RBE 510 – Multi-Robot Systems
Lecture 4: Controlling Groups of Robots

Kevin Leahy

September 2, 2025

Admin

- HW 1 Due Friday

- Office hours Wednesday 3-3:45, UH 250 D

Groups

- Group 1
  - Honor
  - Mann
  - Hart
- Group 2
  - Shah
  - Green
  - Raja
- Group 3
  - Boozer
  - Kaiser
  - Smith
- Group 4
  - Cruse
  - Rosenstein

Paper Presentations

- Presentation dates
  - 9/9
  - 9/16
  - 9/23
  - 9/30

- One paper will be assigned to each group
- Whole class read the paper and be prepared to discuss
- Assigned group plan ~20 minute presentation
- Rubric, paper, and assignment posted this afternoon

Paper Presentation

- Goal:
  - Get a sense of some of the research going on in this field that is outside of what we can cover in class
  - Tie topics from class (trade-offs, complexity, etc.) to current work

- Criteria
  - Explain the main/important ideas to the class
  - Cover assumptions, limitations, trade-offs
  - Discussion/questions for the class

Recap

- Last time:
  - Robot models
    - Motion
    - Comms
    - Processing
  - Generality of models we use
  - Distributed algorithms and complexity

Today

- Back to formations
  - Leader-follower
  - Some variants

- Non-consensus-based approaches to controlling formations

- Reference-frame invariant control

In class so far

- Consensus-based control

$$\dot{x} = -Lx$$

  – Including different assumptions on topologies, information flow, etc.

- Formation control

  – Modified consensus

$$\dot{x} = -Lx + d$$

In class so far

- Goal is static
  - Converge to an average value
  - Converge to a given formation
- How is such a formation determined?
- What if there’s a sequence of tasks to do?

In class so far

- Lots of things are know *a priori*
  - Desired formation
  - Global reference frame/absolute location
- What if this type of information is unavailable?

Leader-Follower Control

Moving a Group of Agents

- So far, we've looked at agents doing something with respect to other agents
  - Reach a consensus
  - Rendezvous
  - Make a formation
  - Etc.

- What if we want to control when and how they move?

- Human input commands?

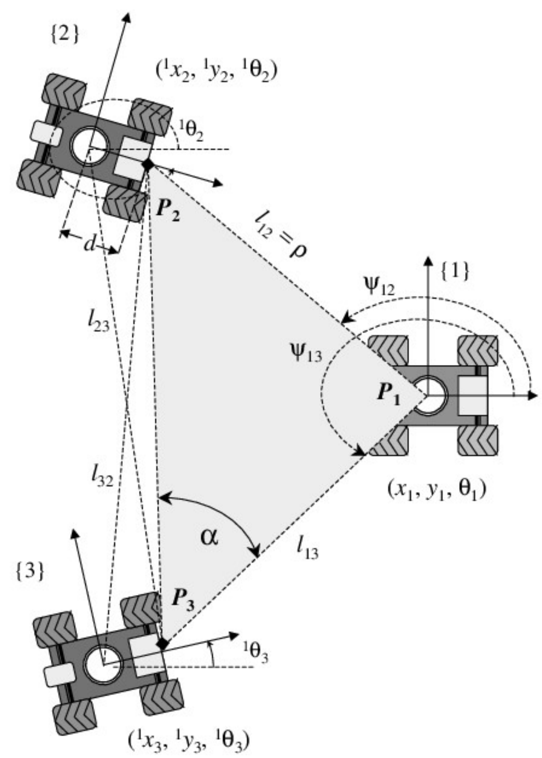
Leader-Follower Networks

- Ji, Muhammad, and Egerstedt 2006
- Ch. 6 of Mesbahi and Egerstedt

- This is a “sheep dog” problem
  - For a heterogenous group with leaders and followers, can we use consensus to control a formation and its location
  - How can the leaders move to force the followers to any desired position?

Leader-Follower Formation Intuition

- One or more agent needs control input
- The rest just perform consensus or formation control



The Model

- n_l leaders
- n_f followers
- Followers run consensus algorithm on position

$$\dot{x} = -Lx$$

- Leaders run a different control law to “herd” the followers
  - What control law should they run?
  - How does it influence the followers

The Model

- $\dot{x} = -Lx$
- Let’s start by splitting the state

$$x = \begin{bmatrix} x_f \\ x_l \end{bmatrix}$$

- What does this do to the Laplacian?

The Laplacian

- Write our new Laplacian as

$$L = \begin{bmatrix} L_f & l_{fl} \\ l_{fl}^T & L_l \end{bmatrix}$$

- $L \in \mathbb{N}^{n \times n}$
- $L_f \in \mathbb{N}^{n_f \times n_f}$
- $L_l \in \mathbb{N}^{n_l \times n_l}$
- $l_{fl} \in \mathbb{N}^{n_f \times n_l}$

Follower Dynamics

- Overall dynamics

$$\dot{x} = -Lx$$

$$\dot{x} = -\begin{bmatrix} L_f & l_{fl} \\ l_{fl}^T & L_l \end{bmatrix} \begin{bmatrix} x_f \\ x_l \end{bmatrix}$$

- What are the follower dynamics?

Follower Equilibria

- $\dot{x}_f = -L_f x_f - l_{fl} x_l$

- Where is equilibrium?

Equilibrium

- $x_f = -L_f^{-1} l_{fl} x_l$

- Can we compute it?

Follower Laplacian

- Follower Laplacian is positive definite (and therefore invertible)

Does it converge?

- Yes!

Quick example

- 4 agents
- $x_l = [0, 3]^T$

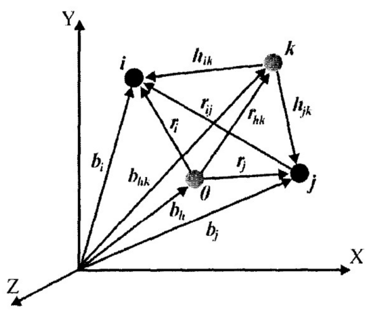
Formation Dynamics

- $\dot{x}_f = -L_f x_f - l_{fl} x_l + d$

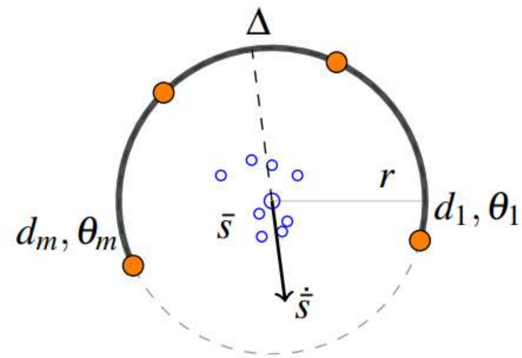
- Equilibria
- $x_f = -L_f^{-1} l_{fl} x_l + L_f^{-1} d$ (unique and stable)

- Reduces to
- $\tilde{x}_f = x_f - L_f^{-1} d$
- $\dot{\tilde{x}}_f = \dot{x}_f$

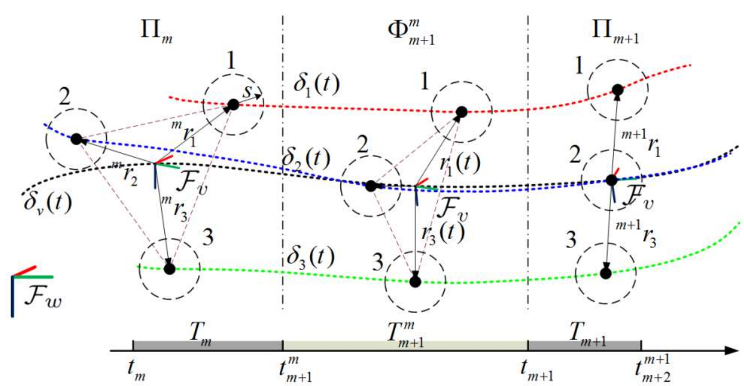
Other approaches



Leonard and Fiorelli 2001



Pierson and Schwager 2015



Zhou and Schwager 2015

Controlling Abstractions

Alternatives to Nearest Neighbor Control

- Based on Belta and Kumar 2004
- One-to-many control regime
- All agents run an identical protocol
- Control is based on the aggregate (global) state of agents instead of local information

Controlling an Abstraction

- Consider N robots, with robot i having state $q_i \in \mathbb{R}^2$
- State is position in *world frame* $\{W\}$
- Single integrator: $\dot{q}_i = u_i$

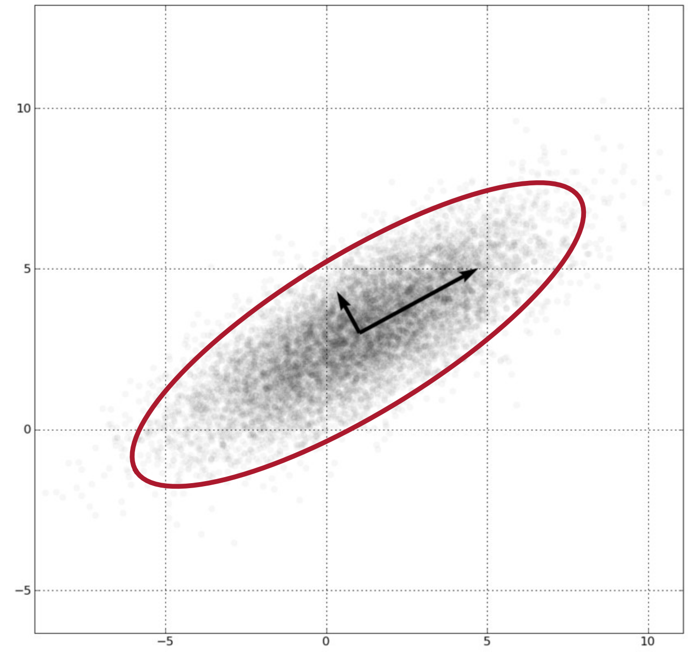
- Collect them together

$$\dot{q} = u$$

Where $q \in Q = \prod_{i=1}^{N} Q_i \in \mathbb{R}^{2N}$

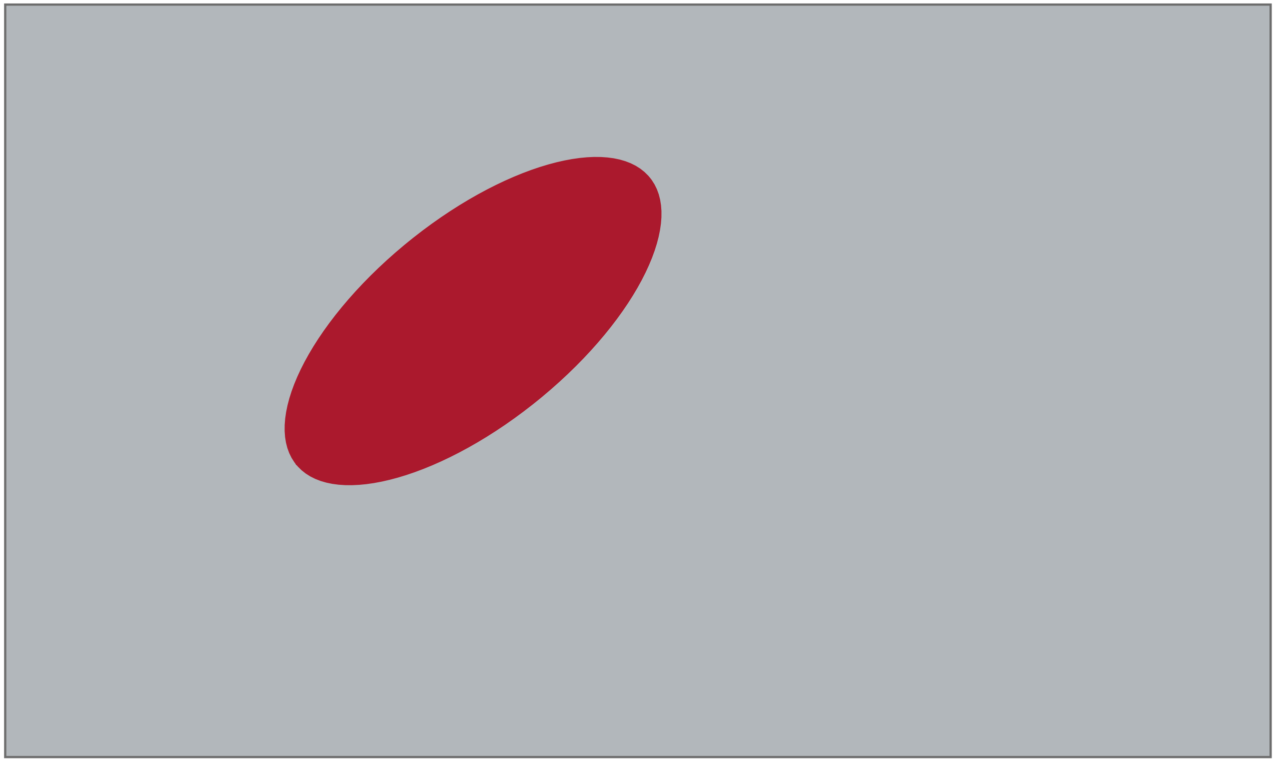
Abstracting the agents

- Consider a bunch of agents distributed in $\mathbb{R}^2$
- Assume they are normally distributed*
- What's a good abstraction for the agents?



*Not technically necessary but it makes the intuition cleaner

Abstraction



- How to describe an ellipse in $\mathbb{R}^2$?
- Can we describe the motion we want the agents to execute in terms of an ellipse?

Abstraction

- Assume agent positions q_i are realizations of a random variable with mean μ and covariance Σ
- If N is sufficiently large, sample mean and covariance converge to true values of the Normal distribution
- Rotation R diagonalizes the covariance
- Covariance matrix has eigenvalues s_1 and s_2

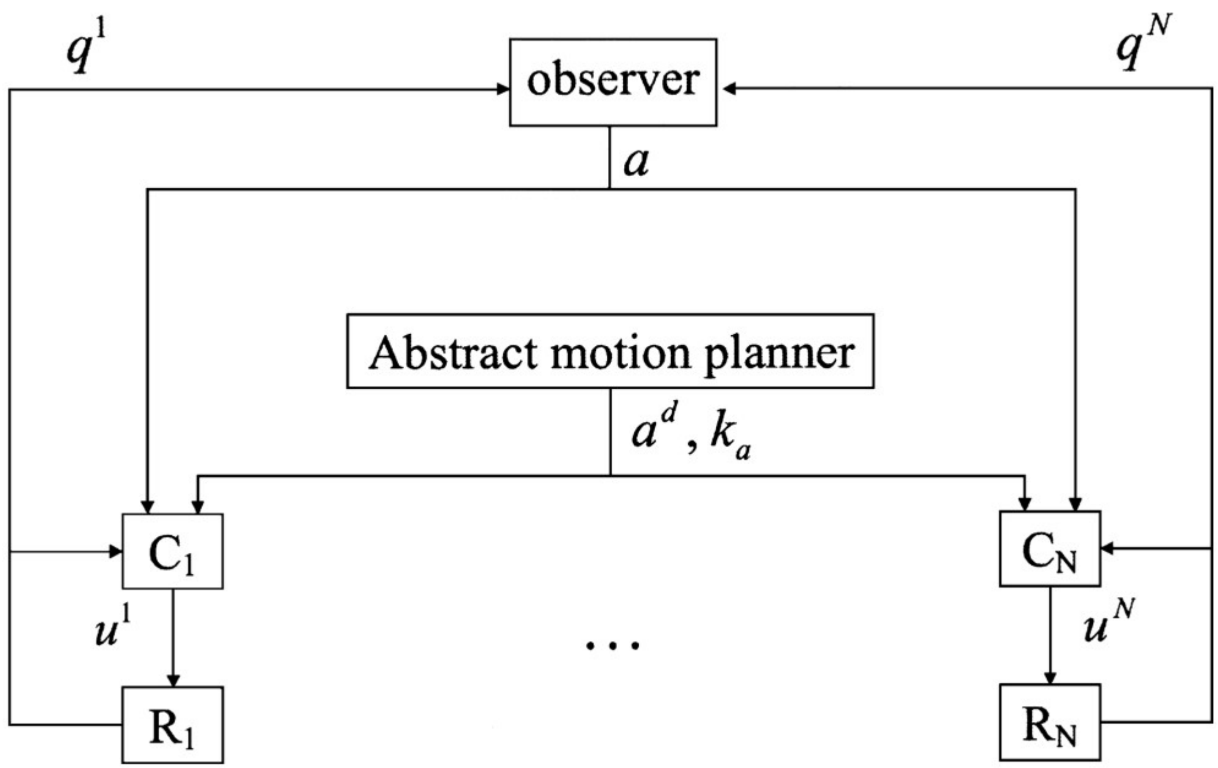
Abstraction

- Can estimate as follows
  - $\mu = \frac{1}{N}\sum_{i=1}^{N} q_i$
  - Defines "equiprobability ellipse" for probability p
    - $c = -2\ln(1-p)$
    - $(x-\mu)^T\Sigma^{-1}(x-\mu) = c$
  - I.e., given a desire to enclose 99% of the robots, we can define a corresponding ellipse that will contain that probability mass consistently

Realism of abstraction

- Can estimate from camera or similar "stand-off" surveillance asset
- Doesn't depend on high-quality sensor or number of agents
- It is an "equipotential" view of the world

Architecture



- Control law C_i of robot R_i depends only on
  - Own state q_i
  - Abstract state a, provided by observer
  - Desired state or trajectory
- No explicit inter-agent dependence

Controlling the Abstraction

- We can describe the desired motion of the ellipse as follows
  - $\dot{\mu} = K_\mu(\mu^d - \mu)$
  - $\dot{\theta} = k_\theta(\theta_d - \theta)$
  - $\dot{s_1} = k_{s_1}(s_1^d - s_1)$
  - $\dot{s_2} = k_{s_2}(s_2^d - s_2)$

- Where $K_\mu \in \mathbb{R}^{2\times 2}$ and $k_\theta, k_{s_{1,2}} > 0$

- Used to guarantee the behavior of the *abstraction* on a trajectory

- How to get the agents to realize the trajectory?

Map to Dynamics

- The derivation involves “simple but rather tedious calculations”

- Overall formation controls are
  - $\dot{q} = X_q^{\mu}\dot{\mu} + \frac{s_1 - s_2}{s_1 + s_2}\dot{\theta}X_q^{\theta} + \frac{\dot{s_1}}{4s_1}X_q^{s_1} + \frac{\dot{s_2}}{4s_2}X_q^{s_2}$

- X_{q} are some matrices whose precise details don’t matter for us

- Note each aspect of the abstract is controllable independently

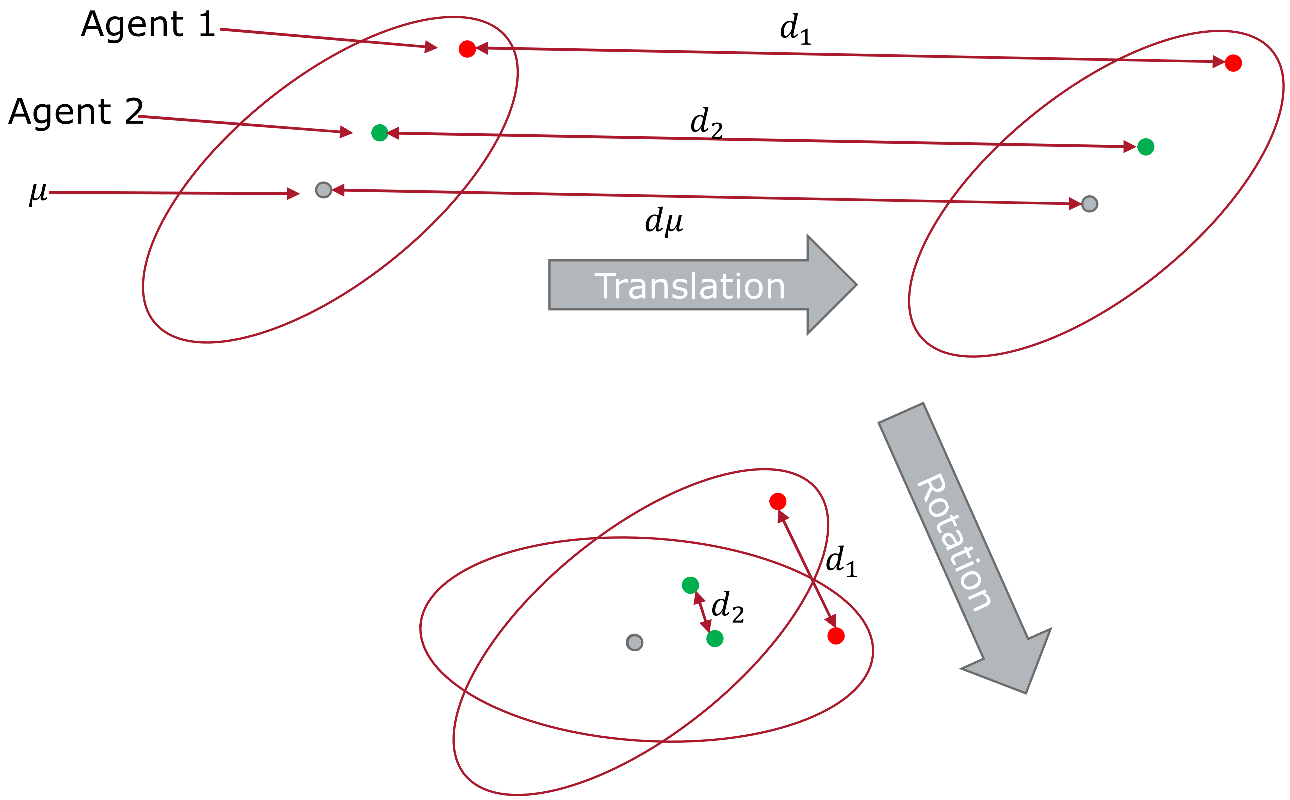
Map to Dynamics

- Convert overall dynamics to individual agent dynamics

$$u_i = \dot{q}_i = \dot{\mu} + \frac{s_1 - s_2}{s_1 + s_2} H_3(q_i - \mu)\dot{\theta} + \frac{1}{4s_1} H_1(q_i - \mu)\dot{s_1} + \frac{1}{4s_2} H_2(q_i - \mu)\dot{s_2}$$

- “Rather centralized”
- Agents need to know their own absolute state
- I.e., position relative to the swarm/abstraction – big assumption!

Agents and Distance from μ



Stability, controllability, etc.

- Some important properties
  - Proposition 2:
    - If a (the abstraction) is bounded, then so are all q_i
    - I.e., if you can control the abstraction, the control laws are well-defined
    - Argument follows that if μ, s_1, and s_2 are bounded, then controls will be bounded
  - Proposition 3:
    - The closed-loop system converges to equilibrium
    - $\dot{a} = 0$ iff $\dot{q}_i = 0$ for all i
    - The rest follows from some Lyapunov and LaSalle-based proofs

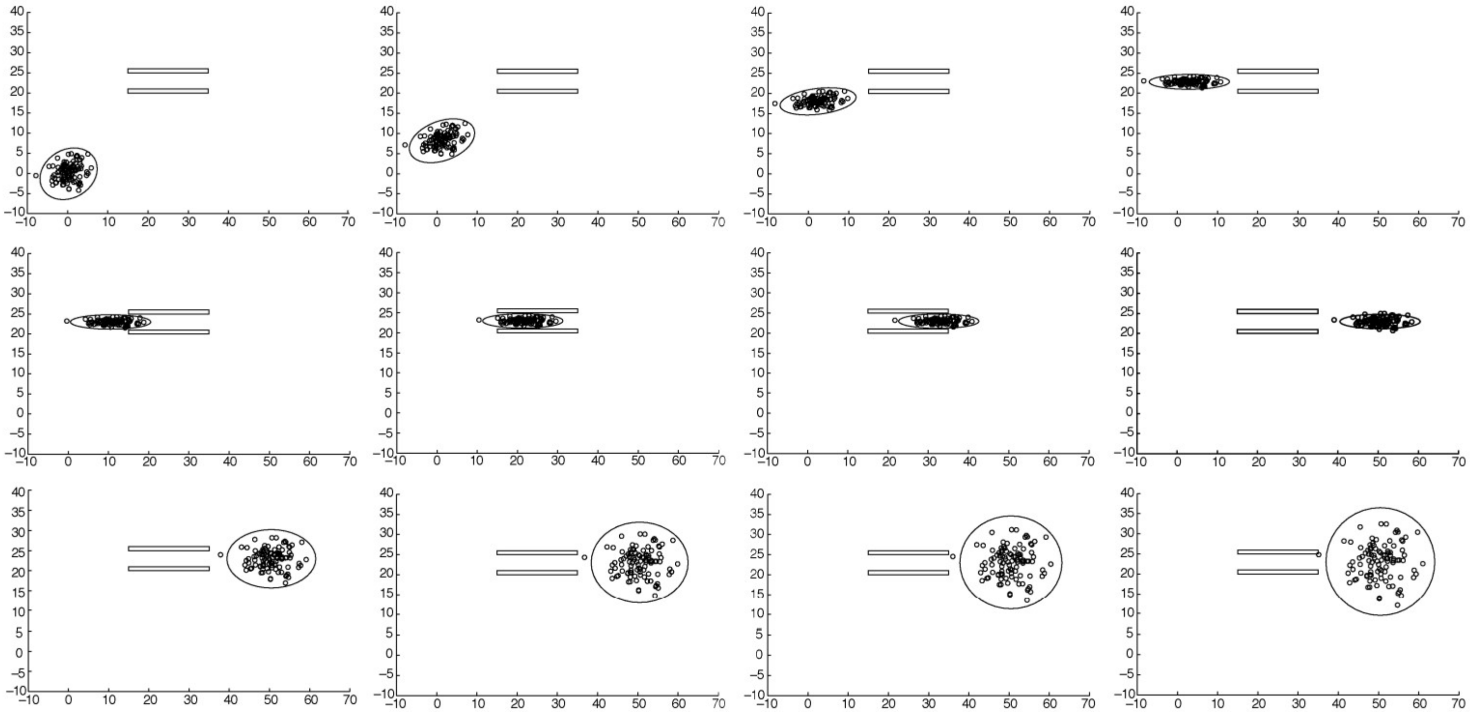
Contractions and Expansions

- If you cede control of orientation (i.e., assume abstraction keeps its initial orientation), then control law

$$\dot{q}_i = \dot{u}_i = \dot{\mu} + \frac{q_i - \mu}{2s}\dot{s}$$

- This allows scaling of the formation via s
- Can switch from scaling mode back to standard mode and vice versa

Example



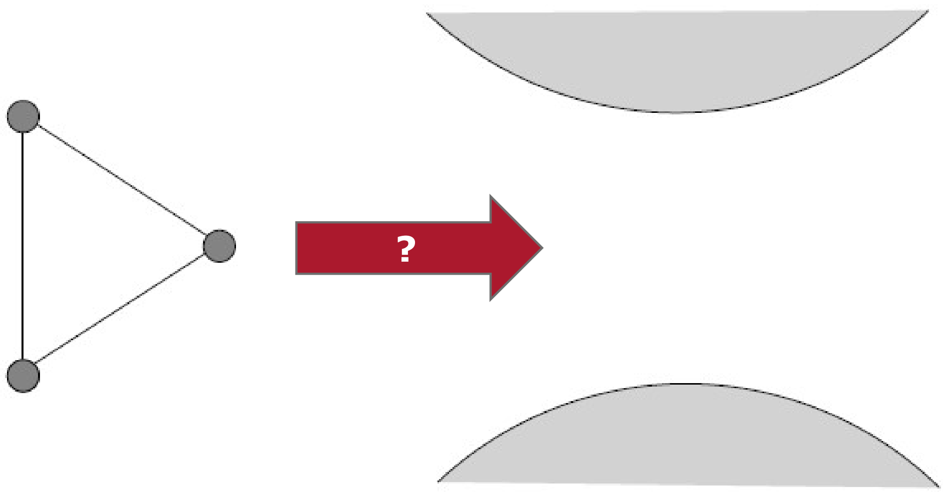
$N = 100$ robots; equiprobability ellipse of 99%

Trade-offs

- Distributed
- Scalable
- Centralized! – Observation and control come from centralized source
- Agents require good self-localization in a global reference frame

Modifications to Formation Control

Navigation with Formations

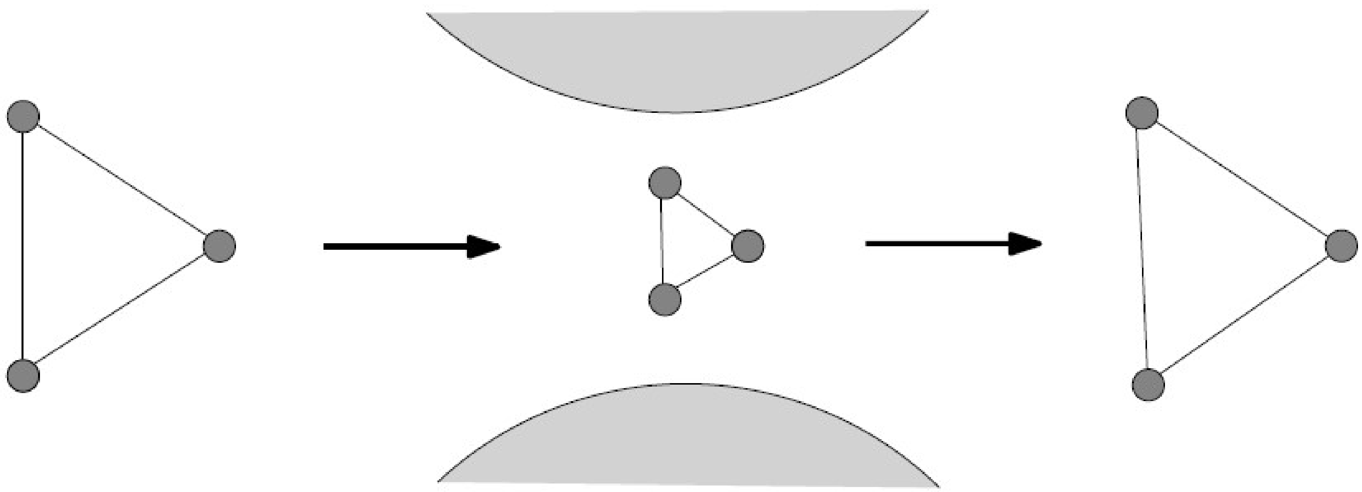


Scaling

- Recall:
  - Formation control: $\dot{x} = -Lx + d$
  - Under conditions:
    - $d_{ij} = -d_{ji}$
    - $1^T d = 0$
- Can scale the entire vector by a term $\alpha \in \mathbb{R}_+$
  - $\dot{x} = -Lx + \alpha d$
  - Does not violate our conditions
- Formation control is therefore **scale invariant**
- But, in the most common case you need to specify the scale

Scaling

- Useful for navigating narrow passages, for example



$\alpha = 1$ $\alpha = 0.5$ $\alpha = 1$

Translation

- Formations are **translation invariant**

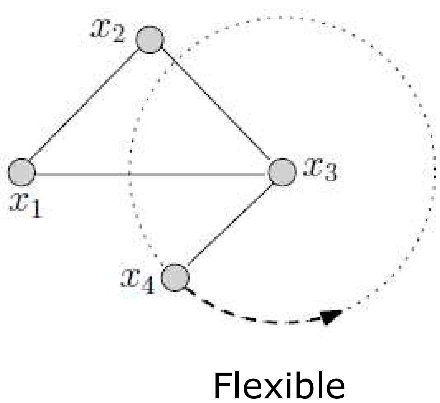
- We already saw this
  - If x^* is a solution to $Lx^* + d = 0$, so is $L(x^* + 1\alpha) + d$
  - If ∃ one solution, ∃ infinitely many solutions

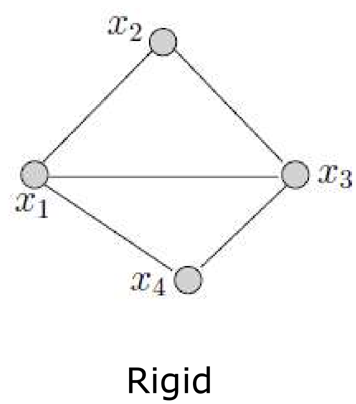
- I.e., the formation can be displaced and still remain a solution to the closed-loop control law

Rigidity

- Imagine a formation already in goal state x^*

- If the only way to translate a single agent is to translate the entire formation, the formation is **rigid**





Rigidity

- Thus, to ensure a feasible formation, the best bet is to ensure
  - $d_{ij} = -d_{ji}$
  - $1^T d = 0$
  - Graph is rigid
- How to guarantee rigidity?

Determining Rigidity

- Assume agents in goal formation x^*
- Apply infinitesimal control input $u = [u_1^T, u_2^T, \ldots, u_n^T]$

- We want the distances between edges to remain constant in that motion

- $\left(\dot{x}_i(t) - \dot{x}_j(t)\right)^T \left(x_i(t) - x_j(t)\right) = 0$ for all $i, j \in E$

Determining Rigidity

- $\left(\dot{x}_i(t) - \dot{x}_j(t)\right)^T \left(x_i(t) - x_j(t)\right) = 0$ for all $i, j \in E$

  Trajectory Constant d_{ij}

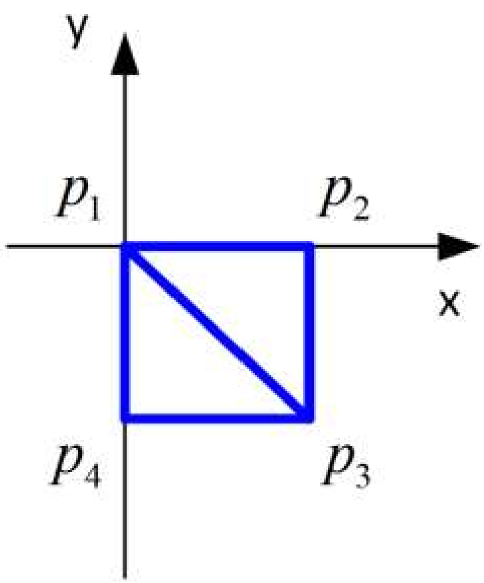
- In matrix form, we write

$$R\big(G(x^*)\big)u = 0$$

- Call $R(G(x^*))$ the **rigidity matrix**

- A framework with $n \geq 2$ points in $\mathbb{R}^2$ is (infinitesimally) rigid iff rank $\mathrm{rank}\left(R\big(G(x^*)\big)\right) = 2n - 3$

Infinitesimal Rigidity

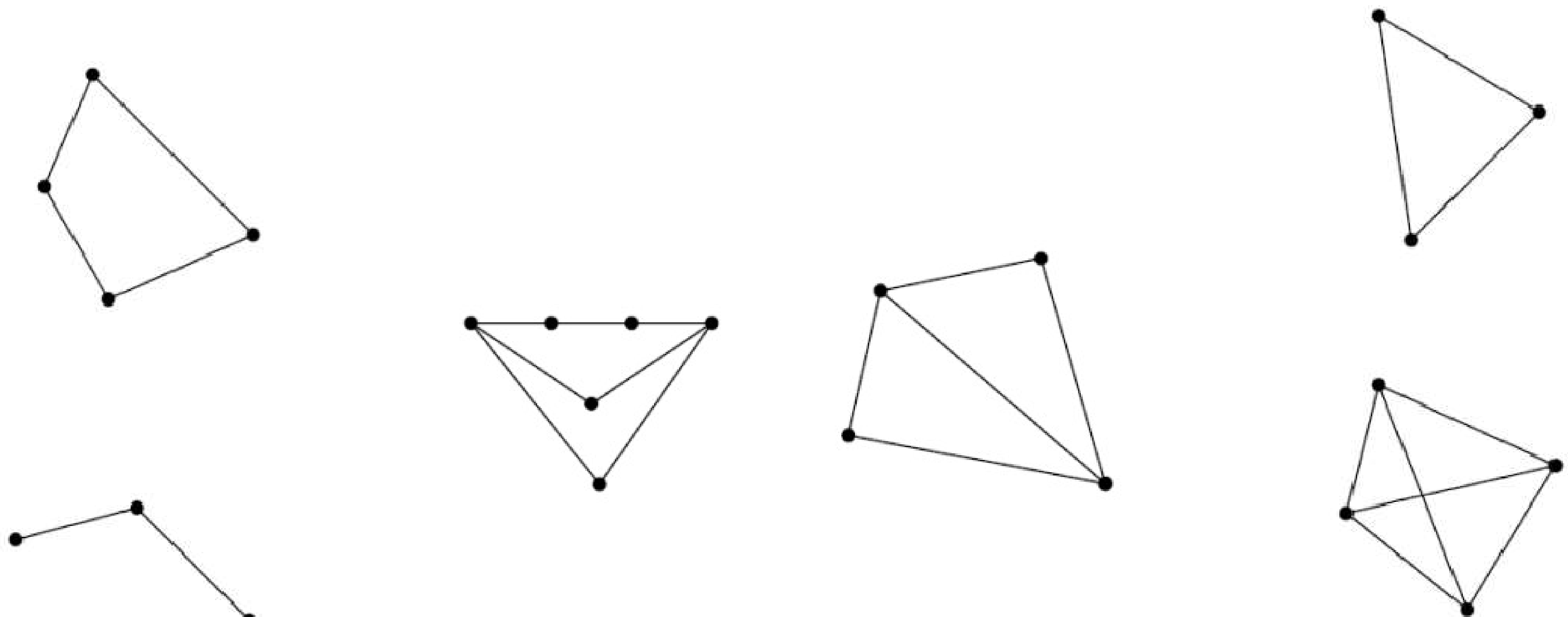


$$R = \begin{bmatrix} (p_1 - p_2)^T & -(p_1 - p_2)^T & 0 & 0 \\ (p_1 - p_3)^T & 0 & -(p_1 - p_3)^T & 0 \\ (p_1 - p_4)^T & 0 & 0 & -(p_1 - p_4)^T \\ 0 & (p_2 - p_3)^T & -(p_2 - p_3)^T & 0 \\ 0 & 0 & (p_3 - p_4)^T & -(p_3 - p_4)^T \end{bmatrix}$$

Infinitesimal Rigidity

$$R = \begin{bmatrix} (p_1 - p_2)^T & -(p_1 - p_2)^T & 0 & 0 \\ (p_1 - p_3)^T & 0 & -(p_1 - p_3)^T & 0 \\ (p_1 - p_4)^T & 0 & 0 & -(p_1 - p_4)^T \\ 0 & (p_2 - p_3)^T & -(p_2 - p_3)^T & 0 \\ 0 & 0 & (p_3 - p_4)^T & -(p_3 - p_4)^T \end{bmatrix}$$

Rigidity



Infinitesimally Rigid

Locally Rigid

Flexible

Globally Rigid

Reference Frame Invariance

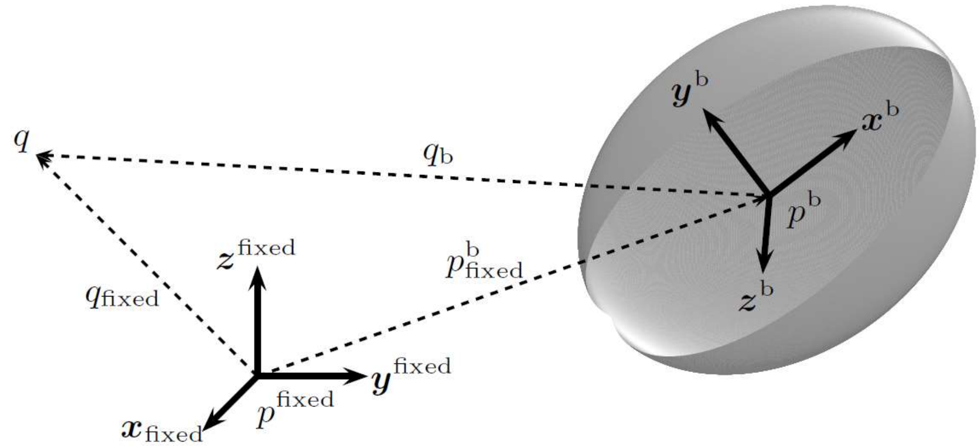
Sources

- Jorge Cortes “Global and robust formation-shape stabilization of relative sensing networks,” *Automatica*, 2009
- Bullo, Cortes, and Martinez Ch. 3
- Krick aet al. “Stabilization of Infinitesimally Rigid Formations of Multi-Robot Networks”, *Conference on Decision and Control*, 2008

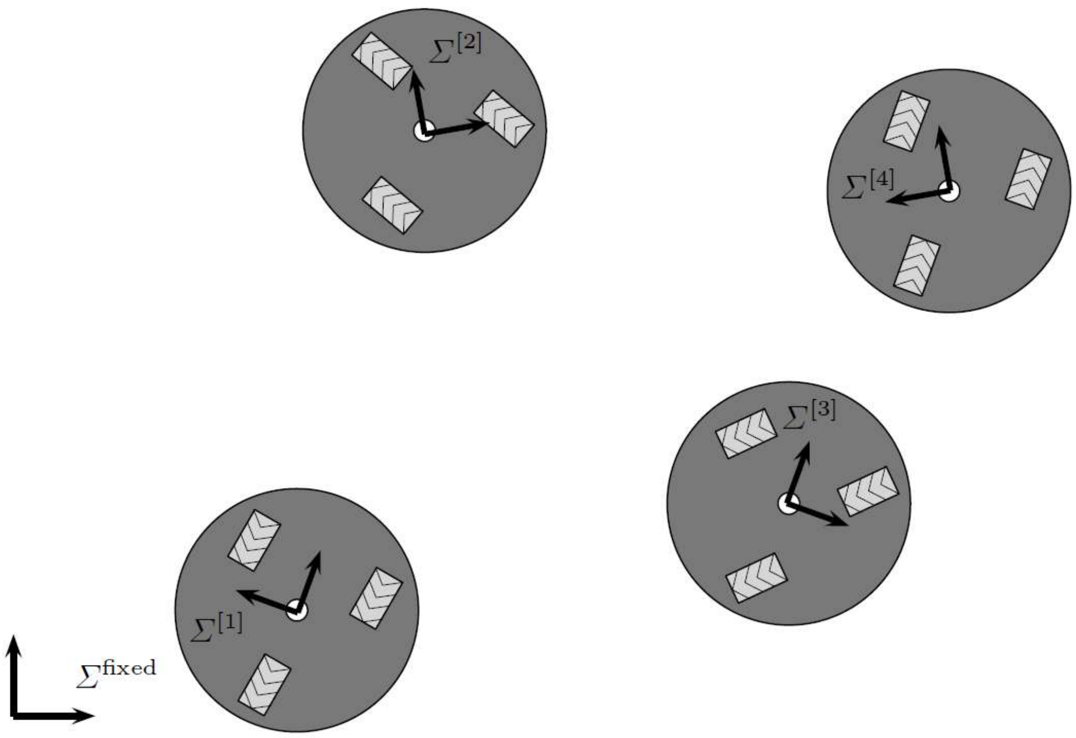
Reference Frames and Kinematics

- Rotation matrices in d-dimensions
  - $SO(d) = \{R \in \mathbb{R}^{d\times d} \mid RR^T = I_d, \det R = 1$

- $q_{fixed} = R^b_{fixed} q_b + p^b_{fixed}$
- $v_{fixed} = R^b_{fixed} v_b$

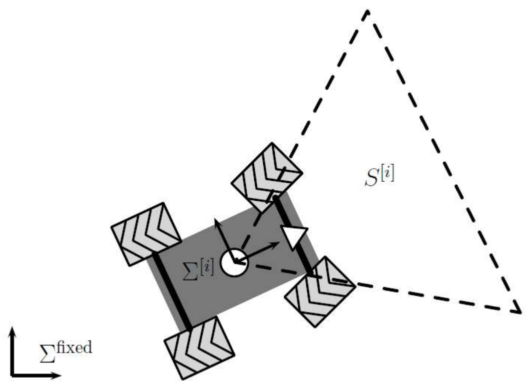


Problem Setup

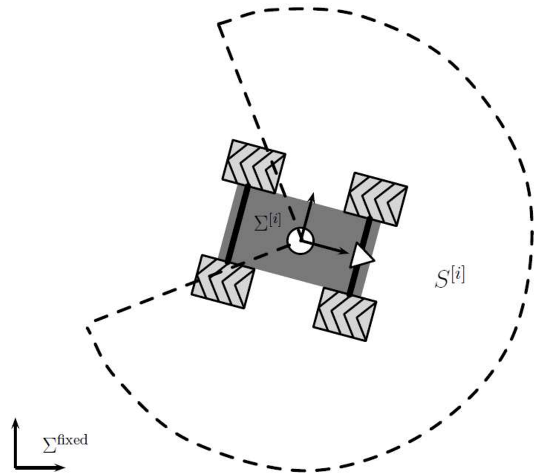


- Each agent i has a body-fixed reference frame, $\Sigma^{[i]}$
- They are all defined relative to Σ^{fixed}, which is unknown to the agents

Problem Setup



Cone-shaped Camera FOV



$270°$ FOV of laser scanner

- Agents are equipped with sensor that return the location of other agents, objects in the environment, etc. *in their own body frame*
- We'll assume a circular footprint, similar to our disc-based comms model

Sensing Model Consequences

1. Robots have no information about the absolute position and orientation of themselves, the other robots, or any part of the environment
   - This significantly relaxes many of the assumptions in examples we've been studying!
   - It also makes them harder to solve
2. The relative sensing capabilities of the robots gives rise to a **sensing graph**, whose edges denote robots within sensing range of a given robot
   - This will (for now) replace our notion of a communication graph
   - The relationship between the two graphs could be more complicated but we ignore that for the time being

Rigid-body Transformations

- Rigid body transformations contain a translation and rotation
- For a set of points $x^* = \{x_1^*, x_2^*, \dots, x_n^*\}$, $Rgd(x^*)$ is $(x_1, x_2, \dots, x_n)$ such that
  - There exists $(q, R) \in \mathbb{R}^d \times SO(d)$ such that $x_i = Rx_i^* + q$
- Note, scalar distance between any two points i, j is the same under such a transformation

Previous versions are not SE(N)-invariant

- Earlier versions don't work this way
  - I.e. $\dot{x} = -Lx + d$ vs $\dot{x} = -LRx + d$ are not equivalent

Control Objective for Rigid Body Transformations

- Now, our goal is to stabilize to a formation in $Rgd(x^*)$
  - Using only *local* measurements in the *local* reference frame
  - Rigidity is important here
  - Control law needs to be invariant under $Rgd(x^*)$ to account for robots' individual reference frames
  - I.e., $\dot{x} = f(x)$ leads to the same result as $\dot{x} = f(Rx + d)$

Stress Function

- Stress function of a graph is

$$Stress(G) = \frac{1}{2} \sum_{(i,j)\in E} \left(\|p_i - p_j\| - d_{ij} \right)^2$$

- For an undirected graph, the gradient of the stress creates feedback law

$$\dot{p}_i = 2 \sum_{j:(i,j)\in E} \left(\|p_i - p_j\| - d_{ij} \right) \frac{p_i - p_j}{\|p_i - p_j\|}$$

Have we seen this before

- Yes! Gradient view of consensus!

The Control Law

$$\dot{p}_i = 2 \sum_{j:(i,j)\in E} \left(\left\| p_i - p_j \right\|^2 - d_{ij} \right) \frac{p_i - p_j}{\left\| p_i - p_j \right\|}$$

Absolute difference in distance

Direction of error

- Homework 2 will ask you to verify some invariance properties of systems like this

The Control Law

- Again, the centroid is invariant
- Control law is independent of global coordinates
- For arbitrary initial conditions, solutions exist and are unique
- But what are solutions?

Equilibria

$$\dot{p}_i = 2 \sum_{j:(i,j)\in E} \left(\|p_i - p_j\| - d_{ij}\right) \frac{p_i - p_j}{\|p_i - p_j\|}$$

Equilibria

- Assumption:
  - Given target formation $\{G, d\}$, assume that $Rgd^{-1}(d) \neq 0$ and framework is infinitesimally rigid at each $p \in Rgd^{-1}(d)$

- Then solutions of the form $\|p_i - p_j\| = d_{ij}$ are *locally* asymptotically stable

Other Approaches

- Many works use common heading consensus (i.e., combine consensus on heading with this type of control law)

Distance- and Bearing-Only Formation Control

- Several researchers in the 2010s worked on bearing- or distance-only formation control
  - (as opposed to relative position formations)

- Major difference is the use of different rigidity measures
  - Harder to generalize for $n > 3$
  - Goal configuration must satisfy certain rigidity properties that we will not discuss in much detail

Wrap Up

Recap

- Formations
  - Scale invariance
  - Translation invariance
  - Rigidity
- Moving a group of agents
  - Leader-follower formations
  - Abstraction control
- Reference-frame invariant control

Next Time:

- Putting It all Together
  - Combining modes for controlling abstractions without a global reference frame
- SE(N) invariance for generalized pairwise interactions
- Next Tuesday: Deployment and Coverage