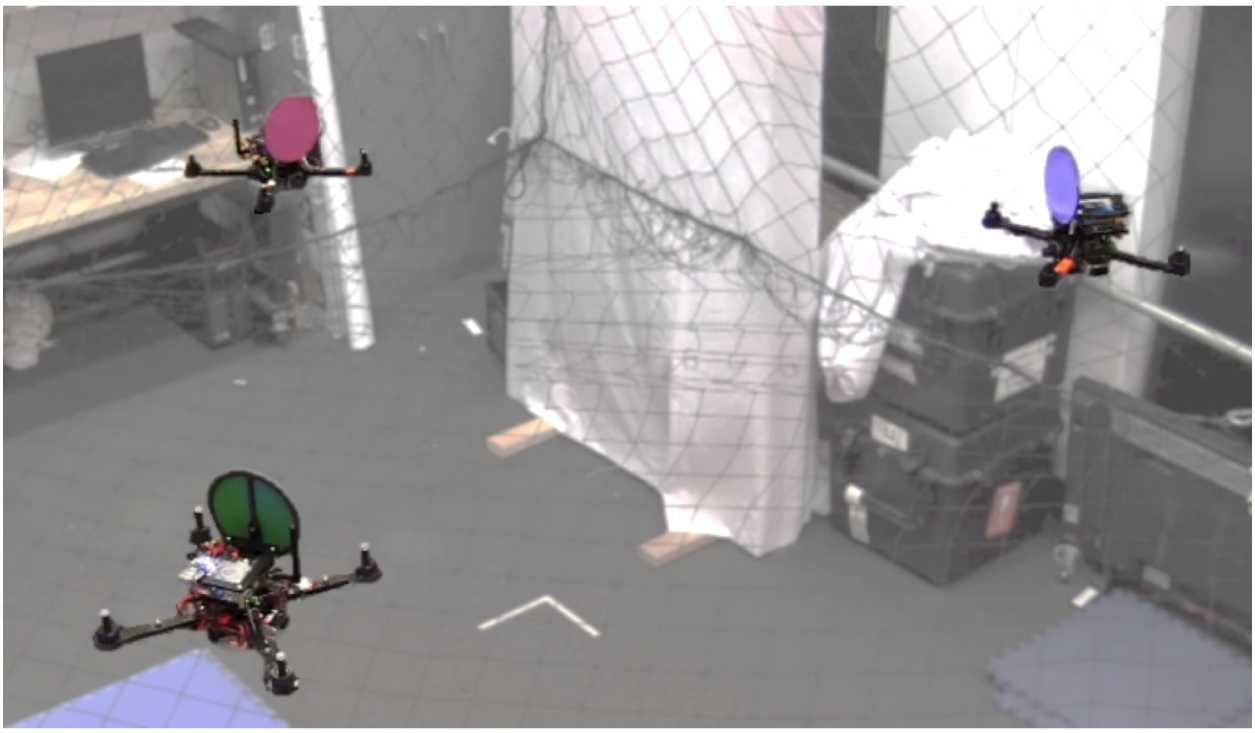


Figure 12. A photograph of the formation of three quadrotors during one of the experiments. A lead robot (top left) is velocity controlled and the other robots maintain the formation as the lead robot moves in the workspace. In a field scenario, the lead robot could run state estimation for the entire formation, which would allow the other robots to free up some payload and computation in order to carry and use other sensors.

length sides (see center column of Figure 13).

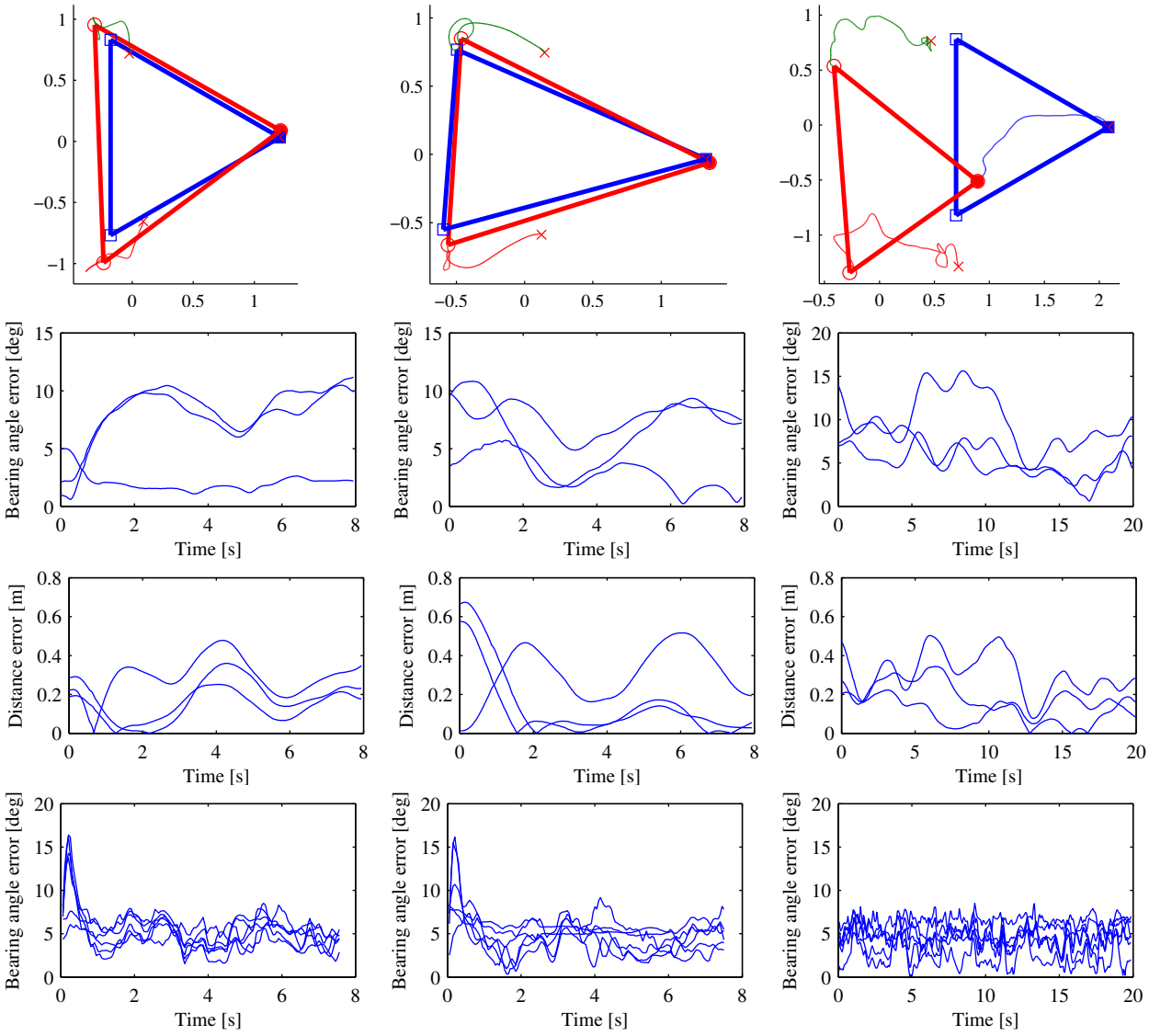
The final experiment also begins with an equilateral triangle, and this formation is maintained throughout the duration of the trial. In this case, the leader is velocity controlled to move in both the horizontal and vertical directions, and the other two robots maintain the formation (see right column of Figure 13).

In all three experiments, the robots are able to achieve and maintain the 3-D formation.

In the first and second experiments, the formation recovers from step inputs, and in the third, the formation follows the leader's motion. In the last row of Figure 13, one can see that the bearing errors that can actually be observed by the robots have a quickly decreasing trend, which then plateaus to around five degrees. This shows that the onboard controller is effective in minimizing these errors, as designed. The five-degrees plateau can be likely attributed to two causes. The first one is noise, whose origin is attributable to two sources: the ellipse fitting procedure in the measurement of the bearings, and the aerodynamic disturbances in the actuation of the control law. The other likely cause for the plateau is due to a miscalibration of one of the onboard cameras (compensating the radial distortion for wide-angle lenses can be a delicate process in practice). This miscalibration produces a non-zero offset between the measured and actual bearing information expressed in metric coordinates. Since the formation was specified in metric coordinates (as opposed to image coordinates), this offset makes the specified formation inconsistent (that is, physically unrealizable), and the controller, intuitively, tries to compensate for this inconsistency. This explanation is also consistent with the plots of the second and third rows of Figure 13, where the errors for some edges are smaller than the others.

Overall, the steady state errors are reasonable, given that they are in the range of 4% of the robots' field of view (5° error over 125° field of view).

These experiments demonstrate the feasibility of the framework for realistic conditions by leveraging onboard sensors, processing, and wireless communication, and its robustness with respect to non-ideal conditions (such as inconsistencies in the formation given to the controller).



(a) Scale change (b) Shape change (c) Motion

Figure 13. Results for the three experiments in the testbed with three quadrotors. (a) experiment where the desired scale of an equilateral triangle was increased; (b) change of shape from an equilateral to an isosceles triangle; (c) group motion initiated by the leader robot. Desired bearing and distances were specified for all the three edges. First row: Projection of the motion onto the xy plane (solid circle, that is, the right-most agent: leader of the formation; thick blue lines and circles: desired formation; red crosses: initial position of the agents; thick red lines and circles: final formation; thin colored lines: agent trajectories). Coordinates are in meters. Second row: angular error between realized and desired bearings. Third row: absolute difference between realized and desired distances. Last row: angular error between measured and realized bearings.

Conclusions and Future Directions

This article presented the vision-based mutual localization and formation control problems for a team of robotic vehicles. The treatment included a summary of the state-of-the-art, a general approach for finding distributed solutions to such problems, notes on practical implementation aspects, and a validation of the approach with simulations and experiments.

Going forward, there are many open problems and research opportunities in bearing-based distributed localization and formation control. This is because these subjects have received relatively less attention with respect to other multi-agent problems considered in the community, such as consensus. For instance, in almost all existing works, the problems are assumed to be time invariant (the topology of the graph, the measurements for the localization problem, and the desired configuration for the formation problem are all fixed). It would be interesting to see if it would be possible to adapt existing results from, for instance, the consensus literature, to this problem. In a similar vein, it would be interesting to investigate whether the ideas from [13] could be used to show exponential convergence for the approach described in this article. Exponential convergence could then be used to show other properties, such as input-to-state stability (ISS).

Other interesting extensions would involve:

- Taking into account field of view constraints in the formation control problem. This is not a trivial matter, as, at the moment, it is not clear if it is even possible to obtain global convergence results or if there are topological obstructions.
- A collision avoidance mechanism (both among the agents and with external objects). A few

existing works address the issue (for instance, [31]), but they all require full relative pose (that is, distance) measurements.

- Unknown/uncertain agent identities, where the identity of the neighbors seen by each agent is not exactly known. This would involve a probabilistic approach for assigning identities, together with the modification of the estimation or control strategy to take into account this uncertainty. This problem has already been solved in a centralized setting by Cognetti et al. [14]; however, a distributed solution and its application to formation control are still open problems.
- Other detection techniques and design for the agents in practical implementations. For instance, instead of relying on color thresholding, one could attempt to directly detect and track the robot in the camera images by using more advanced computer vision techniques. One could also design body enclosures for the quadrotors (for example, a sphere) in order to facilitate detection.
- Higher-order models for the agents with bearing-only measurements in the formation control problem, without requiring the knowledge or the estimation of the relative distances or velocities.

Overall, the authors believe that there is still a large gap between the existing (mostly theoretical) and ideal solutions that are robust enough to be quickly deployable in real applications. The authors hope that this tutorial will inspire existing and new researchers to fill in this gap.

Side Panel: "The geometry of rotations and rigid body motions"

This side panel contains a concise presentation of the geometry of rotations and rigid body motions. The space of d-dimensional rotations is defined as $SO(d) = \{R \in \mathbb{R}^{d\times d} : R^{\mathrm{T}}R = I, \det(R) = 1\}$, while the space of d-dimensional poses is defined as $SE(d) = \{(R, T) : R \in SO(d), T \in \mathbb{R}^d\}$. Given any trajectory $R(t) \in SO(d)$, the tangent $\dot{R}(t)$ lives in the *tangent space* of $SO(d)$ at R, defined as $T_R SO(d) = \{RV : V \in \mathfrak{so}(d)\}$, where $\mathfrak{so}(d)$ is the space of $d \times d$ skew symmetric matrices (see Figure 14). The tangent space is a vector space, so the usual operations of addition and multiplication by scalar between tangent vectors (such as tangents to curves) are well defined. It can also be endowed with an inner product (the *Riemannian metric*) and the corresponding norm, which is given by

$$\|RV\| = \frac{1}{2}\operatorname{tr} V^{\mathrm{T}}V. \tag{27}$$

The gradient of a function on $SO(d)$ is also a tangent vector defined as by (1) (with the Riemannian metric used instead of the standard inner product).

Using Riemannian geometry, one can define the notion of *geodesics* in $SO(d)$, which are the generalization of straight lines to non-Euclidean spaces. Using this notion, one can define a distance $d_{SO(d)}(R_1, R_2)$ which is the minimum angle or the rotation mapping R_1 to R_2. Computationally, this is given by

$$d_{SO(d)}(R_1, R_2) = \arccos\left(\frac{\operatorname{tr}(R_1^{\mathrm{T}}R_2) - 1}{2}\right), \tag{28}$$

Geodesics can also be used to define the *exponential map*, which maps a tangent vector $R_1V \in T_{R_1}SO(d)$ to a rotation R_2 obtained by following the geodesic with tangent R_1V for a length

$\theta = \|R_1 V\| = d_{SO(3)}(R_1, R_2)$. In practice, for $SO(3)$, this map can be implemented using Rodrigues' formula:

$$R_2 = R_1 \left(I + \frac{\sin(\theta)}{\theta} V + \frac{1 - \cos(\theta)}{\theta^2} V^2 \right). \tag{29}$$

The *logarithm map* $\log_{R_1} R_2$ is the inverse of the *exponential map*, and it transforms a rotation R_2 to the vector in the tangent space at R_1 which is tangent to the shortest geodesic connecting R_1 to R_2 and with length equal to $d_{SO(3)}(R_1, R_2)$. In $SO(3)$, this can be computed by

$$\log_{R_1}(R_2) = \frac{\theta}{2\sin(\theta)} R_1 (R - R^{\mathrm{T}}), \tag{30}$$

where again $\theta = d_{SO(3)}(R_1, R_2)$ and $R = R_1^{\mathrm{T}} R_2$. Note that this map is not well defined when $\theta = \pi$.

From the definition of the logarithm map, one can show that the gradient of the distance is given by the negative of the normalized logarithm, that is

$$\mathrm{grad}_{R_1} d_{SO(3)}(R_1, R_2) = -\frac{\log_{R_1}(R_2)}{\|\log_{R_1}(R_2)\|}. \tag{31}$$

Side Panel: "Projective geometry of circular targets"

A circle can be seen as an intersection of a sphere with a plane. In particular, a point p in normalized coordinates belonging to the ellipse that appears on the image plane must satisfy

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}^T \begin{bmatrix} \alpha_1 & \alpha_3 & \alpha_4 \\ \alpha_3 & \alpha_2 & \alpha_5 \\ \alpha_4 & \alpha_5 & \alpha_6 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{p}^T \mathbf{Q} \mathbf{p} \tag{32}$$

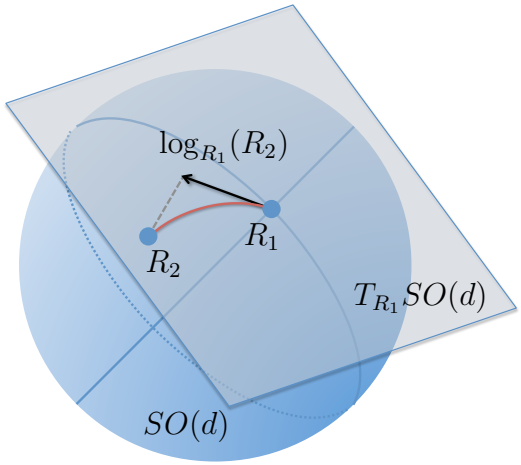


Figure 14. A pictorial representation of the space of rotations and related concepts. The tangent to a curve in the non-Euclidean space $SO(d)$ lies in the tangent space $T_R SO(d)$. Two rotations R_1, R_2 can be connected with a minimum length geodesic, which can then be used to map R_2 to a tangent vector in $T_{R_1}SO(d)$ using the logarithm map $\log_{R_1}(R_1)$.

where $\mathbf{Q}$ is called a *conic* [43] and

$$\begin{aligned}
\alpha_1 =& a_u^2/\|a\|^2 + a_v^2/\|b\|^2, \\
\alpha_2 =& a_u^2/\|b\|^2 + a_v^2/\|a\|^2, \\
\alpha_3 =& a_u a_v/\|a\|^2 - a_u a_v/\|b\|^2, \\
\alpha_4 =& -c_u\alpha_1 - c_v\alpha_3, \\
\alpha_5 =& -c_u\alpha_3 - c_v\alpha_2, \\
\alpha_6 =& \alpha_1 c_u + \alpha_2 c_v + 2\alpha_2 c_u c_v - 1
\end{aligned}$$

are the *conic parameters* expressed in terms of the ellipse center $c = (c_u, c_v)$ as well as the major $a = (a_u, a_v)$ and minor $b = (b_u, b_v)$ axes of the ellipse. The conic $\mathbf{Q}$ for a non-degenerate ellipse has always two eigenvalues that are positive while the remaining one is negative [43]. Without

loss of generality, let the $\mathbf{Q}$ matrix eigenvalues and eigenvectors be $\lambda_0, \lambda_1, \lambda_2$ and $\mathbf{q_0}, \mathbf{q_1}, \mathbf{q_2}$, respectively with $\lambda_2 \leq 0 \leq \lambda_0 \leq \lambda_1$. The normal vector $\mathbf{n}$ to the circular target can be calculated as [43]

$$\mathbf{n} = \sqrt{\frac{\lambda_1 - \lambda_0}{\lambda_1 - \lambda_2}}\mathbf{q_0} + \sqrt{\frac{\lambda_0 - \lambda_2}{\lambda_1 - \lambda_2}}\mathbf{q_2}. \tag{33}$$

The computation of the vector $\mathbf{n}$ can be ambiguous if $\lambda_1 \neq \lambda_2$. This issue can be solved considering that $\mathbf{n}$ must point towards the camera. Once it is computed, the position of the circle $\mathbf{t}$ with respect to the camera can be obtained as

$$\mathbf{t} = \pm\frac{r_0}{\sqrt{-\lambda_1\lambda_2}}\left(\lambda_2\sqrt{\frac{\lambda_1 - \lambda_0}{\lambda_1 - \lambda_2}}\mathbf{q_0} + \lambda_1\sqrt{\frac{\lambda_0 - \lambda_2}{\lambda_1 - \lambda_2}}\mathbf{q_2}\right) \tag{34}$$

where r_0 is the radius of the circular pattern. The sign of the translation can be determined from the fact that the circle must appear in the field of view of the camera.

Side Panel on RANSAC

RANdom SAMple Consensus (RANSAC) is an iterative method, introduced in [23] to estimate parameters of a mathematical model from a set of data which may contain outliers. It is essentially composed of two steps that are iteratively repeated:

- A sample subset containing a minimal number of datapoints (that is, the minimum number of points necessary to determine the model parameters) is randomly selected from the input dataset. The model is then estimated from this sample subset.
- The algorithm checks which elements of the entire dataset are consistent with the model obtained in the previous step. A data element is considered an outlier if it does not fit the model within some error threshold. This threshold defines the maximum deviation

attributable to the effect of noise.

The set of inliers obtained for the fitting model is called a consensus set. The RANSAC algorithm repeats the above two steps until the desired number of iterations is reached. The consensus set with highest number of inliers is then used to produce the final estimate of the model. RANSAC is a non-deterministic algorithm producing reasonable results only with a certain probability. This probability increases with the number of iterations used in the algorithm. In particular, the number of necessary iterations N is given by:

$$N = \frac{\log(1-p)}{\log\left(1-(1-\epsilon)^n\right)} \tag{35}$$

where p represents the probability of success, ϵ the percentage of outliers and n is the number of samples used to estimate the model.

The advantages of RANSAC are its ability to robustly estimate the model parameters in the presence of outliers, and in its applicability to a wide variety of problems. A disadvantage of RANSAC is that the number of iterations necessary to obtain reliable results could be very high. If the number of iterations is limited, the solution obtained may be suboptimal or not even fit the data in an acceptable way. This leads to a trade-off between the number of iterations of the algorithm and the probability that the model obtained is sufficiently close to reality.

References

[1] M. Arie-Nachimson, S. Kovalsky, I. Kemelmacher-Shlizerman, A. Singer, and R. Basri, "Global motion estimation from point matches," in *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, 2012, pp. 81–88.

[2] J. Aspnes, W. Whiteley, and Y. R. Yang, "A theory of network localization," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, 2006.

[3] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Automatic Control*, vol. 14, no. 6, pp. 926–939, 1998.

[4] A. Barton-Sweeney, D. Lymberopoulos, and A. Savvides, "Sensor localization and camera calibration in distributed camera sensor networks," in *International Conference on Broadband Communications, Networks and Systems*, 2006, pp. 1–10.

[5] A. N. Bishop, I. Shames, and B. Anderson, "Stabilization of rigid formations with direction-only constraints," in *IEEE International Conference on Decision and Control*, 2011, pp. 746–752.

[6] A. N. Bishop, T. H. Summers, and B. D. O. Anderson, "Control of triangle formations with a mix of angle and distance constraints," *IEEE International Conference on Control Applications*, pp. 825–830, 2012.

[7] ——, "Stabilization of stiff formations with a mix of direction and distance constraints," in *IEEE International Conference on Control Applications*, 2013, pp. 1194–1199.

[8] A. N. Bishop, M. Deghat, B. D. O. Anderson, and U. Hong, "Distributed formation control with relaxed motion requirements," *International Journal of Robust and Nonlinear Control*, 2014.

[9] F. L. Bookstein, *Morphometric tools for landmark data: geometry and biology*. Cambridge University Press, 1997.

[10] L. Carlone and A. Censi, "From angular manifolds to the integer lattice: Guaranteed orientation estimation with application to pose graph optimization," *IEEE Transactions on Robotics and Automation*, 2014.

[11] Q. Chen and J. Y. S. Luh, "Coordination and control of a group of small mobile robots," in *IEEE International Conference on Robotics and Automation*, 1994, pp. 2315–2320.

[12] N. Chopra and M. W. Spong, "Passivity-based control of multi-agent systems," in *Advances in robot control*. Springer, 2006, pp. 107–134.

[13] S.-J. Chung, U. Ahsun, and J.-J. E. Slotine, "Application of synchronization to formation flying spacecraft: Lagrangian approach," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 2, pp. 512–526, 2009.

[14] M. Cognetti, P. Stegagno, A. Franchi, G. Oriolo, and H. H. Bülthoff, "3D mutual localization with anonymous bearing measurements," in *IEEE International Conference on Robotics and Automation*, 2012.

[15] B. Community, "Boost Library," http://www.boost.org.

[16] D. Crandall, A. Owens, N. Snavely, and D. P. Huttenlocher, "Discrete-continuous optimization for large-scale structure from motion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 3001–3008.

[17] J. P. Desai, J. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," in *IEEE International Conference on Robotics and Automation*, vol. 4, 1998, pp. 2864–2869.

[18] ——, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908, 2001.

[19] D. Devarajan and R. Radke, "Calibrating distributed camera networks using belief propagation," *EURASIP Journal of Applied Signal Processing*, pp. 221–221, 2007.

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

[21] T. Eren, "Formation shape control based on bearing rigidity," *International Journal of Control*, vol. 85, no. 9, pp. 1361–1379, 2012.

[22] T. Eren, W. Whiteley, A. S. Morse, P. N. Belhumeur, and B. D. O. Anderson, "Sensor and network topologies of formations with direction, bearing and angle information between agents," in *IEEE International Conference on Decision and Control*, 2003, pp. 3064–3069.

[23] M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[24] A. Fitzgibbon and R. B. Fisher, "A Buyer's Guide to Conic Fitting," in *British Machine Vision Conference*, 1995, pp. 513–522.

[25] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," in *Robotics: Science and Systems*, 2015.

[26] A. Franchi and P. R. Giordano, "Decentralized control of parallel rigid formations with direction constraints and bearing measurements," in *IEEE International Conference on Decision and Control*, 2012, pp. 5310–5317.

[27] A. Franchi, C. Masone, V. Grabe, M. Ryll, H. H. Bülthoff, and P. R. Giordano, "Modeling and Control of UAV Bearing Formations with Bilateral High-level Steering," *The International Journal of Robotics Research*, vol. 31, no. 12, pp. 1504–1525, 2012.

[28] J. Fredriksson and C. Olsson, "Simultaneous multiple rotation averaging using lagrangian duality," in *Asian Conference on Computer Vision*, 2012.

[29] U. Frese, P. Larsson, and T. Duckett, "A multilevel relaxation algorithm for simultaneous localization and mapping," *IEEE Transactions on Robotics and Automation*, vol. 21, no. 2,

pp. 196–207, 2005.

[30] S. Funiak, C. Guestrin, M. Paskin, and R. Sukthankar, "Distributed localization of networked cameras," in *International Conference on Information Processing in Sensor Networks*, 2006, pp. 34–42.

[31] P. R. Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *The International Journal of Robotics Research*, vol. 32, pp. 299–323, 2013.

[32] A. T. GmbH, "Ascending Technologies GmbH," http://www.asctec.de.

[33] M. V. GmbH, "Matrix Vision GmbH," http://http://www.matrix-vision.com/.

[34] V. M. Govindu, "Combining two-view constraints for motion estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2001, pp. 218–225.

[35] ——, "Lie-algebraic averaging for globally consistent motion estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2004, pp. 684–691.

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

[37] G. Grisetti, R. Kummerle, C. Stachniss, U. Frese, and C. Hertzberg, "Hierarchical optimization on manifolds for online 2D and 3D mapping," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 273–278.

[38] J.-S. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 1999, pp. 318–325.

[39] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, 2004.

[40] R. Hartley, K. Aftab, and J. Trumpf, "L1 rotation averaging using the Weiszfeld algorithm," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[41] I.-A. F. Ihle, M. Arcak, and T. I. Fossen, "Passivity-based designs for synchronized path-following," *Automatica*, vol. 43, no. 9, pp. 1508–1518, 2007.

[42] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 3281–3288.

[43] K. Kanatani and W. Liu, "3D interpretation of conics and orthogonality," *CVGIP: Image Understanding*, vol. 58, no. 3, pp. 286–301, 1993.

[44] D. G. Kendall, "Shape manifolds, procrustean metrics, and complex projective spaces," *Bulletin of the London Mathematical Society*, vol. 16, no. 2, pp. 81–121, 1984.

[45] U. Khan, S. Kar, and J. Moura, "Distributed sensor localization in random environments using minimal number of anchor nodes," *IEEE Transactions on Signal Processing*, vol. 57, no. 5, pp. 2000–2016, May 2009.

[46] K. Konolige, "Large-scale map-making," in *AAAI National Conference on Artificial Intelligence*, 2004, pp. 457–463.

[47] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *IEEE International Conference on Robotics and Automation*, 2011.

[48] G. Loianno, G. Cross, C. Qu, Y. Mulgaonkar, and V. Kumar, "Flying smartphones: Automated flight enabled by consumer electronics," *IEEE Robotics and Automation Magazine*, vol. 22, no. 2, pp. 24–32, 2015.

[49] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous robots*, vol. 4, no. 4, pp. 333–349, 1997.

[50] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, *An Invitation to 3D Vision: From Images to Geometric Models*. Springer Verlag, 2003.

[51] D. Martinec and T. Pajdla, "Robust rotation and translation estimation in multiview reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[52] E. Montijano, D. Zhou, M. Schwager, and C. Sagues, "Distributed formation control without a global reference frame," in *IEEE American Control Conference*, 2014, pp. 3862–3867.

[53] ODROID, "Hardkernel co., Ltd." http://www.hardkernel.com.

[54] K.-K. Oh and H.-S. Ahn, "Formation control and network localization via orientation alignment," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 540–545, 2014.

[55] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[56] R. Olfati-Saber, "Distributed Kalman filtering and sensor fusion in sensor networks," *Networked Embedded Sensing and Control*, pp. 157–167, 2006.

[57] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," *International Journal of Robotics Research*, vol. 32, no. 7, pp. 826–840, 2013.

[58] G. Piovan, I. Shames, B. Fidan, F. Bullo, and B. Anderson, "On frame and orientation localization for relative sensing networks," *Automatica*, vol. 49, no. 1, pp. 206–213, 2013.

[59] B. Servatius and W. Whiteley, "Constraining plane configurations in computer-aided design: Combinatorics of directions and lengths," *SIAM Journal on Discrete Mathematics*, vol. 12, no. 1, pp. 136–153, 1999.

[60] I. Shames, A. Bishop, and B. D. O. Anderson, "Analysis of noisy bearing-only network localization," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 247–252, 2013.

[61] B. Shirmohammadi and C. Taylor, "Self localizing smart camera networks," *ACM Transactions on Sensor Networks*, 2010.

[62] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3D," in *ACM SIGGRAPH*, 2006, pp. 835–846.

[63] G. Stacey and R. Mahony, "A port-Hamiltonian approach to formation control using bearing measurements and range observers," in *IEEE International Conference on Decision and Control*, 2013.

[64] G. Stacey, R. Mahony, and P. Corke, "A bondgraph approach to formation control using relative state measurements," in *IEEE European Control Conference*, 2013, pp. 1262–1267.

[65] P. Tabuada, G. J. Pappas, and P. Lima, "Feasible formations of multi-agent systems," in *IEEE American Control Conference*, vol. 1. IEEE, 2001, pp. 56–61.

[66] R. Tron, "Distributed optimization on manifolds for consensus algorithms and camera network localization," Ph.D. dissertation, The Johns Hopkins University, 2012.

[67] R. Tron and R. Vidal, "Distributed 3-D localization of camera sensor networks from 2-D image measurements," *IEEE Transactions on Automatic Control*, 2014.

[68] R. Tron, J. Thomas, G. Loianno, J. Polin, K. Daniilidis, and V. Kumar, "Vision-based formation control of aerial vehicles," *IEEE Transactions on Robotics and Automation*, (submitted).

[69] R. Tron, B. Afsari, and R. Vidal, "Intrinsic consensus on $SO(3)$ with almost-global convergence," in *IEEE International Conference on Decision and Control*, 2012, pp. 2052–2058.

[70] ——, "Riemannian consensus for manifolds with bounded curvature," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 921–934, 2013.

[71] 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, pp. 3911–3918.

[72] Vicon, "Vicon Motion Systems Ltd." http://www.vicon.com.

[73] P. K. C. Wang, "Navigation strategies for multiple autonomous mobile robots moving in formation," *Journal of Robotic Systems*, vol. 8, no. 2, pp. 177–195, 1991.

[74] D. Zelazo, A. Franchi, and P. R. Giordano, "Rigidity theory in $SE(2)$ for unscaled relative position estimation using only bearing measurements," in *IEEE European Control Conference*, 2014, pp. 2703–2708.

[75] F. Zhang, M. Goldgeier, and P. S. Krishnaprasad, "Control of small formations using shape coordinates," in *IEEE International Conference on Robotics and Automation*, vol. 2, 2003, pp. 2510–2515.

[76] Z. Zhang, "Parameter estimation techniques: A tutorial with application to conic fitting," *Image and Vision Computing*, vol. 15, pp. 59–76, 1997.

[77] S. Zhao and D. Zelazo, "Bearing-based distributed control and estimation in networks of agents," in *IEEE European Control Conference*, 2015.

[78] ——, "Bearing-constrained formation control using bearing measurements," in *Israel Annual Conference on Aerospace Sciences*, 2015.

[79] S. Zhao, F. Lin, K. Peng, B. M. Chen, and T. H. Lee, "Distributed control of angle-constrained cyclic formations using bearing-only measurements," *Systems and Control Letters*, vol. 63, pp. 12–24, 2014.