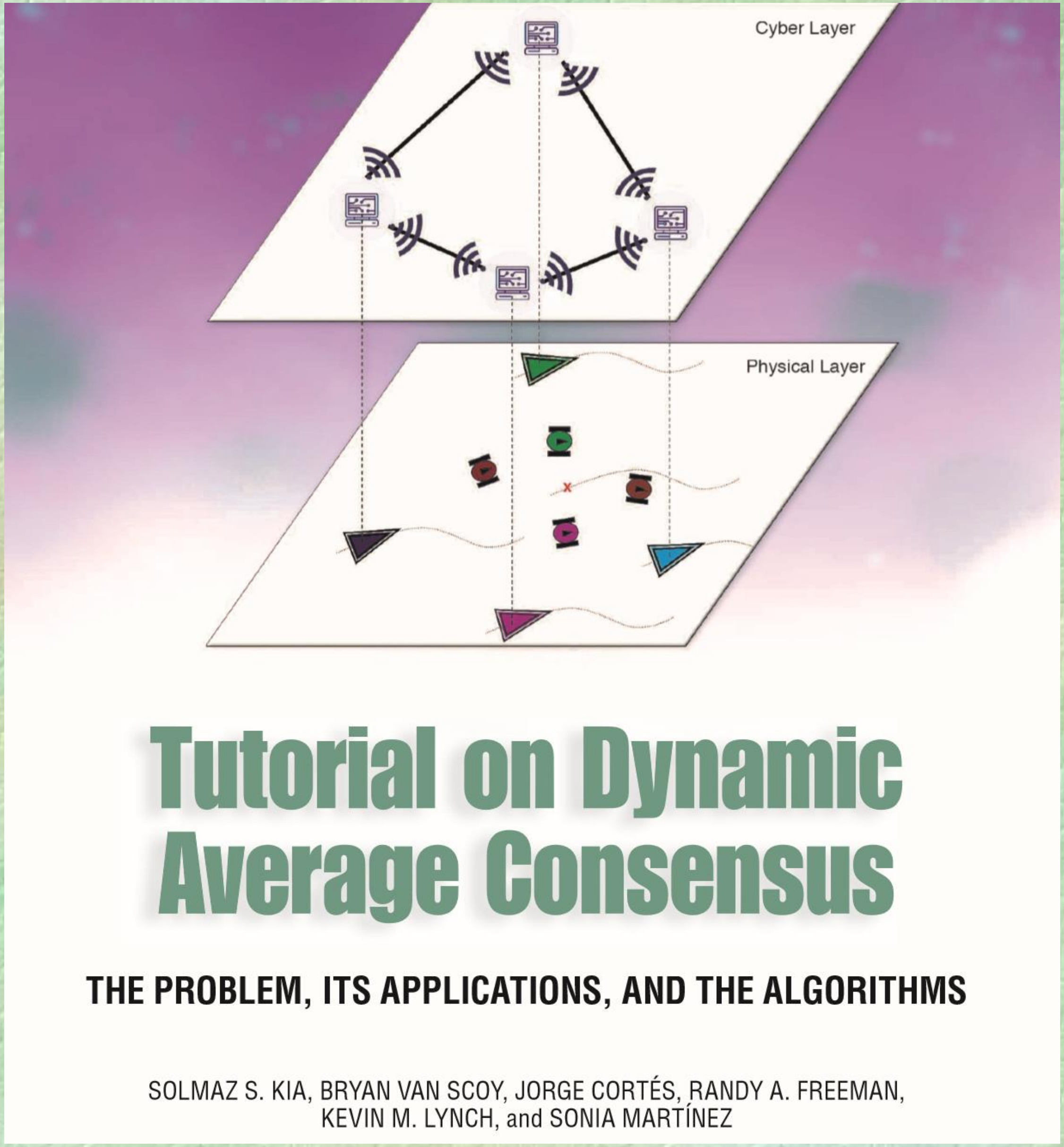
Dynamic Average Consensus

Ali Karimpour
Professor
Ferdowsi University of Mashhad, Iran

Reference

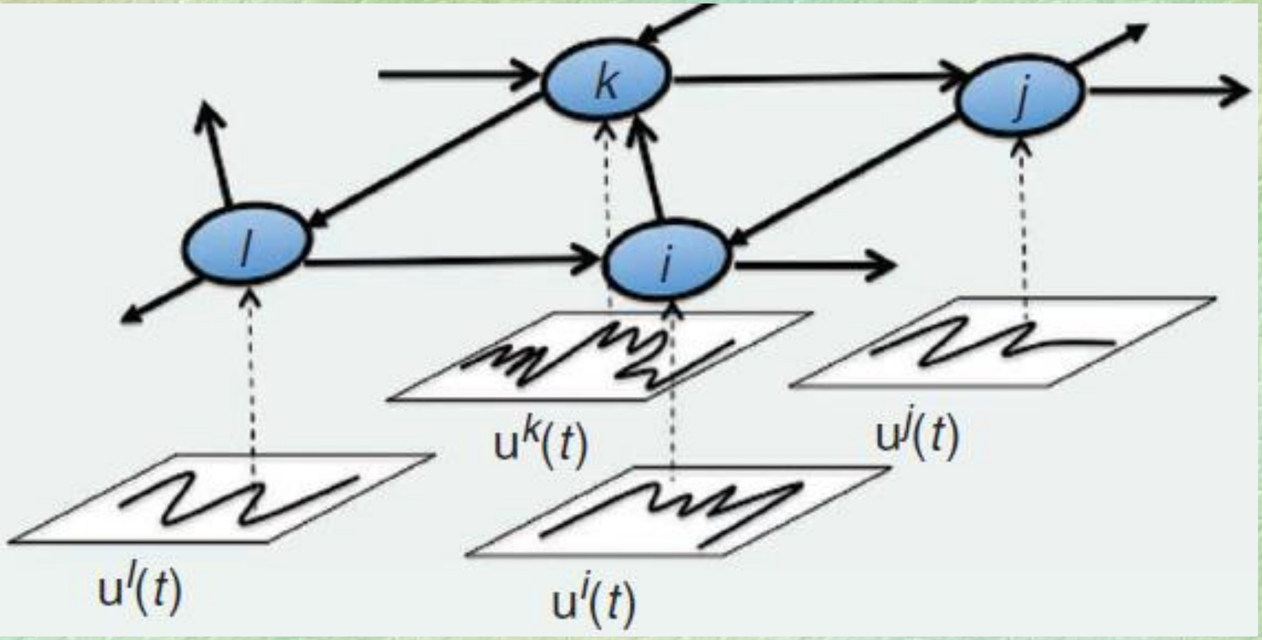

Cyber Layer
Physical Layer
Tutorial on Dynamic Average Consensus
THE PROBLEM, ITS APPLICATIONS, AND THE ALGORITHMS
SOLMAZ S. KIA, BRYAN VAN SCOY, JORGE CORTÉS, RANDY A. FREEMAN,
KEVIN M. LYNCH, and SONIA MARTÍNEZ

Contents

Introduction

The dynamic average consensus problem: The multi-agent network collectively compute the average of the set of time-varying signals.

This problem arises in scenarios with multiple agents, where each one has access to a time-varying signal of interest(for example, a distributed energy resource taking a sequence of frequency measurements in a micro-grid).



Different type of the dynamic average consensus problem:

- Centralized

In this approach all of the information gather in a single place(agent), perform the computation (in other words, calculate the average), and then send the solution back through the network to each agent.

- Flooding

In this approach, every agent act as the centralized agent.

- Distributed

Introduction

Different type of the dynamic average consensus problem:

- Centralized

  1) The algorithm is not robust to failures of the centralized agent
  2) The method is not scalable
  3) Each agent must have a unique identifier
  4) The calculated average is delayed by an amount that grows with the network size
  5) The reference signals from each agent are exposed over the entire network (which is unacceptable in applications involving sensitive data).

- Flooding

  1) All centralized drawback except number 1.

- Distributed

  1) Hard to derive.

Contents

Averaging Consensus

Different type of the average consensus problem:

1) Static Average Consensus

Agents seek to agree on a specific combination of fixed quantities.

Several simple and efficient distributed algorithms with exact convergence guarantees (see Lectures on Network Systems Francesco Bullo)

2) Dynamic Average Consensus

Agents seek to agree on a specific combination of variable quantities.

If there is a static average consensus algorithm that is able to converge infinitely fast then we can use static average consensus.

Solving this problem is challenging because the local interactions among agents involve only partial information, and the quantity that the network seeks to compute is changing as the agents run their routines.

Challenges with Dynamic Problem

Example: Consider a group of six agents with the communication topology of a directed ring which each process described by a fixed value plus a sine wave whose frequency and phase are changing randomly over time. $\omega \sim N(0,0.25), \varphi \sim N(0,(\frac{\pi}{2})^2)$

$$u_1(m) = 1.1\big(2 + \sin\big(\omega(m)t(m) + \varphi(m)\big)\big) - 0.55$$

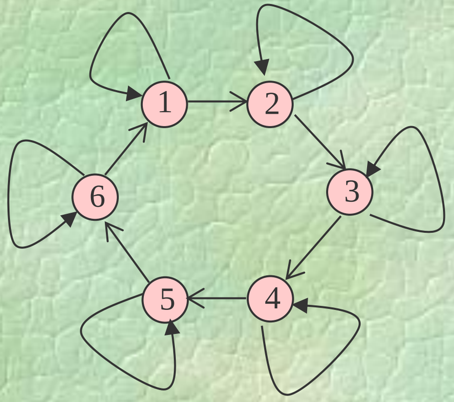
$$u_2(m) = 1\big(2 + \sin\big(\omega(m)t(m) + \varphi(m)\big)\big) + 1$$

$$u_3(m) = 0.9\big(2 + \sin\big(\omega(m)t(m) + \varphi(m)\big)\big) + 0.6$$

$$u_4(m) = 1.05\big(2 + \sin\big(\omega(m)t(m) + \varphi(m)\big)\big) - 0.9$$

$$u_5(m) = 0.96\big(2 + \sin\big(\omega(m)t(m) + \varphi(m)\big)\big) - 0.6$$

$$u_6(m) = 1\big(2 + \sin\big(\omega(m)t(m) + \varphi(m)\big)\big) + 0.4$$



Each agent measure the process according to every 2 sec.

To obtain the average, the following two approaches are used:

1- The standard static discrete-time Laplacian average consensus algorithm.

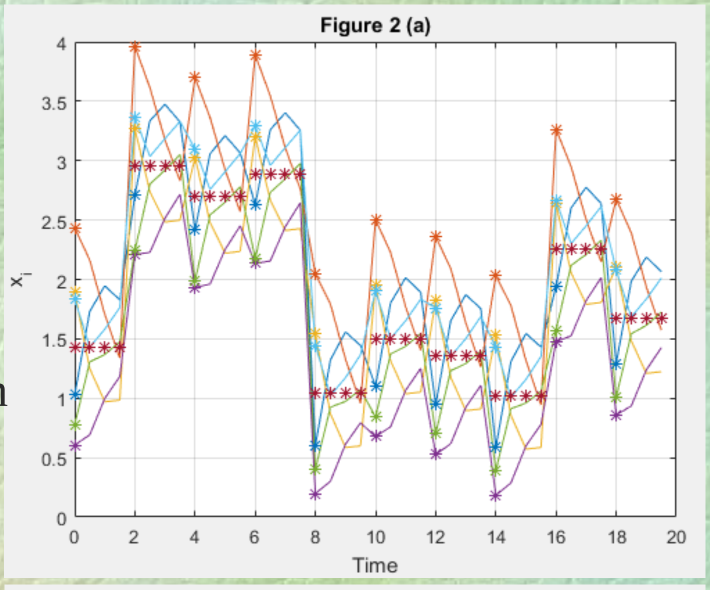
$$x_i(k+1) = x_i(k) - \delta \sum_{j=1}^{n} a_{ij}\big(x_i(k) - x_j(k)\big) \qquad i = 1, 2, \dots, 6$$

2- The dynamic average consensus algorithm [more specifically, strategy (S15)].

Challenges with Dynamic Problem

Figure 2 (a) $\delta = 0.5$ standard static discrete-time Laplacian average consensus algorithm. (3 communication)
(b) 20 communication

States doesn't converge to average.



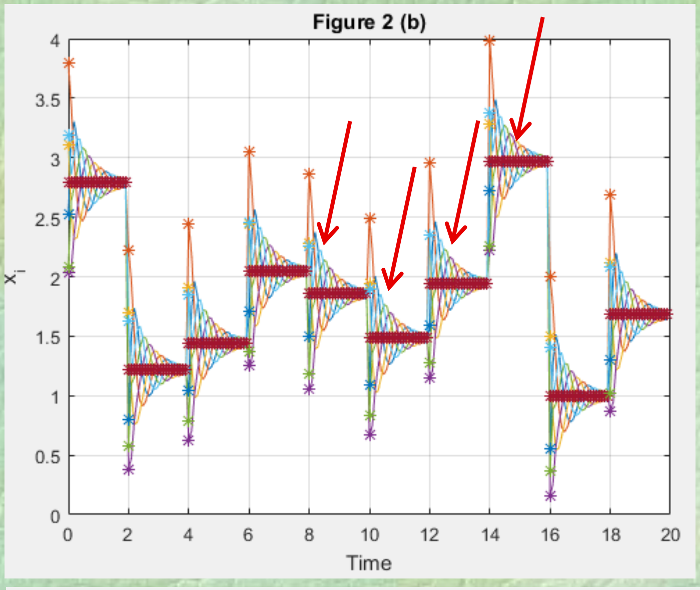
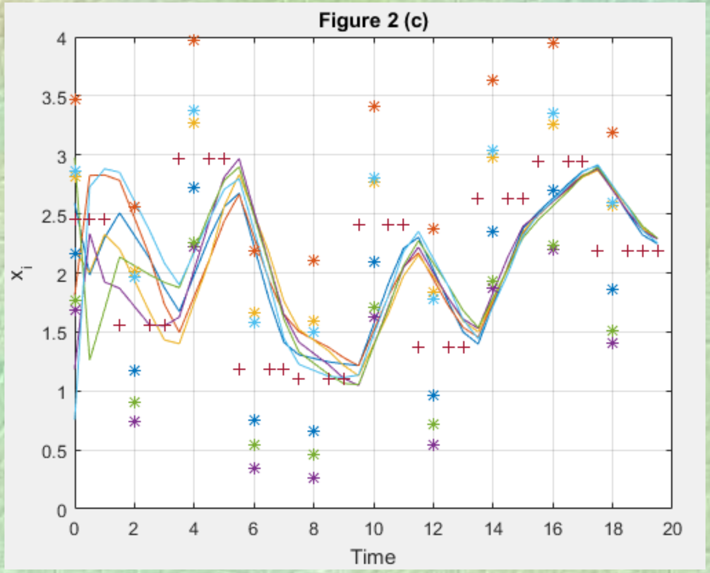
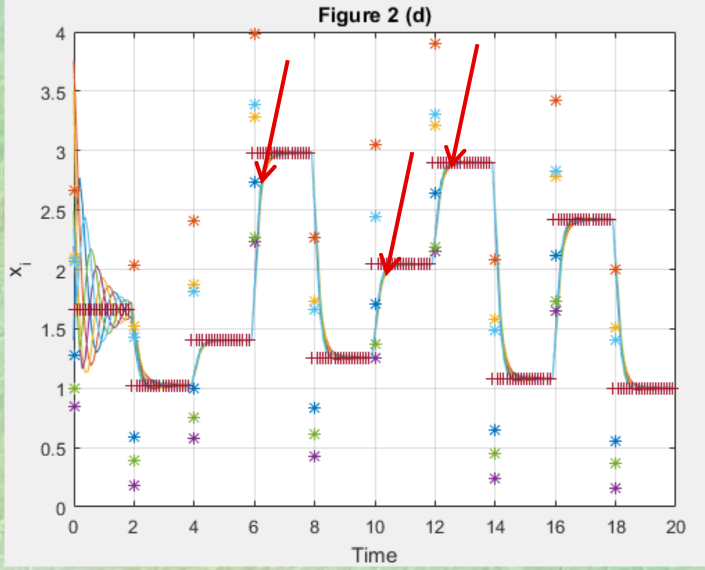


Figure 2 (c) $\delta = 0.5$ dynamic discrete-time Laplacian average consensus algorithm. (3 communication)
(d) 20 communication

All states converge to average.





Contents

Dynamic Average Consensus Problem Formulation

Different type of dynamic averaging problems:

1- Dynamic average consensus in discrete time

2- Dynamic average consensus in continues time

3- Dynamic average consensus in continues time-discrete time

1- Dynamic average consensus in discrete time

$$x^i(t_{k+1}) = c^i(J^i(t_k), \{I^j(t_k)\}_{j \in \mathcal{N}^i_{\text{out}}}), \quad i \in \{1, \ldots, N\},$$

such that $x^i(t_k) \to \mathrm{u}^{\text{avg}}(k)$ as $t_k \to \infty$.

Driving command ($J^i(t_k)$)

Self variables of ith agent (c^i)

Out-neighbors variables of ith agent ($\{I^j(t_k)\}_{j \in \mathcal{N}^i_{\text{out}}}$)

Input: $J^i(k)$ and $\{I^j(k)\}_{j \in \mathcal{N}^i_{\text{out}}}$

Output: $x^i(k+1), J^i(k+1)$, and $I^i(k+1)$

Step 1. $x^i(k+1) \leftarrow c^i(J^i(t_k), \{I^j(t_k)\}_{j \in \mathcal{N}^i_{\text{out}}})$

Step 2. Generate $J^i(k+1)$ and $I^i(k+1)$

Step 3. Broadcast $I^i(k+1)$

Dynamic Average Consensus Problem Formulation

2- Dynamic average consensus in continuous time

$$\dot{x}^i = c^i(J^i(t), \{I^j(t)\}_{j\in\mathcal{N}^i_{\text{out}}}), \quad i \in \{1,\dots,N\},$$

such that $x^i(t) \to \mathsf{u}^{\text{avg}}(t)$ as $t \to \infty$.

Driving command

Self variables of ith agent

Out-neighbors variables of ith agent

3- Dynamic average consensus in continuous time-discrete time

$$\dot{x}^i(t) = c^i(J^i(t), \{I^j(t^j_{k^j})\}_{j\in\mathcal{N}^i_{\text{out}}}), \quad i \in \{1,\dots,N\},$$

Driving command

Self variables of ith agent

such that $x^i(t) \to \mathsf{u}^{\text{avg}}(t)$ as $t \to \infty$.

Out-neighbors variables of ith agent

Dynamic Average Consensus Problem Formulation

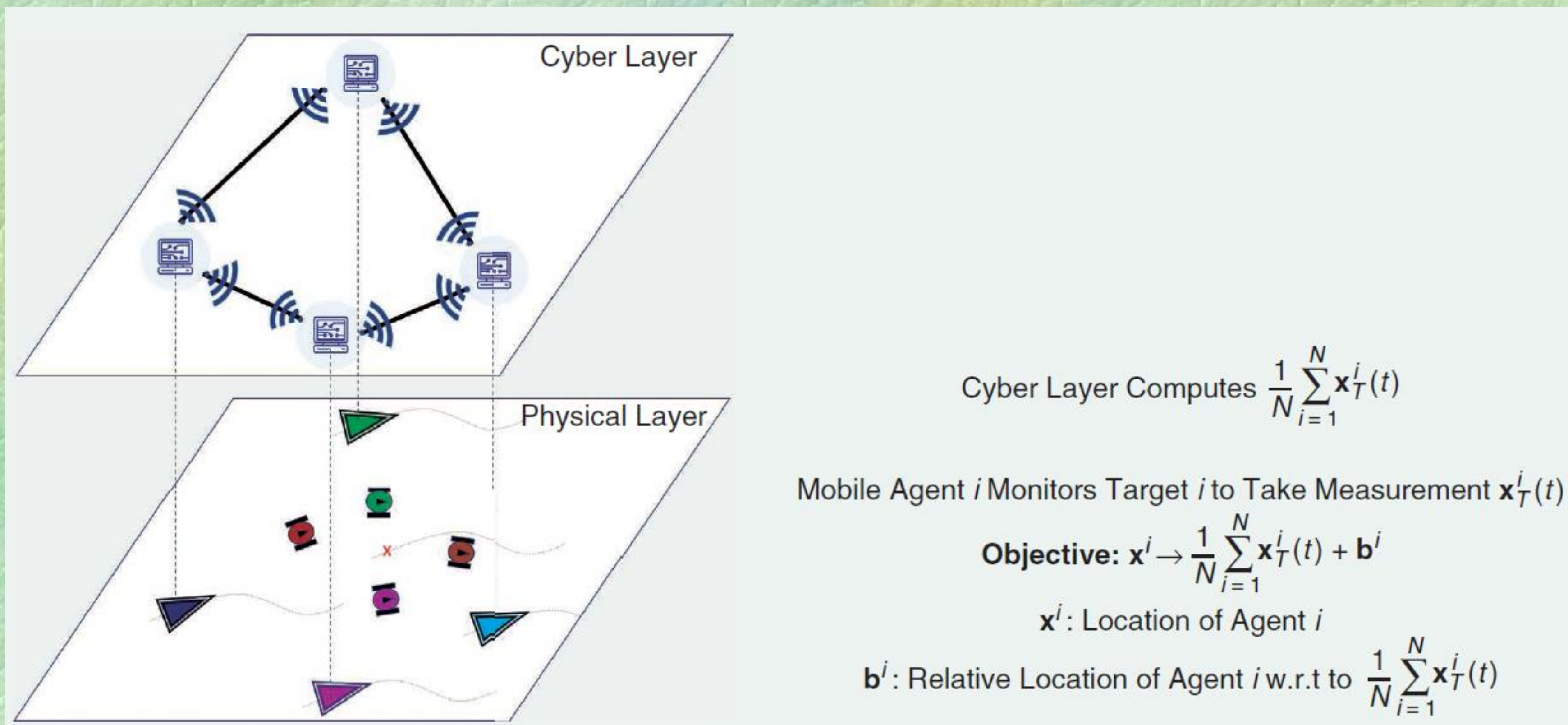
Desirable properties of dynamic average consensus

- Scalability, the amount of computations and resources required on each agent does not grow with the network size of network.
- Robustness, to the disturbances present in practical scenarios, such as communication delays and packet drops, agents entering/leaving the network, initial condition, and noisy measurements
- Correctness, meaning the algorithm converges to the exact average or, alternatively, a formal guarantee can be given about the distance between the estimate and the exact average.

Contents

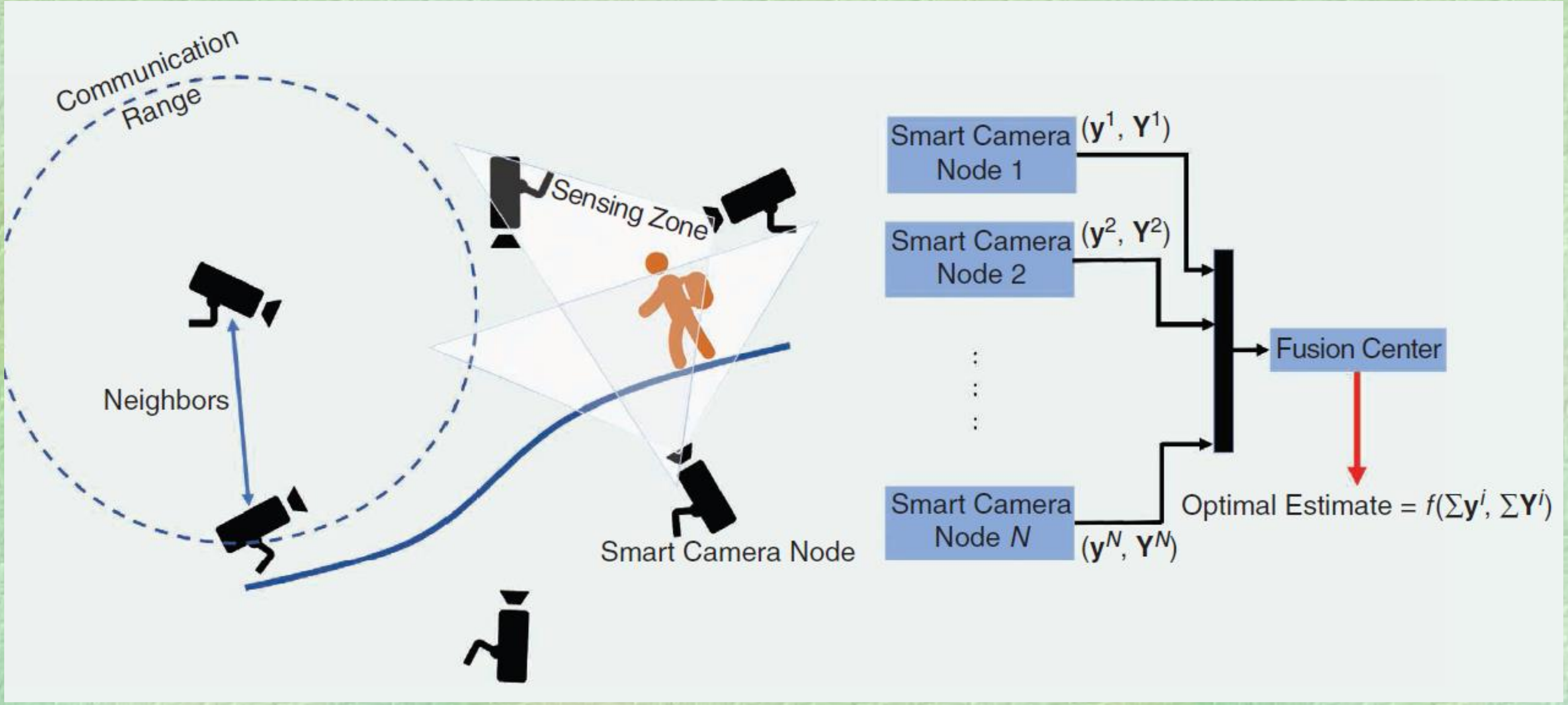
Application of Dynamic Average Consensus in Network Systems

- Distributed Formation Control:



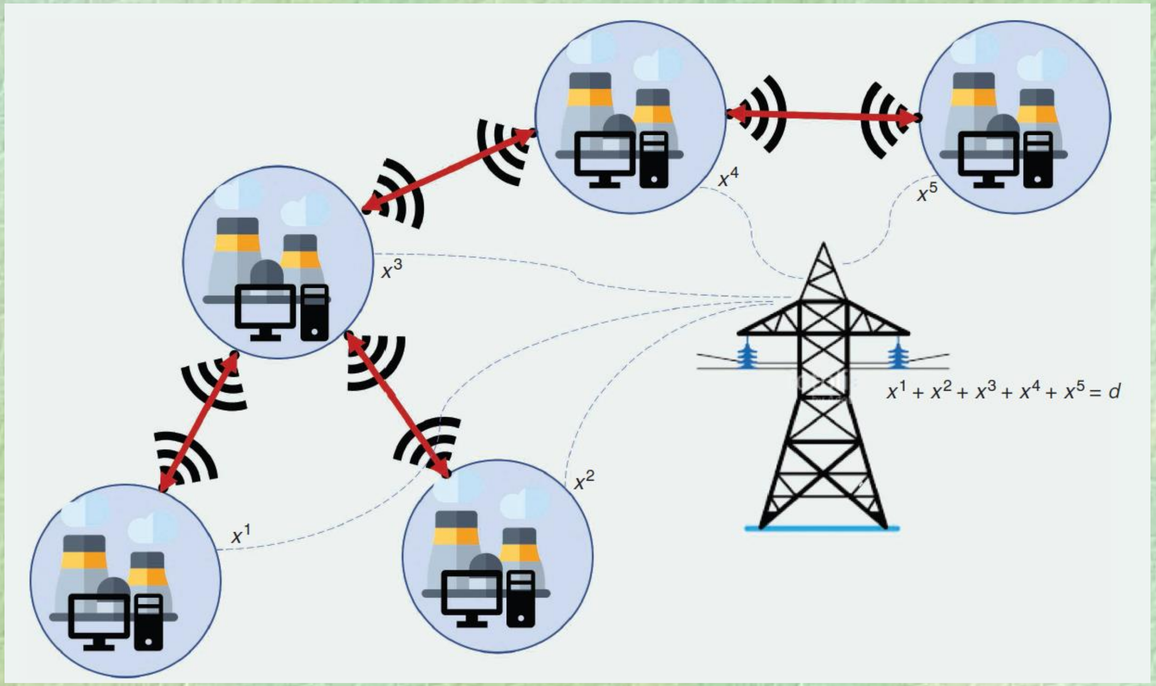
Application of Dynamic Average Consensus in Network Systems

- Distributed State Estimation



Application of Dynamic Average Consensus in Network Systems

- Distributed Resource Allocation



Contents

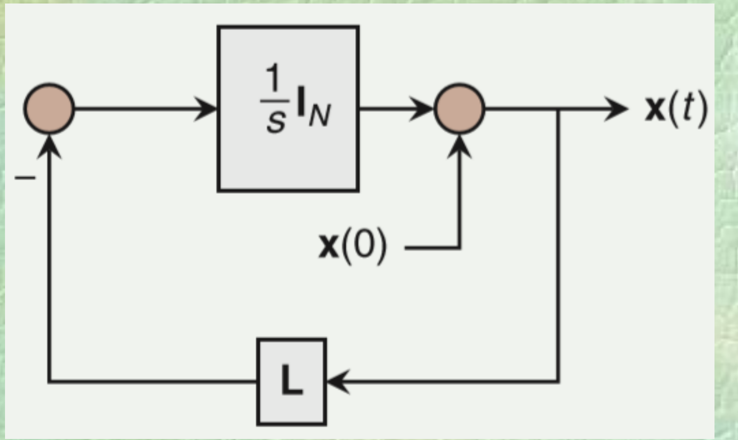
Dynamic Average Consensus in Network Systems

Consensus algorithms to solve the static average consensus problem

- Static average consensus problem in continuous time is:

$$\dot{x}(t) = -\sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right)$$

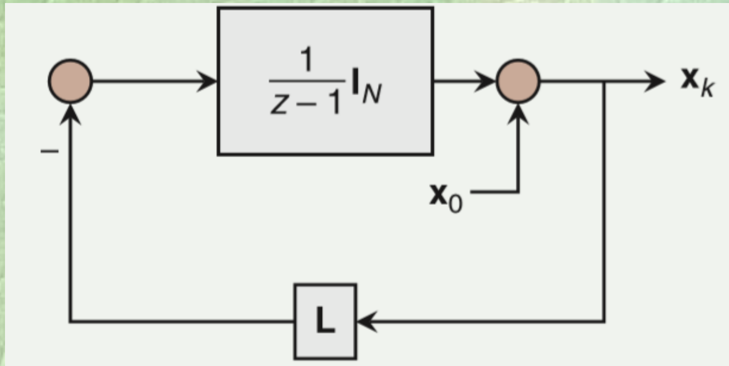
$$\dot{x}(t) = -Lx(t) \Rightarrow x(t) - x(0) = -\int_0^t Lx(t)dt$$



- Static average consensus problem in discrete time is:

$$x^i(k+1) = x^i(k) - \sum_{j=1}^{N} a_{ij}\left(x^i(k) - x^j(k)\right)$$

$$x(k+1) = (I - L)x(k)$$



Dynamic Average Consensus in Network Systems

Theorem 1: Convergence guarantees of the continuous and discrete-time (static average consensus)

Suppose that the communication graph is a constant, strongly connected, and weight-balanced digraph and that the reference signal x(0)=u at each agent is a constant scalar. Then

Continuous time: As $t \to \infty$, every state $x_i(t)$ of the static average consensu algorithm converges to u_{avg} with an exponential rate no worse than the smallest nonzero eigenvalue of Sym(**L**).

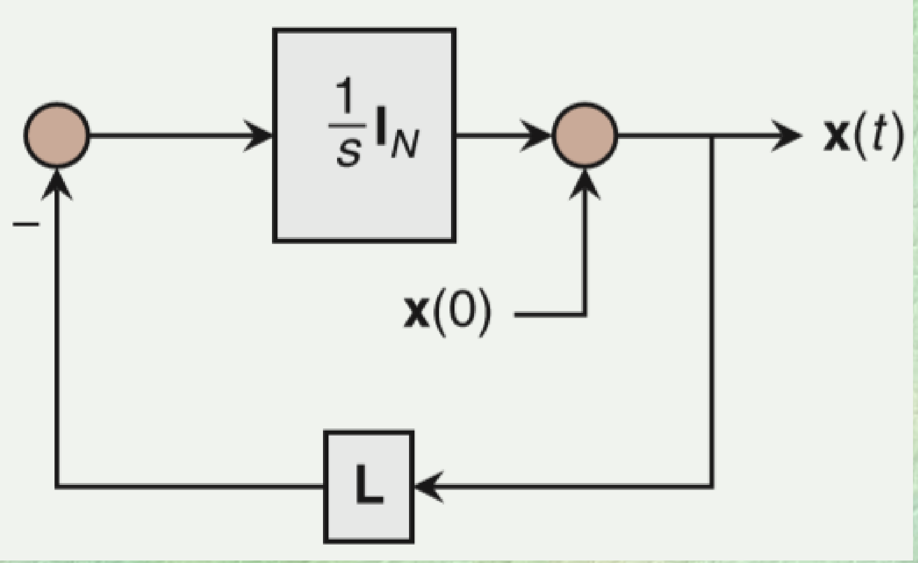
Discrete time: As $k \to \infty$, every state , $x_i(k)$ of the static average consensus algorithm converges to u_{avg} with an exponential rate no worse than $\rho \epsilon (0,1)$, provided that the Laplacian matrix satisfies

$$\rho = \left\| \mathbf{I}_N - \mathbf{L} - \mathbf{1}_N \mathbf{1}_N^\top / N \right\|_2 < 1.$$

Dynamic Average Consensus in Network Systems

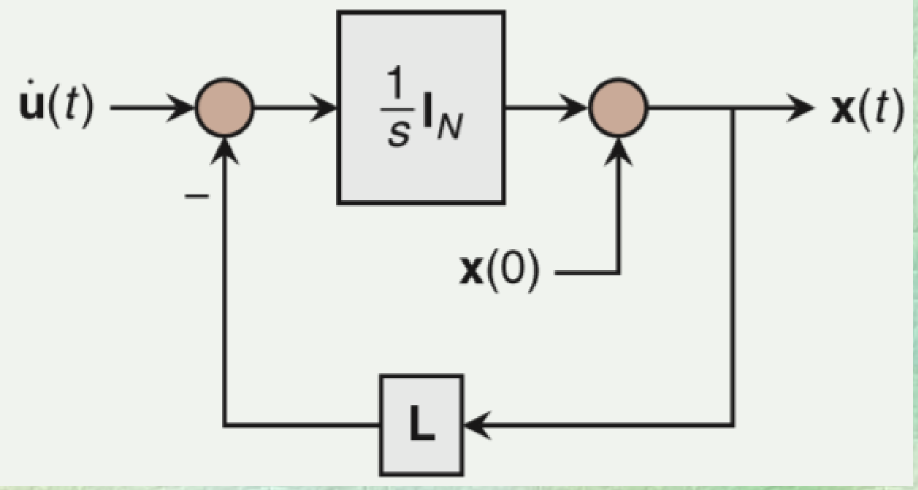
Consensus algorithms to solve the dynamic average consensus problem

- Static average consensus problem in continuous time is:



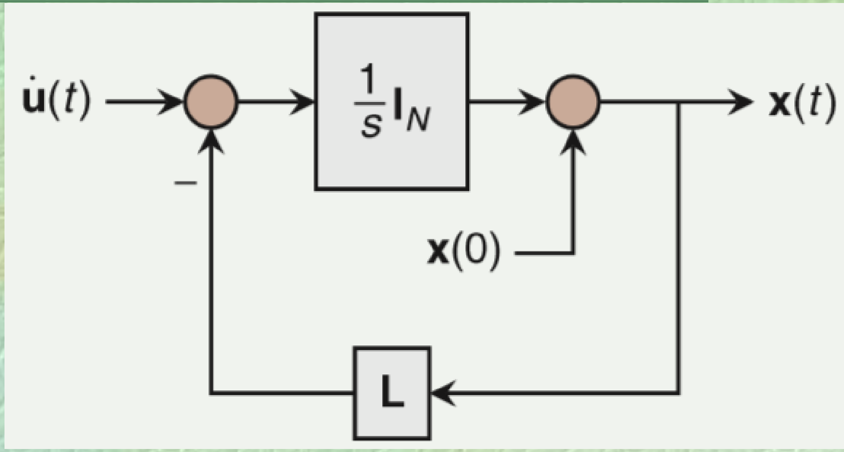
- Dynamic average consensus problem in continuous time is:

$$\dot{x}(t) = -Lx(t) + \dot{u}(t)$$



Dynamic Average Consensus in Network Systems

Dynamic average consensus problem



$$\dot{x}(t) = -Lx(t) + \dot{u}(t)$$

So for every agent we have:

$$\dot{x}^i(t) = -\sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) + \dot{u}^i(t) \qquad (11a)$$

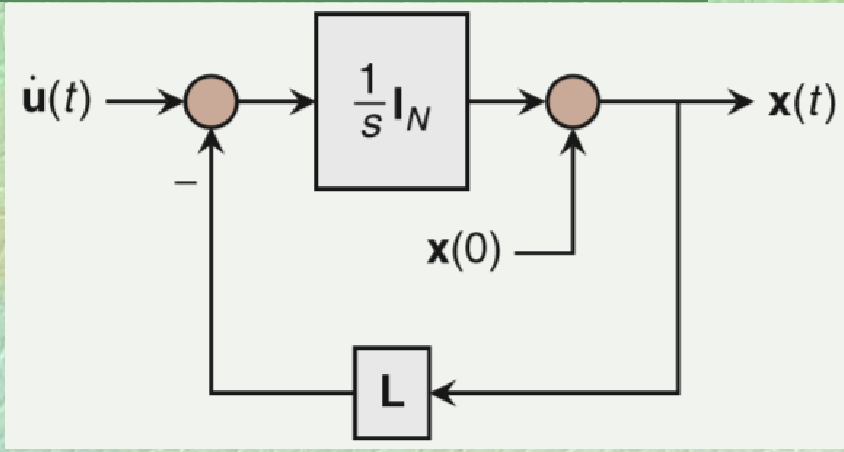
$$x^i(0) = u^i(0) \qquad (11b)$$

Let define: $e^i(t) = x^i(t) - u^{avg}(t)$

$$\begin{bmatrix} \dot{e}^1 \\ \dot{e}^2 \\ .. \\ \dot{e}^N \end{bmatrix} = \begin{bmatrix} .. & .. & .. & .. \\ .. & .. & .. & .. \\ .. & .. & .. & .. \\ .. & .. & .. & .. \end{bmatrix} \begin{bmatrix} e^1 \\ e^2 \\ .. \\ e^N \end{bmatrix} + \begin{bmatrix} \dot{u}^1 \\ \dot{u}^2 \\ .. \\ \dot{u}^N \end{bmatrix}$$

Dynamic Average Consensus in Network Systems

Dynamic average consensus problem



Let define: $e^i(t) = x^i(t) - u^{avg}(t)$

$$\begin{bmatrix} \dot e^1 \\ \dot e^2 \\ \cdot\cdot \\ \dot e^N \end{bmatrix} = \begin{bmatrix} \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot \end{bmatrix} \begin{bmatrix} e^1 \\ e^2 \\ \cdot\cdot \\ e^N \end{bmatrix} + \begin{bmatrix} \dot u^1 \\ \dot u^2 \\ \cdot\cdot \\ \dot u^N \end{bmatrix}$$

Define a new transformation $\bar e = T^T e$

Where $T = \begin{bmatrix} \frac{1}{\sqrt N}\mathbf{1}_N & R \end{bmatrix}$ such that T is unitary ($T^T.T=I$)

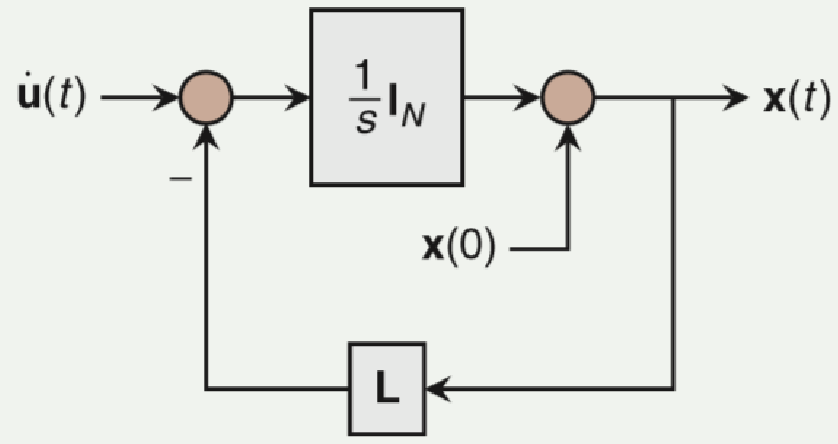
$$\begin{bmatrix} \dot{\bar e}^1 \\ \dot{\bar e}^2 \\ \cdots \\ \dot{\bar e}^N \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ 0 & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ 0 & \cdot\cdot & \cdot\cdot & \cdot\cdot \end{bmatrix} \begin{bmatrix} \bar e^1 \\ \bar e^2 \\ \cdots \\ \bar e^N \end{bmatrix} + RT \begin{bmatrix} 0 \\ \dot u^2 \\ \cdots \\ \dot u^N \end{bmatrix}$$

$$\dot{\bar e}_1 = 0, \quad \bar e_1(t_0) = \frac{1}{\sqrt N}\sum_{j=1}^{N} \left(x^j(t_0) - \mathsf{u}^j(t_0)\right),$$

$$\dot{\bar{\mathbf e}}_{2:N} = -\mathbf R^\top \mathbf L \mathbf R \bar{\mathbf e}_{2:N} + \mathbf R^\top \dot{\mathbf u}, \quad \bar{\mathbf e}_{2:N}(t_0) = \mathbf R^\top \mathbf x(t_0),$$

Dynamic Average Consensus in Network Systems

Dynamic average consensus problem



Define a new transformation $\bar{e} = T^T e$

$$\dot{\bar{e}}_1 = 0, \quad \bar{e}_1(t_0) = \frac{1}{\sqrt{N}} \sum_{j=1}^{N} (x^j(t_0) - \mathrm{u}^j(t_0)),$$

$$\dot{\bar{\mathbf{e}}}_{2:N} = -\mathbf{R}^\top \mathbf{L} \mathbf{R} \bar{\mathbf{e}}_{2:N} + \mathbf{R}^\top \dot{\mathbf{u}}, \quad \bar{\mathbf{e}}_{2:N}(t_0) = \mathbf{R}^\top \mathbf{x}(t_0)$$

The tracking error of each agent over a strongly connected and weight-balanced digraph is

$$|e^i(t)| \le \sqrt{\|\bar{\mathrm{e}}_{2:N}(t)\|^2 + |\bar{e}_1(t)|^2} \le \sqrt{\left(\mathrm{e}^{-\hat{\lambda}_2(t-t_0)}\|\mathbf{\Pi}\mathbf{x}(t_0)\| + \frac{\sup_{t_0 \le \tau \le t}\|\mathbf{\Pi}\dot{\mathbf{u}}(\tau)\|}{\hat{\lambda}_2}\right)^2 + \left(\frac{\sum_{j=1}^{N}(x^j(t_0) - \mathrm{u}^j(t_0))}{\sqrt{N}}\right)^2}$$

Where $\hat{\lambda}_2$ is the smallest non-zero eigenvalue of $Sym(L) = \frac{1}{2}(L + L^T)$ and

$$\mathbf{\Pi} = (\mathbf{I}_N - (1/N)\mathbf{1}_N\mathbf{1}_N^\top)$$

Dynamic Average Consensus in Network Systems

The tracking error of each agent over a strongly connected and weight-balanced digraph is

$$|e^i(t)| \le \sqrt{\|\bar{\mathrm{e}}_{2:N}(t)\|^2 + |\bar{e}_1(t)|^2} \le \sqrt{\left(\mathrm{e}^{-\hat{\lambda}_2(t-t_0)}\|\mathbf{\Pi}\mathbf{x}(t_0)\| + \frac{\sup_{t_0\le\tau\le t}\|\mathbf{\Pi}\dot{\mathbf{u}}(\tau)\|}{\hat{\lambda}_2}\right)^2 + \left(\frac{\sum_{j=1}^{N}(x^j(t_0)-\mathrm{u}^j(t_0))}{\sqrt{N}}\right)^2}$$

$\hat{\lambda}_2$ is the smallest non-zero eigenvalue of $Sym(L) = \frac{1}{2}(L + L^T)$ and $\mathbf{\Pi} = (\mathbf{I}_N - (1/N)\mathbf{1}_N\mathbf{1}_N^\top)$

- First, it highlights the importance of the special initialization
- Perfect asymptotic tracking for reference input signals with decaying rate.
- Perfect asymptotic tracking for unbounded reference signals whose uncommon parts asymptotically converge to a constant value.

$$\left\|\left(\mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top\right)\dot{\mathrm{u}}(\tau)\right\| = \left\|\left(\mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top\right)(\dot{\underline{\mathrm{u}}}(t)\mathbf{1}_N + \dot{\hat{\mathrm{u}}}(t))\right\| = \left\|\left(\mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top\right)\dot{\hat{\mathrm{u}}}(t)\right\|$$

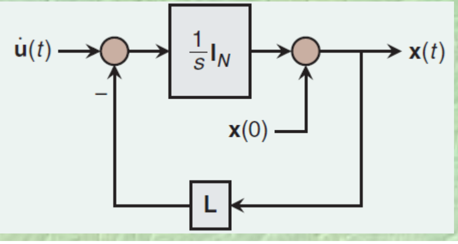
- Finally, the tracking error bound shows that as long as the uncommon part of the reference signals has a bounded rate, then system tracks the average with some bounded error.

Dynamic Average Consensus in Network Systems

$\dot{X}(t) = -LX(t) + \dot{U}(t) \quad x^i(0) = u^i\,(0)$

$$\dot{x}^i(t) = -\sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) + \dot{u}^i\,(t) \qquad (11a)$$

$$x^i(0) = u^i(0) \qquad (11b)$$



$$|e^i(t)| \le \sqrt{\|\bar{\mathrm{e}}_{2:N}(t)\|^2 + |\bar{e}_1(t)|^2} \le \sqrt{\left(\mathrm{e}^{-\hat{\lambda}_2(t-t_0)}\|\mathbf{\Pi x}(t_0)\| + \frac{\sup_{t_0\le\tau\le t}\|\mathbf{\Pi}\dot{\mathbf{u}}(\tau)\|}{\hat{\lambda}_2}\right)^2 + \left(\frac{\sum_{j=1}^{N}(x^j(t_0) - \mathrm{u}^j(t_0))}{\sqrt{N}}\right)^2}, \qquad (14)$$

Theorem 2:

(Convergence of above system over a Strongly connected and weight-balanced digraph)

Let G be a strongly connected and weight-balanced digraph. If $\sum_{j=1}^{N} x^j(t_0) = \sum_{j=1}^{N} \mathrm{u}^j(t_0)$ then the trajectories of system are then bounded and satisfy

$$\lim_{t\to\infty} |x^i(t) - \mathrm{u}^{\mathrm{avg}}(t)| \le \frac{\gamma(\infty)}{\hat{\lambda}_2}, \qquad i \in \{1, \dots, N\},$$

$$\sup_{\tau\in[t,\infty)} \left\| \left(\mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top\right)\dot{\mathbf{u}}(\tau) \right\| = \gamma(t) < \infty.$$

The smallest nonzero eigenvalue of the symmetric part of the graph Laplacian is a measure of connectivity of a graph. For highly connected graphs, it is expected that the diffusion of information across the graph is faster.

Moreover, $\sum_{j=1}^{N} x^j(t) = \sum_{j=1}^{N} \mathrm{u}^j(t)$ for $t \in [t_0, \infty)$.

Introduction II (Basic Notions from Graph Theory)

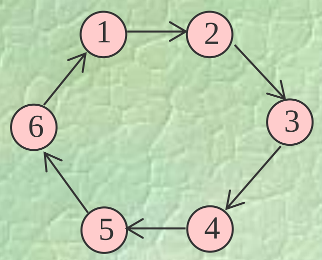
Effect of graph connectivity on eigenvalues of SYM(L)

Consider following graphs

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

eig((L+L')/2) =
0
0.5000
0.5000
1.5000
1.5000
2.0000

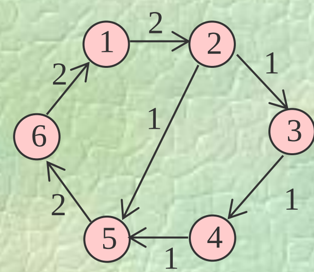
$\hat{\lambda}_2 = 0.5$



$$A=\begin{bmatrix}0&2&0&0&0&0\\0&0&1&0&1&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&2\\2&0&0&0&0&0\end{bmatrix},\quad L=\begin{bmatrix}2&-2&0&0&0&0\\0&2&-1&0&-1&0\\0&0&1&-1&0&0\\0&0&0&1&-1&0\\0&0&0&0&2&-2\\-2&0&0&0&0&2\end{bmatrix}$$

eig((L+L')/2) =
0.0000
0.6340
1.1771
2.0000
2.3660
3.8229

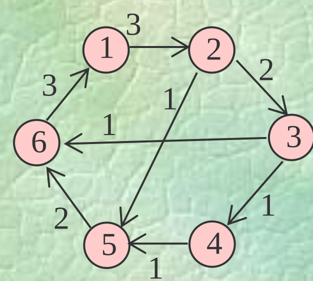
$\hat{\lambda}_2 = 0.634$



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

eig((L+L')/2) =
0.0000
0.9641
1.7929
2.5546
3.2071
5.4813

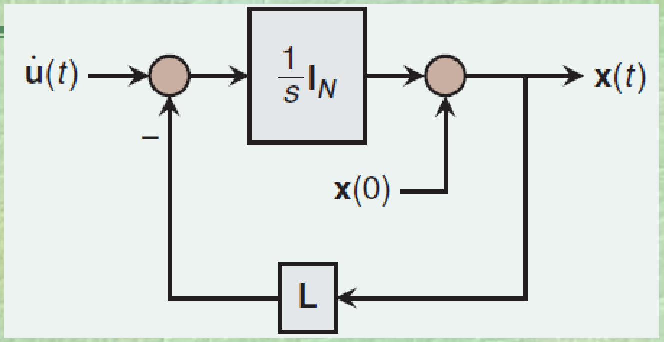
$\hat{\lambda}_2 = 0.964$



Highly connected graph (strongly connected and balance) ⇒ Largest smallest non-zero eigenvalue in symmetric part of the Laplacian

Dynamic Average Consensus in Network Systems

Implementation Challenges of (11)



1) high gain on Laplacian L → Larger $\hat{\lambda}_2$

→ Smaller tracking error

→ In physical systems, increase of the control effort.

$$\dot{X}(t) = -LX(t) + \dot{U}(t) \quad x^i(0) = u^i\ (0)$$

$$\dot{x}^i(t) = -\sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) + \dot{u}^i\ (t)$$

2) It requires the derivative of the reference signals. $x^i(0) = u^i(0)$

$$\lim_{t\to\infty} |x^i(t) - \mathrm{u}^{\mathrm{avg}}(t)| \le \frac{\gamma(\infty)}{\hat{\lambda}_2}, \qquad i \in \{1, \ldots, N\}, \qquad (15)$$

→ Computing the derivative can be costly and prone to error(in applications where the input signals are Measured online)

3) Initialization condition requiring. → Agents must initialize with $x_i(t_0)$=$u_i(t_0)$

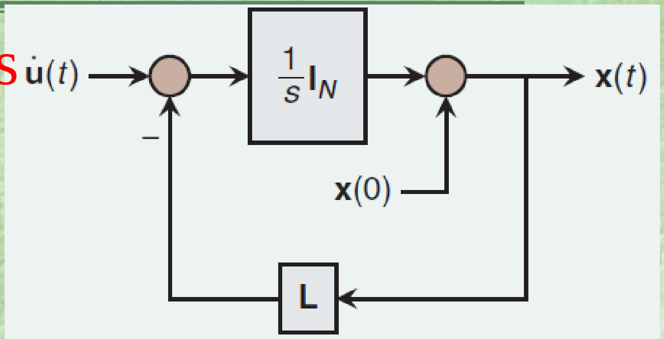
→ any perturbation in u_i results in a steady-state error in the tracking process.

4) If an agent leaves the operation permanently → initialization is no longer valid

→ reinitialization or a steady-state error in their tracking signal.

Dynamic Average Consensus in Network Systems

Implementation Challenges of (11) and solutions



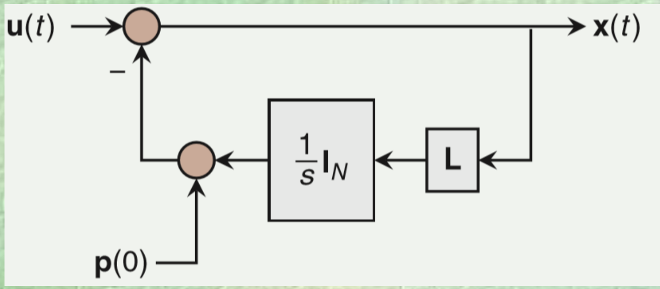
$$\dot{x}^i(t) = -\sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) + \dot{u}^i(t) \qquad (11a)$$

$$x^i(0) = u^i(0) \qquad (11b)$$

Now let: $p_i = \mathrm{u}_i - x_i$

$$\dot{p}^i(t) = \sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) \quad \sum_{j=1}^{N} p^j(t_0) = 0 \qquad (16a)$$

$$x^i(t) = u^i(t) - p^i(t) \qquad (16b)$$



We note that the initialization condition can be easily satisfied if each agent starts at $p^i(0) = 0$. Note that this requirement is mild because p^i is an internal state for agent i and, therefore, is not affected by communication errors.

Dynamic Average Consensus in Network Systems

Time trajectory of states in (11 or 16)

$$\dot{x}^i(t) = -\sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) + \dot{u}^i\,(t) \qquad (11a) \qquad \Rightarrow \sum_{i=1}^{N} \dot{x}^i(t) = \sum_{i=1}^{N} \dot{u}^i(t)$$

$$\Rightarrow \sum_{i=1}^{N} x^i(t) = \sum_{i=1}^{N} u^i(t) + \left(\sum_{i=1}^{N} x^i(t_0) - \sum_{i=1}^{N} u^i(t_0)\right)$$

Thus, if the perturbation on the reference input measurement is removed, then (11) still inherits the adverse effect of the initialization error.

$$\dot{p}^i(t) = \sum_{j=1}^{N} a_{ij}\left(x^i(t) - x^j(t)\right) \quad \sum_{j=1}^{N} p^j(t_0) = 0 \qquad (16a) \qquad \Rightarrow \sum_{i=1}^{N} \dot{p}^i(t) = 0$$

$$\Rightarrow \sum_{i=1}^{N} p^i(t) = \sum_{i=1}^{N} p^i(t_0)$$

So, for the case of the alternative algorithm (16) when the perturbations are removed, then (16) recovers the convergence guarantee of the perturbation-free case.

Dynamic Average Consensus in Network Systems

Presence of Additive Reference Input Perturbations in (11 or 16)

Lemma 1: Convergence of (16) Over a Strongly Connected and Weight-Balanced Digraph in the Presence of Additive Reference Input Perturbations

Let G be a strongly connected and weight-balanced digraph. Suppose $w^i(t)$ is an additive perturbation on the measured reference input signal $u^i(t)$. Then if $\sum_{j=1}^{N} p^j(t_0) = 0$ the trajectories of system 16 are then bounded and satisfy

$$\lim_{t \to \infty} |x^i(t) - \mathrm{u}^{\mathrm{avg}}(t)| \leq \frac{\gamma(\infty) + \omega(\infty)}{\hat{\lambda}_2}, \qquad i \in \{1, \ldots, N\},$$

where

$$\gamma(t) = \sup_{\tau \in [t,\infty)} \|(\mathbf{I}_N - (1/N)\mathbf{1}_N \mathbf{1}_N^\top)\dot{\mathbf{u}}(\tau)\| < \infty \qquad \omega(t) = \sup_{\tau \in [t,\infty)} \|(\mathbf{I}_N - (1/N)\mathbf{1}_N \mathbf{1}_N^\top)\dot{\mathbf{w}}(\tau)\| < \infty$$

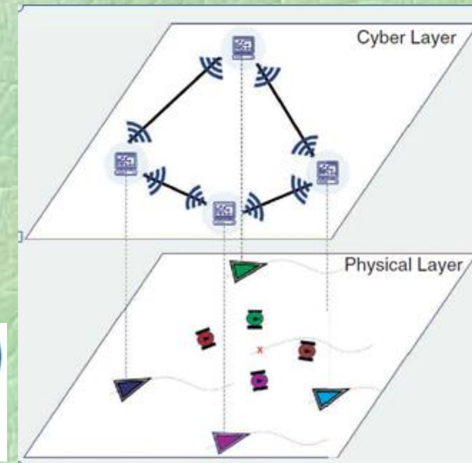
Moreover, $\sum_{j=1}^{N} p^j(t) = 0 \ for \ t \in [t_0, \infty)$

The perturbation w^i in Lemma 1 can also be regarded as a bounded communication perturbation. Therefore, (16) [and similarly (11)] is considered naturally robust to bounded communication error.

Dynamic Average Consensus in Network Systems

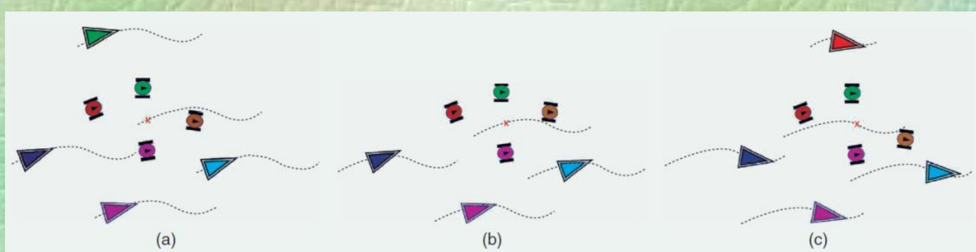
Implementation of eq(16) for following system.

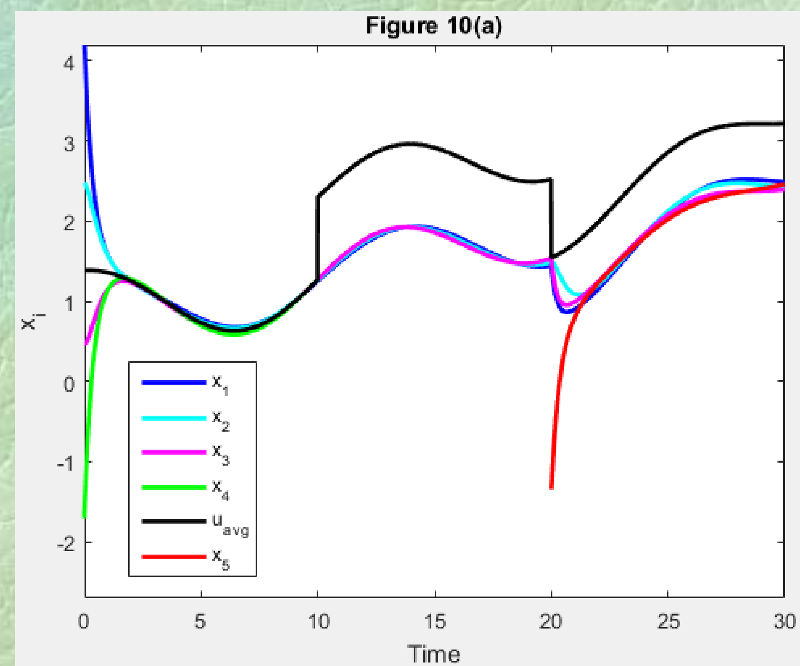
Study of agent departure and arrival (departure agent 4 at 10^s and arriving agent five at 20^s and $p^5(20)=0$)



Objectives are moving by following equations and agents want to track there mean.

$$x^l_T(t) = (t/20)^2 + 0.5\sin\left((0.35 + 0.05l)\,t + (5-l)\frac{\pi}{5}\right) + 4 - 2(l-1), \quad l \in \{1, 2, 3, 4\}$$





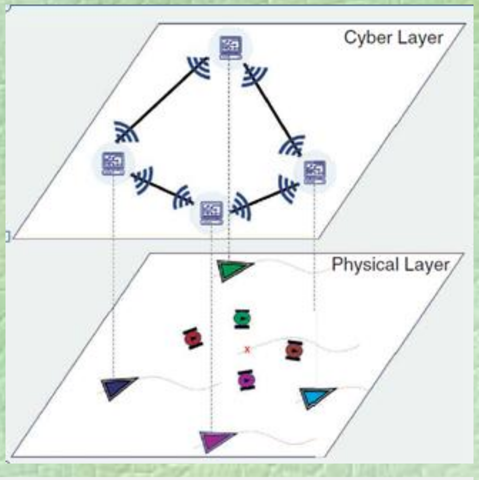
Drawbacks:

- Steady state error when the agents change.

Dynamic Average Consensus in Network Systems

Implementation of eq(16) for following system.

Perturbation of input signals $u^1(t)=x^1(t) - 4\cos t$ at $t \epsilon [0,2]$, and $t \epsilon [3,5]$

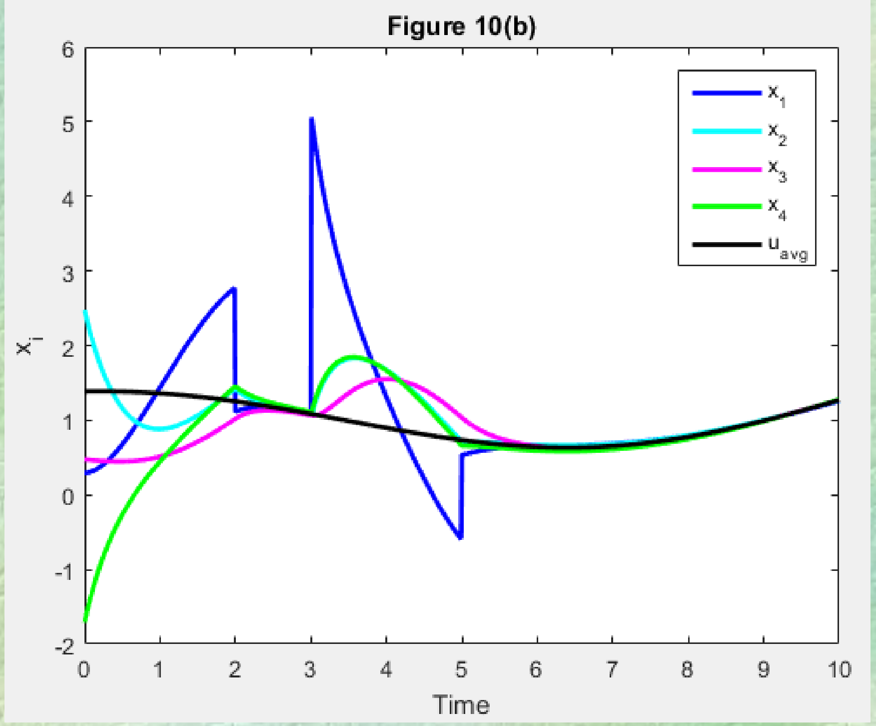


Objectives are moving by following equations and agents want to track there mean.

$$x_T^l(t) = (t/20)^2 + 0.5\sin\left((0.35 + 0.05l)t + (5-l)\frac{\pi}{5}\right) + 4 - 2(l-1), \quad l \in \{1, 2, 3, 4\}$$

Despite the perturbation, including the initial measurement error of $u^1(0)=x^1(0)-4$

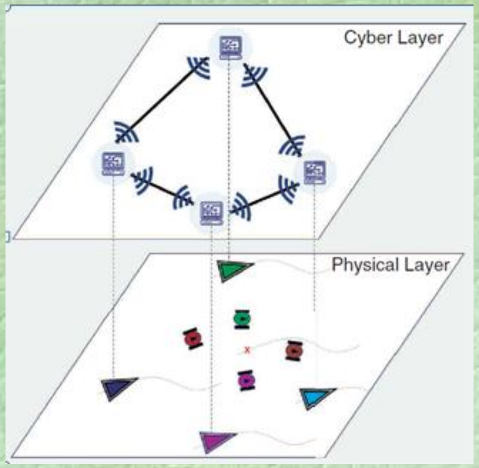
Fortunately noise compensated very good.



Dynamic Average Consensus in Network Systems

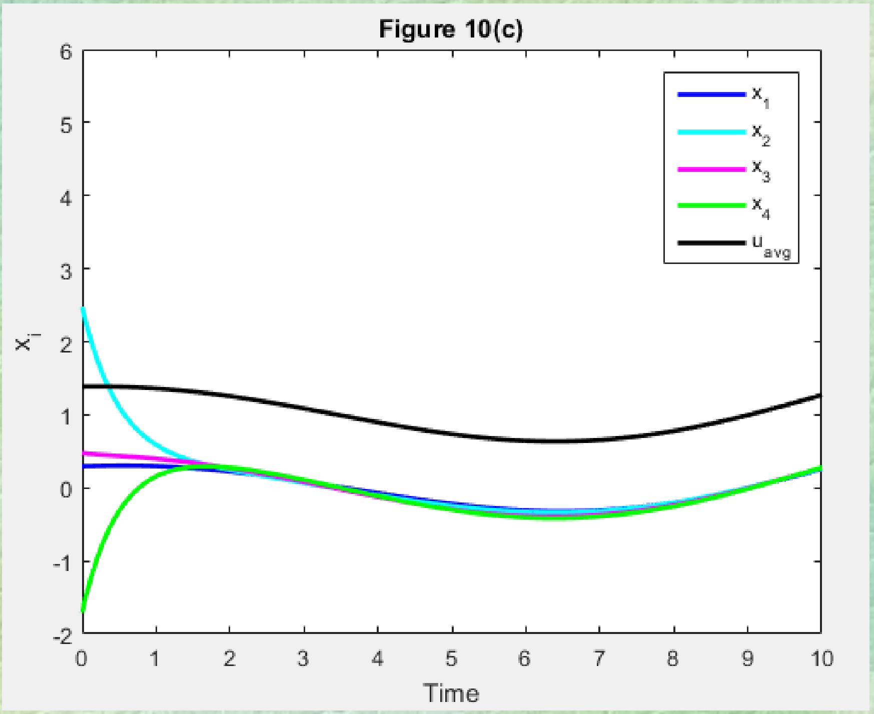
Implementation of eq(11) for following system.

Initialization error for agent one $u_1(0) = x_1(0) - 4$



Objectives are moving by following equations and agents want to track there mean.

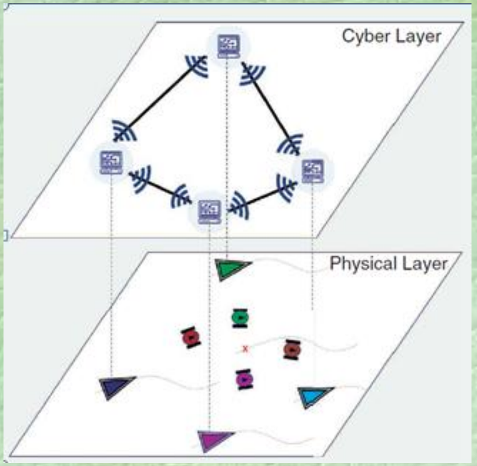
$$x_T^l(t) = (t/20)^2 + 0.5\sin\left((0.35 + 0.05l)t + (5-l)\frac{\pi}{5}\right) + 4 - 2(l-1), \quad l \in \{1, 2, 3, 4\}$$



Drawbacks:

- Steady state error when it start by nonzero initial condition.

Dynamic Average Consensus in Network Systems

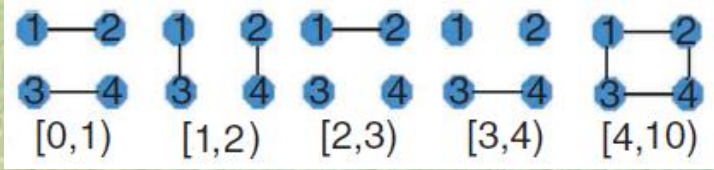


Implementation of eq(11)/eq(16) for following system.

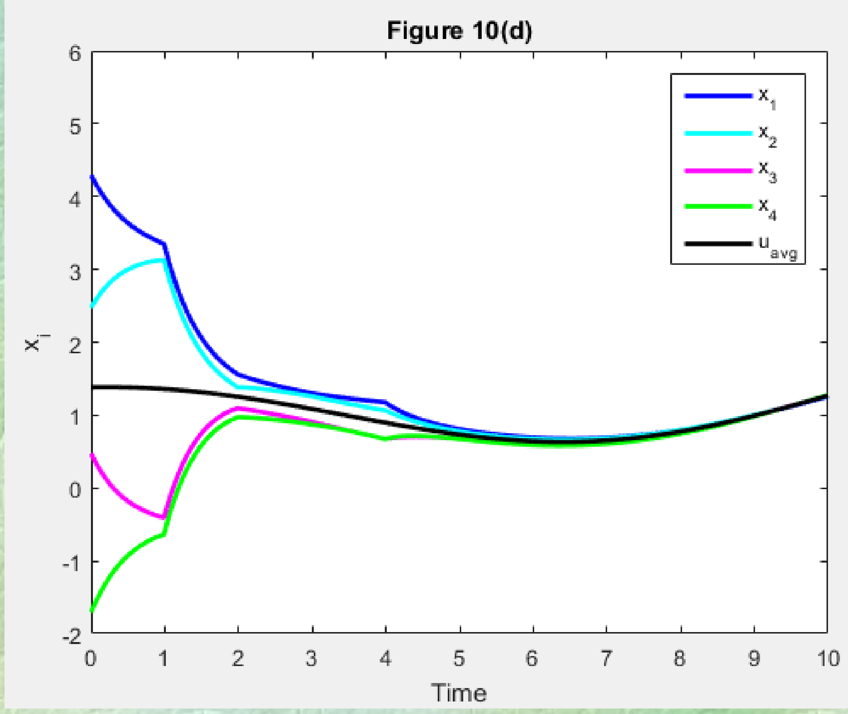
Objectives are moving by following equations and agents want to track there mean.

$$x_T^l(t) = (t/20)^2 + 0.5\sin\left((0.35 + 0.05l)t + (5-l)\frac{\pi}{5}\right) + 4 - 2(l-1), \quad l \in \{1, 2, 3, 4\}$$

Scenario 4: Graph topology changes in different times as:



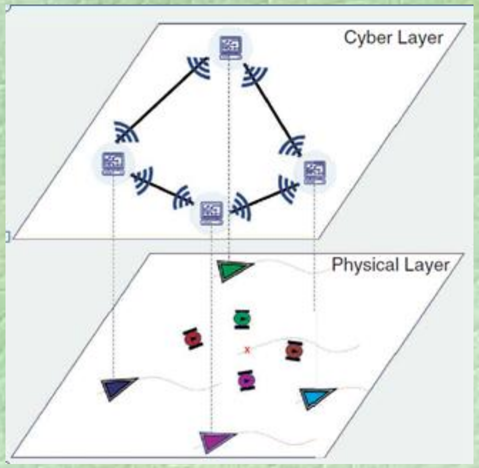
Fortunately graph changes compensated.

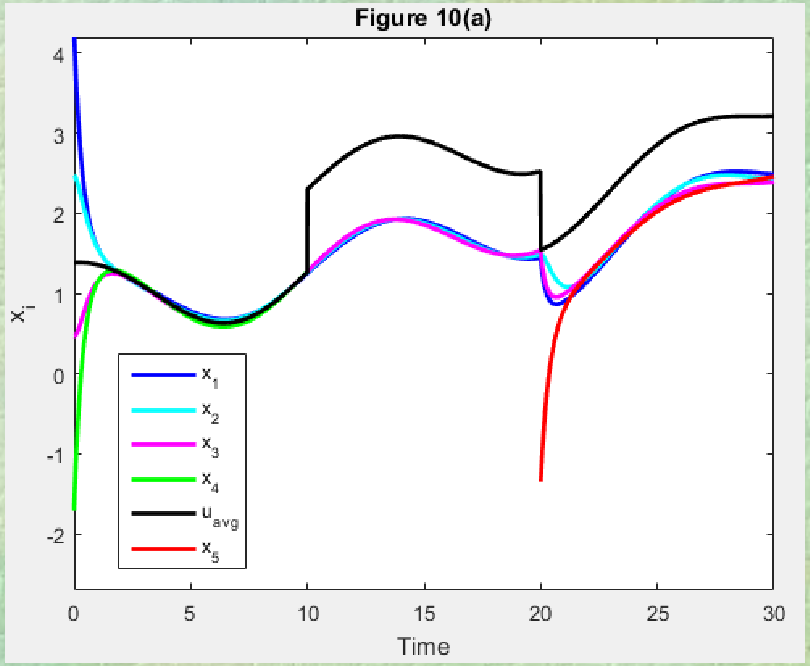


Dynamic Average Consensus in Network Systems

Implementation of eq(16) for following system.

Study of agent departure and arrival (departure agent 4 at 10^s and arriving agent five at 20^s $p^5(20)=0$)





Discussion:

- Initialization is very important in the study of departure or arriving.

We need Robustness to Initialization and Permanent Agent Dropout

Continuous-Time Dynamic Average Consensus Algorithms

Robustness to Initialization and Permanent Agent Dropout

$$\dot{q}^i(t) = -\sum_{j=1}^{N} b_{ij}(x^i - x^j), \quad (19a)$$

$$\dot{x}^i = -\alpha(x^i - u^i) - \sum_{j=1}^{N} a_{ij}(x^i - x^j) + \sum_{j=1}^{N} b_{ji}(q^i - q^j) + \dot{u}^i, \quad (19b)$$

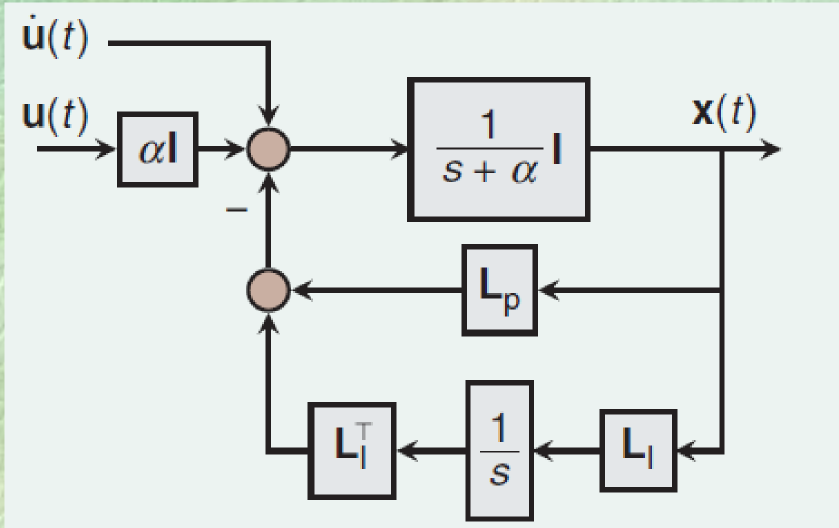
$$q^i(t_0),\, x^i(t_0) \in \mathbb{R}, \qquad i \in \{1, \ldots, N\}, \quad (19c)$$

$$\dot{\mathbf{q}} = -\mathbf{L}_I \mathbf{x}, \quad (20a)$$

$$\dot{\mathbf{x}} = -\alpha(\mathbf{x} - \mathbf{u}) - \mathbf{L}_p \mathbf{x} + \mathbf{L}_I^{\top} \mathbf{q} + \dot{\mathbf{u}}, \quad (20b)$$

$$\dot{\mathbf{x}} = -\alpha(x - \mathbf{u}) - \mathbf{L}_p \mathbf{x} - \mathbf{L}_I^{\top} \int_{t_0}^{t} \mathbf{L}_I \mathbf{x}(\tau)\, d\tau + \mathbf{L}_I^{\top} \mathbf{q}(t_0) + \dot{\mathbf{u}}.$$

In (19), the agents are allowed to use two different adjacency matrices, $[a_{ij}]$ and $[b_{ij}]$, so that they have an extra degree of freedom to adjust the tracking performance of the algorithm.

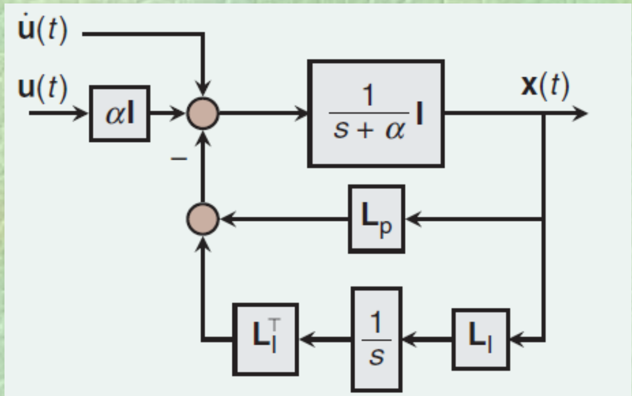


Draw back of (19):

It doesn't work for directed graph since of L_I^T in equation (20b).

Continuous-Time Dynamic Average Consensus Algorithms

Robustness to Initialization and Permanent Agent Dropout



Theorem 3: Convergence of (19)

Let $\mathbf{L}_P$ and $\mathbf{L}_I$ be Laplacian matrices corresponding to strongly connected and weight-balanced digraphs. Let $\gamma(t) = \sup_{\tau \in [t,\infty)} \|(\mathbf{I}_N - (1/N)\mathbf{1}_N\mathbf{1}_N^\top)\dot{\mathbf{u}}(\tau)\| < \infty$

Starting from any initial condition x(0) , q(0) and for any $\alpha \in R_{>0}$ the trajectories of (19) satisfy

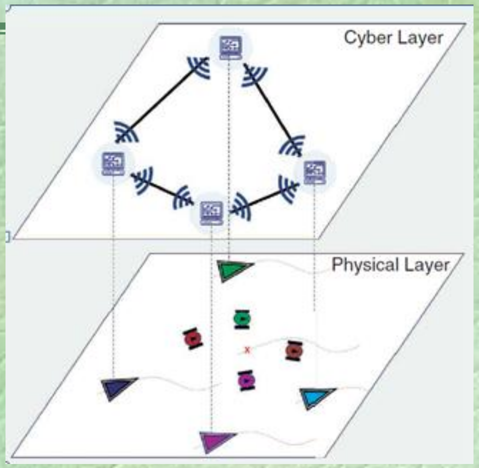
$$\lim_{t\to\infty} |x^i(t) - \mathrm{u}^{\mathrm{avg}}(t)| \le \frac{\kappa\|\mathbf{B}\|\gamma(\infty)}{\underline{\lambda}}, \quad i \in \{1,\ldots,N\},$$

Moreover

$$\sum_{j=1}^{N} x^j(t) = \sum_{j=1}^{N} \mathrm{u}^j(t) + \mathbf{e}^{-\alpha(t-t_0)}\left(\sum_{j=1}^{N} x^j(t_0) - \sum_{j=1}^{N} \mathrm{u}^j(t_0)\right),$$

for $t \in [t_0, \infty)$

Continuous-Time Dynamic Average Consensus Algorithms



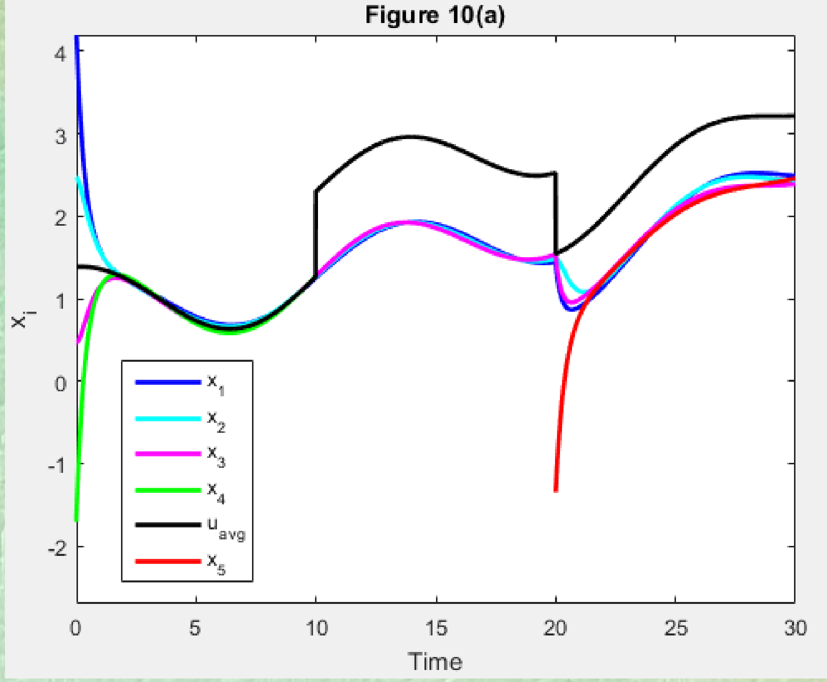
Comparing eq.(19) with eq.(16) on agent departure and arrival

Objectives are moving by following equations and agents want to track there mean.

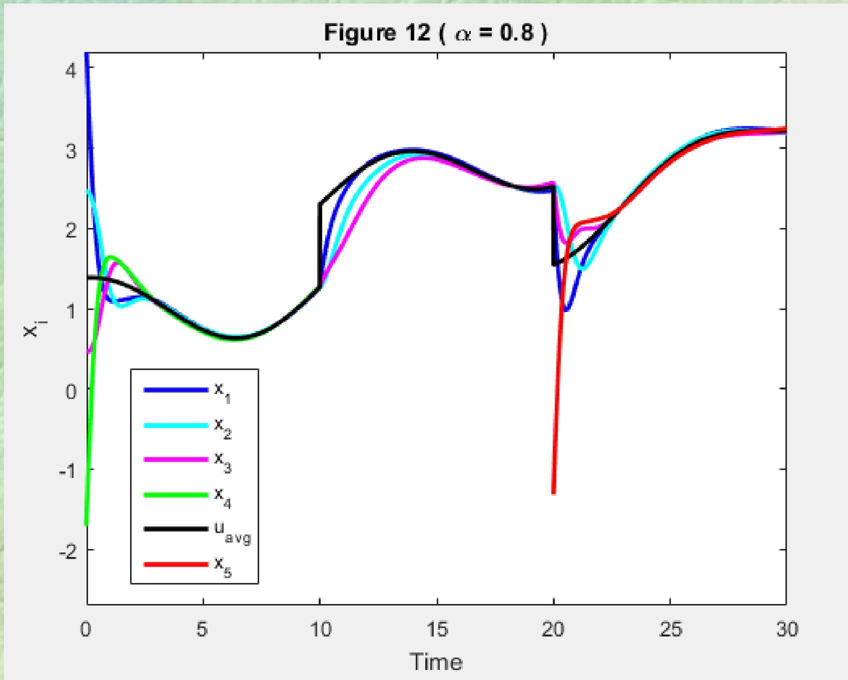
$$x_T^l(t) = (t/20)^2 + 0.5\sin\left((0.35 + 0.05l)t + (5-l)\frac{\pi}{5}\right) + 4 - 2(l-1), \quad l \in \{1, 2, 3, 4\}$$

Study of agent departure and arrival (departure agent 4 at 10^s and arriving agent five at 20^s and $p^5(20)=0$)

Simulation for eq. (16)



Simulation for eq. (19) or (20)



Continuous-Time Dynamic Average Consensus Algorithms

Controlling the Rate of Convergence

All consensus algorithms presented in (11) or (16) and also (19) are that the rate of convergence is dictated by network topology as well as some algorithm parameters.

However, in some applications, the task is not just to obtain the average of the dynamic inputs but rather to physically track this value, possibly with limited control authority.

To allow the network to prespecify its desired worst rate of convergence β, *first-order-input dynamic consensus* (*FOI-DC*) proposed.

$$\begin{cases} \epsilon \dot{q}^i = -\sum_{j=1}^{N} b_{ij}(z^i - z^j), \\ \epsilon \dot{z}^i = -(z^i + \beta \mathsf{u}^i + \dot{\mathsf{u}}^i) - \sum_{j=1}^{N} a_{ij}(z^i - z^j) + \sum_{j=1}^{N} b_{ji}(q^i - q^j), \end{cases} \tag{24a}$$

$$\dot{x}^i = -\beta x^i - z^i, \quad i \in \{1, \ldots, N\}. \tag{24b}$$

Continuous-Time Dynamic Average Consensus Algorithms

Controlling the Rate of Convergence

A small value for ϵ in (24a) leads to fast dynamic.

$$\begin{cases} \epsilon\dot{q}^i = -\sum_{j=1}^{N} b_{ij}(z^i - z^j), \\ \epsilon\dot{z}^i = -(z^i + \beta u^i + \dot{u}^i) - \sum_{j=1}^{N} a_{ij}(z^i - z^j) + \sum_{j=1}^{N} b_{ji}(q^i - q^j), \end{cases} \quad (24a)$$

$$\dot{x}^i = -\beta x^i - z^i, \quad i \in \{1,\ldots,N\}. \quad (24b)$$

The slow dynamics (24b) then uses the signal generated by the fast dynamics to track the average of the reference signal across the network at a prespecified smaller rate β.

$$\left|e^i(t)\right| \leq \mathbf{e}^{-\beta(t-t_0)}\left|e^i(t_0)\right| + \frac{\kappa}{\beta}\sup_{t_0\leq\tau\leq t}\left(\mathbf{e}^{-\epsilon^{-1}\underline{\lambda}(t-t_0)}\left\|\begin{bmatrix}\mathbf{y}(t_0)\\ \mathbf{e}_z(t_0)\end{bmatrix}\right\| + \frac{\epsilon\|\bar{\mathbf{B}}\|}{\underline{\lambda}}\sup_{t_0\leq\tau\leq t}\left\|\beta\dot{\mathbf{u}}(\tau) + \ddot{\mathbf{u}}(\tau)\right\|\right)$$

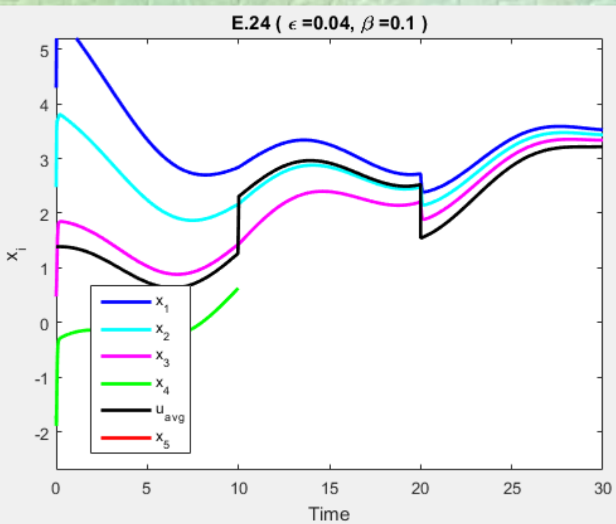
No need to initialization.

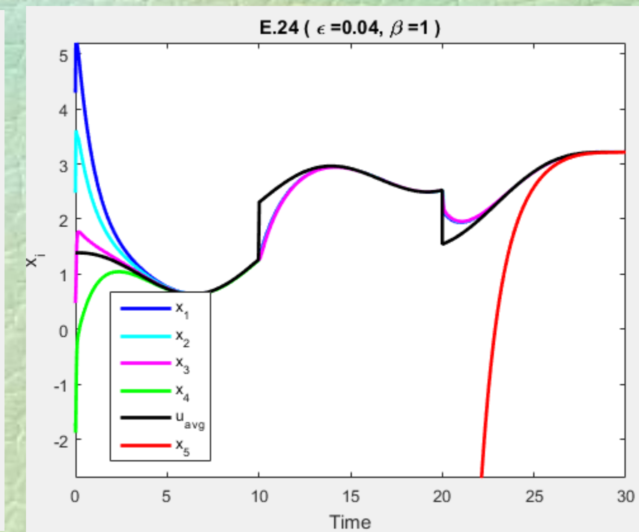
For small value of ϵ (fast dynamic in 24a)

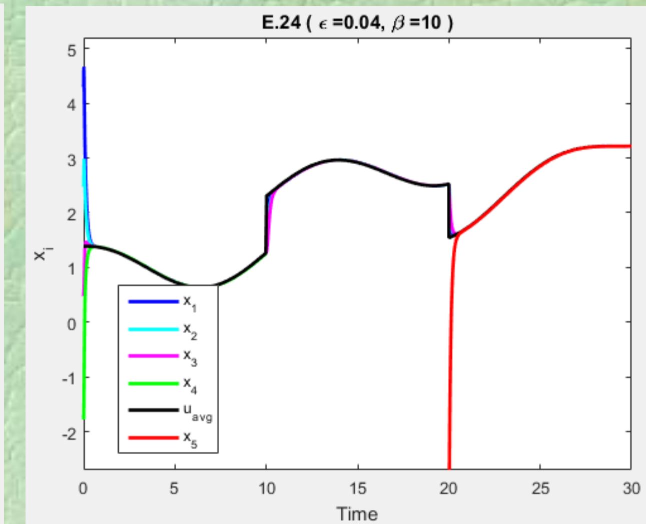
$$\left|e^i(t)\right| \leq \mathbf{e}^{-\beta(t-t_0)}\left|e^i(t_0)\right|$$

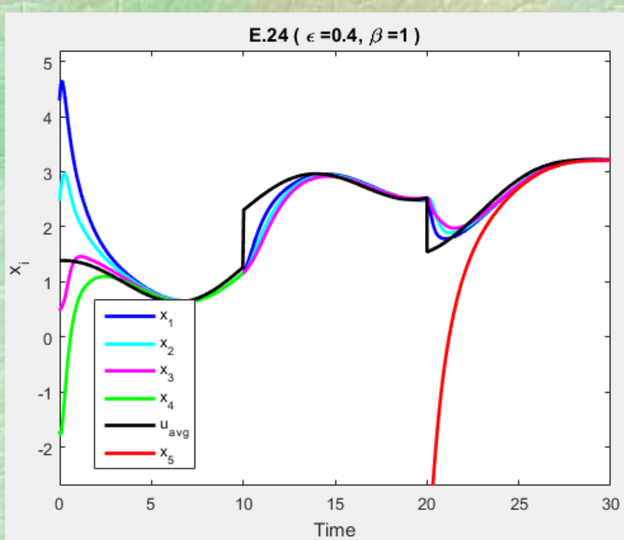
Continuous-Time Dynamic Average Consensus Algorithms

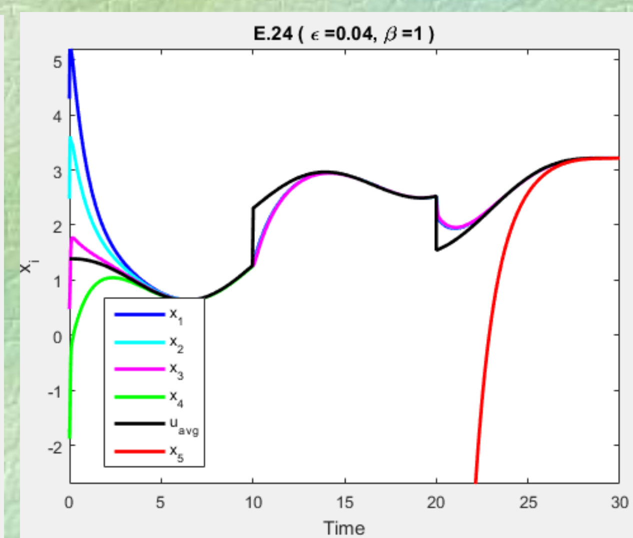
Controlling the Rate of Convergence

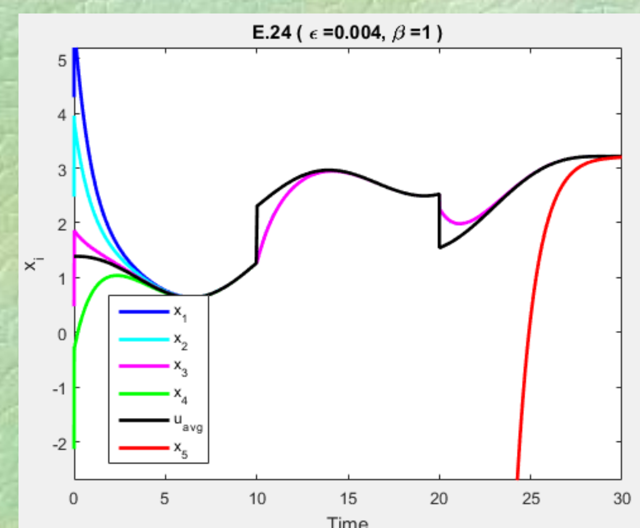












Continuous-Time Dynamic Average Consensus Algorithms

An Alternative Algorithm for Directed Graphs

As observed, eq.(19/20) is not implementable over directed graphs because of (L_I^T) .

$$\dot{\mathbf{q}} = -\mathbf{L}_{\mathrm{I}}\mathbf{x}, \tag{20a}$$

$$\dot{\mathbf{x}} = -\alpha(\mathbf{x} - \mathbf{u}) - \mathbf{L}_{\mathrm{P}}\mathbf{x} + \mathbf{L}_{\mathrm{I}}^{\top}\mathrm{q} + \dot{\mathbf{u}}, \tag{20b}$$

Authors of [19], proposed a modified proportional and integral agreement feedback dynamic average consensus algorithm whose implementation does not require the agents to know their respective columns of the Laplacian (L_I^T). This algorithm is:

$$\dot{q}^i = \alpha\beta\sum_{j=1}^{N} \mathsf{a}_{ij}(x^i - x^j), \tag{25a}$$

$$\dot{x}^i = -\alpha(x^i - \mathsf{u}^i) - \beta\sum_{j=1}^{N} \mathsf{a}_{ij}(x^i - x^j) - q^i + \dot{\mathsf{u}}^i, \tag{25b}$$

$$x^i(t_0), q^i(t_0) \in \mathbb{R} \text{ s.t. } \sum_{j=1}^{N} q^j(t_0) = 0, \tag{25c}$$

Equivalently:

$$\dot{\mathbf{x}} = -\alpha(\mathbf{x} - \mathbf{u}) - \beta\mathbf{L}\mathbf{x} - \alpha\beta\int_{t_0}^{t} \mathbf{L}\mathbf{x}(\tau)\,\mathrm{d}\tau - \mathbf{q}(t_0) + \dot{\mathbf{u}},$$

An Alternative Algorithm for Directed Graphs

$$\dot{q}^i = \alpha\beta \sum_{j=1}^{N} \mathsf{a}_{ij}(x^i - x^j), \tag{25a}$$

$$\dot{x}^i = -\alpha(x^i - \mathsf{u}^i) - \beta \sum_{j=1}^{N} \mathsf{a}_{ij}(x^i - x^j) - q^i + \dot{\mathsf{u}}^i, \tag{25b}$$

$$x^i(t_0), q^i(t_0) \in \mathbb{R} \ \text{ s.t. } \sum_{j=1}^{N} q^j(t_0) = 0, \tag{25c}$$

$$\dot{\mathbf{x}} = -\alpha(\mathbf{x} - \mathbf{u}) - \beta\mathbf{L}\mathbf{x} - \alpha\beta\int_{t_0}^{t} \mathbf{L}\mathbf{x}(\tau)\,\mathbf{d}\tau - \mathbf{q}(t_0) + \dot{\mathbf{u}},$$

Theorem 4: Convergence of (25) Over Strongly Connected and Weight-Balanced Digraphs for Dynamic Input Signals [19]

Let G be a strongly connected and weight-balanced digraph. If $\sum_{j=1}^{N} q^j(t_0) = 0$ then for any $\alpha, \beta \in R_{>0}$ the the trajectory of (25) satisfy

$$\lim_{t\to\infty} \left| x^i(t) - \mathsf{u}^{\text{avg}}(t) \right| \le \frac{\gamma(\infty)}{\beta\hat{\lambda}_2}, \qquad i \in \{1, \ldots, N\}, \tag{26}$$

where $\gamma(t) = \sup_{\tau\in[t,\infty)} \|(\mathbf{I}_N - (1/N)\mathbf{1}_N\mathbf{1}_N^{\top})\dot{\mathbf{u}}(\tau)\| < \infty$ and the convergence rate to the error bound is $\min\{\alpha, \beta Re(\lambda_2)\}$.

Continuous-Time Dynamic Average Consensus Algorithms

An Alternative Algorithm for Directed Graphs

Compact form of (25) $\dot{\mathbf{x}} = -\alpha(\mathbf{x}-\mathbf{u}) - \beta\mathbf{L}\mathbf{x} - \alpha\beta\int_{t_0}^{t}\mathbf{L}\mathbf{x}(\tau)\,d\tau - \mathbf{q}(t_0) + \dot{\mathbf{u}},$

Compact form of (11) $\dot{\mathbf{x}}(t) = -\mathbf{L}x(t) + \dot{\mathbf{u}}(t),$

Compact form of (16) $\dot{p}(t) = -\mathbf{L}x(t) + \dot{\mathbf{u}}(t), \qquad x(t) = \mathbf{u}(t) - p(t).$

Remark1: $U(s)$ → X(s) is the same for all of systems(11, 16 and 25)

Remark2: Both 25 and 16 (unlike 11) enjoy robustness to reference signal measurement perturbations and naturally preserves the privacy of the input of each agent against adversaries.

Remark3: Specifically, an adversary with access to the time history of all network communication messages cannot uniquely reconstruct the reference signal of any agent in (25), which is not the case for (16) and (11)??!!.

Continuous-Time Dynamic Average Consensus Algorithms

Review of different continuous-time dynamics.

$$\dot{x}^i(t) = -\sum_{j=1}^{N} a_{ij}(x^i(t) - x^j(t)) + \dot{\mathrm{u}}^i(t), \quad i \in \{1,\ldots,N\},$$
$$x^i(0) = \mathrm{u}^i(0). \tag{11}$$

$$\dot{q}^i(t) = -\sum_{j=1}^{N} b_{ij}(x^i - x^j), \tag{19}$$
$$\dot{x}^i = -\alpha(x^i - u^i) - \sum_{j=1}^{N} a_{ij}(x^i - x^j) + \sum_{j=1}^{N} b_{ji}(q^i - q^j) + \dot{\mathrm{u}}^i,$$
$$q^i(t_0),\, x^i(t_0) \in \mathbb{R}, \qquad i \in \{1,\ldots,N\},$$

$$\begin{cases} \epsilon\dot{q}^i = -\sum_{j=1}^{N} b_{ij}(z^i - z^j), \\ \epsilon\dot{z}^i = -(z^i + \beta\mathrm{u}^i + \dot{\mathrm{u}}^i) - \sum_{j=1}^{N} a_{ij}(z^i - z^j) + \sum_{j=1}^{N} b_{ji}(q^i - q^j), \end{cases} \tag{24}$$
$$\dot{x}^i = -\beta x^i - z^i, \quad i \in \{1,\ldots,N\}.$$

$$\dot{q}^i = \alpha\beta \sum_{j=1}^{N} \mathrm{a}_{ij}(x^i - x^j), \tag{25}$$
$$\dot{x}^i = -\alpha(x^i - \mathrm{u}^i) - \beta \sum_{j=1}^{N} \mathrm{a}_{ij}(x^i - x^j) - q^i + \dot{\mathrm{u}}^i,$$
$$x^i(t_0),\, q^i(t_0) \in \mathbb{R} \text{ s.t. } \sum_{j=1}^{N} q^j(t_0) = 0,$$

Algorithm	(11)	(19)	(24)	(25)
$J^i(t)$	$\{x^i(t), \dot{\mathrm{u}}(t)\}$	$\{x^i(t), q^i(t), \mathrm{u}(t)\}$	$\{x^i(t), z^i(t), q^i(t), \mathrm{u}(t), \dot{\mathrm{u}}(t)\}$	$\{x^i(t), q^i(t), \mathrm{u}(t), \dot{\mathrm{u}}(t)\}$
$\{I^j(t)\}_{j \in \mathcal{N}^i_{\mathrm{out}}}$	$\{x^j(t)\}_{j \in \mathcal{N}^i_{\mathrm{out}}}$	$\{x^j(t), q^i(t)\}_{j \in \mathcal{N}^i_{\mathrm{out}}}$	$\{z^j(t), v^j(t)\}_{j \in \mathcal{N}^i_{\mathrm{out}}}$	$\{x^j(t), q^i(t)\}_{j \in \mathcal{N}^i_{\mathrm{out}}}$
Initialization requirement	$x^i(0) = \mathrm{u}^i(0)$	None	None	$\sum_{j=1}^{N} q^i(0) = 0$

Discrete-Time Dynamic Average Consensus Algorithms

Implementing of continuous-time algorithms on practical cyber-physical systems requires continuous communication between agents.

This requirement is not feasible in practice due to constraints on the communication bandwidth. To address this issue, the discrete-time dynamic average consensus algorithms where the communication among agents occurs only at discrete-time steps are studied.

In continuous time, the parameters may be scaled to achieve any desired convergence rate, whereas in discrete time, the parameters must be carefully chosen to ensure convergence.

Here, a simple method using root locus techniques for choosing the parameters to optimize the convergence rate is provided. It is also shown how to further accelerate the convergence by introducing extra dynamics into the dynamic average consensus algorithm.

TABLE 2 The arguments of the driving command in (2) for the reviewed discrete-time dynamic average consensus algorithms together with their initialization requirements.

Algorithm	(27)	(29)	(30)	(31)
$J^i(t)$	$\{\mathrm{u}_k^i, p_k^i\}$	$\{\mathrm{u}_k^i, p_k^i, p_{k-1}^i\}$	$\{\mathrm{u}_k^i, p_k^i, q_k^i\}$	$\{\mathrm{u}_k^i, p_k^i, p_{k-1}^i, q_k^i, q_{k-1}^i\}$
$\{I^j(t)\}_{j \in \mathcal{N}_{\text{out}}^i}$	$\{x_k^j\}_{j \in \mathcal{N}_{\text{out}}^i}$	$\{x_k^j\}_{j \in \mathcal{N}_{\text{out}}^i,}$	$\{x_k^j, p_k^j\}_{j \in \mathcal{N}_{\text{out}}^i}$	$\{x_k^j, p_k^j\}_{j \in \mathcal{N}_{\text{out}}^i}$
Initialization requirement	$\sum_{j=1}^{N} p_0^j = 0$	$\sum_{j=1}^{N} p_0^j = 0$	None	None

Discrete-Time Dynamic Average Consensus Algorithms

For simplicity of exposition, assume the communication graph is constant, connected, and undirected so,

Laplacian matrix is then symmetric ⟹ Real eigenvalues ⟹ $\lambda_1=0, \lambda_2>\lambda_1, \ldots, \lambda_n$

Suppose we have lower and upper bounds on λ_2 and λ_n.

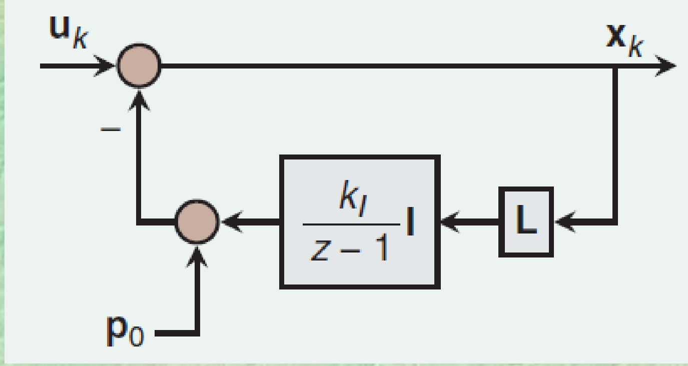
$$\dot{p}^i(t) = \sum_{j=1}^{N} a_{ij}(x^i(t) - x^j(t)), \quad \sum_{j=1}^{N} p^j(t_0) = 0, \quad (16a)$$

$$x^i(t) = \mathsf{u}^i(t) - p^i(t). \quad (16b)$$

⟹

$$p^i_{k+1} = p^i_k + k_I \sum_{j=1}^{N} a_{ij}(x^i_k - x^j_k), \quad p^i_0 \in \mathbb{R}, \quad (27a)$$

$$x^i_k = \mathsf{u}^i_k - p^i_k, \quad (27b)$$

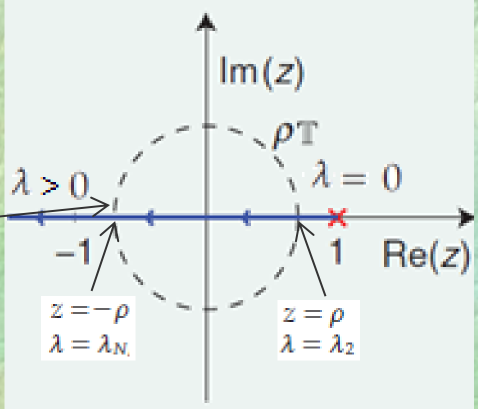


$$0 = z\mathbf{I} - (\mathbf{I} - k_I\mathbf{L}).$$

The characteristic equation corresponding to the eigenvalue of **L** is then

$$0 = 1 + \lambda \frac{k_I}{z-1}.$$

Good convergence if eigenvalues are in the circle



$$k_I = \frac{2}{\lambda_2 + \lambda_N} \quad \text{and} \quad \rho = \frac{\lambda_N - \lambda_2}{\lambda_N + \lambda_2}.$$

Discrete-Time Dynamic Average Consensus Algorithms

While the previous choice of parameters optimizes the convergence rate, even faster convergence can be achieved by introducing extra dynamics into the dynamic average consensus algorithm.

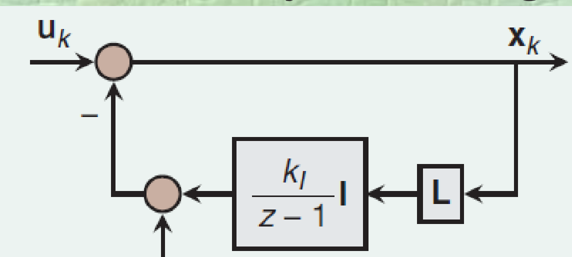


Figure 13(a)

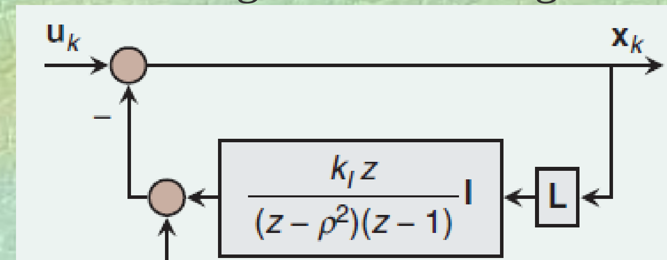
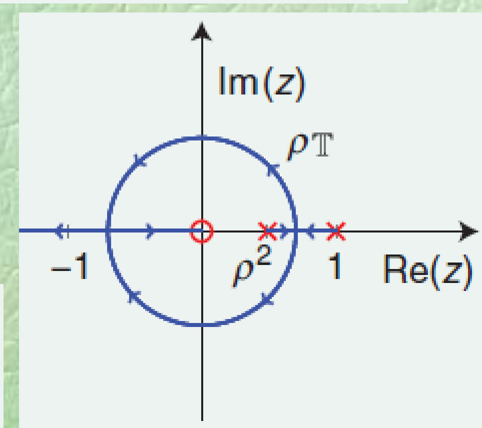


Figure 13(c)

$$p^i_{k+1} = (1+\rho^2)p^i_k - \rho^2 p^i_{k-1} + k_I \sum_{j=1}^{N} a_{ij}(x^i_k - x^j_k),$$

$$p^i_0 \in \mathbb{R}, \quad i \in \{1,\dots,N\}, \qquad (29a)$$

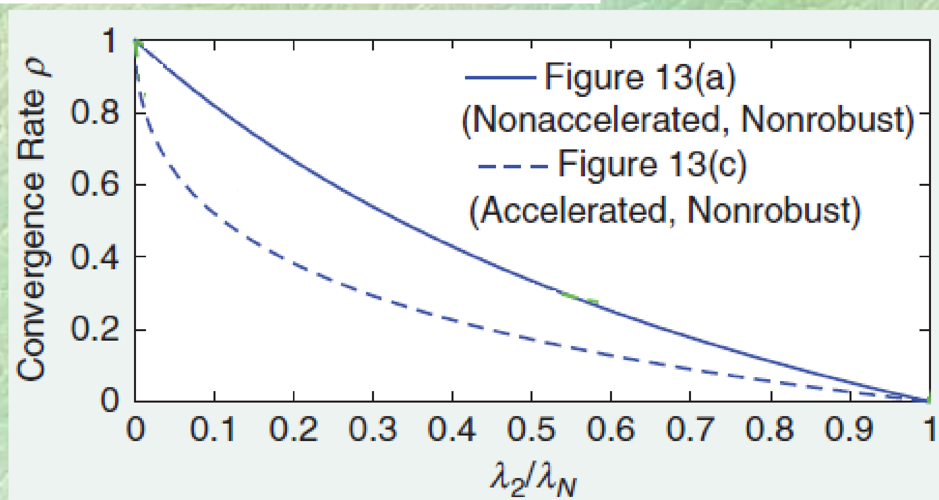
$$x^i_k = u^i_k - p^i_k. \qquad (29b)$$



(b)

$$k_I = \frac{4}{(\sqrt{\lambda_2}+\sqrt{\lambda_N})^2}$$

$$\rho = \frac{\sqrt{\lambda_N}-\sqrt{\lambda_2}}{\sqrt{\lambda_N}+\sqrt{\lambda_2}}.$$



Discrete-Time Dynamic Average Consensus Algorithms

Robust Dynamic Average Consensus Algorithms

$$\dot{q}^i(t) = -\sum_{j=1}^{N} b_{ij}(x^i - x^j), \quad (19a)$$

$$\dot{x}^i = -\alpha(x^i - u^i) - \sum_{j=1}^{N} a_{ij}(x^i - x^j) + \sum_{j=1}^{N} b_{ji}(q^i - q^j) + \dot{u}^i, \quad (19b)$$

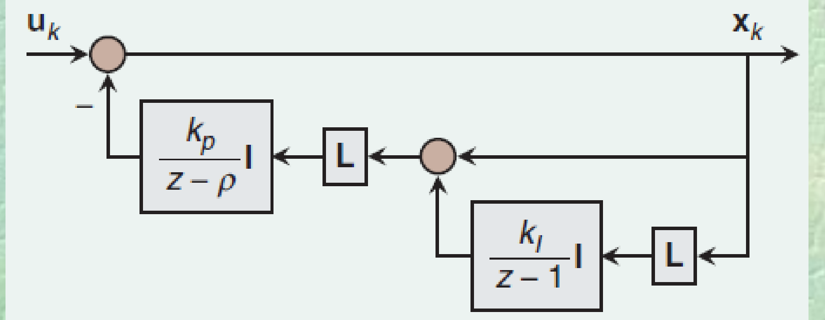
$$q^i(t_0), x^i(t_0) \in \mathbb{R}, \qquad i \in \{1, \ldots, N\}, \quad (19c)$$

⇒

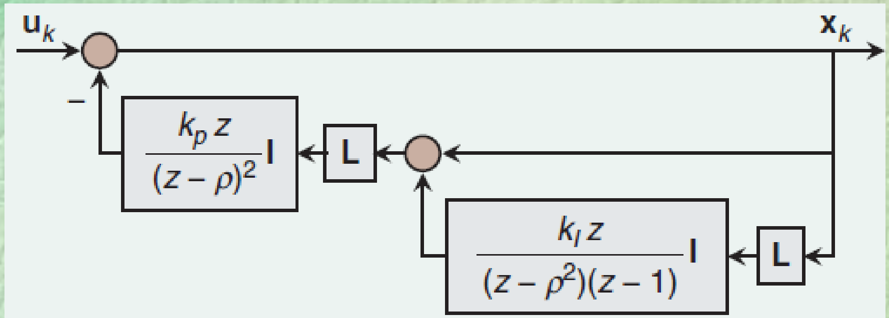
$$q^i_{k+1} = \rho q^i_k + k_p \sum_{j=1}^{N} a_{ij}((x^i_k - x^j_k) + (p^i_k - p^j_k)), \quad (30a)$$

$$p^i_{k+1} = p^i_k + k_I \sum_{j=1}^{N} a_{ij}(x^i_k - x^j_k), \quad (30b)$$

$$x^i_k = u^i_k - q^i_k, \qquad p^i_0, q^i_0 \in \mathbb{R}, \qquad i \in \{1, \ldots, N\}, \quad (30c)$$



Accelerated version using extra dynamics, given by



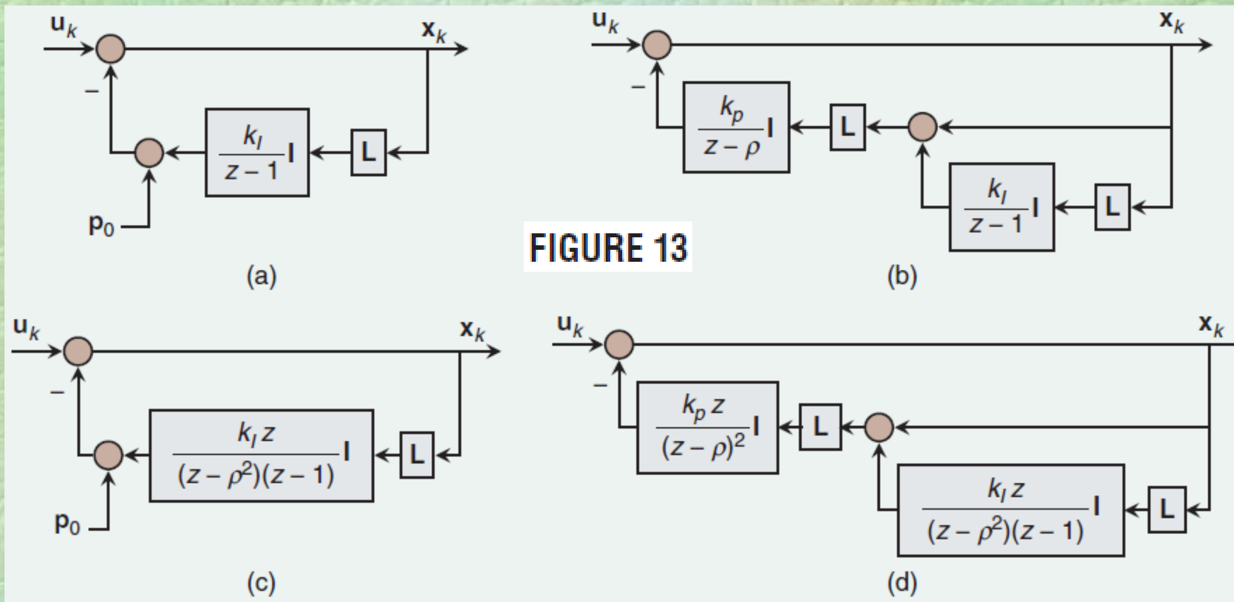
$$q^i_{k+1} = 2\rho q^i_k - \rho^2 q^i_{k-1} + k_p \sum_{j=1}^{N} a_{ij}((x^i_k - x^j_k) + (p^i_k - p^j_k)), \quad (31a)$$

$$p^i_{k+1} = (1 + \rho^2) p^i_k - \rho^2 p^i_{k-1} + k_I \sum_{j=1}^{N} a_{ij}(x^i_k - x^j_k), \quad (31b)$$

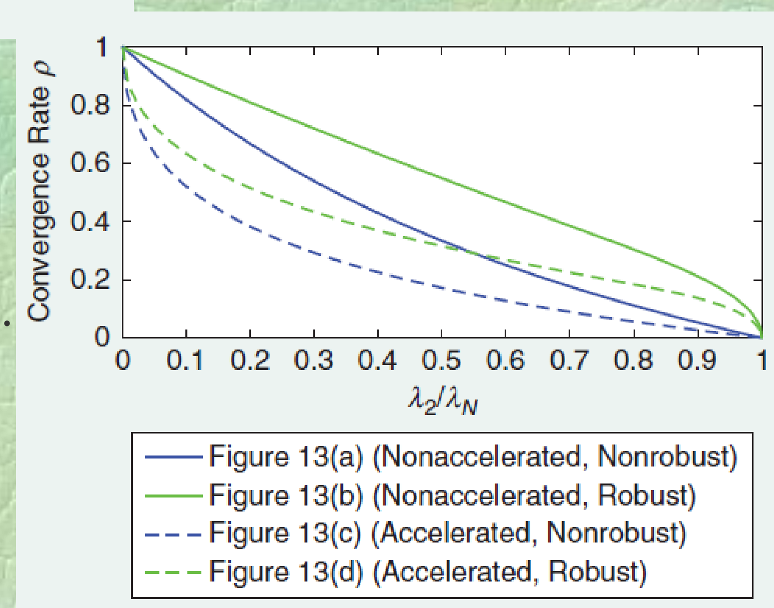
$$x^i_k = u^i_k - q^i_k, \qquad p^i_0, q^i_0 \in \mathbb{R}, \qquad i \in \{1, \ldots, N\}, \quad (31c)$$

Details of the results for this two algorithms can be found in [57].

Discrete-Time Dynamic Average Consensus Algorithms



Although the convergence rates of the standard and accelerated PI dynamic average consensus algorithms (Figure 13 b and d) are slower than those of (Figure 13 a and c), respectively, they have the additional advantage of being robust to initial conditions.



Discrete-Time Dynamic Average Consensus Algorithms

Theorem 5: Optimal Convergence Rates of Discrete-Time Dynamic Average Consensus Algorithms

Let G be a connected, undirected graph. Suppose the reference signal $\mathbf{u}_i$ at each agent is a constant scalar. Consider the dynamic average consensus algorithms in Figure 13, with the parameters chosen according to Table 3 [the algorithms in Figure 13(a) and (c) are initialized such that the average of the initial integrator states is zero]. The agreement states x_i converge to $\mathbf{u}^{avg}$ exponentially with rate ρ.

TABLE 3 The parameter selection for the dynamic average consensus algorithms of Figure 13 as a function of the minimum and maximum nonzero Laplacian eigenvalues λ_2 and λ_N, respectively, with $\lambda_r := \lambda_2/\lambda_N$. N/A: not applicable.

	ρ	k_I	k_p
Figure 13(a)	$\frac{\lambda_N-\lambda_2}{\lambda_N+\lambda_2}$	$\frac{2}{\lambda_2+\lambda_N}$	N/A
Figure 13(b)	$\begin{cases}\frac{8-8\lambda_r+\lambda_r^2}{8-\lambda_r^2}, & 0<\lambda_r\le 3-\sqrt{5}\\ \frac{\sqrt{(1-\lambda_r)(4+\lambda_r^2(5-\lambda_r))}-\lambda_r(1-\lambda_r)}{2(1+\lambda_r^2)}, & 3-\sqrt{5}<\lambda_r\le 1\end{cases}$	$\frac{1-\rho}{\lambda_2}$	$\frac{1}{\lambda_N}\frac{\rho(1-\rho)\lambda_r}{\rho+\lambda_r-1}$
Figure 13(c)	$\frac{\sqrt{\lambda_N}-\sqrt{\lambda_2}}{\sqrt{\lambda_N}+\sqrt{\lambda_2}}$	$\frac{4}{(\sqrt{\lambda_2}+\sqrt{\lambda_N})^2}$	N/A
Figure 13(d)	$\begin{cases}\frac{6-2\sqrt{1-\lambda_r}+\lambda_r-4\sqrt{2-2\sqrt{1-\lambda_r}+\lambda_r)}}{2+2\sqrt{1-\lambda_r}-\lambda_r}, & 0<\lambda_r\le 2(\sqrt{2}-1)\\ \frac{-3-2\sqrt{1-\lambda_r}+\lambda_r+2\sqrt{2+2\sqrt{1-\lambda_r}-\lambda_r}}{-1-2\sqrt{1-\lambda_r}+\lambda_r}, & 2(\sqrt{2}-1)<\lambda_r\le 1\end{cases}$	$\frac{(1-\rho)^2}{\lambda_2}$	$(2+2\sqrt{1-\lambda_r}-\lambda_r)k_I$

Perfect Tracking Using a Prior Knowledge of the Input Signals

The design of the dynamic average consensus algorithms described in the discussion so far does not require prior knowledge of the reference signals and is therefore broadly applicable.

↓

The convergence guarantees of these algorithms are strong only when the reference signals are constant or slowly varying. The error of such algorithms can be large, however, when the reference signals change quickly in time.

This section describes dynamic average consensus algorithms, which are capable of tracking fast time-varying signals with either zero or small steady-state error.

In each case, their design assumes some specific information about the nature of the reference signals.

1) Reference signals have a known model,

2) Reference signals are band limited,

3) Reference signals have bounded derivatives.

References

[1]- “Tutorial on Dynamic Average Consensus” IEEE Control System Magazine, June 2019

Appendix (Basic Notions from Graph Theory)

A weighted digraph is a triplet G = (V, E, A), where (V, E) is a digraph and $A \in R^{N\times N}$ is a weighted adjacency matrix with the property that $a_{ij} > 0$ if $(i, j)\in E$ otherwise $a_{ij} = 0$.

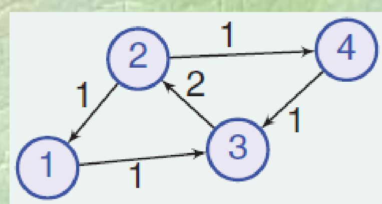
$E \in V\times V$ is the edge set. An edge from i to j, denoted by (i,j) means that agent *j* can send information to agent *i*.

The out-degree matrix $\mathbf{D}^{out}$ is the diagonal matrix with entries $\mathbf{D}_{ii}^{out} = d^{out}(i)$, for all $i \in V$.

A digraph is weight balanced if, at each node $i\in V$, the weighted out-degree and weighted in-degree coincide.

The (out-) Laplacian matrix is **L** = **D**out - **A**.

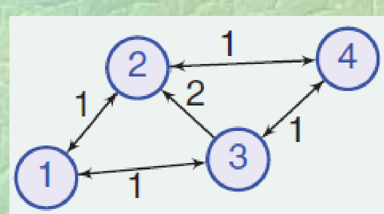
A weight balanced digraph:



$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ -1 & 2 & 0 & -1 \\ 0 & -2 & 2 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

A weighted digraph is undirected if $a_{ij} = a_{ji}$ for all $i, j \in V$.

A weight undirected graph



$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 4 & -2 & -1 \\ -1 & -2 & 4 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}$$

Appendix (Basic Notions from Graph Theory)

Property of Laplacian matrix:

Based on the structure of **L**, at least one of the eigenvalues of **L** is zero and the rest of them have nonnegative real parts.

For strongly connected digraphs, rank(**L**) = N - 1.

In a strongly connected digraph it is possible to reach any node starting from any other node by traversing edges in the direction(s) in which they point.

For strongly connected and weight-balanced digraphs, denote the eigenvalues of Sym(**L**) = (**L** + **L**^{T}) /2 one of them is zero and others nonnegative real.

For strongly connected and weight-balanced digraphs,

$$0 < \hat{\lambda}_2 \mathbf{I} \le \mathbf{R}^\top \mathrm{Sym}(\mathbf{L}) \mathbf{R} \le \hat{\lambda}_N \mathbf{I},$$

where $\mathbf{R} \in \mathbb{R}^{N \times (N-1)}$ satisfies $[(1/\sqrt{N}) \mathbf{1}_N \;\; \mathbf{R}][(1/\sqrt{N}) \mathbf{1}_N \;\; \mathbf{R}]^\top = [(1/\sqrt{N}) \mathbf{1}_N \;\; \mathbf{R}]^\top [(1/\sqrt{N}) \mathbf{1}_N \;\; \mathbf{R}] = \mathbf{I}_N.$

Intuitively, the Laplacian matrix can be viewed as a diffusion operator over the graph.

$$[\mathbf{L}\mathbf{x}]_i = \sum_{j \in \mathcal{V}} a_{ij}(x^i - x^j),$$

The smallest nonzero eigenvalue $\hat{\lambda}_2$ of the symmetric part of the graph(strongly connected ansd balance) Laplacian is a measure of connectivity of a graph. (next slide)