Section 3
Consensus Control

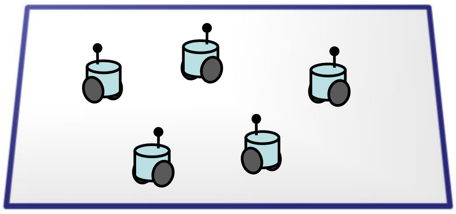
Shun-ichi Azuma
Nagoya University
www.ctrl.mae.nagoya-u.ac.jp

Outline

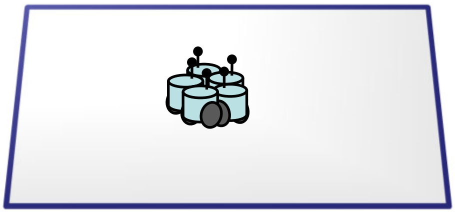
What is Consensus?

- **What is consensus?**
  - Consider a multi-agent system with n agents.
  - Each agent is a **dynamical system**, i.e., which has a state.
  - Each agent can **obtain information on other agents** and it updates its state.
  - **Consensus**: All the agents **reach an agreement** about their states.

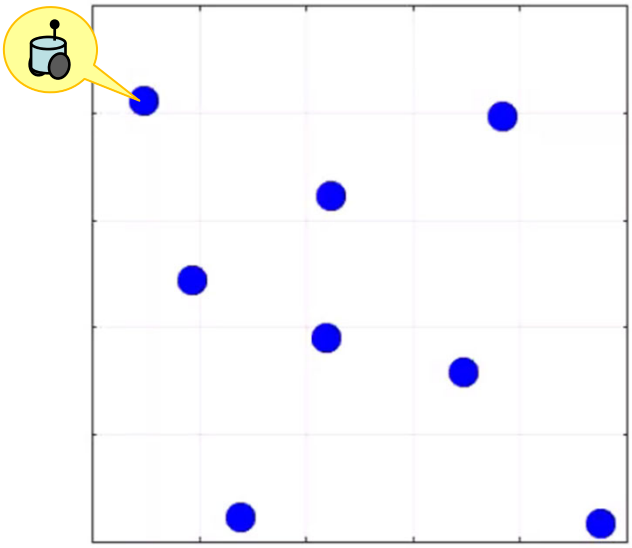
If the states are the positions ...



after a while

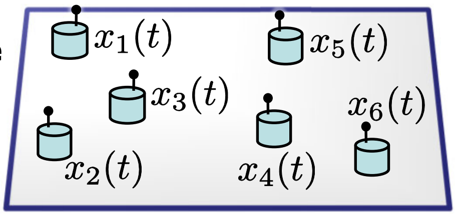


- **Consensus in 2D State Space**

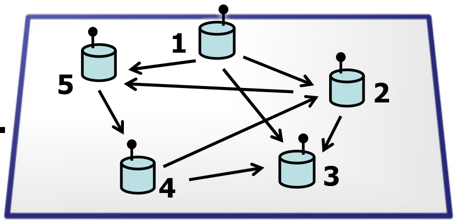


Multi-agent Systems to Be Studied

- **Agents**
  - Consider n agents, which are called agent i (i=1,2,…,n)
  - Agent i is described by
    $\dot{x}_i(t) = f_i(x_i(t), u_i(t))$
    where
    - $x_i(t) \in \mathrm{R}^m$ is the state,
    - $u_i(t) \in \mathrm{R}^l$ is the input,
    - f_i is a function.



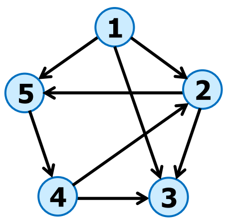
- **Neighbors**
  - Each agnet can obtain the information of some other agents to determine its input.
  - An agent about which agent i has the information is called a neighbor of agent i.
  - $\mathcal{N}_i$: Set of the neighbors of agent i



$\mathcal{N}_1 = \emptyset$ $\quad \mathcal{N}_2 = \{1,4\}$
$\mathcal{N}_3 = \{1,2,4\}$ $\quad \mathcal{N}_4 = \{5\}$
$\mathcal{N}_5 = \{1,2\}$

- **Network**
  - The resulting information flow is represented by a network, which is expressed by the graph $G = (\mathcal{V}, \mathcal{E})$ where

$$\mathcal{V} := \{1, 2, \ldots, n\}$$
$$\mathcal{E} := \{(j,i) \in \mathcal{V} \times \mathcal{V} : j \in \mathcal{N}_i\}$$



Consensus

- **Definition of Consensus**

Definition

If the system satisfies

$$\lim_{t\to\infty}(x_i(t)-x_j(t))=0 \quad \forall(i,j)\in\mathcal{V}\times\mathcal{V}$$

for every $(x_1(0),x_2(0),\ldots,x_n(0))\in\mathrm{R}^{mn}$,
then we call that the system achieves consensus.

  - Consensus means that $x_1(t)=x_2(t)=\cdots=x_n(t)$ asymptotically holds.

- **Consensus Value**
  - If the system achieves consensus and $\lim_{t\to\infty}x_i(t)=\alpha$ for some i, α is called the consensus value (or agreement).

- **A Variety of Consensus Values**

(1) Average Consensus: $\alpha=\frac{1}{n}\sum_{i=1}^{n}x_i(0)$

(2) Geometric-mean Consensus: $\alpha=\left(\prod_{i=1}^{n}x_i(0)\right)^{\frac{1}{n}}$

(3) Maximum Consensus: $\alpha=\max_{i\in\{1,2,\ldots,n\}}x_i(0)$

(4) Minimum Consensus: $\alpha=\min_{i\in\{1,2,\ldots,n\}}x_i(0)$

(5) Leader-follower Consensus : $\alpha=x_l(0)$
(when agent l is the leader)

Consensus Problem

- **Consensus Problem**
  - For the multi-agent system, find $u_i \; (i = 1, 2, \ldots, n)$ such that **the system achieves consensus** (or consensus with a specific consensus value).

Outline

3.1 Consensus Problem

3.2 Consensus Controllers

3.3 Analysis of Consensus Control

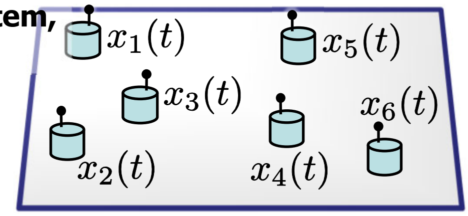
3.4 Python Implementation

Consensus Controllers

- **Multi-agent System to Be Considered**
  - **Consider a multi-agent system, where agent i is given by**

    $\dot{x}_i(t) = u_i(t)$

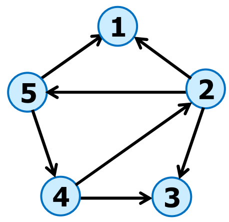
    for the **scalar** $x_i(t) \in \mathrm{R}$ **.**



  - $\mathcal{N}_i$ **: Set of the neighbors of agent i**
  - $G = (\mathcal{V}, \mathcal{E})$ **: Network**

- **Consensus Controllers**
  - $u_i(t) = \sum_{j \in \mathcal{N}_i} (x_j(t) - x_i(t))$

- **Example of** $u_i(t) = \sum_{j \in \mathcal{N}_i} (x_j(t) - x_i(t))$



$$u_1(t) = \sum_{j \in \mathcal{N}_1} (x_j(t) - x_1(t))$$

$$= (x_2(t) - x_1(t)) + (x_5(t) - x_1(t))$$

$$u_2(t) = x_4(t) - x_2(t)$$

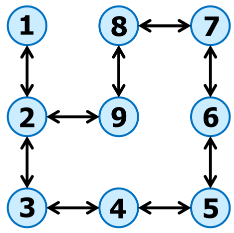
$$u_3(t) = (x_2(t) - x_3(t)) + (x_4(t) - x_3(t))$$

$$u_4(t) = x_5(t) - x_4(t)$$

$$u_5(t) = x_2(t) - x_5(t)$$

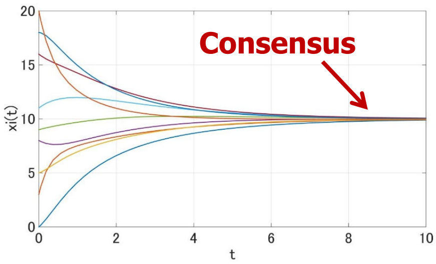
Performance of Consensus Controllers

- Consider the multi-agent system with the right network and the corresponding consensus controller.



- Time response of the states for the initial states

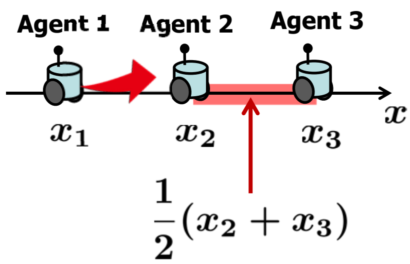
$$x_1(0)=0,\ x_2(0)=3$$
$$x_3(0)=5,\ x_4(0)=8$$
$$x_5(0)=9,\ x_6(0)=11$$
$$x_7(0)=16,\ x_8(0)=18$$
$$x_9(0)=20$$



Idea of Consensus Controllers

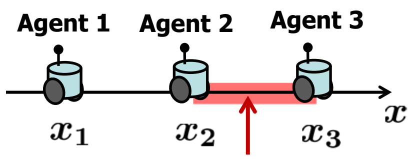
Consensus by 3 agents

- Consider 3 agents on the line, where their states represent the positions.
- How should each agent move for consensus?



- Agent 1 goes to the region between agents 2 and 3.
- Let the representative point of the region be the average of the positions of the two agents, i.e., $\frac{1}{2}(x_2 + x_3)$
- In the local coordinate of Agent 1, it is $\frac{1}{2}(x_2 + x_3) - x_1$

- In the local coordinate of Agent 1,



$$\frac{1}{2}(x_2 + x_3) - x_1$$
$$= \frac{1}{2}\left((x_2 - x_1) + (x_3 - x_1)\right) = \frac{1}{2}\sum_{j\in\{2,3\}}(x_j - x_1)$$

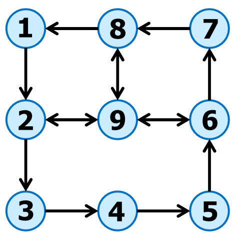
- The consensus controller

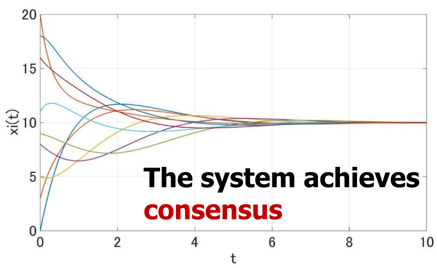
$$u_1(t) = \sum_{j\in\mathcal{N}_1}(x_j(t) - x_1(t))$$

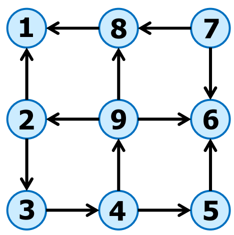
moves agent 1 to the direction to the average position of agents 2 and 3.

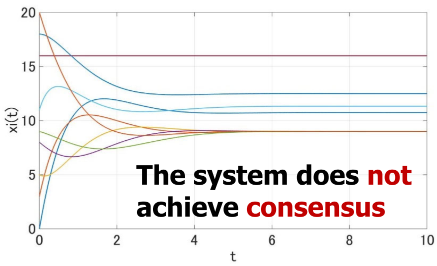
⇨ The consensus controller moves each agent to the direction to the average position of neighbors.

Two Cases for Consensus Controllers





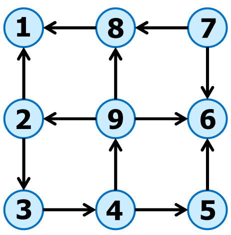


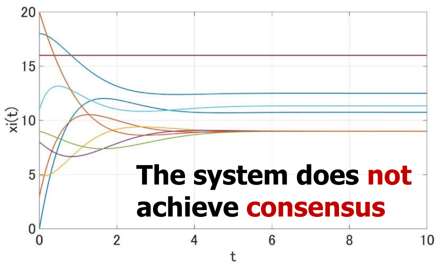


Outline

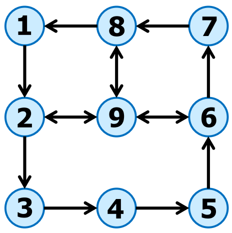
Two Basic Questions for Consensus Controllers

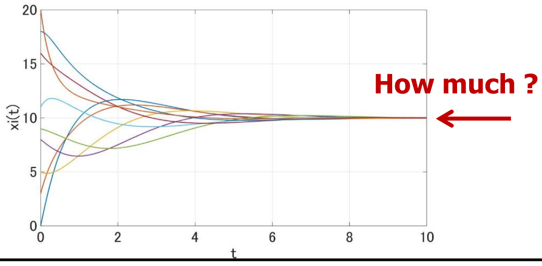
- **Q1: What system achieves consensus?**





- **Q2: If consensus, how much is the consensus value?**





- **How to answer them?**

(Step 1) Derive the differential equation representing the collective dynamics

$$\dot{x}(t) = F(x(t))$$

where $x(t) := [x_1(t)\ x_2(t)\ \cdots\ x_n(t)]^\top$

(Step 2) Analyze the solution of the differential equation.

$$\lim_{t\to\infty} x(t)\ ?$$

If the system achieves consensus, then

$$\lim_{t\to\infty} x(t) = \alpha \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

Derivation of Collective Dynamics

- **Dynamics of Agent i**
  - **Original dynamics :** $\dot{x}_i(t) = u_i(t)$
  - **Consensus Controller:** $u_i(t) = \sum_{j\in\mathcal{N}_i} (x_j(t) - x_i(t))$

⇨ **Controlled Dynamics:** $\dot{x}_i(t) = \sum_{j\in\mathcal{N}_i} (x_j(t) - x_i(t))$

  - **For the element** $a_{ij} \in \{0, 1\}$ **of the adjacency matrix of the graph G, we have** $a_{ij} = \begin{cases} 0 & \text{if } j \notin \mathcal{N}_i \\ 1 & \text{if } j \in \mathcal{N}_i \end{cases}$
  - **Therefore,** $\dot{x}_i(t) = \sum_{j=1}^{n} a_{ij}(x_j(t) - x_i(t))$

- **By dividing the right-hand side into the x_j term and x_i term , we have**

$$\begin{aligned}\dot{x}_i(t) &= \sum_{j\in\mathcal{N}_i} a_{ij}(x_j(t) - x_i(t)) \\ &= -\sum_{j\in\mathcal{N}_i} a_{ij}x_i(t) + \sum_{j\in\mathcal{N}_i} a_{ij}x_j(t)\end{aligned}$$

- **By considering it for all the agents, we eventually obtain**

$$\begin{cases} \dot{x}_1(t) = -\sum_{j=1}^n a_{1j}x_1(t) + \sum_{j=1}^n a_{1j}x_j(t) \\ \dot{x}_2(t) = -\sum_{j=1}^n a_{2j}x_2(t) + \sum_{j=1}^n a_{2j}x_j(t) \\ \qquad\vdots \\ \dot{x}_n(t) = -\sum_{j=1}^n a_{nj}x_n(t) + \sum_{j=1}^n a_{nj}x_j(t) \end{cases}$$

- **We rewrite it in a vector form....**

- **A vector form representation of collective dynamics**

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} = -\begin{bmatrix} \sum_{j=1}^n a_{1j} & 0 & \cdots & 0 \\ 0 & \sum_{j=1}^n a_{2j} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \sum_{j=1}^n a_{nj} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} + \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \cdots & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix}$$

Degree matrix D

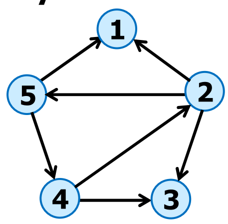
Adjacency matrix A

- **By letting** $x(t) := [x_1(t)\ x_2(t)\ \cdots\ x_n(t)]^\top \in \mathbf{R}^n$

$$\dot{x}(t) = -(D - A)x(t) \longrightarrow \dot{x}(t) = -Lx(t)$$

⇨ $\dot{x}(t) = -Lx(t)$

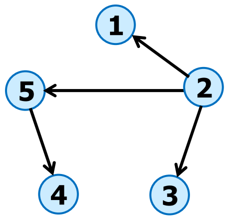
- **Collective Dynamics**
  - $\dot{x}(t) = -Lx(t)$ **: A linear system**
  - **The convergence property is characterized by the eigenvalues and eigenvectors of L.**
- **Review from Section 2**
  - **A spanning tree of a graph G: A subgraph of G which is a tree including all the vertices of G.**



  - **If L has a spanning tree,**

$$\lim_{t\to\infty} e^{-Lt} = \left(\frac{1}{v_1 1_n}\right) 1_n v_1$$

where v_1 is the left-eigenvector of L associated with zero eigenvalue.



- **Convergence Property of Collective Dynamics**
  - **The collective dynamics:** $\dot{x}(t) = -Lx(t)$
  - **The solution:** $x(t) = e^{-Lt}x(0)$
  - **If L has a spanning tree,**

$$\lim_{t\to\infty} x(t) = \left(\lim_{t\to\infty} e^{-Lt}\right) x(0)$$

$$= \left(\frac{1}{v_1 1_n}\right) 1_n \underbrace{v_1 x(0)}_{\text{scalar}} = \underbrace{\left(\frac{v_1 x(0)}{v_1 1_n}\right)}_{\text{scalar}} 1_n$$

⇨ **Consensus at** $\dfrac{v_1 x(0)}{v_1 1_n}$

- **Consensus Condition and Consensus Value**

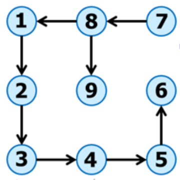
Theorem 1

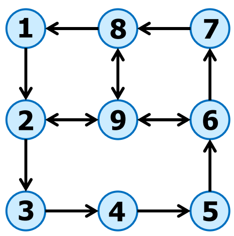
(i) The system achieves consensus iff G has a spanning tree.

(ii) If the system achieves consensus, the consensus value is $\alpha = \dfrac{v_1 x(0)}{v_1 \mathbf{1}_n}$.

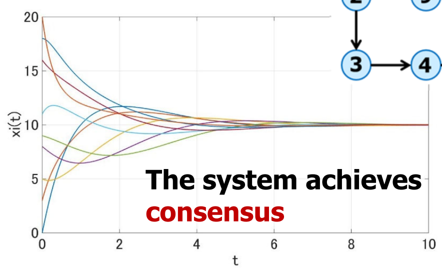
- **The sufficiency of (i) and (ii) are given by the discussion based on the case where G has a spanning tree.**
- **The necessity of (i) can be proven by showing that we find an initial state which does not result in consensus.**

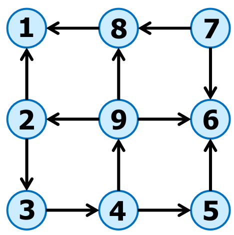
Example 1



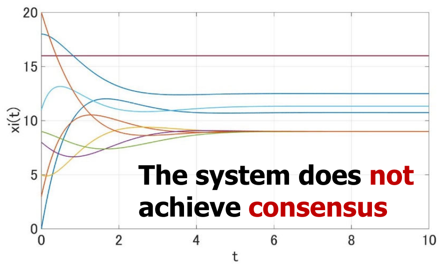


has a spanning tree



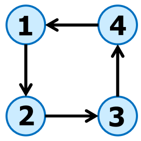


has no spanning tree



Example 2

- **Consider a multi-agent system with the network in the right figure, which achieves consensus.**



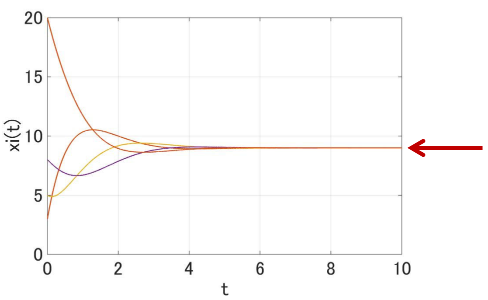
$x_1(0)=3,\ x_2(0)=5$
$x_3(0)=8,\ x_4(0)=20$

- **How much is the consensus value for the initial state?**
- **Graph Laplacian**

$$L = \begin{bmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

- **The left-eigenvector is** $v_1 = [1 \ \ 1 \ \ 1 \ \ 1]$.
- $\alpha = \dfrac{v_1 x(0)}{v_1 \mathbf{1}_n} = \dfrac{1\times 3 + 1\times 5 + 1\times 8 + 1\times 20}{1+1+1+1} = 9$

- **Time response for**
  $x_1(0)=3,\ x_2(0)=5$
  $x_3(0)=8,\ x_4(0)=20$
- **Consensus value = 9**



Remarks on An Undirected Network Case

- Assume that G is undirected.
- **The system achieves consensus iff G is connected**, because, in the undirected case, G has a spanning tree iff G is connected.
- **The consensus value is equal to the average of the initial states (average consensus)**, because $v_1 = 1_n^\top$, i.e.,

$$\alpha = \frac{v_1 x(0)}{v_1 1_n} = \frac{1_n^\top x(0)}{1_n^\top 1_n} = \frac{1}{n}\sum_{i=1}^{n} x_i(0)$$

Outline

3.1 Consensus Problem

3.2 Consensus Controllers

3.3 Analysis of Consensus Control

3.4 Python Implementation

Python Code

- **Sample code: 3_consensus.py**

  https://drive.google.com/drive/folders/1Y0Fu2b3tDSOXsP_YV5NpaGGPeujlrkf1?usp=sharing

Python Code

```
from scipy.integrate  import odeint
import numpy as np
import matplotlib.pyplot as plt
```

Part 1: Definition of a multi-agent system as an ODE

```
def MAS(x,t,N):

    dxdt = [0] * len(N)
    u = [0] * len(N)

    # Definition of agent i
    for i in range(len(N)):

        # Computation of the control input of agent i
        dif = []

        for j in N[i]:
            dif.append(x[j] - x[i])

        u[i] = sum(dif)

        # Dynamics of agent i
        dxdt[i] = u[i]

    return dxdt
```

$$u_i(t) = \sum_{j \in \mathcal{N}_i} (x_j(t) - x_i(t))$$

$$\dot{x}_i(t) = u_i(t)$$

Part 2: Solve the ODE

```
N = [[2], [3,5], [4], [1,2], [6], [2]]
x0 = [-1, 2, 6, 3, -3, 1]
t = np.arange(0, 5, 0.001)
N.insert(0,[])
x0.insert(0,0)
x = odeint(MAS, x0, t, args=(N,))
```

Part 3: Plot the result on a graph

```
plt.plot(t,np.delete(x, 0, 1))
plt.xlabel('t')
plt.ylabel('xi')
plt.grid()
plt.show()
```