Applications
- Consensus
  - formation control
  - distributed estimation and computation
  - sensor networks
  - distributed aperture imaging
  - animal flocking.

Topics
- Introduction to graph theory
  - Adjacency and Laplacian matrices
- Consensus protocol
  - Spectral graph theory and convergence analysis

Lecture based off:

https://murray.cds.caltech.edu/images/murray.cds/1/1e/Eeci-sp09_L4_graphtheory.pdf

ONLINE NOTES: Please add numerical / code examples of all of the above. We can make this lecture into a case study, but keep things on the smaller & more manageable side throughout, and then end with a larger example. For the case study, if you are able, can also add theorems for directed graphs and highlight oscillations & what can go wrong if there are cycles, be sure to connect spectrum of $L$ to qualitative behavior of consensus protocol.

# Introduction to Graph Theory and Consensus

Many applications in engineering involve coordinating groups of agents to behave in a desireable way. Examples abound in a wide range of areas:

- Transportation: air traffic control and intelligent transportation systems
- Military: distributed aperture imaging and battlespace management
- Scientific: animal coordination and group opinion dynamics (e.g. on social networks)
- Sensor networks: adaptive ocean sampling, building sensor networks in green buildings
- General networks: communication, power, and supply chain networks

Many, but not all, of these problems can be posed as consensus problems. In this case, we assume there are $n$ agents, with each agent state $x_i(t) \in \mathbb{R}$ evolving according to the following differential equation:

$$\dot{x}_i = f_i(x_i) + \sum_{j \in N_i} u_i(x_i, x_j), \quad i = 1, \ldots, n \quad (*)$$

In equation $(*)$:

- the function $f_i : \mathbb{R} \to \mathbb{R}$ defines the dynamics of agent $i$, i.e., how the internal state of agent $i$ evolves in the absence of other agents.
- the neighbor set $N_i$ of agent $i$ are those agents that are directly connected to agent $i$, as specified by a communication graph $G$ with nodes $i = 1, \ldots, n$ corresponding to agents, and edges defining inter-agent communication.
- the coordination rule $u_i : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, which we are to design so that the collective state of the agents $\underline{x}(t) = (x_1(t), \ldots, x_n(t))$ converges to a desired goal state $\underline{x}^*$.

In many settings of interest, each agent is a computer that is trying to compute or estimate a shared value. This is a common model adopted in distributed optimization and learning (want to compute a shared prediction), sensor networks (compute a shared estimate), and opinion dynamics (compute shared opinion). In these settings, it makes sense to set the local dynamics function $f_i(x) = 0$, so that an agent's state is determined solely by interactions with its neighbors. Agent $i$'s dynamics then become

$$\dot{x}_i = \sum_{j \in N_i} u_i(x_i, x_j), \quad i = 1, \ldots, n \quad (**)$$

A common goal for such systems is for all agents to converge to the same value, i.e., that eventually, $x_1 = x_2 = \cdots = x_n = \eta$, for $\eta$ some value. If this happens, system $(**)$ is said to achieve consensus with consensus value $\eta$.

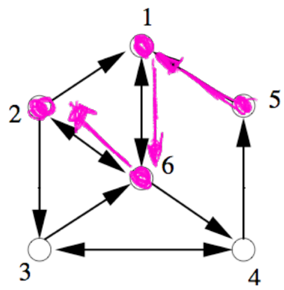
Studying the behavior of system $(**)$ will require us to bring together all of the ideas that we've seen over the past few lectures on eigenvalues, dynamical systems, and spectral decompositions of symmetric matrices. We start with a summary of relevant tools from graph theory, which will help us model the flow of information in $(**)$.

# A Primer on Graph Theory

We have used graphs before, but we pause now to describe formal linear algebraic representations thereof.

We begin with the basic definition of a graph, which is defined as a pair $G=(\mathcal{V},\mathcal{E})$ that consists of a set of vertices $\mathcal{V}$ and a set of edges $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$. A vertex $v_i\in\mathcal{V}$ is a node in the graph, and an edge $e_{ij}=(v_i,v_j)\in\mathcal{E}$ is an arc connecting node $i$ to node $j$ (note sometimes this convention is reversed).

Example:



$$\mathcal{V}=\{1,2,3,4,5,6\}$$
$$\mathcal{E}=\{(1,6),(2,1),(2,3),(2,6),(6,2),(3,4),(3,6),(4,3),(4,5),(5,1),(6,1),(6,2),(6,4)\}$$

Path $\pi$ with $\mathcal{V}_\pi=\{v_5,v_1,v_2,v_6\}$ and $\mathcal{E}_\pi=\{(v_5,v_1),(v_1,v_2),(v_2,v_6)\}$ (see def'n below)

Some important notation and terminology for graphs include:

- the order of a graph is the number of nodes $|\mathcal{V}|$
- Nodes $v_i$ and $v_j$ are adjacent if there exists $e=(v_i,v_j)\in\mathcal{E}$ (i.e., if node $i$ is directly connected to node $j$ via an edge $e$).
- An adjacent node $v_j$ for a node $v_i$ is called a neighbor of $v_i$.
- The set of all neighbors of $v_i$ is denoted $\mathcal{N}_i$
- A graph $G$ is called complete if all nodes are adjacent.

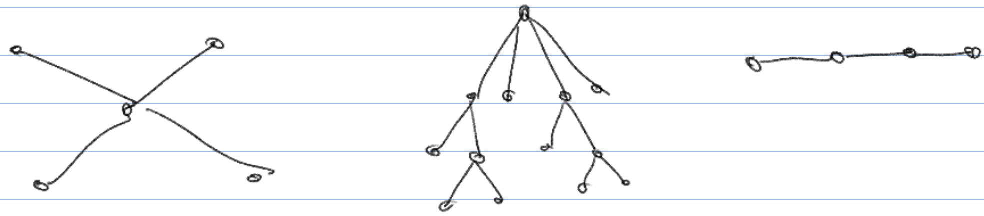
We will focus our study on the class of undirected graphs, that is graphs satisfying the property that if $e_{ij}\in\mathcal{E}$ then $e_{ji}\in\mathcal{E}$. In words, this means that all edges are bidirectional: if $v_j$ is a neighbor of $v_i$, then $v_i$ is a neighbor of $v_j$. Although we won't focus on them, a graph that is not undirected is called directed.

A fundamental question associated with the study of the consensus system (~~(4)~~) is under what properties of information sharing between agents can we guarantee that consensus is achieved. Intuitively, if each agent talks to "enough" other agents, we expect consensus to happen. This intuition can be formalized through the notion of graph connectivity.

## Connectedness of Graphs

We start by defining a path in $G$, which is a subgraph $\pi=(\mathcal{V}_\pi,\mathcal{E}_\pi)$, with distinct nodes $\mathcal{V}_\pi=\{v_1,v_2,\dots,v_m\}$ and $\mathcal{E}_\pi=\{(v_1,v_2),(v_2,v_3),\dots,(v_{m-1},v_m)\}$. The length of a path $\pi$ is defined as $|\mathcal{E}_\pi|=m-1$. For example, in the graph above, the path $\pi$ defined by $\mathcal{V}_\pi=\{v_5,v_1,v_6,v_2\}$ and $\mathcal{E}_\pi=((v_5,v_1),(v_1,v_6),(v_6,v_2))$ is the path that goes from node 5 to node 1 to node 6 to node 2. It has length 3, the number of edges traversed. An undirected graph $G$ is called connected if there exists a path $\pi$ between any two distinct nodes of $G$.

Example: These graphs are connected (all edges are bidirectional):



This graph, composed of two disconnected subgraphs, is not:

## Matrices Associated with a Graph

In our study of network flow problems, we encountered the incidence matrix associated with a graph. This is but one matrix we can associate with a graph $G$, and while the incidence matrix is convenient for encoding flow conservation, we will see that the Graph Laplacian Matrix is more natural when defining consensus algorithms. We begin by defining the adjacency matrix $A \in \mathbb{B}^{n \times n}$ of graph $G$ of order $n$ by:

$$a_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in \mathcal{E}, \quad i,j = 1, \dots, n. \\ 0 & \text{otherwise} \end{cases}$$

The degree matrix $\Delta$ of a graph $G$ is a diagonal $n \times n$ matrix with diagonal elements $\Delta_{ii}$ specified by the number of edges leaving node $i$, also called the out-degree of $v_i$, which we'll denote by $\text{out}(v_i)$:

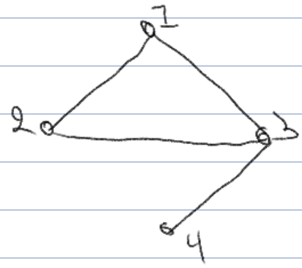
$$\Delta_{ii} = \text{out}(v_i), \quad i = 1, \dots, n.$$

The Laplacian matrix $L$ of a graph is defined as $L = \Delta - A$. An important property of the Laplacian is that its rows all sum to zero, and that if a graph is undirected, then its adjacency matrix and its Laplacian are both symmetric.

Example: the adjacency matrix and Laplacian for the directed graph above are:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \qquad L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & 0 & 0 & -1 \\ 0 & 0 & 2 & -1 & 0 & -1 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 \\ -1 & -1 & 0 & -1 & 0 & 3 \end{bmatrix}$$

The adjacency and Laplacian for the undirected graph



are $A = \begin{array}{c|c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \hline \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$ and $L = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$.

## Consensus Protocols

Consider a collection of $N$ agents that communicate along a set of undirected links described by a graph $G$. Each agent has state $x_i(t) \in \mathbb{R}$, with initial value $x_i(0)$, and together, they wish to determine the average of the initial states $\text{avg}(\underline{x}(0)) = \frac{1}{N}\sum_{i=1}^{N} x_i(0)$.

The agents implement the following consensus protocol:

$$\dot{x}_i = \sum_{j \in N_i} (x_j - x_i) = -|N_i|(x_i - \text{avg}(X_{N_i})),$$

where $\text{avg}(X_{N_i}) = \frac{1}{|N_i|}\sum_{j \in N_i} x_j$ is the average of the states of the neighbors of agent $i$. This equivalent to the first order homogeneous linear ordinary differential equation:

$$\dot{\underline{x}} = -L\underline{x}. \quad \text{(AVG)}$$

Based on our previous analysis of such systems, we know that the solution to (AVG) is given by

$$\underline{x}(t) = c_1 e^{\lambda_1 t}\underline{v}_1 + \cdots + c_n e^{\lambda_n t} v_n, \qquad \underline{x}(0) = [\underline{v}_1 \cdots \underline{v}_n]\underline{c}. \quad \text{(SOL)}$$

where $(\lambda_i, \underline{v}_i)$, $i = 1, \ldots, n$, are the eigenvalue/eigenvector pairs of the negative graph Laplacian $-L$. Thus, the behavior of the consensus system (AVG) is determined by the spectrum of $L$. We will spend the rest of today's lecture on understanding the following theorem:

**Theorem:** If the graph $G$ defining the consensus system (AVG) is connected, then the state of the agents converges to $x_i^* = \text{avg}(\underline{x}(0))$ exponentially quickly.

This result is extremely intuitive! It says that so long as the information at one node can eventually reach every other node in the graph, then we can achieve consensus via the protocol (AVG). Let's try to understand why. As in last class we order the eigenvalues of $-L$ in descending order: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.

Our first observation is that $\lambda = 0$, $\underline{v} = \underline{1}$ is an eigenvalue/eigenvector pair for $-L$. This follows from the fact that each row of $L$ sums to 0, and so:

$$-L\,\underline{1} = \underline{0} = 0\,\underline{1}.$$

A fact that we'll show is true later is that the eigenvalues of $L$ are all nonnegative, and thus we have that $\lambda_i \leq 0$ for the eigenvalues of $-L$. As such, we know that $\lambda = 0$ is a largest eigenvalue of $-L$: hence we label them $\lambda_1 = 0$, $\underline{v}_1 = \underline{1}$.

Next, we recall that for an undirected graph, the Laplacian $L$ is symmetric, and hence is diagonalized by an orthonormal eigenbasis $-L = Q\Lambda Q^T$, where $Q = [\underline{u}_1 \cdots \underline{u}_n]$ is an orthogonal matrix composed of orthonormal eigenvectors of $L$, and $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Although we do not know $\underline{u}_2, \ldots, \underline{u}_n$, we know that $\underline{u}_1 = \frac{\underline{v}_1}{\|\underline{v}_1\|} = \frac{1}{\sqrt{N}}\underline{1}$.

We can therefore rewrite (SOL) as:

$$\underline{x}(t) = c_1 e^{0t}\frac{1}{\sqrt{N}}\underline{1} + c_2 e^{\lambda_1 t}\underline{u}_1 + \cdots + c_n e^{\lambda_n t}\underline{u}_n$$

$$= c_1 \frac{1}{\sqrt{N}}\underline{1} + c_2 e^{\lambda_1 t}\underline{u}_1 + \cdots + c_n e^{\lambda_n t}\underline{u}_n \qquad (*)$$

where now we can compute $\underline{c}$ by solving $\underline{x}(0) = Q\underline{c} \Rightarrow \underline{c} = Q^T\underline{x}(0)$, as $Q$ is an orthogonal matrix

Let's focus on computing $c_1$:

$$c_1 = \underline{u}_1^T\underline{x}(0) = \frac{1}{\sqrt{N}}\underline{1}^T\underline{x}(0) = \frac{1}{\sqrt{N}}\sum_{i=1}^{N} x_i(0).$$

Plugging this back into $(*)$, we get:

$$\underline{x}(t) = \frac{1}{N}\sum_{i=1}^{N} x_i(0)\cdot\underline{1} + c_2 e^{\lambda_2 t}\underline{u}_2 + \cdots + c_n e^{\lambda_n t}\underline{u}_n$$

$$= \operatorname{avg}(\underline{x}(0))\underline{1} + c_2 e^{\lambda_2 t}\underline{u}_2 + \cdots + c_n e^{\lambda_n t}\underline{u}_n \qquad (**).$$

This is very exciting! We have shown that the solution $\underline{x}(t)$ to (AVG) is composed of a sum of the goal consensus state $\underline{x}^* = \operatorname{avg}(\underline{x}(0))\underline{1}$ and exponential functions $c_i e^{\lambda_i t}\underline{u}_i$, $i = 2, \ldots, n$, evolving in the subspace $\underline{u}_1^{\perp}$ orthogonal to the consensus direction $\frac{1}{\sqrt{N}}\underline{1}$. Thus, if we can show that $\lambda_2, \ldots, \lambda_n < 0$, we will have established our result.

To establish this result, we start by stating a widely used theorem for bounding localizing eigenvalues.

---

Theorem (Gershgorin's Disk Theorem): Let $A \in \mathbb{R}^{n \times n}$, and define the radius

$$r_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|$$

as the absolute row sum with entry $a_{ii}$ deleted. Then all eigenvalues of $A$ are located in the union of $n$ disks:

$$G(A) = \bigcup_{i=1}^{n} G_i(A), \quad G_i(A) = \{ z \in \mathbb{C} \mid |z - a_{ii}| \leq r_i \}$$

In the case of symmetric matrices, we can restrict the $G_i(A)$ to the real line:

$$G_i(A) = \{ \lambda \in \mathbb{R} \mid |\lambda - a_{ii}| \leq r_i \}$$

---

Example: Consider $A = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$. Gershgorin's disk theorem tells us that the eigenvalues $\lambda_1$ and $\lambda_2$ are contained within the set

$$G(A) = \{ \lambda \in \mathbb{R} \mid |\lambda - 3| \leq 1 \},$$

or equivalently that $2 \leq \lambda_2 \leq \lambda_1 \leq 4$. As we've computed in previous examples, $\lambda_1 = 4$ and $\lambda_2 = 2$, which indeed do lie within $G(A)$.

Let's apply this theorem to a graph Laplacian $L$. The diagonal elements of $L = \Delta - A$ are given by $\Delta_{ii} = out(v_i)$, the out-degree of node $i$. Further, the radii $r_i = out(v_i)$ as well, as $a_{ij} = 1$ if node $i$ is connected to node $j$, and $0$ otherwise. Therefore, for row $i$, we have the following Gershgorin intervals:

$$G_i(L) = \{ \lambda \in \mathbb{R} \mid |\lambda - out(v_i)| \leq out(v_i) \}.$$

These are intervals of the form $[0, 2\,out(v_i)]$, and therefore the union $G(L) = \bigcup_{i=1}^{n} G_i(L) = [0, 2d_{max}]$, where $d_{max} = \max_i out(v_i)$ is the maximal out degree of a node in the graph. Taking the negative of everything, we conclude that $G_i(-L) = [-2d_{max}, 0]$.

This tells us that $\lambda_i \leq 0$ for $i = 1, 2, \ldots, n$, for the eigenvalues of $-L$. This is almost what we wanted. We still need to show that only $\lambda_1 = 0$, and that $\lambda_n \leq \cdots \leq \lambda_2 < 0$. To answer this question, we rely on the following proposition:

**Proposition:** The algebraic multiplicity of the 0 eigenvalue of a graph Laplacian $L$ is equal to the number of connected components in the graph. In particular, if the graph $G$ is connected, then only $\lambda_1 = 0$, and $\lambda_n \leq \cdots \lambda_2 < \lambda_1 = 0$.

Unfortunately proving this result would take us too far astray. Instead, we highlight the intuitive nature of the result in terms of the consensus system (AVG). This proposition tells us that if the communication graph $G$ is strongly connected, i.e., if everyone's information eventually reaches everyone, then $\underline{x}(t) \to \underline{x}^* = \text{avg}(\underline{x}(0))$ at a rate governed by the slowest decaying mode $e^{\lambda_2 t}$.

In contrast, suppose the graph $G$ is disconnected, and consists of the disjoint union of two connected graphs $G_1 = (V_1, \mathcal{E}_1)$ and $G_2 = (V_2, \mathcal{E}_2)$, i.e., $G = (V_1 \cup V_2, \mathcal{E}_1 \cup \mathcal{E}_2)$ and $V_1 \cap V_2 = \emptyset$ and $\mathcal{E}_1 \cap \mathcal{E}_2 = \emptyset$. Then if we run the consensus protocol (AVG) on $G$, the system effectively decouples into two parallel systems, each evolving on their own graph and blissfully unaware of the other:

$$\dot{\underline{x}}_1 = -L_1 \underline{x}_1 \quad \text{and} \quad \dot{\underline{x}}_2 = -L_2 \underline{x}_2. \quad (*)$$

Here we use $\underline{x}_1$ to denote the state of agents in $G_1$, with Laplacian $L_1$, and similarly for $\underline{x}_2$. By the above discussion, if $L_1$ and $L_2$ are both strongly connected, then $\underline{x}_i(t) \to \underline{x}_i^* = \text{avg}(\underline{x}_i(0))\,\underline{1}$, and $\lambda = 0, \underline{v} = \underline{1}$ is an eigenvalue/vector pair for each graph.

If we now consider the joint graph $G$, composed of the two disjoint graphs $G_1$ and $G_2$, we don't expect behavior to change: each consensus protocol $\dot{\underline{x}}_i = -L_i \underline{x}_i$ will evolve as it did before.

To see how this manifests in the algebraic multiplicity of the 0 eigenvalue of $L = \begin{bmatrix} L_1 & \\ & L_2 \end{bmatrix}$, note that for the composite system with state $\underline{x} = \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix}$, we have the consensus dynamics:

$$\begin{bmatrix} \dot{\underline{x}}_1 \\ \dot{\underline{x}}_2 \end{bmatrix} = \begin{bmatrix} -L_1 & \\ & -L_2 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix},$$

which has $\lambda_1 = 0$ with $\underline{v}_1 = \begin{bmatrix} \underline{1} \\ \underline{0} \end{bmatrix}$ and $\lambda_2 = 0$ with $\underline{v}_2 = \begin{bmatrix} 0 \\ \underline{1} \end{bmatrix}$ so that

$$\begin{bmatrix} \underline{x}_1^* \\ \underline{x}_2^* \end{bmatrix} = \begin{bmatrix} \underline{1} \\ 0 \end{bmatrix} \text{avg}(\underline{x}_1(0)) + \begin{bmatrix} 0 \\ \underline{1} \end{bmatrix} \text{avg}(\underline{x}_2(0)).$$

This is of course expected, as all we have done is rewrite $(*)$ using block vectors and matrices — we have not changed anything about the consensus protocol.