# 3 Consensus in multi-agent systems

*This chapter introduces the consensus problem for multi-agent systems and gives solutions to this problem for various communication constraints. The main consensus condition requires that the communication structure has to possess a spanning tree. The results of this chapter provide the basis for the synchronisation methods described in Chapter 4.*

## 3.1 Consensus problem

The consensus problem is an important step towards the control of multi-agent systems. A networked controller should make the state $x_i(t)$ of all agents to asymptotically approach a common value $\bar{x}$

$$\lim_{t\to\infty} x_i(t) = \bar{x}, \quad i = 1, 2, ..., N,$$

which is called the *consensus value*. A crucial aspect of this problem is the fact that $\bar{x}$ is not prescribed by the control task but appears as the result of "negotiations" among the agents. It will be shown later how $\bar{x}$ depends upon the initial states of the agents and the network structure.

The consensus problem is characterised by the following assumptions (Fig. 3.1):

- The agents $P_i$, $(i = 1, 2, ..., N)$ have integrator dynamics and, thus, are described by a single differential equation

$$P_i: \quad \dot{x}_i(t) = u_i(t), \quad x_i(0) = x_{i0}, \tag{3.1}$$

  in which $x_i(t)$ denotes the scalar state and $u_i(t)$ the scalar input of the $i$-th agent.

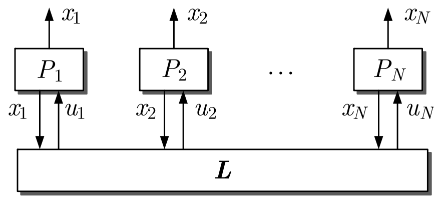


**Fig. 3.1:** System $\overline{\Sigma}$ considered in the consensus problem

- The local controllers $C_i$, $(i = 1, 2, ..., N)$ act as proportional controllers that feed back the differences between the agent state $x_i(t)$ and the states $x_j(t)$, $(j \in \mathcal{N}_i)$ of the neighbouring agents:
$$C_i: \quad u_i(t) = -\sum_{j=1, j\neq i}^{N} a_{ij}(x_i(t) - x_j(t)). \tag{3.2}$$
The coefficients $a_{ij}$ are the elements of the adjacency matrix $\boldsymbol{A}_{\mathcal{G}}$ of the communication graph. Hence, the sum on the right-hand side of eqn. (3.2) is effective for all neighbours $j \in \mathcal{N}_i$ of the agent $P_i$, which leads to the equivalent representation
$$C_i: \quad u_i(t) = -\sum_{j\in\mathcal{N}_i} a_{ij}(x_i(t) - x_j(t)). \tag{3.3}$$
Equations (3.2) and (3.3) are also called the *consensus protocol*. They specify the interactions among the agents. These interactions will be represented later on in terms of the Laplacian matrix $\boldsymbol{L}$ of the communication graph as indicated in Fig. 3.1.

These assumptions simplify the general block diagram in Fig. 1.10 on p. 15 of multi-agent systems to the diagram in Fig. 3.1. Since the consensus value $\bar{x}$ is unknown in advance, the controllers (3.3) cannot react on the control error $\bar{x} - x_i(t)$ and the well-known theory of tracking control is not applicable to solve the consensus problem.

All agents together represent a set of $N$ integrators

$$P: \quad \dot{\boldsymbol{x}}(t) = \boldsymbol{u}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{3.4}$$

with the state $\boldsymbol{x} \in \mathrm{IR}^N$ and the input vector $\boldsymbol{u} \in \mathrm{IR}^N$. The networked controller is a proportional controller with $N$ inputs and $N$ outputs

$$C: \quad \boldsymbol{u}(t) = -\boldsymbol{L}\boldsymbol{x}(t), \tag{3.5}$$

where the controller matrix $\boldsymbol{L}$ turns out to be the Laplacian matrix of the communication network defined in Section 2.2.5. The $i$-th row of eqn. (3.5) reads as

$$C_i: \quad u_i(t) = -\sum_{j=1}^{N} l_{ij}x_j(t), \tag{3.6}$$

which is an alternative representation of the consensus protocol (3.2), in which the elements of the Laplacian matrix $L$ appear rather than the elements of the adjacency matrix $\boldsymbol{A}_{\mathcal{G}}$.

The overall system (3.4), (3.5) is a linear dynamical system $\overline{\Sigma}$ of $N$–th order with the state-space representation

$$\text{Overall system } \overline{\Sigma}: \quad \dot{\boldsymbol{x}}(t) = -\boldsymbol{L}\boldsymbol{x}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{3.7}$$

and the block diagram of Fig. 3.1. This equation is said to represent a *consensus system*, a consensus network, the consensus dynamics or the Laplacian flow.

**Example 3.1** *Consensus problem*

Consider the consensus problem for a system with the communication graph $\vec{\mathcal{G}}$ shown in Fig. 3.2. Since the agent $P_1$ gets communicated only the state $x_3(t)$ of the agent $P_3$, its input is determined according to eqn. (3.2) as

$$u_1(t) = -a_{13}(x_1(t) - x_3(t)).$$

Similarly, agent $P_3$ has the input

$$u_3(t) = -a_{32}(x_3(t) - x_2(t)).$$

Agent $P_2$ knows the states of the two other agents, its input is determined as the weighted sum of the state differences

$$u_2(t) = -a_{21}(x_2(t) - x_1(t)) - a_{23}(x_2(t) - x_3(t)).$$

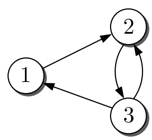


**Fig. 3.2:** Communication graph considered in Example 3.1

The re-formulation of the consensus protocol as

$$\begin{pmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{pmatrix} = -\begin{pmatrix} a_{13}x_1(t) - a_{13}x_3(t)) \\ (a_{21} + a_{23})x_2(t) - a_{21}x_1(t) - a_{23}x_3(t) \\ a_{32}x_3(t) - a_{32}x_2(t) \end{pmatrix}$$

shows that indeed the Laplacian matrix

$$\boldsymbol{L} = \begin{pmatrix} a_{13} & 0 & -a_{13} \\ -a_{21} & a_{21} + a_{23} & -a_{23} \\ 0 & -a_{32} & a_{32} \end{pmatrix}$$

appears in the representation (3.6) of the consensus protocol and in the overall system (3.7):

$$\overline{\Sigma}: \quad \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{pmatrix} = -\begin{pmatrix} a_{13} & 0 & -a_{13} \\ -a_{21} & a_{21} + a_{23} & -a_{23} \\ 0 & -a_{32} & a_{32} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix}. \quad \square$$

The consensus of a multi-agent system is defined as follows:

**Definition 3.1 (Consensus)**
*The system (3.7) is said to reach a consensus if for all initial states* $x_{i0}$, $(i = 1, 2, ..., N)$ *the following relation holds for some scalar* $\bar{x}$*:*

$$\lim_{t \to \infty} \boldsymbol{x}(t) = \bar{x}\boldsymbol{1}. \tag{3.8}$$

Later on, the consensus value $\bar{x}$ turns out to depend upon the initial states of the agents and upon the communication structure $\boldsymbol{L}$ of the networked controller. The main problem to be solved now is to elaborate conditions on the communication network so that a consensus is reached.

There are several reasons why the consensus problem has attracted a lot of attention:

- The consensus problem concerns a direct relation between the structural properties described by the graph Laplacian $\boldsymbol{L}$ and the behaviour of a networked dynamical system. Equation (3.7) shows that the graph matrix $\boldsymbol{L}$ appears in the model of a dynamical system. Hence, the consensus problem brings together graph-theoretical and systems-theoretical properties and shows how the *collective dynamics* of the agents $P_i$ within the overall system $\overline{\Sigma}$ depend upon the network structure $\boldsymbol{L}$.

- The consensus problem appears as a simplified version of several practical problems, in particular as a basic problem of *cooperative control* of physically coupled or uncoupled subsystems. For example, formation control of robots can be treated as a consensus problem, in which the robots should assume prescribed relative positions (Section 3.3.6).

- The consensus problem has its origin in the fields of *distributed computing* and management science, where the agents (processors, team members) should reach an agreement with respect to an important quantity. Several iterative algorithms can be posed as the discrete-time version of the system $\overline{\Sigma}$, which will be introduced in Section 3.4. A consensus should be reached by iterating the two-step procedure of first exchanging information among the agents and second determining the average of the current local information.

The consensus protocol (3.3) can be interpreted as a simple form of a *cooperative controller*, because its implementation is only possible if all agents are willing to cooperate, that is, to send their state information to their neighbours and to change their own state in dependence upon the information received from other agents. In this way, they reach the common state $\bar{x}\boldsymbol{1}$.

This chapter focusses on the following question:

> Under what conditions on the network structure represented by the Laplacian matrix $\boldsymbol{L}$ does the networked system $\overline{\Sigma}$ with arbitrary initial state $\boldsymbol{x}_0$ reach a consensus?

Furthermore, it shows how the consensus value $\bar{x}$ depends upon the network structure and the initial states of the agents and evaluates how quickly the agents approach this value. As the graph under consideration can be weighted, the Laplacian matrix $\boldsymbol{L}$ does not only say which agents are connected with which other agents but it also provides the weighting factors $l_{ij}$ for all existing edges $(j \to i)$. Hence, the consensus problem can be treated simultaneously for directed, undirected and weighted graphs.

## 3.2 Continuous-time consensus

### 3.2.1 Basic convergence results

This section investigates under what conditions all agents within the networked system $\overline{\Sigma}$ reach asymptotically a common consensus value $\bar{x}$. In the model (3.7) of the overall system $\overline{\Sigma}$, $\boldsymbol{L}$ is the Laplacian matrix of a directed graph $\overrightarrow{\mathcal{G}}$ that represents the communication structure of the controller $C$. This section derives conditions on the graph $\overrightarrow{\mathcal{G}}$ that ensure that the system $\overline{\Sigma}$ reaches a consensus according to Definition 3.1.

**Preliminary analysis of the system $\overline{\Sigma}$.** If the matrix $\boldsymbol{L}$ had only eigenvalues with positive real part, the solution to the convergence problem would be very easy, because in this ficticious situation the state $\bar{\boldsymbol{x}} = \boldsymbol{0}$ would be the only equilibrium state of an asymptotically stable system (3.7). However, the matrix $\boldsymbol{L}$ has at least one vanishing eigenvalue $\lambda_1$, it may have multiple vanishing eigenvalues, and in both cases the system $\overline{\Sigma}$ has an infinite number of equilibrium points. Therefore, the consensus problem cannot be solved by applying methods for proving the asymptotic stability of the linear system (3.7), but it requires to extend the stability analysis towards systems that are stable, but not asymptotically stable.

Since the matrix $\boldsymbol{L}$ has a vanishing eigenvalue with a right eigenvector $\boldsymbol{1}$, all $\boldsymbol{x} \in \mathrm{I\!R}^N$ with

$$x_1 = x_2 = ... = x_N = \bar{x} \in \mathrm{I\!R} \tag{3.9}$$

are an equilibrium state of $\overline{\Sigma}$, because for these states eqns. (3.7) and (3.9) lead to $\dot{\boldsymbol{x}} = \boldsymbol{0}$. As in all of these states a consensus is reached, the set of all of them is called the *agreement set* or the set of the collective decision:

$$\text{Agreement set:} \quad \mathcal{A} = \{\bar{x}\boldsymbol{1} \mid \bar{x} \in \mathrm{I\!R}\}. \tag{3.10}$$

**Consensus condition.** The following theorem states the basic convergence result of the consensus problem.

**Theorem 3.1 (Necessary and sufficient condition for continuous-time consensus)**
*Consider a multi-agent system $\overline{\Sigma}$ described by eqn. (3.7). The structure of the system is represented by the weighted directed graph $\overrightarrow{\mathcal{G}}$ with the Laplacian matrix $\boldsymbol{L}$. For all initial states $x_{i0}$, $(i = 1, 2, ..., N)$ the system $\overline{\Sigma}$ reaches a consensus according to eqn. (3.8) for some consensus value $\bar{x} \in \mathrm{I\!R}$ if and only if the graph $\overrightarrow{\mathcal{G}}$ has a spanning tree.*

That is, a consensus is reached whenever the communication graph has an agent from which there exist paths to all other agents. Note that the graph $\overrightarrow{\mathcal{G}}$ can have arbitrary nonnegative weights. For unweighted graphs all weights are set to 1. For undirected graphs the Laplacian matrix is symmetric.

There are several proofs of this theorem all of which are based on Theorem 2.1, which ensures that the second smallest eigenvalue $\lambda_2$ of the Laplacian matrix $\boldsymbol{L}$ is positive if and only if the corresponding graph $\overrightarrow{\mathcal{G}}$ has a spanning tree. The main arguments of these proofs are listed here because they help to understand the consensus problem:

- If the vanishing eigenvalue $\lambda_1 = 0$ of the matrix $\boldsymbol{L}$ is single, the agreement set $\mathcal{A}$ is the one-dimensional eigenspace spanned by the right eigenvector $\boldsymbol{1}$ that belongs to $\lambda_1$. Hence, every equilibrium state of the system belongs to $\mathcal{A}$ and ensures consensus.

- The eigenvalue $\lambda_1 = 0$ is single only if the graph $\overrightarrow{\mathcal{G}}$ has a spanning tree. Hence, consensus can occur only in networked systems that possess such a tree. This requirement is intuitively clear, because a tree is necessary for the existence of an agent the state of which influences, directly or indirectly, the state of all other agents. If the system does not have a spanning tree, the graph consists of two or more subgraphs that have a spanning tree each. Such systems have two or more agents without a path between them. Since they cannot communicate with each other they cannot agree about a consensus and no consensus is possible.

- The fact that the state $\boldsymbol{x}(t)$ of the system $\overline{\Sigma}$ reaches the agreement set $\mathcal{A}$ asymptotically can be proved by using the Lyapunov function

  $$v(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{T}}(t)\boldsymbol{x}(t)$$

  and by showing that the derivative $\dot{v}(t) = \frac{\mathrm{d}V}{\mathrm{d}\boldsymbol{x}}^{\mathrm{T}}\dot{\boldsymbol{x}}(t)$ along the trajectory of the system (3.7) is negative as long as the system has not yet reached the set $\mathcal{A}$.

The following proof decomposes the system $\overline{\Sigma}$ into an asymptotically stable part and a first-order system, which describes the convergence into a consensus state.

**Proof.** The structure of the following proof is highlighted to make the main ideas applicable for the more general problem of synchronisation considered in Section 4.3.

**1. Representation of the consensus error.** The state transformation

$$\underbrace{\begin{pmatrix} x_1(t) \\ e_2(t) \\ \vdots \\ e_N(t) \end{pmatrix}}_{\tilde{\boldsymbol{x}}} = \underbrace{\left(\begin{array}{c|c} 1 & \mathbf{0}^{\mathrm{T}} \\ \hline -\boldsymbol{1} & \boldsymbol{I} \end{array}\right)}_{\boldsymbol{T}^{-1}} \begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_N(t) \end{pmatrix} \quad \text{with} \quad \boldsymbol{T} = \left(\begin{array}{c|c} 1 & \mathbf{0}^{\mathrm{T}} \\ \hline \boldsymbol{1} & \boldsymbol{I} \end{array}\right) \tag{3.11}$$

that includes the $(N-1 \times N-1)$–unity matrix $\boldsymbol{I}$ is applied to the system (3.7) to get the new representation

$$\dot{\tilde{\boldsymbol{x}}}(t) = -\boldsymbol{T}^{-1}\boldsymbol{L}\boldsymbol{T}\tilde{\boldsymbol{x}}(t). \tag{3.12}$$

The new state variables $e_i(t) = x_i(t) - x_1(t)$, $(i = 2, 3, ..., N)$ represent the distance of the states of the agents $P_2$,..., $P_N$ from the state of the agent $P_1$. If the matrix $\boldsymbol{L}$ is decomposed as

$$\boldsymbol{L} = \begin{pmatrix} l_{11} & \boldsymbol{l}_{12}^{\mathrm{T}} \\ \boldsymbol{l}_{21} & \boldsymbol{L}_{22} \end{pmatrix},$$

where $\boldsymbol{L}_{22}$ is an $(N-1 \times N-1)$–matrix, $\boldsymbol{l}_{21}$ and $\boldsymbol{l}_{12}$ are $(N-1)$–vectors and $l_{11}$ is a scalar, one gets

$$\boldsymbol{T}^{-1}\boldsymbol{L}\boldsymbol{T} = \left(\begin{array}{c:c} 0 & \boldsymbol{l}_{12}^{\mathrm{T}} \\ \hdashline \boldsymbol{0} & \tilde{\boldsymbol{L}}_{22} \end{array}\right) \quad \text{with} \quad \tilde{\boldsymbol{L}}_{22} = \boldsymbol{L}_{22} - \boldsymbol{1}\, \boldsymbol{l}_{12}^{\mathrm{T}} \tag{3.13}$$

after having used the property (2.29) of $\boldsymbol{L}$.

**2. Properties of the matrix $\tilde{\boldsymbol{L}}_{22}$.** Due to the structure of the transformed matrix (3.13), the eigenvalues $\lambda_1\{\tilde{\boldsymbol{L}}_{22}\}$,..., $\lambda_{N-1}\{\tilde{\boldsymbol{L}}_{22}\}$ of $\tilde{\boldsymbol{L}}_{22}$ coincide with the eigenvalues $\lambda_2\{\boldsymbol{L}\}$,..., $\lambda_N\{\boldsymbol{L}\}$ of $\boldsymbol{L}$:

$$\lambda_i\{\tilde{\boldsymbol{L}}_{22}\} = \lambda_{i+1}\{\boldsymbol{L}\}, \quad i = 1, 2, ..., N-1. \tag{3.14}$$

Hence, all eigenvalues of $\tilde{\boldsymbol{L}}_{22}$ have positive real part if and only if the graph $\overrightarrow{\mathcal{G}}$ has a spanning tree.

**3. Convergence analysis.** The overall system $\overline{\Sigma}$ in eqn. (3.12) can be decomposed into the first-order subsystem

$$\dot{x}_1(t) = -\boldsymbol{l}_{12}^{\mathrm{T}} \begin{pmatrix} e_2(t) \\ e_3(t) \\ \vdots \\ e_N(t) \end{pmatrix}, \quad x_1(0) = x_{10}$$

and the remaining $N-1$ subsystems

$$\begin{pmatrix} \dot{e}_2(t) \\ \dot{e}_3(t) \\ \vdots \\ \dot{e}_N(t) \end{pmatrix} = -\tilde{\boldsymbol{L}}_{22} \begin{pmatrix} e_2(t) \\ e_3(t) \\ \vdots \\ e_N(t) \end{pmatrix}, \quad \begin{pmatrix} e_2(0) \\ e_3(0) \\ \vdots \\ e_N(0) \end{pmatrix} = \begin{pmatrix} x_{20} - x_{10} \\ x_{30} - x_{10} \\ \vdots \\ x_{N0} - x_{10} \end{pmatrix}. \tag{3.15}$$

If the graph $\overrightarrow{\mathcal{G}}$ has a spanning tree, the eigenvalues $\lambda_2$,..., $\lambda_N$ of $\boldsymbol{L}$ have positive real part and

$$\lim_{t\to\infty} \left\| \begin{pmatrix} e_2(t) \\ e_3(t) \\ \vdots \\ e_N(t) \end{pmatrix} \right\| = 0 \tag{3.16}$$

holds for all initial states $\boldsymbol{x}_0$ of the overall system and implies

$$\lim_{t\to\infty} |\dot{x}_1(t)| = 0$$

and

$$\lim_{t\to\infty} x_1(t) = \bar{x}$$

for some scalar $\bar{x}$. After the back-transformation

$$\boldsymbol{x}(t) = \underbrace{\left(\begin{array}{c:c} 1 & \boldsymbol{0}^{\mathrm{T}} \\ \hdashline \boldsymbol{1} & \boldsymbol{I} \end{array}\right)}_{\boldsymbol{T}} \begin{pmatrix} x_1(t) \\ e_2(t) \\ \vdots \\ e_N(t) \end{pmatrix} = x_1(t)\boldsymbol{1} + \begin{pmatrix} 0 \\ e_2(t) \\ \vdots \\ e_N(t) \end{pmatrix}$$

the appearance of a consensus is proved:

$$\lim_{t\to\infty} \boldsymbol{x}(t) = \bar{x}\boldsymbol{1}.$$

On the other hand, if there does not exist any spanning tree in $\overrightarrow{\mathcal{G}}$, $\lambda_2 = 0$ holds, which means that eqn. (3.16) is violated for at least one initial state of the system $\overline{\Sigma}$. Then for this initial state, the back-transformation shows that no consensus is reached, which proves the necessity of the spanning tree for a consensus. □

For later investigations it is important to see that the matrix $\tilde{\boldsymbol{L}}_{22}$ defined in eqn. (3.13) can be obtained from the Laplacian matrix $\boldsymbol{L}$ as

$$\begin{aligned}\tilde{\boldsymbol{L}}_{22} &= \boldsymbol{L}_{22} - \boldsymbol{1}\,\boldsymbol{l}_{12}^{\mathrm{T}} \\ &= \underbrace{\begin{pmatrix} -\boldsymbol{1}_{N-1} & \boldsymbol{I}_{N-1}\end{pmatrix}}_{\boldsymbol{U}} \boldsymbol{L} \underbrace{\begin{pmatrix} \boldsymbol{0}_{N-1}^{\mathrm{T}} \\ \boldsymbol{I}_{N-1}\end{pmatrix}}_{\boldsymbol{W}}\end{aligned} \tag{3.17}$$

with the $(N-1 \times N)$–matrix $\boldsymbol{U}$ and the $(N \times N-1)$–matrix $\boldsymbol{W}$ that have the property

$$\boldsymbol{U}\boldsymbol{W} = \boldsymbol{I}_{N-1}.$$

The matrices $\boldsymbol{U}$ and $\boldsymbol{W}$ are independent of $\boldsymbol{L}$. $\tilde{\boldsymbol{L}}_{22}$ is called the *reduced Laplacian matrix*.

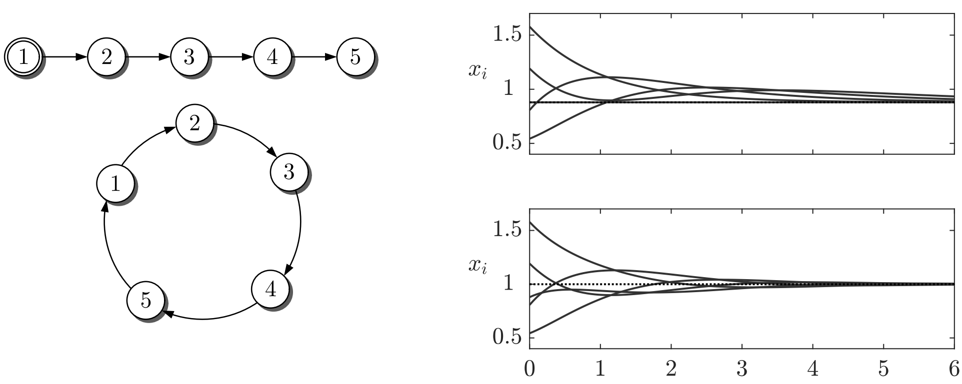


**Fig. 3.3:** Structure and consensus behaviour of the systems considered in Example 3.2

**Example 3.2** *Consensus in networked systems*

The two systems with five agents and the structure of Fig. 3.3 (left) possess a spanning tree. Hence, their agents reach a consensus.

The right part of Fig. 3.3 shows the behaviour of both systems for the same initial state. In the path graph used in the upper part of the figure, the agent $P_1$ determines the consensus value: $\bar{x} = x_{10}$

and the other agents approach this value in approx. 6 time units. As the lower part of the figure shows, the consensus is reached faster in the ring graph and the consensus value $\bar{x} = 1$ is the average of the initial states of all agents, because in the ring each agent sends information about its state, directly or indirectly, to all other agents. □

### 3.2.2 Consensus value

This section answers the question how the consensus value $\bar{x}$ depends upon the initial state $\boldsymbol{x}_0$ of the overall system and upon the communication structure described by the Laplacian matrix $\boldsymbol{L}$. As a preliminary result, an important property $\overline{\Sigma}$ is described by the following lemma:

**Lemma 3.1 (Invariant property of consensus systems)**
*The scalar*

$$y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}(t)$$

*is invariant under the movement of the system $\overline{\Sigma}$ described by eqn. (3.7), where $\boldsymbol{w}^{\mathrm{T}} \geq \boldsymbol{0}^{\mathrm{T}}$ is a left eigenvector of the Laplacian matrix $\boldsymbol{L}$ for the vanishing eigenvalue $\lambda_1 = 0$.*

That is, the scalar $y$ remains constant along the trajectory $\boldsymbol{x}(t)$ of the system and, in particular, equals $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_0$:

$$y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}(t) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_0, \quad t \geq 0. \tag{3.18}$$

This property is valid even if the system $\overline{\Sigma}$ does not satisfy the consensus condition stated in Theorem 3.1.

**Proof of Lemma 3.1.** Assume that the signal $y(t) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}(t)$ is time-dependent with the derivative

$$\begin{aligned}\dot{y}(t) &= \boldsymbol{w}^{\mathrm{T}}\dot{\boldsymbol{x}}(t)\\ &= -\boldsymbol{w}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{x}(t).\end{aligned}$$

Since $\boldsymbol{w}^{\mathrm{T}}$ is an eigenvector for the vanishing eigenvalue $\lambda_1$, one gets

$$\dot{y}(t) = -\boldsymbol{0}^{\mathrm{T}}\boldsymbol{x}(t) = 0,$$

which proves the lemma. □

The lemma can be used to determine the consensus value $\bar{x}$. Under the condition of Theorem 3.1 the system $\overline{\Sigma}$ asymptotically reaches a consensus

$$\boldsymbol{x}(t) \stackrel{t\to\infty}{\longrightarrow} \bar{x}\boldsymbol{1}.$$

As $y$ is invariant along the trajectory $\boldsymbol{x}(t)$, the relation

$$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_0 = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}(t) \stackrel{t\to\infty}{\longrightarrow} \boldsymbol{w}^{\mathrm{T}}\bar{x}\boldsymbol{1}$$

and, hence,

$$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_0 = \bar{x}\,\boldsymbol{w}^{\mathrm{T}}\boldsymbol{1}$$