Snapshots of modern mathematics
from Oberwolfach

№ 17/2019

Formation Control and Rigidity Theory

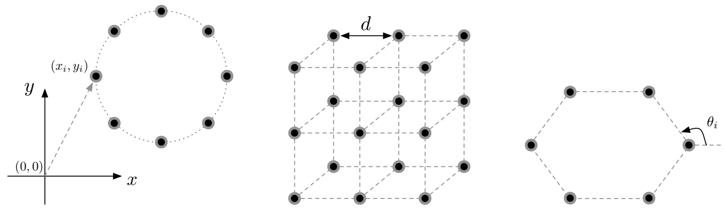
Daniel Zelazo[1] • Shiyu Zhao[2]

Formation control is one of the fundamental coordination tasks for teams of autonomous vehicles. Autonomous formations are used in applications ranging from search-and-rescue operations to deep space exploration, with benefits including increased robustness to failures and risk mitigation for human operators. The challenge of formation control is to develop distributed control strategies using vehicle onboard sensing that ensures the desired formation is obtained. This snapshot describes how the mathematical theory of rigidity has emerged as an important tool in the study of formation control problems.

1 Introduction

The basic task of formation control is to drive a team of unmanned vehicles to some desired spatial configuration. This configuration, which is called the *formation*, can be defined in a number of ways. One natural way could be

[1] Daniel Zelazo is supported by Israel Science Foundation (grant No. 1490/30) and the German-Israeli Foundation for Scientific Research and Development.
[2] Shiyu Zhao is supported by National Natural Science Foundation of China (Grant No. 61903308).



(a) A ring formation specified in a global coordinate system.

(b) A 3D grid formation specified by distances.

(c) A hexagonal formation specified by bearings.

Figure 1: Formations can be specified using global or relative state information.

to simply specify the absolute position of each vehicle in a global coordinate system. Every vehicle can then move to its specified position and the formation is obtained. However, in real-world applications, this approach may not be possible. For instance, in underwater or deep space applications, a global coordinate system may be unavailable. The vehicles must then depend on *relative sensing*, that is, to obtain the correct position, the vehicles rely only on information gathered from each other. In this setting, a formation may be specified by the relative positions of each vehicle, the distances between each vehicle, or the bearings between vehicles. Examples of some different formation types are shown in Figure 1.

In this snapshot, we will discuss formations specified by relative sensing. One of the advantages of these methods is that each vehicle can make decisions on where to move using information that is sensed *locally*, that is, without the need for a centralized coordinator.

2 The formation control problem

A common way to simplify the study of formations is to model each vehicle as a *kinematic point mass* [7, 8]. Kinematics is the study of the geometry of motion of objects, and a point mass means we do not consider the shape and size of the vehicle, but rather assume for simplicity's sake that we have a point in space that represents the vehicle. So, we consider a team of n vehicles and denote by $\mathbf{p}_i \in \mathbb{R}^d$ the position of vehicle i in a d-dimensional Euclidean space. In practical applications, the vehicles are modelled in either 2-dimensional or 3-dimensional spaces, so in all that follows, you can think of d as either equal to 2 or 3. The spatial configuration $\mathbf{p}$ of all the vehicles together will be denoted by a vector in $\mathbb{R}^{nd}$. For example, if we have a configuration of four vehicles at the corners of a square, say $\mathbf{p}_1 = (0,0), \mathbf{p}_2 = (1,0), \mathbf{p}_3 = (1,1)$ and $\mathbf{p}_4 = (0,1)$, so $n = 4$ and $d = 2$, we will write $\mathbf{p} = (0,0,1,0,1,1,0,1) \in \mathbb{R}^8$.

We will assume that each vehicle is able to sense certain quantities that are a function of their relative states. This could be the distance $d_{i,j}$ between vehicle i and j which is defined by

$$d_{i,j}^2 = \|\mathbf{p}_i - \mathbf{p}_j\|^2 = (p_{i,1} - p_{j,1})^2 + (p_{i,2} - p_{j,2})^2 + \cdots + (p_{i,d} - p_{j,d})^2. \quad (1)$$

Here, $\|\mathbf{x}\| = \sqrt{x_1^2 + \cdots + x_d^2}$ is the Euclidean *norm*, or size, of the vector $\mathbf{x} \in \mathbb{R}^d$. In other words, the distance $d_{i,j}$ is the length of the straight line segment between the vehicles i and j. Another possibility is the *bearing* $\mathbf{g}_{i,j}$ from vehicle i to vehicle j, which is defined to be the vector

$$\mathbf{g}_{i,j} = \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|}.$$

The bearing vector between two vehicles is illustrated in Figure 2. Note that $\mathbf{g}_{i,j} = -\mathbf{g}_{j,i}$, when measured in a common frame.

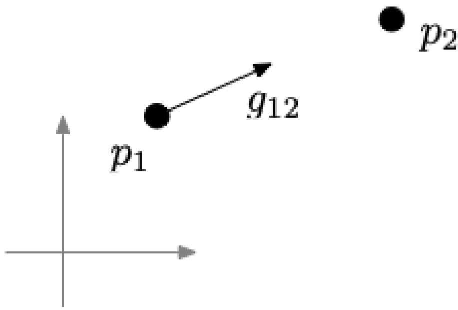


Figure 2: Bearing between agents.

The sensing and communication information of a multi-vehicle system can be encoded in a *graph* [5]. Here, in the context of graph theory, we mean something different from the graph of a function pictured as a curve in the plane. A graph, denoted $\mathcal{G}$, is defined by a set of *vertices*, $\mathcal{V}$, and a set of *edges*, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. For example, looking at diagram (c) in Figure 1, we can label the vertices from the left-hand top corner going clockwise by $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ and then we can see that there exists an edge, for instance, between vertices 2 and 3, but not between 2 and 4.

In the context of autonomous vehicles, the graph is called the *interaction graph*, and we define it by associating vehicle i to the vertex $v_i \in \mathcal{V}$ in the graph, and by saying that vehicle i has access to a relative measurement with vehicle

j if and only if $(v_i, v_j) \in \mathcal{E}$, that is, there is an edge in the graph between the vertices v_i and v_j. The combination of the interaction graph and the spatial configuration of the vehicle team, $(\mathcal{G}, \mathbf{p})$, is referred to as a *framework* [2].

With the above set-up, we are now prepared to define the formation control problem. Suppose that the vector $\mathbf{p}^* \in \mathbb{R}^{nd}$ represents one specific configuration of the vehicles in the desired formation, called the *nominal configuration.* Then we let $\mathcal{F}(\mathbf{p})$ denote the set of all possible desired formations. To be more precise, we write

$$\mathcal{F}(\mathbf{p}) = \left\{\mathbf{p} \in \mathbb{R}^{nd} \,|\, F(\mathbf{p}) = F(\mathbf{p}^*)\right\},$$

where F is a function on the space $\mathbb{R}^{nd}$ that extracts the information necessary for us to decide if a given configuration of vehicles is in the desired formation.

To make this clearer, suppose we take as an example the formation objective to be for four vehicles to form a square, with the shape of the formation specified by distance between the vehicles. The special point $\mathbf{p}^*$ could be as above, that is, with coordinates in the plane given by $\mathbf{p}_1^* = (0, 0), \mathbf{p}_2^* = (1, 0), \mathbf{p}_3^* = (1, 1)$ and $\mathbf{p}_4^* = (0, 1)$, and the function $F : \mathbb{R}^8 \to \mathbb{R}^6$ can then be defined by specifying the distance $d_{i,j}$ (as defined in (1) above) between each pair of vehicles, that is, by setting

$$F(\mathbf{p}) := (d_{1,2}, d_{2,3}, d_{3,4}, d_{1,4}, d_{1,3}, d_{2,4}).$$

So, the function F is defined by the length of the sides of the quadrilateral formed by the four vehicles, but also the diagonals, so as to ensure we obtain a square. Thus, the set $\mathcal{F}(\mathbf{p})$ represents all configurations $\mathbf{p}$ obtained by the rigid-body rotations and translations of the nominal configuration $\mathbf{p}^*$.

Problem 1 (Formation Control) *For a team of n vehicles with an interaction graph $\mathcal{G}$, design a control u_i for the velocity vector of each vehicle, such that the following properties are satisfied:*

i) *The control u_i is* distributed*, that is, it is not centrally controlled, and is a function only of the relative states between its neighbors defined by the graph $\mathcal{G}$.*
ii) *The control u_i is* stabilizing*, which means, roughly speaking, that small changes in input values lead to small changes in outputs.*
iii) *The set $\mathcal{F}(\boldsymbol{p})$ is such that vehicles in any generic starting position will in time eventually converge to a formation that lies in $\mathcal{F}(\boldsymbol{p})$, in other words, we want the set $\mathcal{F}(\boldsymbol{p})$ to be "asymptotically stable".*

A comprehensive survey of approaches for solving Problem 1 can be found in [9]. Stabilizing control laws for distance based formations were originally proposed in [7], and recent results for bearing based formations can be found in [11].

An important question to be addressed for any strategy attempting to solve Problem 1 is how to define the function $F(\mathbf{p})$ and thus the set $\mathcal{F}(\mathbf{p})$ such that the formation corresponds to the desired *formation shape*. In the next section, we present a mathematical theory known as *rigidity theory* that can be used to understand how to specify a formation shape unambiguously.

3 Rigidity Theory

The theory of rigidity is a way of characterising the "stiffness" or "flexibility" of structures formed by rigid bodies connected by flexible links or hinges. A structure is said to be *rigid* if it does not bend or otherwise move when it is subjected to an outside force. The origins of rigidity theory date back to a conjecture by Leonard Euler (1707–1783) in 1776 [4], which states that solid figures "can undergo change only to the extent that they are not undamaged or closed on all sides". The proof and development of this notion of deformation of solids includes results by Augustin-Louis Cauchy (1789–1857) [3] and, more recently, Henneberg [6] and Asimov [2]. In this section, we present an overview of the theory for bearing-constrained rigidity. For an excellent discussion on distance-constrained rigidity and its connection to Problem 1, we refer instead to [1].

3.1 Bearing-Constrained Rigidity

The basic problem in the theory of bearing-constrained rigidity is to decide whether or not a framework can be uniquely determined up to a translation and a scaling factor given the bearings between each pair of neighbors in the framework. Equivalently, we can ask whether two frameworks with the same inter-neighbor bearings have the same shape.

We first define some necessary notations. For a framework $(\mathcal{G}, \mathbf{p})$, we recall that the relative bearing from $\mathbf{p}_j$ to $\mathbf{p}_i$ is defined to be

$$\mathbf{g}_{i,j} = \frac{\mathbf{p}_i - \mathbf{p}_j}{\|\mathbf{p}_i - \mathbf{p}_j\|}, \quad \forall (i,j) \in \mathcal{E}. \tag{2}$$

Let $\mathbb{R}^{d\times d}$ be the space of all square matrices of size $d \times d$ with real entries, and let $I_d \in \mathbb{R}^{d\times d}$ denote the *identity* matrix. For any nonzero vector $\mathbf{x} \in \mathbb{R}^d$, for $d \geq 2$, we define the *orthogonal projection operator* $P : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ by setting

$$P(\mathbf{x}) = I_d - \frac{\mathbf{x}}{\|\mathbf{x}\|}\frac{\mathbf{x}^T}{\|\mathbf{x}\|}, \tag{3}$$

where here the "T" indicates the *transpose*, where we are considering a vector to be a matrix with one column and its transpose to be a matrix with one row.

For notational simplicity, we will write $P_{\mathbf{x}} = P(\mathbf{x})$. Let us consider the example of a vector $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$:

$$\begin{aligned} P_{\mathbf{x}} = P(x_1, x_2) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{\|\mathbf{x}\|^2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{\|\mathbf{x}\|^2} \begin{bmatrix} (x_1)^2 & x_1 x_2 \\ x_2 x_1 & (x_2)^2 \end{bmatrix} \\ &= \frac{1}{\|\mathbf{x}\|^2} \begin{bmatrix} (x_2)^2 & -x_1 x_2 \\ -x_2 x_1 & (x_1)^2 \end{bmatrix}. \end{aligned}$$

It can be verified that $P_{\mathbf{x}}^T = P_{\mathbf{x}}$ and $P_{\mathbf{x}}^2 = P_{\mathbf{x}}$. This latter equality is what is meant by a function being a projection. Two vectors $\mathbf{x}$ and $\mathbf{y}$ are said to be *orthogonal* if their scalar product $\mathbf{x} \cdot \mathbf{y}$ is equal to zero. To see what this means geometrically, recall that $\mathbf{x} \cdot \mathbf{y} = \|\mathbf{x}\|\|\mathbf{y}\| \cos\theta$, where θ is the angle between $\mathbf{x}$ and $\mathbf{y}$. Thus, if the scalar product is 0, the vectors are at an angle of 90°. The operator $P_{\mathbf{x}}$ can loosely be thought of as mapping any vector $\mathbf{y}$ into the part of $\mathbf{y}$ that is orthogonal to $\mathbf{x}$. You can check this with the 2-dimensional example given above, by calculating first $P_{\mathbf{x}}(\mathbf{y})$ for an arbitrary vector $\mathbf{y}$ and then working out the scalar product of $P_{\mathbf{x}}(\mathbf{y})$ and $\mathbf{x}$. It follows that two nonzero vectors $\mathbf{x}$ and $\mathbf{y} \in \mathbb{R}^d$ are parallel if and only if $P_{\mathbf{x}}(\mathbf{y}) = 0$ (or equivalently $P_{\mathbf{y}}(\mathbf{x}) = 0$), since a vector parallel to $\mathbf{x}$ has no component in the direction orthogonal to $\mathbf{x}$.

We are now prepared to define the fundamental concepts in bearing rigidity.

Definition 1 (Bearing Equivalency and Congruency) *We say that frameworks $(\mathcal{G}, \boldsymbol{p})$ and $(\mathcal{G}, \boldsymbol{p}')$ are* bearing equivalent *if $P_{(\boldsymbol{p}_i - \boldsymbol{p}_j)}(\boldsymbol{p}'_i - \boldsymbol{p}'_j) = 0$ for all $(i, j) \in \mathcal{E}$, and are* bearing congruent *if $P_{(\boldsymbol{p}_i - \boldsymbol{p}_j)}(\boldsymbol{p}'_i - \boldsymbol{p}'_j) = 0$ for all $i, j \in \mathcal{V}$.*

In other words, bearing equivalence means that the vectors defined by corresponding edges in each interaction graph are parallel. Bearing congruency means that the vectors between every corresponding pair (i, j) and (i', j') of vehicles are parallel, regardless of whether there is an edge between them in the interaction graph. Bearing congruency is a stronger property, since, by definition, bearing congruency implies bearing equivalency. The converse, however, is not true, as illustrated in Figure 3, where we see that the vectors that would be along the diagonals of each formation are not parallel.

Definition 2 (Bearing Rigidity) *A framework $(\mathcal{G}, \boldsymbol{p})$ is* bearing rigid *if there exists a constant $\epsilon > 0$ such that any framework $(\mathcal{G}, \boldsymbol{p}')$ that is bearing equivalent to $(\mathcal{G}, \boldsymbol{p})$ and satisfies $\|\boldsymbol{p}' - \boldsymbol{p}\| < \epsilon$ is also bearing congruent to $(\mathcal{G}, \boldsymbol{p})$.*

This means that if we have a framework and we alter it just a little into a framework that is bearing equivalent to the original, where here a "little"

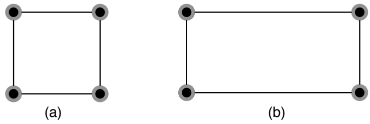


Figure 3: The two frameworks are bearing equivalent but *not* bearing congruent.

alteration is one such that the distance of one to the other is smaller than ϵ, then the altered framework has to also have the stronger property of being bearing congruent to the original.

Definition 3 (Global Bearing Rigidity) *A framework* $(\mathcal{G}, \boldsymbol{p})$ *is* globally bearing rigid *if any framework that is bearing equivalent to* $(\mathcal{G}, \boldsymbol{p})$ *is also bearing congruent to* $(\mathcal{G}, \boldsymbol{p})$*.*

This is just as in the previous definition, but now we don't need to specify that we only move a little between frameworks. Referring again to Figure 3, we see that the square framework in part (a) is not globally bearing rigid.

We next define *infinitesimal bearing rigidity*, which is one of the most important concepts in the theory of bearing-constrained rigidity. At this point, we need to assume that the interaction graph has an *orientation*, that is, that each edge has a direction associated to it, so we can say that an edge starts at some vertex v_i and ends at v_j. Suppose that there are a total of m edges and let $H \in \mathbb{R}^{m \times n}$ be the matrix defined by setting for the entries h_{ki},

$$h_{ki} := \begin{cases} 1, & \text{if edge } k \text{ starts at vertex } v_i \\ -1, & \text{if edge } k \text{ ends at vertex } v_i \\ 0, & \text{otherwise.} \end{cases}$$

Now let $\bar{H}$ be a new, bigger, matrix that is obtained from H by replacing each entry h_{ij} of H by $h_{ij} \cdot I_d$. Formally, this is known as the "Kronecker product" of H and I_d.[3]

We define the *bearing function* $F_B : \mathbb{R}^{dn} \to \mathbb{R}^{dm}$ by setting

$$F_B(\mathbf{p}) = \big(\mathbf{g}_1, \dots, \mathbf{g}_m\big) \in \mathbb{R}^{dm},$$

where the element $\mathbf{g}_k$ denotes the bearing vector along the edge k, that is, if the kth edge starts at v_i and ends at v_j, $\mathbf{g}_k = \mathbf{g}_{ij}$, as defined in Equation (2), and, for the same k, let us also define the notation $\mathbf{e}_k = \mathbf{p}_j - \mathbf{p}_i$. This means

[3] For more details and examples, we refer to https://en.wikipedia.org/wiki/Kronecker_product.

that $F_B(\mathbf{p})$ is just the concatenation of all the bearings into one vector. The *bearing rigidity matrix* $R(\mathbf{p})$ is defined to be the *Jacobian*, or matrix of partial derivatives, of the bearing function:

$$R(\mathbf{p}) = \frac{\partial F_B(\mathbf{p})}{\partial \mathbf{p}} \in \mathbb{R}^{dm \times dn} \\ = \operatorname{diag}\left(\frac{P_{\mathbf{g}_k}}{\|\mathbf{e}_k\|}\right)\bar{H}, \tag{4}$$

where $\operatorname{diag}\left(\frac{P_{\mathbf{g}_k}}{\|\mathbf{e}_k\|}\right)$ refers to the square matrix in $\mathbb{R}^{md \times md}$ with the blocks $P_{\mathbf{g}_1}/\|\mathbf{e}_1\|, \ldots, P_{\mathbf{g}_m}/\|\mathbf{e}_m\|$ along the diagonal and 0s everywhere else. The second inequality in (4) comes from applying the chain rule for differentiation to the rigidity function. Intuitively, we have that the bearing rigidity matrix takes into account both the orthogonal projections and the orientation of the edges in the interaction graph.

We would like now to consider how the bearing function changes with a small change in the positions of the vehicles, that is, we are interested in $F_B(\mathbf{p} + \delta\mathbf{p})$, where $\delta\mathbf{p}$ is a small variation of the configuration $\mathbf{p}$. Using the Taylor expansion for vector-valued functions, we can write an approximation

$$F_B(\mathbf{p} + \delta\mathbf{p}) \approx F_B(\mathbf{p}) + R(\mathbf{p})\delta\mathbf{p}.$$

If $R(\mathbf{p})\delta\mathbf{p} = 0$, then we say that $\delta\mathbf{p}$ is an *infinitesimal bearing motion* of $(\mathcal{G}, \mathbf{p})$. Furthermore, an infinitesimal bearing motion is called *trivial* if it corresponds to a translation and a scaling of the entire framework. These motions are called trivial because they clearly result in identical bearing functions: translating, shrinking or expanding a framework cannot change the bearings between the component vehicles, independently of the interaction graph.

Definition 4 (Infinitesimal Bearing Rigidity) *A framework is said to be* infinitesimally bearing rigid *if all the infinitesimal bearing motions are trivial.*

In summary, an infinitesimally bearing rigid framework means the only first-order changes to the positions of a framework that preserve the bearings are the translations and scalings. A framework that is not infinitesimally bearing rigid may allow some other motions that would still result in the same bearing function (that is, might have $R(\mathbf{p})\delta\mathbf{p} = 0$ even when $\delta\mathbf{p}$ is not trivial).

To illustrate this definition, let us consider two different sensing graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ for the example of four vehicles with the objective of forming a square, as above. The graphs are shown in Figure 4. Directly from the figure, we see that for the framework $(\mathcal{G}_1, \mathbf{p})$ the desired bearing vectors take the form

$$F_{B_1}(\mathbf{p}^*) = (0, -1, 1, 0, \sqrt{2}/2, -\sqrt{2}/2, 1, 0, 0, -1),$$

whereas for the framework $(\mathcal{G}_2, \mathbf{p})$ we have one less desired bearing to maintain and so the bearing vector is

$$F_{B_2}(\mathbf{p}^*) = (0, -1, 1, 0, 1, 0, 0, -1).$$

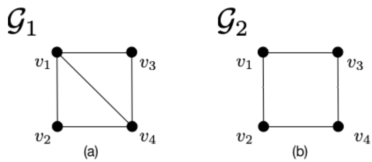


Figure 4: Two different sensing graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ on the vertex set $\mathcal{V} = \{v_1, v_2, v_3, v_4\}$.

For the graph $\mathcal{G}_1$, the only infinitesimal bearing motions are trivial, that is, translations or scalings. Observe though, that for the graph $\mathcal{G}_2$ there is another infinitesimal motion that leaves the bearing function unchanged. Indeed, if we change the square to a rectangle by leaving the two vehicles at v_1 and v_2 stationary and translating the other two by some fixed amount to the right, it is straightforward to verify that the bearing function F_{B_2} at these two configurations is equal. So $(\mathcal{G}_2, \mathbf{p})$ is not an infinitesimally bearing rigid framework.

Up to this point, we have introduced all the fundamental concepts in the bearing-constrained rigidity theory. We next explore the properties of these concepts.

For any undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, denote by $\mathcal{G}^\kappa$ the graph with an edge between every pair of vertices from $\mathcal{V}$, and by $R^\kappa(\mathbf{p})$ the bearing rigidity matrix of the framework $(\mathcal{G}^\kappa, \mathbf{p})$. The next result gives necessary and sufficient conditions for bearing equivalency and bearing congruency, and we note that the zeroes in the statement and the proof are zero vectors and zero matrices.

Theorem 1 *Two frameworks $(\mathcal{G}, \boldsymbol{p})$ and $(\mathcal{G}, \boldsymbol{p}')$ are bearing equivalent if and only if $R(\boldsymbol{p})\boldsymbol{p}' = 0$, and bearing congruent if and only if $R^\kappa(\boldsymbol{p})\boldsymbol{p}' = 0$.*

Proof. Referring to Equation (4), we have that

$$R(\mathbf{p})\mathbf{p}' = \frac{\text{diag} P_{\mathbf{g}_k}}{\|e_k\|} \bar{H}(\mathbf{p}')^T = \frac{\text{diag} P_{\mathbf{g}_k}}{\|e_k\|} (\|e_1'\|\mathbf{g}_1', \ldots, \|e_m'\|\mathbf{g}_m')^T.$$

It then follows that

$$R(\mathbf{p})\mathbf{p}' = 0 \text{ if and only if } P_{\mathbf{g}_k}\mathbf{g}_k' = 0, \forall k \in \{1, \ldots, m\}.$$

Therefore, by Definition 1, the two frameworks are bearing equivalent if and only if $R(\mathbf{p})\mathbf{p}' = 0$. It can be analogously shown that frameworks are bearing congruent if and only if $R^{\kappa}(\mathbf{p})\mathbf{p}' = 0$. ■

Since any infinitesimal bearing motion $\delta\mathbf{p}$ is such that $R(\mathbf{p})\delta\mathbf{p} = 0$ by definition, and since also $R(\mathbf{p})\mathbf{p} = 0$ (this follows from the fact that $P_{\mathbf{x}}(\mathbf{x}) = 0$ for every $\mathbf{x} \in \mathbb{R}^d$), we have that $R(\mathbf{p})(\mathbf{p} + \delta\mathbf{p}) = R(\mathbf{p})\mathbf{p} + R(\mathbf{p})\delta\mathbf{p} = 0$, and so Theorem 1 implies that $\mathcal{G}(\mathbf{p} + \delta\mathbf{p})$ is bearing equivalent to $\mathcal{G}(\mathbf{p})$. We now provide a necessary and sufficient condition for global bearing rigidity (GBR). Before stating the result, define for a matrix $M \in \mathbb{R}^{m\times n}$, the set $\mathrm{Null}(M)$ to be all those vectors $\mathbf{v} \in \mathbb{R}^n$ such that $Mv = 0$. For interested readers, the proof (and a further equivalent condition) can be found in [11].

Theorem 2 (Condition for GBR) *A framework $(\mathcal{G}, \boldsymbol{p})$ in $\mathbb{R}^d$ is globally bearing rigid if and only if* $\mathrm{Null}(R^{\kappa}(\boldsymbol{p})) = \mathrm{Null}(R(\boldsymbol{p}))$.

The following result shows that bearing rigidity (BR) and global bearing rigidity are equivalent notions.

Theorem 3 (Condition for BR) *A framework $(\mathcal{G}, \boldsymbol{p})$ in $\mathbb{R}^d$ is bearing rigid if and only if it is globally bearing rigid.*

These results then lead to algebraic conditions on the bearing rigidity matrix for infinitesimal bearing rigidity (IBR). Let us give just one of the easier to state results here, and refer again to [11]. Here the *rank* of a matrix A is defined to be the maximum number of linearly independent[4] columns of A.

Theorem 4 (Condition for IBR) *For a framework $(\mathcal{G}, \boldsymbol{p})$ in $\mathbb{R}^d$, the following statements are equivalent:*

(a) *$(\mathcal{G}, \boldsymbol{p})$ is infinitesimally bearing rigid;*

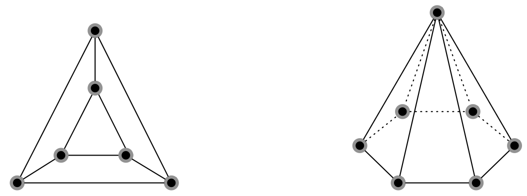
(b) $\mathrm{rank}(R(\boldsymbol{p})) = dn - d - 1$.

One of the powerful consequences of infinitesimal bearing rigidity is that these frameworks uniquely determine the shape of the framework. Figure 5 shows an example of a non-infinitesimally and infinitesimally bearing rigid framework. The conditions of Theorem 4 can be verified for these examples. In fact, infinitesimal bearing rigidity implies both bearing rigidity (Definition 2) and global bearing rigidity (Definition 3).

Theorem 5 *Infinitesimal rigidity implies bearing rigidity and global bearing rigidity. Furthermore, an infinitesimal bearing rigid framework is uniquely determined up to a translational and a scaling factor.*

---

[4] For more details, refer to https://en.wikipedia.org/wiki/Rank_(linear_algebra).



(a) A 2D framework that is not IBR.

(b) A 3D framework that is IBR.

Figure 5: An example of a framework that is not infinitesimally bearing rigid and one that is infinitesimally bearing rigid.

These results on bearing rigidity become essential when considering Problem 1. To guarantee that the set of desired configurations $\mathcal{F}(p)$ corresponds to unique shapes, we must ensure the infinitesimal bearing rigidity of the framework. In the next section, we present a distributed control law that solves Problem 1 using only bearing measurements, and some numerical simulations for the two frameworks $(\mathcal{G}_1, \mathbf{p})$ and $(\mathcal{G}_2, \mathbf{p})$ given above that demonstrate that any control law we may choose for a set of vehicles may not guarantee that it will converge to the desired formation shape if there are non-trivial motions of the framework that preserve the bearing formation.

4 Bearing-Only Formation Control

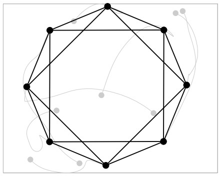
Given an understanding of how formation shapes can be uniquely determined by bearing measurements, we are now prepared to present a solution to the formation control problem. With this in mind, suppose that we have an infinitesimally bearing rigid framework $(\mathcal{G}, p^*)$, and the set $\mathcal{F}(p)$ of desired formations specified by bearings:

$$\mathcal{F}(\mathbf{p}) = \left\{ \mathbf{p} \in \mathbb{R}^{nd} \mid \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|} = \frac{\mathbf{p}_j^* - \mathbf{p}_i^*}{\|\mathbf{p}_j^* - \mathbf{p}_i^*\|} = \mathbf{g}_{ij}^*,\, \forall (v_i, v_j) \in \mathcal{E} \right\}.$$

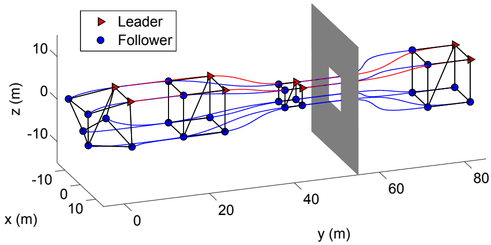
Then the proposed formation control law is given by

$$u_i(t) = -\sum_{(v_i, v_j) \in \mathcal{E}} P_{\mathbf{g}_{i,j}(t)} \mathbf{g}_{i,j}^*,\, v_i \in \mathcal{V}.$$

Here, $P_{\mathbf{g}_{i,j}(t)}$ is the orthogonal projection operator, as defined in (3). Observe that this control law is indeed distributed, as each neighbor only relies on the measured bearing to its neighbors and the desired bearing angle. The control also has a geometric interpretation, since $P_{\mathbf{g}_{ij}(t)}\mathbf{g}_{ij}^*$ is orthogonal to $\mathbf{g}_{ij}(t)$, where we recall that this means they are at an angle of 90° to each other. Thus, the control law attempts to reduce the bearing error between agents i and j.



(a) Bearing formation control.



(b) Formation maneuvering.

Figure 6: Examples of formation control.

This control law leads to an almost global *exponentially stable* system. By this we mean that for "almost any" initial starting position, the vehicles will achieve the desired formation exponentially quickly. In fact, the system has two isolated equilibrium points. The stable equilibrium corresponds precisely to the desired formation shape, while the other equilibriums are unstable and correspond to certain point reflections of the formation. For details on the stability analysis of this system, the reader is referred to [11].

Figure 6(a) shows an example of the proposed control law where a group of agents are tasked at forming the regular polygon shown. The trajectories of the agents are shown by the grey lines, and the final formation is marked by the black lines. This bearing formation control framework can also be extended to allow for leaders to drive the formation and control its scale [10]. This is demonstrated in Figure 6(b).

We finally demonstrate numerical simulations for the same control law given above and the interaction graphs $\mathcal{G}_1$ and $\mathcal{G}_2$. In both cases we aim to drive the formation to a square specified by the desired bearings in $F_{B_1}(\mathbf{p}^\star)$ and $F_{B_2}(\mathbf{p}^\star)$ respectively. We assume the same initial conditions for both simulations.

Figure 7 shows the resulting trajectories for the infinitesimally bearing rigid framework. It can be verified that the formation converges to a square, where each side has length ~ 3.79.

Figure 8 shows the resulting trajectories for the non-rigid framework. It can clearly be seen that the resulting formation converges to a rectangle.

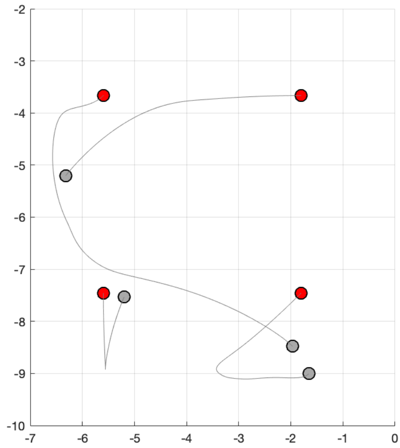


Figure 7: Trajectories for an infinitesimally bearing rigid framework.

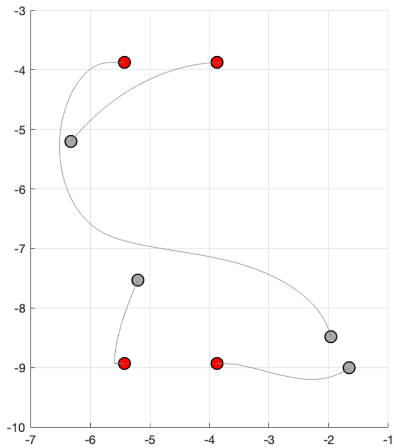


Figure 8: Trajectories for non-infinitesimally bearing rigid framework.

Finally, we show in Figure 9 the total bearing error function along the trajectories for both examples. That is, we plot the function

$$e(t) = \|F_B(\mathbf{p}(t)) - F_B(\mathbf{p}^{\star})\|^2.$$

As can be seen, both cases minimize the bearing error. However, only the infinitesimally rigid case ensures the final formation is indeed the desired one.

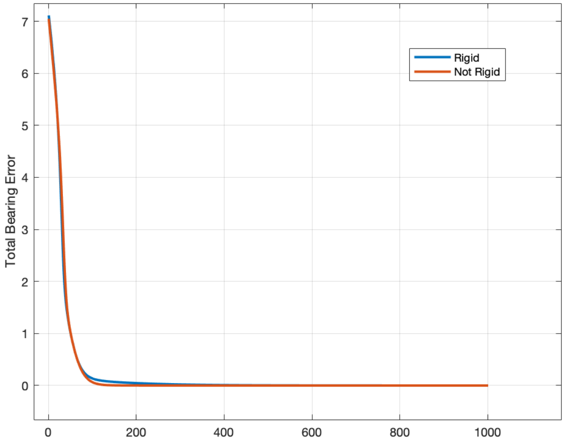


Figure 9: Total bearing error.

References

[1] B. D. O. Anderson, C. Yu, B. Fidan, and J. M. Hendrickx, *Rigid graph control architectures for autonomous formations*, IEEE Control Systems Magazine **28** (2008), no. 6, 48–63.

[2] L. Asimow and B. Roth, *The rigidity of graphs*, Transactions of the American Mathematical Society **245** (1978), 279–289.

[3] A. L. Cauchy, *Sur les polygones et polyèdres, second mémoire*, Journal de l'École polytechnique **9** (1813), 87–98.

[4] L. Euler, *Continuatio fragmentorum ex adversariis mathematicis depromptorum*, Opera Postuma Mathematica e Physica, Vol. 1, St. Petersburg Academy of Sciences, 1862.

[5] C. Godsil and G. F. Royle, *Algebraic graph theory*, Springer, New York, 2001.

[6] L. Henneberg, *Die graphische statik der starren systeme*, B. G. Teubner, Leipzig, 1911.

[7] L. Krick, M. E. Broucke, and B. A. Francis, *Stabilization of infinitesimally rigid formations of multi-robot networks*, International Journal of Control **82** (2009), no. 3, 423–439.

[8] J. A. Marshall, M. E. Broucke, and B. A. Francis, *Formations of vehicles in cyclic pursuit*, IEEE Transactions on Automatic Control **49** (2004), no. 11, 1963–1974.

[9] K.K̇. Oh, M.-C. Park, and H.-S. Ahn, *A survey of multi-agent formation control*, Automatica **53** (2015), 424–440.

[10] S. Zhao and D. Zelazo, *Translational and scaling formation maneuver control via a bearing-based approach*, IEEE Trans. on Control of Networked Systems (2015), no. 99, 1.

[11] ———, *Bearing rigidity and almost global bearing-only formation stabilization*, IEEE Transactions on Automatic Control **61** (2016), no. 5, 1255–1268.

Daniel Zelazo *is an assistant professor of aerospace engineering at the Technion - Israel Institute of Technology.* Shiyu Zhao *is post-doctoral researcher at the University of California - Riverside.*

Mathematical subjects
Discrete Mathematics and Foundations

Connections to other fields
Engineering and Technology

DOI
10.14760/SNAP-2019-017-EN

Snapshots of modern mathematics from Oberwolfach provide exciting insights into current mathematical research. They are written by participants in the scientific program of the Mathematisches Forschungsinstitut Oberwolfach (MFO). The snapshot project is designed to promote the understanding and appreciation of modern mathematics and mathematical research in the interested public worldwide. All snapshots are published in cooperation with the IMAGINARY platform and can be found on www.imaginary.org/snapshots and on www.mfo.de/snapshots.

ISSN 2626-1995

Mathematisches Forschungsinstitut Oberwolfach

Member of Leibniz Leibniz Association

IMAGINARY open mathematics