Solution of a Distributed Linear System Stabilisation Problem

NICTA Linear Systems Workshop

ANU

Brian Anderson (ANU/NICTA)
with contributions from:
Brad Yu, Baris Fidan, Soura Dasgupta, Steve Morse

Overview and Key Contribution

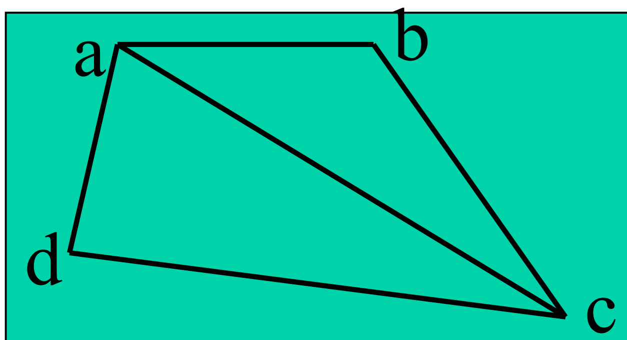
- In linear algebra terms, we consider the problem: given a real square matrix A, when and how can we find a real diagonal matrix Λ such that ΛA is stable (eigenvalues with negative real part)
- We explain the origins of the problem: distributed formation control.

Outline

- **Formations and Rigidity**
- Controlling a formation for rigidity: architecture
- Controlling a formation for rigidity: Equations of motion
- Stabilising a matrix by diagonal multiplication.
- Conclusions and questions

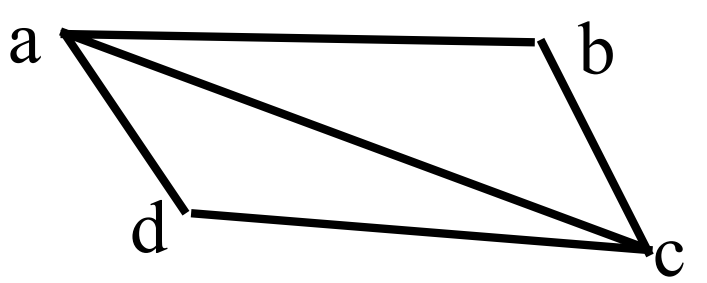
Rigid Formations

- A formation is a collection of agents (point agents for us) in two or three dimensional space
- A formation is *rigid* if the distance between each pair of agents does not change over time
- Rigidity can be secured by maintaining the distance between a subset of the agent pairs
- In a rigid formation, normally only ***some distances*** are explicitly maintained, with the rest being consequentially maintained.

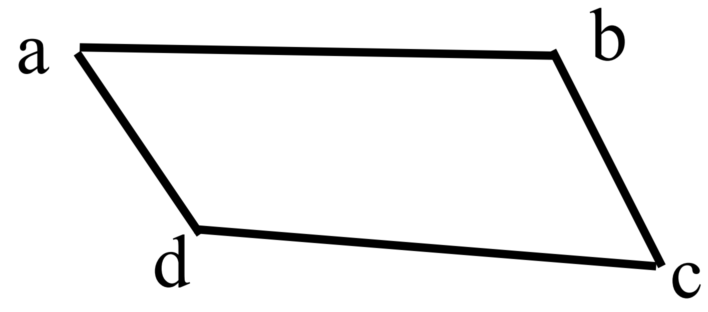


The distances ab,bc,cd,ad and ac are explicitly maintained and the distance bd is maintained as a result of the topology.

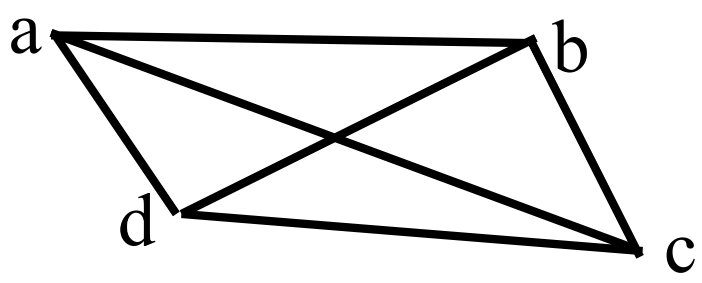
Rigid Formations



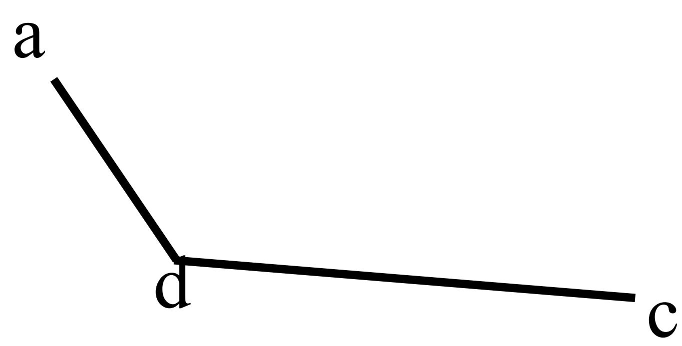
MINIMALLY RIGID



NONRIGID



RIGID, BUT NOT MINIMALLY RIGID



NONRIGID

Rigid Formations

- Consider a point formation $F = (\{p_1, p_2, \ldots p_n\}, L)$ with m maintenance links defined by $(i,j) \in L$, moving along a trajectory $p(t)$ with each distance $d_{ij} = || p_i - p_j ||$ constant. Along such a trajectory:

$$(p_i - p_j)^T \frac{d}{dt}(p_i - p_j) = 0 \ \ \forall \ (i,j) \in L$$

Rigid Formations

- Consider a point formation $F = (\{p_1, p_2, \ldots p_n\}, L)$ with m maintenance links defined by $(i,j) \in L$, moving along a trajectory $p(t)$ with each distance $d_{ij} = || p_i - p_j ||$ constant. Along such a trajectory:

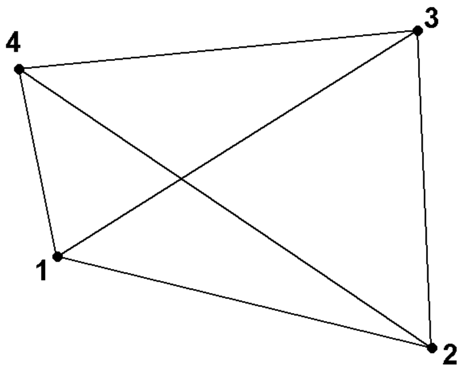
$$(p_i - p_j)^T \frac{d}{dt}(p_i - p_j) = 0 \;\; \forall \; (i,j) \in L$$

- These scalar equations may be gathered as:

$$R(p)\frac{dp}{dt} = 0$$

where $R(p)$ is $m \times 2n$

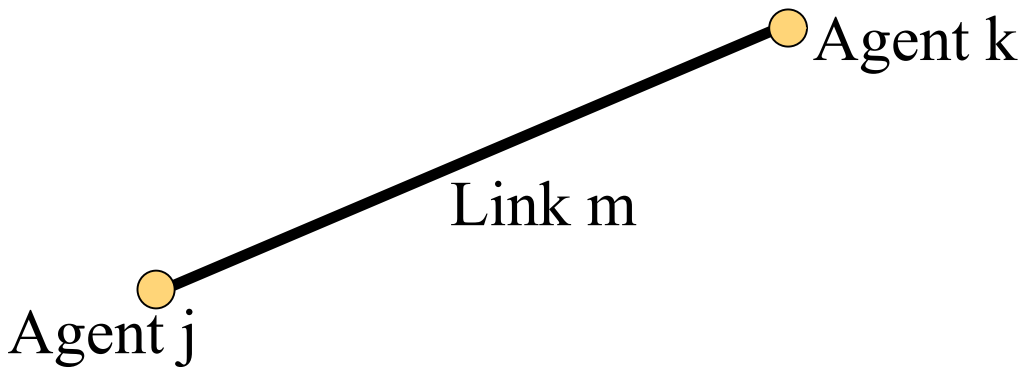
Rigid Formations



Sample two dimensional Rigidity Matrix--a ***Matrix Net*** $\sum x_i M_i + y_i N_i$ in coordinates of points.

	v_1	v_2	v_3	v_4
$(1,2)$	$x_1 - x_2 \quad y_1 - y_2$	$x_2 - x_1 \quad y_2 - y_1$	0	0
$(1,3)$	$x_1 - x_3 \quad y_1 - y_3$	0	$x_3 - x_1 \quad y_3 - y_1$	0
$(1,4)$	$x_1 - x_4 \quad y_1 - y_4$	0	0	$x_4 - x_1 \quad y_4 - y_1$
$(2,3)$	0	$x_2 - x_3 \quad y_2 - y_3$	$x_3 - x_2 \quad y_3 - y_2$	0
$(2,4)$	0	$x_2 - x_4 \quad y_2 - y_4$	0	$x_4 - x_2 \quad y_4 - y_2$
$(3,4)$	0	0	$x_3 - x_4 \quad y_3 - y_4$	$x_4 - x_3 \quad y_4 - y_3$

General form of Rigidity Matrix



$$\begin{array}{rc} \text{row } m & \begin{bmatrix} & \vdots & & \vdots & \\ \cdots & (p_j - p_k)^T & \cdots & (p_k - p_j)^T & \cdots \\ & \vdots & & \vdots & \end{bmatrix} \\ & \begin{array}{cc} \text{block column } j & \text{block column } k \end{array} \end{array}$$

Rigid Formations

In a **rigid** two-dimensional formation, the only motions possible are translation (2 directions) and rotation. Hence nullspace of $R(p)$ has dimension 3.

- $R(p)$ has $2n$ columns

Theorem: Assume F is a two-dimensional formation with $n \geq 3$ points. F is rigid iff

$$\text{rank } R(p) = 2n - 3$$

Note that R has the same rank for all p except a set of measure zero. So almost all formations with the same graph are either rigid or not rigid. We can speak of the graph being *generically rigid* or not.

Outline

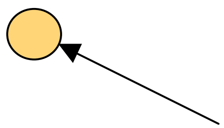
- Formations and Rigidity
- **Controlling a formation for rigidity: architecture**
- Controlling a formation for rigidity: Equations of motion
- Stabilising a matrix by diagonal multiplication.
- Conclusions and questions

Controlling for Rigidity

- To maintain a distance between two agents, either:
  - The task can be one for which both agents are responsible, or
  - One agent of the pair only can be given the task of control (which would seem to be more efficient)
- Control schemes for the two types turn out to be very different
- One-way control is effectively new, and will be addressed here.
- Our algorithms *sense* relative position of neighbours, and *control* distance

Leader First-Follower Formations

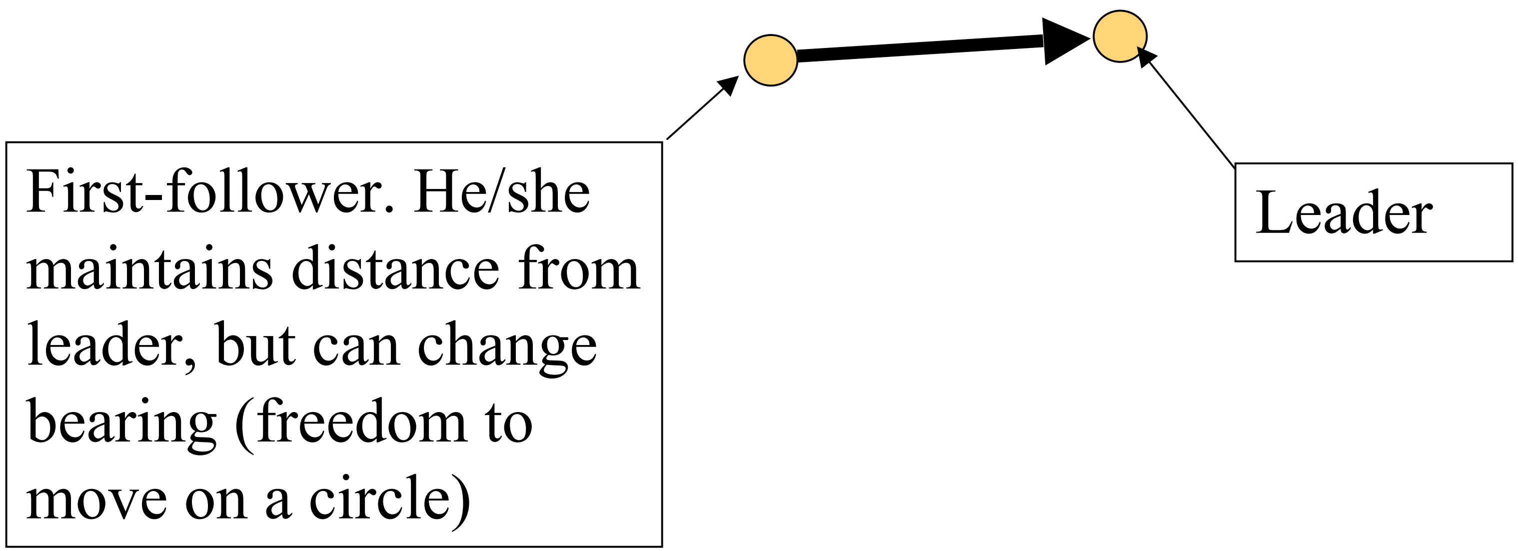
- We shall consider
  - Formations with n agents and the minimum number of links for maintaining rigidity--which is 2n-3
  - Formations with a *leader first-follower structure*



Leader: he/she can go anywhere, and does not have to look at anyone.

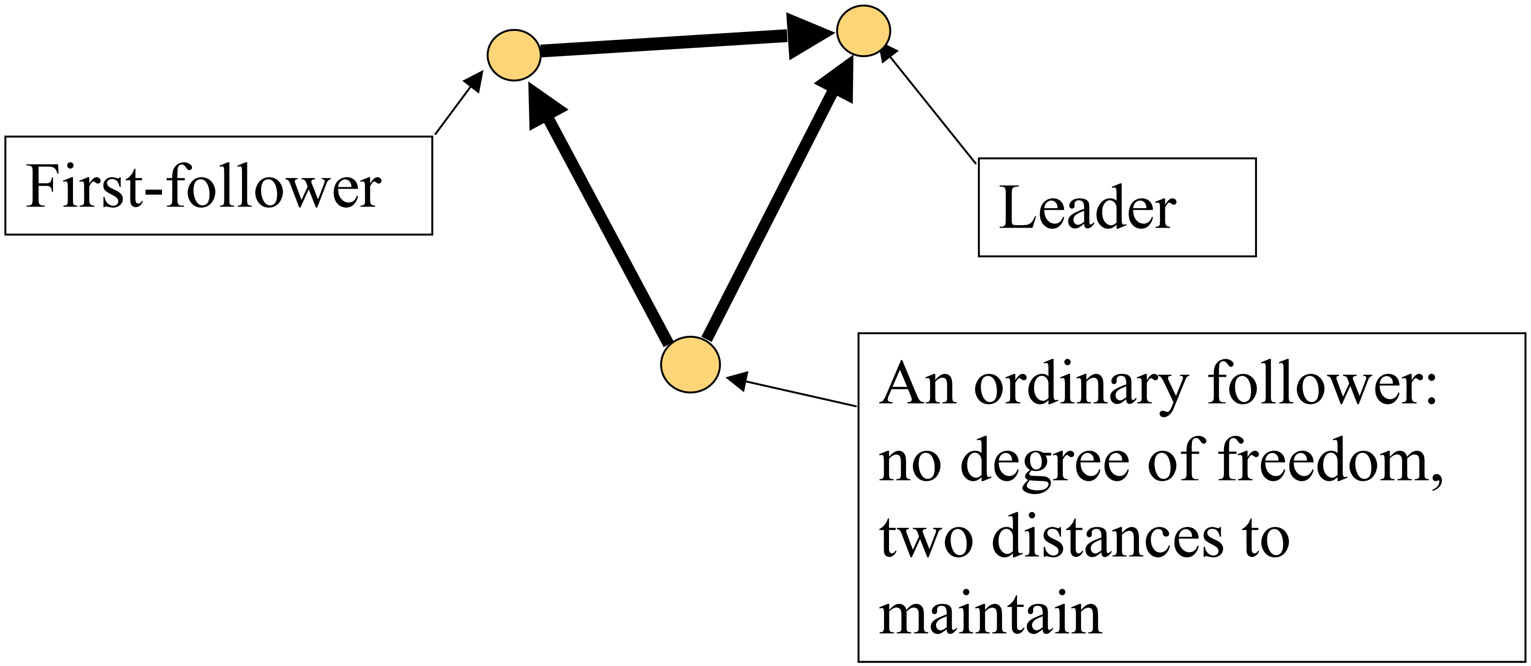
Leader First-Follower Formations

- We shall consider
  - Formations with n agents and the minimum number of links for maintaining rigidity--which is 2n-3
  - Formations with a *leader first-follower structure*



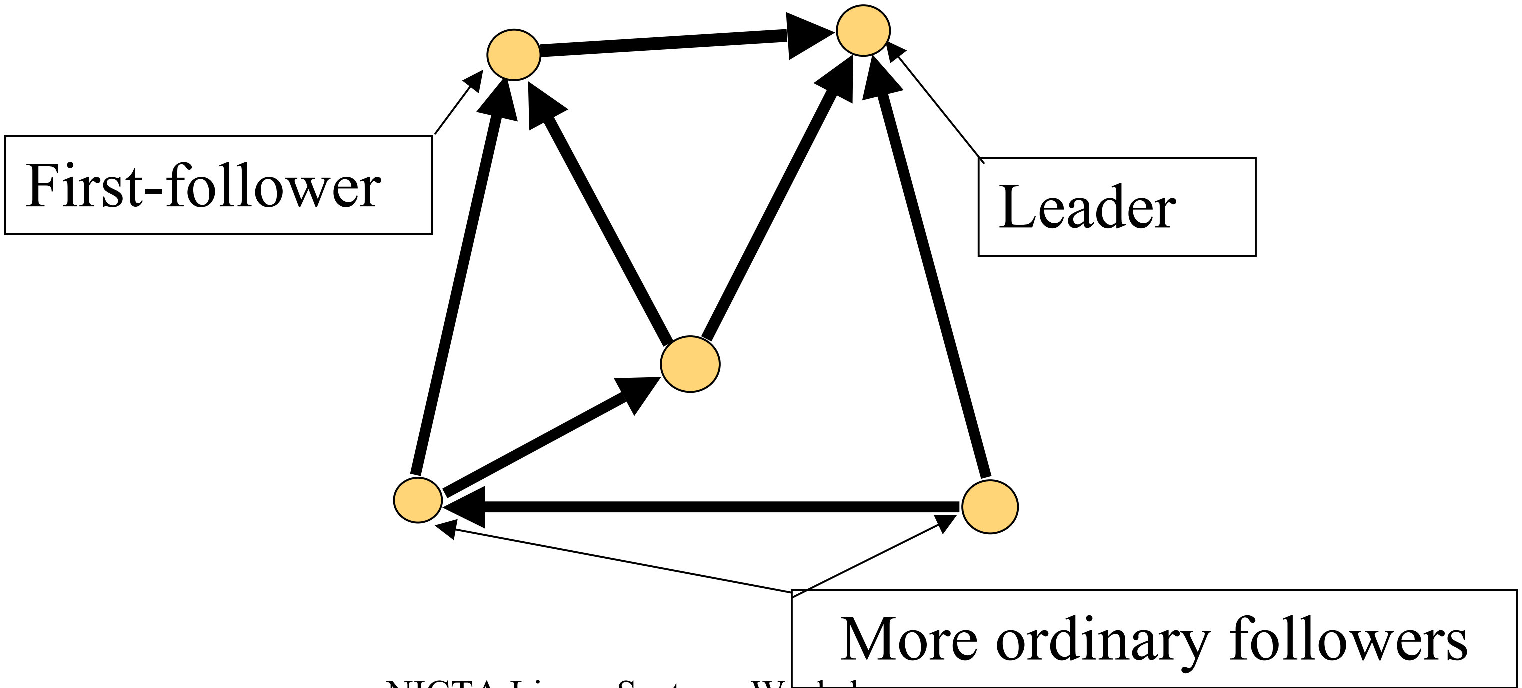
Leader First-Follower Formations

- We shall consider
  - Formations with n agents and the minimum number of links for maintaining rigidity--which is 2n-3
  - Formations with a *leader first-follower structure*

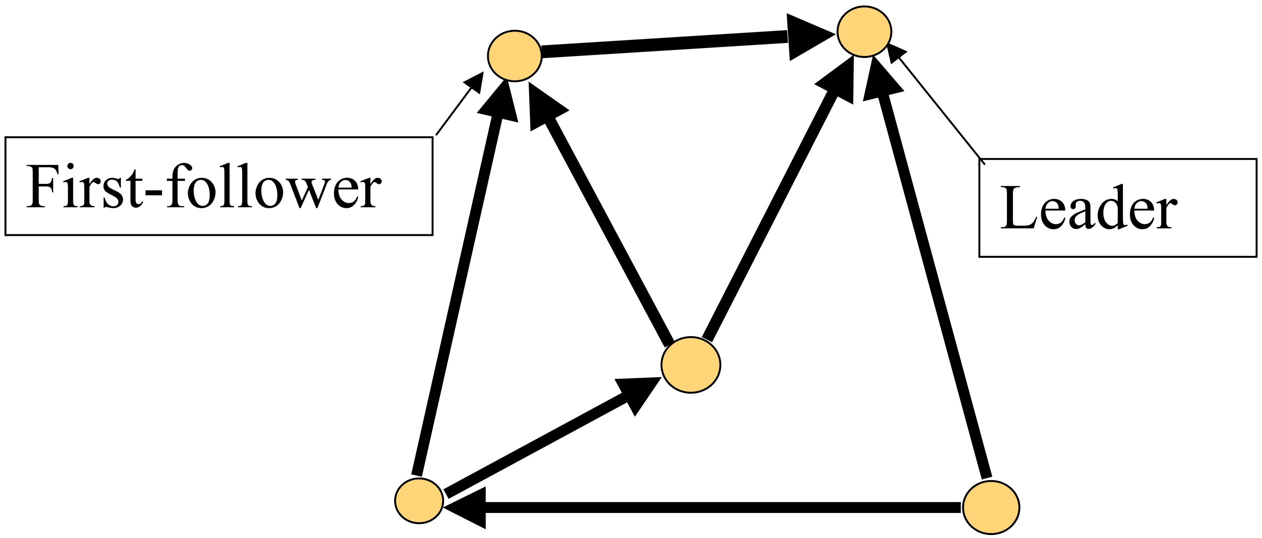


Leader First-Follower Formations

- We shall consider
  - Formations with n agents and the minimum number of links for maintaining rigidity--which is 2n-3
  - Formations with a *leader first-follower structure*



Leader First-Follower Formations

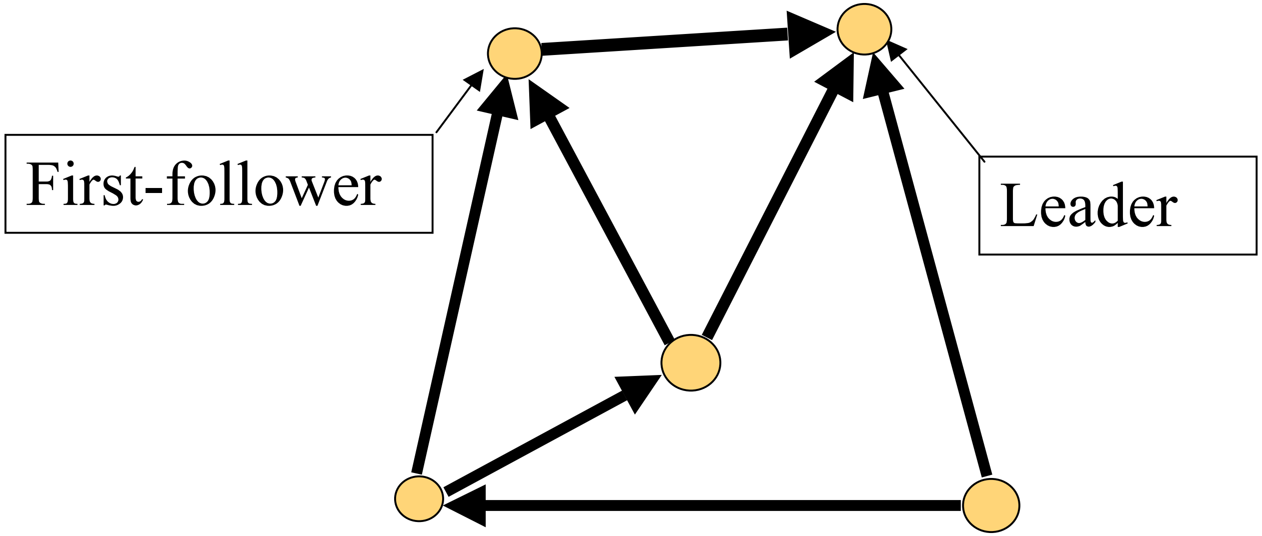


Above figure represents formation by a **directed graph**

The leader and first follower give translational and rotational freedom to the formation; all other agents must be ordinary followers, with two constraints, to ensure rigidity

Any such digraph yields a **minimally rigid formation** -- check 2n-3 edges.

Leader First-Follower Formations



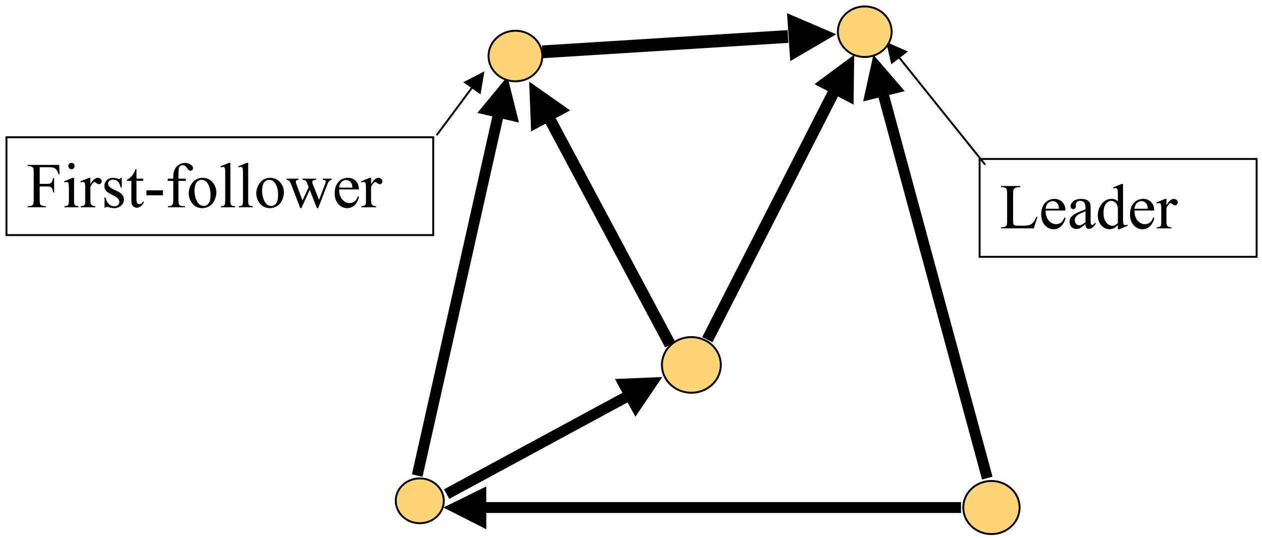
The above digraph is acyclic.

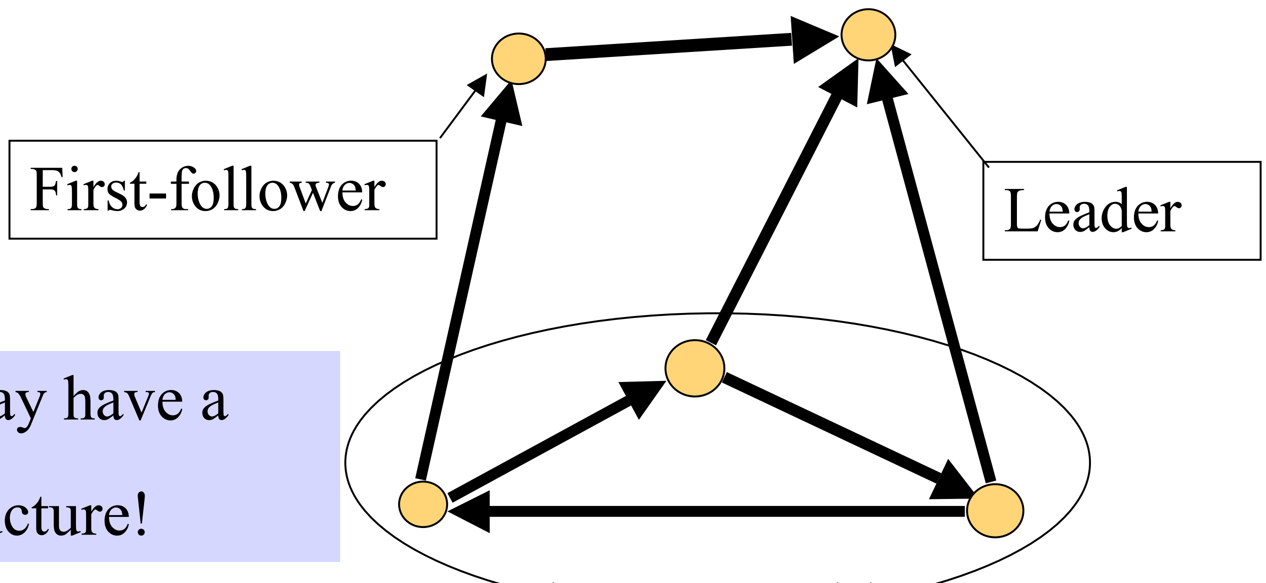
However, we can contemplate graphs with cycles

Intuitively, one can see that acyclic formation rigidity control is easy:

- Acyclicity induces an order.
- Design *local* controllers in order, getting triangularly coupled system

Leader First-Follower Formations





But we may have a cyclic structure!

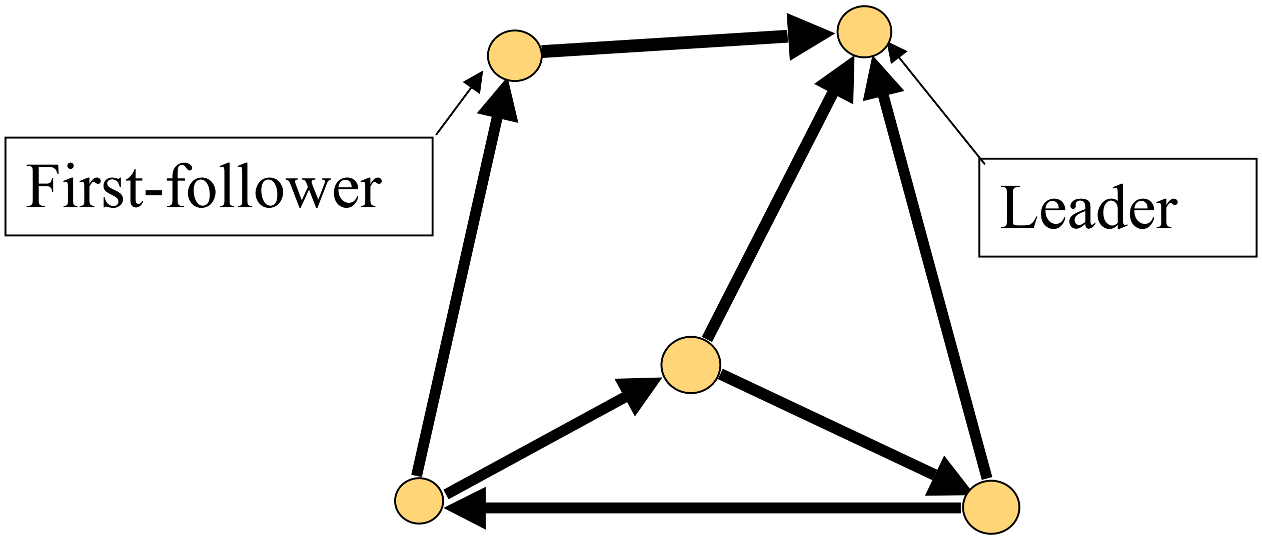
Leader First-Follower Formations

Rigidity control of structure with cycle is certainly harder.

There is a clear feedback mechanism, and so worry about instability.

Our work has focussed on this case.

We shall provide a small-signal (linearised system) analysis.



Outline

- Formations and Rigidity
- Controlling a formation for rigidity: architecture
- **Controlling a formation for rigidity: Equations of motion**
- Stabilising a matrix by diagonal multiplication.
- Conclusions and questions

Getting to equations

- Suppose that the formation is correctly located before time 0, with agent j at position p_{0j}.
- At time 0, we find that all agents are displaced from their correct positions. Agent j is displaced by δp_j (which is a 2-vector).
- We separate gross motion of the formation (translation and rotation) from correction of its shape. For shape correction,
  - The leader will not move
  - The first follower will correct his/her distance from the leader, but otherwise not move
  - The other followers will seek to correct their distances.

Motion of ‘other’ followers

- Suppose agent j has to maintain distance from agents k and m.
- It looks at these agents and, noting their present position (displaced from the nominal by δp_k and δp_m), figures where it should have to move in order to restore the distances to the correct value.This needs *relative positions*.
- This position will be displaced from the nominal because agents k and m are displaced from the nominal. Identify this target position displacement from the nominal as $\delta^* p_j$, and note that this is a function of δp_k and δp_m.
- Agent j then moves to reduce the distance from this target position, *assuming p_k and p_m do not move*.

Motion of ‘other’ followers

- This position will be displaced from the nominal because agents k and m are displaced from the nominal. Identify this target position displacement from the nominal as $\delta^* p_j$, and note that this is a function of δp_k and δp_m.
- Agent j then moves to reduce the distance from this target position, *assuming p_k and p_m do not move* :

$$\dot{\delta p_j} = A_j(\delta^* p_j(\delta p_k, \delta p_m) - \delta p_j)$$

Here, to get the distance reduction property, we need

$$A_j + A_j^T > 0$$

Motion of ‘other’ followers

$$\delta \dot{p}_j = A_j(\delta^* p_j(\delta p_k, \delta p_m) - \delta p_j)$$

$$A_j + A_j^T > 0$$

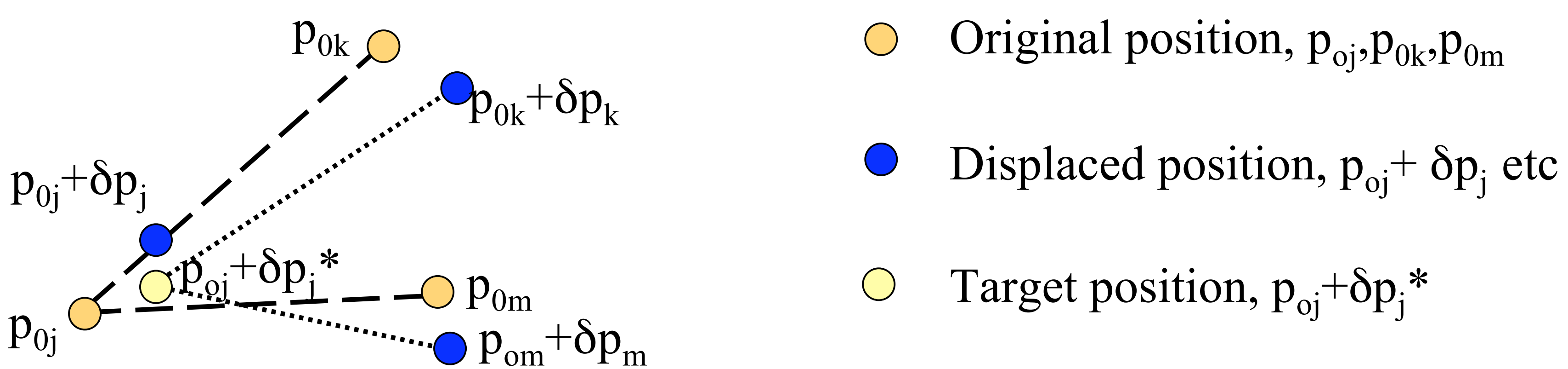
We need to properly evaluate $\delta^* p_j(\delta p_k, \delta p_m)$

as a linear expression in $\delta p_k, \delta p_m$

Motion of 'other' followers

We need to properly evaluate $\delta^* p_j(\delta p_k, \delta p_m)$

as a linear expression in $\delta p_k, \delta p_m$



$$||p_{0j} - p_{0m}||^2 = ||[p_{0j} + \delta p_j^*] - [p_{0m} + \delta p_m]||^2$$
$$||p_{0j} - p_{0k}||^2 = ||[p_{0j} + \delta p_j^*] - [p_{0k} + \delta p_k]||^2$$

$$[p_{0j} - p_{0m}]^T[\delta p_j^* - \delta p_m] = 0 \quad [p_{0j} - p_{0k}]^T[\delta p_j^* - \delta p_k] = 0$$

Motion of 'other' followers

$$[p_{0j} - p_{0m}]^T[\delta p_j^* - \delta p_m] = 0 \quad [p_{0j} - p_{0k}]^T[\delta p_j^* - \delta p_k] = 0$$

$$\delta p_j^* = \begin{bmatrix} (p_{0j} - p_{0m})^T \\ (p_{0j} - p_{0k})^T \end{bmatrix}^{-1} \begin{bmatrix} (p_{0j} - p_{0m})^T & 0_{1\times 2} \\ 0_{1\times 2} & (p_{0j} - p_{0k})^T \end{bmatrix} \begin{bmatrix} \delta p_m \\ \delta p_k \end{bmatrix}$$

Since

$$\delta \dot{p}_j = A_j(\delta^* p_j(\delta p_k, \delta p_m) - \delta p_j)$$

there follows

$$\delta \dot{p}_j = -A_j \begin{bmatrix} (p_{0j} - p_{0m})^T \\ (p_{0j} - p_{0k})^T \end{bmatrix}^{-1} \begin{bmatrix} (p_{0j} - p_{0m})^T & (p_{0m} - p_{0j})^T & 0_{1\times 2} \\ (p_{0j} - p_{0k})^T & 0_{1\times 2} & (p_{0k} - p_{0j})^T \end{bmatrix} \begin{bmatrix} \delta p_j \\ \delta p_m \\ \delta p_k \end{bmatrix}$$

Motion of ‘other’ followers

$$\dot{\delta p_j} = -A_j \underbrace{\begin{bmatrix} (p_{0j}-p_{0m})^T \\ (p_{0j}-p_{0k})^T \end{bmatrix}^{-1}} \begin{bmatrix} (p_{0j}-p_{0m})^T & (p_{0m}-p_{0j})^T & 0_{1\times 2} \\ (p_{0j}-p_{0k})^T & 0_{1\times 2} & (p_{0k}-p_{0j})^T \end{bmatrix} \begin{bmatrix} \delta p_j \\ \delta p_m \\ \delta p_k \end{bmatrix}$$

With A_j, this is an adjustable diagonal matrix

This 2 by 6 matrix is a submatrix of the rigidity matrix.

$$\begin{array}{ll} \text{row m} \rightarrow & \begin{bmatrix} & \vdots & & \vdots & \\ \cdots & (p_j - p_k)^T & \cdots & (p_k - p_j)^T & \cdots \\ & \vdots & & \vdots & \end{bmatrix} \end{array}$$

block column j corresponding to agent j

block column k corresponding to agent k

Motion of First Follower

- First follower needs to correct its distance from leader
- Impose the additional requirement that it never moves on a line orthogonal to the line joining it to the leader.
- With p_j and p_k denoting the coordinates of the first follower and leader, we obtain:

$$[p_{0j} - p_{0k}]^T [\delta p_j^* - \delta p_k] = 0$$

$$\begin{bmatrix} -y_{0j} + y_{0k} & x_{0j} - x_{0k} \end{bmatrix} [\delta p_j^* - \delta p_j] = 0$$

There results:

$$\dot{\delta p_j} = -A_j \begin{bmatrix} x_{0j} - x_{0k} & y_{0j} - y_{0k} \\ -y_{0j} + y_{0k} & x_{0j} - x_{0k} \end{bmatrix}^{-1} \begin{bmatrix} (p_{0j} - p_{0k})^T & -(p_{0j} - p_{0k})^T \\ 0_{2\times 1} & 0_{2\times 1} \end{bmatrix} \begin{bmatrix} \delta p_j \\ \delta p_k \end{bmatrix}$$

Motion of FF and leader

$$\dot{\delta p_j} = -A_j \underbrace{\begin{bmatrix} x_{0j} - x_{0k} & y_{0j} - y_{0k} \\ -y_{0j} + y_{0k} & x_{0j} - x_{0k} \end{bmatrix}}_{\text{Adjustable 2 by 2 matrix}}^{-1} \begin{bmatrix} (p_{0j} - p_{0k})^T & -(p_{0j} - p_{0k})^T \\ 0_{2\times 1} & 0_{2\times 1} \end{bmatrix} \begin{bmatrix} \delta p_j \\ \delta p_k \end{bmatrix}$$

Again part of rigidity matrix plus zero row

- For the purposes of controlling formation shape, the leader (agent k say) does not move:

$$\dot{\delta p_k} = 0$$

Formation motion

- Number the agents from 1 to n, with the first follower and leader as agents (n-1) and n
- Number the edges so that edges (2j-1) and 2j are *out-edges* of vertex j, for j = 1,2,…(n-2). Edge 2n-3 goes from first follower to leader.
- Putting the various equations together, we get:

$$\frac{d}{dt}\begin{bmatrix} \delta p_1 \\ \delta p_2 \\ \vdots \\ \delta p_{n-1} \\ \delta p_n \end{bmatrix} = \Lambda \begin{bmatrix} R \\ 0_{3\times 2n} \end{bmatrix} \begin{bmatrix} \delta p_1 \\ \delta p_2 \\ \vdots \\ \delta p_{n-1} \\ \delta p_n \end{bmatrix}$$

R is the rigidity matrix and Λ is block 2×2 and adjustable.

Formation motion

$$\frac{d}{dt}\begin{bmatrix} \delta p_1 \\ \delta p_2 \\ \vdots \\ \delta p_{n-1} \\ \delta p_n \end{bmatrix} = \Lambda \begin{bmatrix} R \\ 0_{3\times 2n} \end{bmatrix} \begin{bmatrix} \delta p_1 \\ \delta p_2 \\ \vdots \\ \delta p_{n-1} \\ \delta p_n \end{bmatrix}$$

R is the rigidity matrix and Λ is block 2×2 and adjustable.

- Three eigenvalues at origin corresponds to no motion of the leader, and motion of first follower restricted to occurring on line joining first follower to leader
- Choose coordinates so that y coordinate axis is on line perpendicular to that joining first follower to leader.
- Then we can drop some terms

Formation motion

$$\frac{d}{dt}\begin{bmatrix} \delta\xi_1 \\ \delta\xi_2 \\ \vdots \\ \delta\xi_{2n-3} \end{bmatrix} = \hat{\Lambda}\hat{R}\begin{bmatrix} \delta\xi_1 \\ \delta\xi_2 \\ \vdots \\ \delta\xi_{2n-3} \end{bmatrix} + \text{constant}$$

First (2n-3) rows and columns of Λ. Still block diagonal and adjustable

First (2n-3) columns of R

Coordinates of p_i are ξ_{2i-1}, ξ_{2i}.

The constant excitation comes from the initial and thereafter constant displacment of the leader and one coordinate of the first follower

Formation motion: one last thing

$$\frac{d}{dt}\begin{bmatrix}\delta\xi_1\\ \delta\xi_2\\ \vdots\\ \delta\xi_{2n-3}\end{bmatrix} = \hat{\Lambda}\hat{R}\begin{bmatrix}\delta\xi_1\\ \delta\xi_2\\ \vdots\\ \delta\xi_{2n-3}\end{bmatrix} + \text{constant}$$

First (2n-3) rows and columns of Λ. Still block diagonal and adjustable

First (2n-3) columns of R

The special assignations of edge ordering and vertex ordering ensure that generically, all leading principal minors are nonzero. (Tricky graph theory result)

Key question: Can we stabilize through choice of the (block) diagonal matrix?

Nontriviality

- The problem is nontrivial.
- It is easy to find examples where choice of 'obvious' diagonal matrix (based on simplest local control law below) leads to instability.

$$\dot{\delta p_j} = (\delta^* p_j(p_k, p_m) - \delta p_j)$$

Outline

- Formations and Rigidity
- Controlling a formation for rigidity: architecture
- Controlling a formation for rigidity: Equations of motion
- **Stabilising a matrix by diagonal multiplication.**
- Conclusions and questions

Linear Algebra Problem

- In linear algebra terms, we consider the problem: given a real square matrix A, when and how can we find a real diagonal matrix Λ such that ΛA is stable (eigenvalues with negative real part)
- Note that we have not exploited fact that Λ can be taken as *block* diagonal.
- Problem has been considered for **complex** A and Λ and a condition is known such that one can assign eigenvalues (Friedland 1975)
- Eigenvalue assignment problem is an example of *inverse eigenvalue problem;* far more results exist for complex case.

Linear Algebra Problem

- In linear algebra terms, we consider the problem: given a real square matrix A, when and how can we find a real diagonal matrix Λ such that ΛA is stable (eigenvalues with negative real part)
- We cannot hope for an eigenvalue assignment result. Consider

$$A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

- Through real diagonal scaling one can never get a matrix with a complex eigenvalue.
- Same is true if 0 is replaced by small nonzero value.

Linear Algebra Problem

- In linear algebra terms, we consider the problem: given a real square matrix A, when and how can we find a real diagonal matrix Λ such that ΛA is stable (eigenvalues with negative real part)
- If all k by k principal minors of an n by n matrix are zero, the coefficient of s^{n-k} in the characteristic polynomial is zero. Diagonal scaling cannot change a zero principal minor into a nonzero one. Therefore a necessary condition for stabilisability is that there exists at least one k by k principal minor which is nonzero
- Friedman result on eigenvalue assignment for complex case is that assignment is achievable when all leading principal minors are nonzero.
- **Our result for the real case is that stabilizability is achievable, again when all leading principal minors are nonzero.**
  - **Note that this property holds generically as a result of graph theory considerations!**

Outline of proof

Proof is by induction. Base step is trivial. Suppose that an (r-1) by (r-1) real matrix with nonzero leading principal minors can be stabilised by real diagonal multiplication. Let A be r by r with nonzero leading principal minors.

$$A = \begin{bmatrix} A_{11} & a_{12} \\ a_{21}^T & a_{22} \end{bmatrix}$$

Choose diagonal Λ_1 such that $\Lambda_1 A_{11}$ has all eigenvalues with negative real parts (using induction). Now consider the DE

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \epsilon^{-1}\Lambda_1 A_{11} x_1 + \epsilon^{-1}\Lambda_1 a_{12} x_2 \\ \lambda_2 a_{21}^T x_1 + \lambda_2 a_{22} x_2 \end{bmatrix}$$

This is stable for suitably small ε iff (singular perturbation theory)

$$\lambda_2 [a_{22} - a_{21}^T A_{11}^{-1} a_{12}] < 0$$

Leading principal minor condition assures λ_2 exists.

Outline

- Formations and Rigidity
- Controlling a formation for rigidity: architecture
- Controlling a formation for rigidity: Equations of motion
- Stabilising a matrix by diagonal multiplication.
- **Conclusions and questions**

Conclusions and Questions

- We have demonstrated an example of a nontrivial decentralized control problem, with a nontrivial solution and an associated nontrivial problem of linear algebra.
- Associated linear algebra problems:
  - What more can be done utilizing a block diagonal structure for Λ?
  - What is the set of achievable eigenvalues, or what can be said about the set? Could one achieve arbitrary real eigenvalues?
  - When can one ensure that the eigenvalues are **not** widely dispersed?

Conclusions and Questions

- Associated control problems:
  - Can one characterize the set of linear stabilising laws?
  - Can one characterize the set of nonlinear stabilising laws?
  - Can one work with large perturbations?
  - Can one work with other formation leader structures?
  - Can one gracefully integrate formation shape control and formation motion control to an objective?
  - What formations will be hard to control (i.e. need large signals or suffer badly from noise)?
  - Can one generalize from point agents?
  - Can one handle three dimensional problems?
  - Can one use graph theory properties to simplify, e.g. decomposition of a directed graph into cyclic parts?

Questions?