Cooperative Control of Multi-Agent Systems

Hideaki Ishii
Dept. Computational Intelligence & Systems Science
ishii@dis.titech.ac.jp

Advanced Topics in Mathematical Information Sciences I
Jul 10th, 2015
Introduction

Control of multi-agent systems

- Active research in the area of systems control (2000~)
- Keywords: Distributed control/algorithms, Communication networks, Remote control over networks,…
Consensus problem

- One of the basic problems for multi-agent systems
- Initiated the research trend in this area
- Systems control approach: Theory-based with applications

In this lecture

- Basics of multi-agent consensus
What is consensus?

Flocks of fish/birds

Load balancing among servers

Formation of autonomous robots

Sensor networks
Example 1: Autonomous robots

- Cluster of small robots for planetary exploration
  - High flexibility and reliability at low cost
  - Communication is limited by on-board power

- Array antenna
  - Multiple antennas coupled for directed transmission
  - Formation of robots based on distributed control laws

![Diagram of robot formation](image)
Example 2: Sensor networks

- Spatially distributed autonomous sensors with wireless communication capability
- **Problem:** When each sensor measures an unknown parameter $+$ noise, want to find the average of all measurements.
Consensus problem

- Network of agents without a leader
- Each agent communicates with others and updates its state
- All agents should arrive at the same (unspecified) state

Achieve global objectives through local interaction!
Some history (1): Boids

- Flocking of birds: Formation flying without a leader
- What are the simple control laws for each bird?
- Simulation-based study by Raynolds

Three rules
- Separation
- Alignment
- Cohesion

Raynolds (1987)
Some history (2): Model by Vicek et al.

- Proposed a mathematical model of agents’ dynamics
  - Each agent moves on a plane at constant speed
  - Align with the directions of neighboring agents
- Flocking behavior was observed by simulation

Vicek et al. (1995)

Analytic results by Jadbabaie et al.

- Proved that all agents converge to the same direction if there is sufficient connectivity structure
  - Motivated control researchers to study multi-robot problems

Jadbabaie, Lin, & Morse (2003), Tsitsiklis & Bertsekas (1989)
Network of agents
Network of agents

Info can be sent from 4 to 2
Connectivity in multi-agent systems

- Represented as a graph
  - Node set \( \mathcal{V} = \{1, 2, \ldots, N\} \Rightarrow \) Indices for the agents
  - Edge set \( \mathcal{E} \subset \mathcal{V} \times \mathcal{V} \Rightarrow \) Communication among the agents

Info can be sent from 4 to 2
\( \Leftrightarrow (4, 2) \in \mathcal{E} \)
Connectivity in multi-agent systems

- Neighbor set $\mathcal{N}_i$ ⇒ Indices of agents that can send info to agent $i$

- Example: For agent 2

$$\mathcal{N}_2 = \{3, 4\} = \{j \in \mathcal{V} : (j, 2) \in \mathcal{E}\}$$
Basics of graphs

Types: Directed/Undirected

Nodes $i$ and $j$ are connected

$\Leftrightarrow$ Agent $j$ is reachable from $i$ by following edges

Graph is (strongly) connected

$\Leftrightarrow$ Any two nodes are connected
Protocol for distributed algorithms

- At time $k$, agent $i$ does the following:
  1. Sends its value $x_i(k)$ to the neighbor agents
  2. Updates its value based on the received info and obtains $x_i(k + 1)$
Average consensus

- **Problem:** Find a distributed algorithm satisfying the two conditions:

  1. All agents converge to the same value.
     
     \[ |x_i(k) - x_j(k)| \to 0, \quad k \to \infty, \quad \forall i, j = 1, 2, \ldots, N \]

  2. The value is the average of the initial values.
Algorithms in this lecture

Two classes of consensus problems

1. Real-valued
2. Integer-valued (Quantized)

- Algorithms may be deterministic or probabilistic

- Graph structure: Undirected and connected
Average consensus (1)

Real-valued case
Real-valued average consensus

- Each agent has a real value $x_i(k)$
- Average consensus

$$x_i(k) \rightarrow \frac{1}{N} \sum_{j=1}^{N} x_j(0), \quad k \rightarrow \infty, \quad \forall i = 1, 2, \ldots, N$$

Average of initial values

- Example

  Initial values 1 2 2

  ![Diagram](image)

  Ave = 1.666
Distributed algorithm

- Update scheme for agent $i$:

$$x_i(k + 1) = W_{ii}x_i(k) + \sum_{j \in \mathcal{N}_i} W_{ij}x_j(k)$$

where

$$W_{ij} = \begin{cases} 
\frac{1}{1 + \max\{d_i, d_j\}} & \text{if } j \in \mathcal{N}_i \\
1 - \sum_{\ell \in \mathcal{N}_i} W_{i\ell} & \text{if } i = j \\
0 & \text{Otherwise}
\end{cases}$$

$$d_i = |\mathcal{N}_i| \quad \text{Number of neighbors for agent } i$$

- Can be implemented in a distributed manner

Xiao, Boyd, Lall (2005)
Example

Init. values 1 2 2

Ave = 1.666

- Update scheme for agent 1:

\[ x_1(k + 1) = \frac{2}{3} x_1(k) + \frac{1}{3} x_2(k) \]

\[ = 1 - \frac{1}{3} = \frac{1}{1 + \max\{1, 2\}} \]

- Update scheme for agent 2:

\[ x_2(k + 1) = \frac{1}{3} x_2(k) + \frac{1}{3} x_1(k) + \frac{1}{3} x_3(k) \]

\[ = 1 - \frac{1}{3} - \frac{1}{3} = \frac{1}{1 + \max\{1, 2\}} = \frac{1}{1 + \max\{1, 2\}} \]
Example

Init. values 1 2 2

Ave = 1.666

Distributed algorithm:

\[ x_1(k + 1) = \frac{2}{3} x_1(k) + \frac{1}{3} x_2(k) \]

\[ x_2(k + 1) = \frac{1}{3} x_2(k) + \frac{1}{3} x_1(k) + \frac{1}{3} x_3(k) \]

\[ x_3(k + 1) = \frac{2}{3} x_3(k) + \frac{1}{3} x_2(k) \]
Example

Init. values 1 2 2

Ave = 1.666

Consensus!
Example

Init. values

\[ x(k+1) = \begin{bmatrix} 2/3 & 1/3 & 0 \\ 1/3 & 1/3 & 1/3 \\ 0 & 1/3 & 2/3 \end{bmatrix} x(k), \quad x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \]

Each element is nonnegative, and

- Sum of elements in each row = 1 ⇒ Row stochastic
- Sum of elements in each column = 1 ⇒ Column stochastic

Ave = 1.666
General form of the algorithm

$$x(k + 1) = Wx(k)$$

$$x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_N(k) \end{bmatrix}$$

- **Property 1**
  - Because $W$ is row stochastic,
    $$W1 = 1$$
    where $1 = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$
  - The matrix has eigenvalue 1
  - Corresponding eigenvector is a (scalar multiple of)
    vector 1: $c1$
General form of the algorithm

\[ x(k + 1) = W x(k) \]

\[ x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_N(k) \end{bmatrix} \]

- Property 2
  - Because \( W \) is column stochastic,
    \[ 1^T W = 1^T \]
    where \( 1 = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \)
  - Thus
    \[ \sum_{i=1}^{N} x_i(k + 1) = 1^T x(k + 1) = 1^T W x(k) \]
    \[ = 1^T x(k) = \sum_{i=1}^{N} x_i(k) \]

Sum of all elements is invariant!
Average vector

- By properties 1 and 2,

For eigenvalue 1, the eigenvector is in the form \( x^* = c \mathbf{1} \)

and satisfies

\[
\sum_{i=1}^{N} x_i^* = \sum_{i=1}^{N} x_i(0)
\]

Hence

\[
x_i^* = \frac{1}{N} \sum_{i=1}^{N} x_i(0) \quad \forall i = 1, \ldots, N
\]

- The desired average!

However, there may be other vectors as the eigenvector.

- If the graph is connected, then it is unique.

(by the Perron-Frobenius theorem)
Convergence of the algorithm

\[ x(k + 1) = Wx(k) \quad \text{with} \quad x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_N(k) \end{bmatrix} \]

- **Computation via power method**
  - The state \( x(k) \) converges to the eigenvector \( x^* \)

- **Result:** If the network of agents forms a connected graph, then average consensus is achieved:

\[
x_i(k) \to \frac{1}{N} \sum_{j=1}^{N} x_j(0), \quad k \to \infty, \quad \forall i = 1, 2, \ldots, N
\]
Autonomous mobile robots: Rendezvous

- 10 agents
- Random graph:
  - Initial positions are uniformly distributed
  - Neighbors are agents within radius $r$
Radius $r=0.8$  
# of edges 35
- Radius $r=0.6$  
  # of edges 27
- The graph is a subgraph of the previous one.
Radius $r=0.38$  
\# of edges 11

Disconnected !
Recap

Average consensus: Real-valued case

- True average
- Connected graph
- Matrix theory
Average consensus (2)

Integer-valued (quantized) case
Quantized average consensus

- Each agent's value $x_i(k)$ is an integer

- **What's different:**
  - True average of $N$ integers ≠ integer
  - Approximation of the average is not unique
  - Convergence in finite time is possible (i.e., not asymptotic)

Init. values 1 2 2 Ave = 1.666

![Diagram showing the quantized average consensus with initial values 1, 2, and 2, and an average of 1.666.]

Kashap, Basar, Srikant (2007)
Probabilistic communication

Gossip algorithm

- Agents decide to communicate at a random time with randomly chosen neighbor.
- To each edge, assign a probability to be chosen.

- No need of a common clock.
  
  (asynchronous communication)

Problem:

Find a distributed algorithm such that

1. Each agent’s value is always an integer
2. Sum of all agents’ value is constant
3. For sufficiently large $k$, the agents achieve average consensus, that is,

$$x_i(k) = \left\lfloor \frac{1}{N} \sum_{j=1}^{N} x_j(0) \right\rfloor$$

or

$$\frac{1}{N} \sum_{j=1}^{N} x_j(0)$$

Kashap, Basar, Srikant (2007)
Quantized gossip algorithm

■ At time \( k \), one edge \((i, j)\) is randomly chosen.

■ Agents \( i, j \) update their values to \( x_i(k + 1), x_j(k + 1) \) by
  ■ If \( x_i(k) = x_j(k) \), then the values stay the same.
  ■ If \(|x_i(k) - x_j(k)| = 1\), then exchange the values (Swapping)
  ■ Otherwise, if \( x_i(k) < x_j(k) \), then let
    \[ x_i(k + 1) = x_i(k) + 1 \]
    \[ x_j(k + 1) = x_j(k) - 1 \]
    1. Sum of both values remains the same
    2. Their difference is reduced

![Graph with edges and nodes indicating the gossip algorithm process.](Image)
Quantized gossip algorithm

Result:

The algorithm achieves quantized average consensus with probability 1 in finite time.

Two important properties:

- Swapping
- Probabilistic algorithm
Example 1 (Swapping)

- For each edge, the difference in values is at most 1.
- The average is unknown from local info.
- By swapping, consensus is possible.
  - Agents with values 1 and 3 become neighbors (with prob. 1).
Example 2 (Probabilistic algorithm)

Example of a deterministic algorithm: Periodic comm.

- Only swapping occurs, thus no consensus.
- Under probabilistic comm., convergence in a few steps.
Recap

Average consensus: Quantized-valued case

- Approximate average
- Gossip algorithm – Probabilistic but always correct
- Theory of Markov chain
- Performance at the order of $O(N^2)$
Summary

- Multi-agent systems and consensus problems
- Graph representation of network structures
- Distributed algorithms: Deterministic vs Probabilistic
- Update schemes for different agent values
  (real, quantized, and binary)

New challenges

- Performance
- Communication (time delay, data rate, graph,…)
- Dynamics of the agents (high dim., nonlinear,…)
Consensus problem

- Network of agents without a leader
- Each agent communicates with others and updates its state
- All agents should arrive at the same (unspecified) state

Achieve global objectives through local interaction!