

First-Order Average Consensus for Cooperative Control Problems Using Novel Weighting Strategies

Furugh Mirali* Antonio Mendez Gonzalez* Herbert Werner*

* *Institute of Control Systems, Hamburg University of Technology, Germany*
 (e-mail: {furugh.mirali}, {antonio.mendez}, {h.werner}@tuhh.de)

Abstract: In this paper we show how average consensus can be guaranteed in cooperative control schemes for multi-agent systems, even when normalised or weighted Laplacians are used. First we present weighting strategies for improving the performance that can be applied in a distributed manner (only local information is used), based on a recently proposed importance matrix. Then we modify the first-order consensus protocol, such that average consensus is guaranteed. It is then shown how this scheme can be used to improve the performance of cooperative control schemes. For this purpose, the effect of a formation reference on the consensus states is explicitly stated. A simulation result with second order LTI agent models illustrates the proposed approach.

© 2017, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: Multi-agent systems, average consensus, distributed, networks, cooperative control

1. INTRODUCTION

Achieving consensus in a network of mobile agents that can communicate with each other is fundamental in many applications. In general, these agents do not have central knowledge about the overall network but only about the local neighbourhood. With the help of consensus protocols and communication among each other consensus can be achieved (see e.g. Reynolds (1987); Olfati-Saber (2006)).

Using robust control schemes for cooperative control goes hand in hand with some sort of normalisation of the graph Laplacian, in order to guarantee robust stability. On the other hand, as a consequence of the normalisation, only consensus (not average consensus) can be guaranteed, because the normalisation destroys the symmetry of the Laplacian.

For the continuous-time case, Ren and Atkins (2005) propose different weights on communication links, however they do not specify how agents should locally decide on how to set these weights and therefore average consensus is not reached, thus only convergence to a common value is guaranteed.

In this paper we extend the cooperative control scheme such that we guarantee average consensus even when there is a formation reference. In order to do this, we first show how the formation reference affects consensus and how the cooperative control loop has to be changed to guarantee average consensus. We assume that there is no global designer and that each individual agent constructs the corresponding row entry of a weighted Laplacian in a distributed manner. We present a distributed algorithm for constructing the generalised weighted Laplacian, Fiedler (1975), by using an importance matrix as proposed by Mirali and Werner (2017). Furthermore, we show how to locally construct the left Perron vector for the weighted Laplacians in order to extend existing first-order average consensus protocols to a much more general case. Finally, we apply these weighting strategies to a cooperative control problem and show that using adequate weights on each communication

channel can improve the convergence speed in formation control.

This paper is structured as follows: In Section 2 we collect some facts from graph theory for modelling the interconnection structure of a network. In Section 3 we review the first-order continuous-time consensus protocols. Then we present different weighting schemes for constructing the row entries of a weighted Laplacians in a distributed manner. In addition we present an adaptation of the first-order discrete-time consensus protocol introduced by Olshevsky and Tsitsiklis (2011) for continuous-time to guarantee average consensus using weighted Laplacians. In Section 4 we show how to apply the proposed consensus protocols to a cooperative control problem. We will extend the cooperative control loop such that we guarantee average consensus even when the weighted Laplacians are used instead of the standard Laplacian. Then in Section 5 we present simulation results. Finally in Section 6 we present conclusions.

2. PRELIMINARIES

This section reviews some aspects of graph theory for modelling a communication network (see e.g. Diestel (2010); Chung (1997)) and gives an overview about the notation used in this paper.

Let $\mathcal{V} = \{v_1, \dots, v_N\}$ and $\mathcal{E} = \{e_1, \dots, e_M\}$ be the set of vertices and the set of undirected edges of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, respectively. The communication network can be represented with the help of the *adjacency matrix* $A \in \mathbb{R}^{N \times N}$

$$A = \begin{cases} 1 & \{v_i, v_j\} \in \mathcal{E}, \\ 0 & \text{else.} \end{cases} \quad (1)$$

Let \mathcal{N}_i be the set of all vertices v_j such that $\{v_i, v_j\} \in \mathcal{E}$ and the degree d_i be the cardinality of \mathcal{N}_i . The *degree matrix* is defined as $D = \text{diag}(d_1, d_2, \dots, d_N)$, where $\text{diag}(\cdot)$ means diagonal matrix. The *graph Laplacian* is defined as $L = D - A$. An identity matrix of size $N \times N$ is denoted by I_N . A collection

* Antonio Mendez G. acknowledges the support from the National Council of Science and Technology (CONACYT) in Mexico and the German Academic Exchange Service (DAAD).

of N identical subsystems is denoted by $\hat{G}(s) = I_N \otimes G(s)$, where \otimes is the Kronecker product.

3. CONTINUOUS-TIME AVERAGE CONSENSUS PROTOCOLS

In this section we briefly review the continuous-time first-order average consensus protocol proposed by Olfati-Saber et al. (2007). Then we present a continuous-time adaptation of the discrete-time consensus protocol introduced by Olshevsky and Tsitsiklis (2011), where average consensus is guaranteed while using the *equal neighbour model*, i.e. weighting incoming information equally. This corresponds to using non-symmetric edge weights, in contrast to the work by Xiao and Boyd (2004). Given an interconnection structure, we will present a distributed algorithm for constructing a *generalised Laplacian* introduced by Fiedler (1975) using the *importance matrix* S and the weighting strategies introduced by Mirali and Werner (2017). We will call L_S the *weighted Laplacian* defined by the importance matrix S .

3.1 First-Order Average Consensus

The linear first-order consensus protocol proposed by Olfati-Saber and Murray (2004) is defined as

$$\dot{x}_i(t) = u_i(t), \quad (2)$$

$$u_i(t) = \sum_{v_j \in \mathcal{N}_i} a_{ij}(x_j - x_i), \quad i = 1, \dots, N, \quad (3)$$

where a_{ij} is the corresponding (i, j) -entry of the adjacency matrix A . This can be written as

$$\dot{x}(t) = -Lx(t). \quad (4)$$

It is well-known that

$$\lim_{t \rightarrow \infty} x(t) = \frac{\mathbf{1}\mathbf{1}^T}{N}x(0) \quad (5)$$

holds if and only if

- (i) $L\mathbf{1} = \mathbf{0}$,
- (ii) $\mathbf{1}^T L = \mathbf{0}^T$,
- (iii) $\lambda_2(L) > 0$,

where $\mathbf{1}$ and $\mathbf{0}$ are column vectors with all entries equal to one and zero, respectively and $\lambda_2(\cdot)$ is the second smallest eigenvalue. The standard Laplacian $L = D - A$ of an undirected graph fulfills these conditions always, see e.g. Chung (1997). If the agents weight incoming channels differently preserving the row sum to be zero, the symmetry of the resulting weighted Laplacian L_S will be lost and the protocol in (4) will not converge to the average but to a weighted average, i.e.

$$\lim_{t \rightarrow \infty} x(t) = \mathbf{1}\pi_S^T x(0), \quad (6)$$

where π_S is the left Perron vector of L_S . To overcome this problem Olshevsky and Tsitsiklis (2011) proposed *two parallel passes* for the corresponding discrete-time case using the equal neighbour model. In order to extend the first-order consensus protocol in (4) using this idea, we first introduce a distributed procedure for constructing the weighted Laplacians. Mirali and Werner (2017) introduced the importance matrix S for constructing a stochastic weighting matrix for discrete-time first-order consensus protocols. Here we extend this to the continuous-time case and give a procedure to construct weighted Laplacians in a distributed manner.

Let $S = \text{diag}(s_1, \dots, s_N)$ be a diagonal matrix and $s_i > 0$ be an importance measure of agent v_i . Examples for specific importance measures will be discussed later. First each agent is communicating its importance measure to its neighbours. Then the off-diagonal non-zero row entries of the weighted Laplacian are set to the corresponding negative importance measure of the neighbours. The diagonal entries are set to the sum of the importance measures of the neighbours, to maintain the row sum to be zero. The distributed construction of the weighted Laplacian is summarised in Algorithm 1. The weighted Laplacian can be written as

$$L_S = \text{diag}(AS\mathbf{1}) - AS. \quad (7)$$

It is easy to verify that the corresponding left Perron vector π_S with $\mathbf{1}^T \pi_S = 1$ is

$$\pi_S = \frac{S\mathbf{1}}{\mathbf{1}^T S\mathbf{1}}. \quad (8)$$

If the row entries are scaled with a constant R_i , i.e.

$$L_{SR} = R(\text{diag}(AS\mathbf{1}) - AS), \quad (9)$$

where R is a diagonal matrix with the positive entries (R_1, \dots, R_N) , then the left Perron vector is

$$\pi_{SR} = \frac{R^{-1}S\mathbf{1}}{\mathbf{1}^T R^{-1}S\mathbf{1}}. \quad (10)$$

Note that the eigenvalues of L_S and consequently of L_{SR} are real and nonnegative. To see this, use a similarity transformation $\tilde{L}_S = TL_S T^{-1}$ with $T = S^{\frac{1}{2}}$. Then we get

$$\tilde{L}_S = S^{\frac{1}{2}}L_S S^{-\frac{1}{2}} = \text{diag}(AS\mathbf{1}) - S^{\frac{1}{2}}AS^{\frac{1}{2}}.$$

Since $S^{\frac{1}{2}}AS^{\frac{1}{2}}$ is symmetric and eigenvalues do not change under similarity transformation, the eigenvalues of L_S are real. With the help of the Gerschgorin circle theorem it can be easily shown that the eigenvalues are all nonnegative. R can be used for scaling down the row entries of the weighted Laplacians in order to guarantee stability of a feedback loop for cooperative control.

Now we present an adaptation of the discrete-time consensus protocol presented by Mirali and Werner (2017) (based on Olshevsky and Tsitsiklis (2011)) using weighted Laplacians for the continuous-time case that guarantees average consensus.

Algorithm 2 consists of two linear first-order consensus protocols

$$\begin{aligned} \dot{y}(t) &= -L_S y(t), & y(0) &= S^{-1}\mathbf{1}, \\ \dot{z}(t) &= -L_S z(t), & z(0) &= S^{-1}x(0), \end{aligned} \quad (11)$$

Algorithm 1 Distributed construction of the non-zero row entries of L_S

Require: non-zero row entries of A are known, each agent v_i has an importance measure s_i

Ensure: $L_S\mathbf{1} = \mathbf{0}$, $\lambda_2(L_S) > 0$

```

1: function WEIGHTING(  $A$  )
2:   for every agent  $v_i$  do
3:     ▷ send  $s_i$  to  $v_j \in \mathcal{N}_i$ 
4:     for every neighbour  $v_j \in \mathcal{N}_i$  do
5:       ▷ receive  $s_j$ 
6:        $L_S(i, j) \leftarrow -s_j$ 
7:     end for
8:      $L_S(i, i) \leftarrow \sum_{j \in \mathcal{N}_i} s_j$ 
9:   end for
10:  return  $L_S$ 
11: end function

```

running in parallel using the additional two states $y(t)$ and $z(t)$.

Theorem 1. Algorithm 2 guarantees convergence to average consensus.

A short proof, following the same lines as Olshevsky and Tsitsiklis (2011) and Mirali and Werner (2017), is given in the Appendix. Olshevsky and Tsitsiklis (2011) proposed the *equal neighbour model*, where every agent is weighting incoming information equally, i.e the importance matrix $S = I$. The corresponding weighted Laplacian using Algorithm 1 is equal to the standard Laplacian L .

3.2 Weighting Strategies

Here we review the four weighting strategies proposed by Mirali and Werner (2017).

Degree neighbour model This model considers the degree of agent v_i as importance measure s_i and thus D as the importance matrix S .

Eigenvector neighbour model Motivated by Bertrand and Moonen (2013) within this model every agent uses its eigenvector centrality as an importance measure. The eigenvector centrality $c_i = [c_{\max}]_i$ of agent v_i is defined by

$$Ac_{\max} = \lambda_{\max} c_{\max}, \quad (12)$$

where λ_{\max} is the maximum eigenvalue corresponding to the eigenvector c_{\max} of A . Mahony and Absil (2003) proposed a continuous-time power algorithm for calculating the principal eigenvector of a given matrix. This is illustrated in Fig. 1, where $f_A(c)$ is the Rayleigh quotient

$$f_A(c) = \frac{c^T A c}{c^T c},$$

Algorithm 2 Two parallel passes using L_S and guaranteeing average consensus

Require: non-zero row entries of A are known, each agent v_i has two additional states y_i and z_i

Ensure: $\lim_{t \rightarrow \infty} x_i(t) = \frac{1}{N} \sum_{j=1}^n x_j(0)$

```

1: function AVGCONS( $A, x(0)$ )
2:    $L_S \leftarrow$  WEIGHTING( $A$ )
3:   for every agent  $v_i$  do
4:      $y_i(0) \leftarrow \frac{1}{s_i}$ 
5:      $z_i(0) \leftarrow \frac{x_i(0)}{s_i}$ 
6:   end for
7:   loop
8:     for every agent  $v_i$  do
9:        $\triangleright$  send  $y_i$  and  $z_i$  to  $v_j \in \mathcal{N}_i$ 
10:       $\dot{y}_i(t) \leftarrow \sum_{j \in \mathcal{N}_i \cup \{i\}} [-L_S]_{ij} y_j(t)$ 
11:       $\dot{z}_i(t) \leftarrow \sum_{j \in \mathcal{N}_i \cup \{i\}} [-L_S]_{ij} z_j(t)$ 
12:       $x_i(t) \leftarrow \frac{z_i(t)}{y_i(t)}$ 
13:     end for
14:   end loop
15:   return  $x(t)$ 
16: end function

```

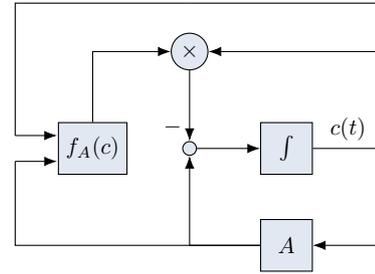


Fig. 1. Continuous-time power algorithm

$c(0)$ is set arbitrarily and \times is the element-wise multiplication. Assuming that each agent v_i is storing c_i , we can, considering (7) set S to $C = \text{diag}(c_1, \dots, c_n)$. Note that we assume that every agent has a-priori knowledge of the corresponding value in the principal eigenvector of A .

Degree-eigenvector neighbour model The *degree neighbour model* and the *eigenvector neighbour model* combined give the importance matrix

$$S = CD. \quad (13)$$

Here we are again assuming that each agent knows its corresponding eigenvector centrality c_i .

State neighbour model Each agent constructs the importance matrix from the distance of its initial value to those of its neighbours, i.e.

$$\delta_i = \frac{1}{d_i} \sum_{v_j \in \mathcal{N}_i} |x_i(0) - x_j(0)|. \quad (14)$$

Then we take the diagonal matrix

$$[X]_{ii} = \begin{cases} \delta_i & \delta_i \geq 1, \\ x_i(0) & \delta_i < 1 \vee x_i(0) > 1, \\ 1 & \delta_i < 1 \vee x_i(0) < 1, \end{cases} \quad (15)$$

to be the importance matrix S .

4. APPLICATION TO COOPERATIVE CONTROL

In this section we propose a cooperative control scheme that guarantees average consensus when using non-standard Laplacians, i.e. normalised Laplacians or the proposed weighted Laplacians.

Consider the cooperative control loop shown in Fig. 2, where $\hat{P}(s)$ is a group of N identical agents, $\hat{K}(s)$ denotes the associated (a priori designed) controllers, $\hat{H}(s) = \hat{P}(s)\hat{K}(s)$ and L represents the interconnection structure. The signal r is a formation reference, e_L is the consensus error, u is the control input and y is the measured output. The closed-loop system $T(s)$ is given by

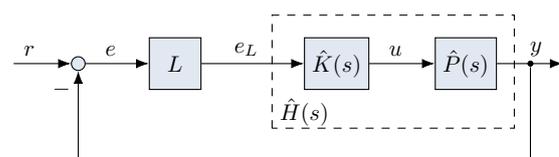


Fig. 2. Cooperative control loop

$$\hat{T}(s) \left\{ \begin{bmatrix} \dot{\xi} \\ y \end{bmatrix} = \begin{bmatrix} \hat{A}_H - \hat{B}_H L \hat{C}_H & \hat{B}_H L \\ \hat{C}_H & 0 \end{bmatrix} \begin{bmatrix} \xi \\ r \end{bmatrix}, \right. \\ \left. = \begin{bmatrix} \hat{A}_P - \hat{B}_P \hat{D}_K L \hat{C}_P & \hat{B}_P \hat{C}_K & \hat{B}_P \hat{D}_K L \\ -\hat{B}_K L \hat{C}_P & \hat{A}_K & \hat{B}_K L \\ \hat{C}_P & 0 & 0 \end{bmatrix} \begin{bmatrix} \xi \\ r \end{bmatrix}. \right. \quad (16)$$

In a first step we show how the formation reference affects the consensus. We make the following assumptions. The formation reference $r(t)$ is constant in time. The controller $K(s)$ is designed such that closed-loop system is marginally stable, i.e. there is one pole at zero due to the integrator dynamics of A_P . In addition, we assume that the plant is controllable and that the measured output y is a position. For simplicity we consider single input single output systems.

Lemma 1. When the above assumptions are fulfilled the plant states depend linearly on the formation reference r .

Proof. See Appendix.

Now we show how the formation reference affects consensus.

Lemma 2. The steady state value of the position states is

$$\lim_{t \rightarrow \infty} \xi^{\text{pos}}(t) = p_1 q_1^T \xi^{\text{pos}}(0) + (I - p_1 q_1^T) r, \quad (17)$$

where ξ^{pos} is the initial position.

Proof. See Appendix.

This can be used to extend the cooperative control loop such that we can guarantee average consensus even when using normalised Laplacians or the proposed weighted Laplacians. Motivated by Olshevsky and Tsitsiklis (2011) we next propose an algorithm consisting of two parallel processes and using a filter to guarantee average consensus. In addition, we will modify the two parallel processes, such that there is no division involved. Experience shows that there are situations where the second process crosses zero and then the position will grow arbitrary fast due to the division. To overcome these difficulties, we propose the following algorithm.

Algorithm 3. (1) Initialisation of the additional states at time $t = 0$ with

$$g_i(0) = \frac{s_i}{R_i}, \quad h_i^{\text{pos}}(0) = \frac{R_i}{s_i} \xi_i^{\text{pos}}(0), \quad (18)$$

where $g \in \mathbb{R}^N$ and $h \in \mathbb{R}^{N(n_p+n_k)}$ with n_p and n_k being the number of plant and controller states, respectively. The remaining states are not changed.

(2) The position reference is transformed with $F = RS^{-1}$ as $r_F = Fr$.

(3) The dynamics of the two processes are

$$\dot{g}(t) = -Lg(t), \quad y_g(t) = g(t), \quad (19)$$

$$\begin{bmatrix} \dot{h} \\ y_h \end{bmatrix} = \begin{bmatrix} \hat{A}_H - \hat{B}_H L_S \hat{C}_H & \hat{B}_H L_S \\ \hat{C}_H & 0 \end{bmatrix} \begin{bmatrix} h \\ 0 \end{bmatrix}. \quad (20)$$

(4) Each agent updates its position values with

$$y_i(t) = y_{g_i} y_{h_i}^{\text{pos}} + r_i - r_F y_{g_i}, \quad (21)$$

i.e. only the position states will be multiplied with $y_g(t)$.

This algorithm is illustrated in Fig. 3.

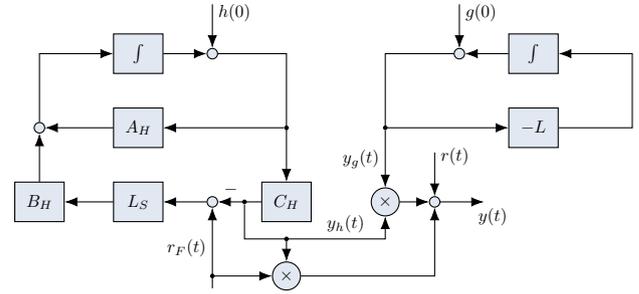


Fig. 3. Extended cooperative control loop

Theorem 2. Algorithm 3 guarantees convergence of the formation centered at the average of the initial positions.

Proof. See Appendix.

5. SIMULATION RESULTS

This section presents simulation results to illustrate the proposed algorithm. We consider $N = 10$ agents $P(s)$ that are modelled by the second-order linear SISO system given in (22), which describes the dynamics of a mobile vehicle in one dimension:

$$P(s) \left\{ \begin{bmatrix} v_i \\ a_i \\ y_i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ v_i \\ u_i \end{bmatrix}, \quad (22)$$

where x , v and a are the position, velocity and acceleration, respectively, of a point particle, u is the control input and y is the measured output. The initial positions for the agents are $x(0) = [7 \ 10 \ 3 \ -3 \ 0 \ 9 \ -10 \ -7 \ -1 \ 1]^T$, for which the average is $x_{\text{avg}} = 0.9$ and the initial velocities are 0. The agents interact through a fixed communication graph as shown in Fig. 4. In order to guarantee stability we normalise the weighted Laplacians and the standard Laplacian, such that the diagonal entries are equal to one. This can be done by each agent individually with the help of the scaling matrix R . The controller is chosen as

$$K(s) = \frac{b_4 s^4 + b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}, \quad (23)$$

where

$$a_0 = 9.107 \cdot 10^7, \quad a_1 = 3.812 \cdot 10^6,$$

$$a_2 = 5.895 \cdot 10^4, \quad a_3 = 399.3$$

$$b_0 = 63.4, \quad b_1 = 7.166 \cdot 10^9,$$

$$b_2 = 2.855 \cdot 10^8, \quad b_3 = 4.72 \cdot 10^6, \quad b_4 = 2.971 \cdot 10^4.$$

For the proposed algorithm we use a normalised weighted Laplacian using the *state neighbour model* for the first process and for the second process we use a scaled version of the standard Laplacian $L_k = kL$, where $k = 10$. This can be done since the second process is used for scaling the output of the first process and is a linear first-order consensus protocol. This is compared to the standard normalised Laplacian. Fig. 5 shows

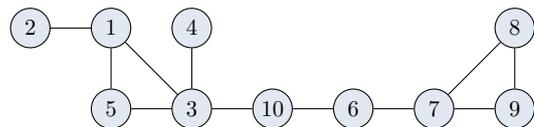


Fig. 4. Undirected communication topology

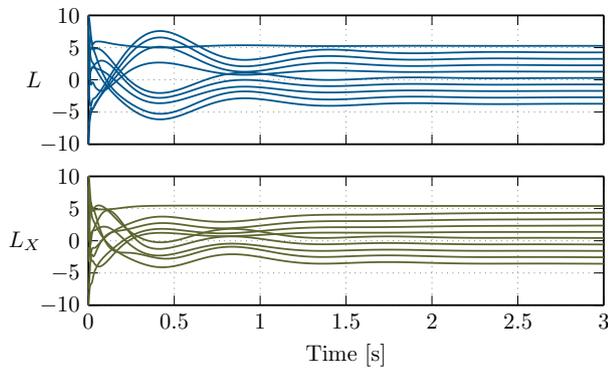


Fig. 5. Time response of agent positions with (L_X , state neighbour model) and without (L) weighting, and controller $K(s)$ as in (23)

the comparison of the time response of the positions. Using the standard normalised Laplacian, the average of the position states is converging to 0.7727; whereas with the proposed algorithm the formation is indeed centered at the average value of 0.9.

6. CONCLUSION

The recently introduced concept of an importance matrix is a central tool for generating novel non-symmetric weighting techniques; these weighted Laplacian schemes can be incorporated into consensus protocols and cooperative control loops. In both cases, the convergence speed can be improved while average consensus is guaranteed. It has been shown that by weighting the information exchange between agents, the performance can be increased without re-tuning the controller. Moreover, since the closed-loop eigenvalues are directly influenced by the eigenvalues of the Laplacian, this influence has to be taken into account when weights are introduced; this issue can lead to instability; therefore, normalised Laplacians are used.

REFERENCES

- Bertrand, A. and Moonen, M. (2013). Seeing the bigger picture: How nodes can learn their place within a complex ad hoc network topology. *IEEE Signal Processing Magazine*, 30(3), 71–82.
- Chung, F.R.K. (1997). *Spectral graph theory*, volume 92. Published for the Conference Board of the mathematical sciences by the American Mathematical Society, Providence and R.I.
- Diestel, R. (2010). *Graph theory*, volume 173 of *Graduate texts in mathematics*. Springer, Heidelberg and New York, 4th ed. edition.
- Fiedler, M. (1975). A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czechoslovak Mathematical Journal*, 25(4), 619–633.
- Mahony, R. and Absil, P.A. (2003). The continuous-time rayleigh quotient flow on the sphere. *Linear algebra and its applications*, 368, 343–357.
- Massioni, P. and Verhaegen, M. (2009). Distributed control for identical dynamically coupled systems: A decomposition approach. *IEEE Transactions on Automatic Control*, 54(1), 124–135.
- Mirali, F. and Werner, H. (2017). Distributed weighting strategies for improved convergence speed of first-order consensus. In *American Control Conference*.
- Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3), 401–420.
- Olfati-Saber, R., Fax, J.A., and Murray, R.M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1), 215–233.
- Olfati-Saber, R. and Murray, R.M. (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9), 1520–1533.
- Olshchvsky, A. and Tsitsiklis, J. (2011). Convergence speed in distributed consensus and averaging. *SIAM Review*, 53(4), 747–772.
- Ren, W. and Atkins, E. (2005). Second-order consensus protocols in multiple vehicle systems with local interactions. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 15–18.
- Reynolds, C.W. (1987). Flocks, herds and schools: a distributed behavioural model. *Computer Graphics*, 21(4), 25–34.
- Xiao, L. and Boyd, S.P. (2004). Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1), 65–78.

Appendix A. PROOFS

Here we give proofs for Lemma 1, 2 and Theorem 1, 2.

Proof. (*Theorem 1.*) First note that considering 11 the following

$$y(t) = e^{-L_S t} y(0), \quad z(t) = e^{-L_S t} z(0) \quad (\text{A.1})$$

holds and that L_S has an eigenvalue equal to zero, since the row sum is zero. The eigenvalues of the weighted Laplacian are real and nonnegative. Using this property of the weighted Laplacian and initialising $y_i(0)$ as $z_i(0)$ described in line 4 and 5 of Algorithm 1, the limits of (11) are

$$\lim_{t \rightarrow \infty} y_i(t) = \sum_{j=1}^N \frac{s_j}{\mathbf{1}^T S \mathbf{1}} \frac{1}{s_j} = \frac{N}{\mathbf{1}^T S \mathbf{1}},$$

$$\lim_{t \rightarrow \infty} z_i(t) = \sum_{j=1}^N \frac{s_j}{\mathbf{1}^T S \mathbf{1}} \frac{x_j(0)}{s_j} = \sum_{j=1}^N \frac{x_j(0)}{\mathbf{1}^T S \mathbf{1}},$$

and hence

$$\lim_{t \rightarrow \infty} x_i(t) = \lim_{t \rightarrow \infty} \frac{z_i(t)}{y_i(t)} = \frac{1}{N} \sum_{j=1}^N x_j(0). \quad \blacksquare$$

Proof. (*Lemma 1.*) The feedback loop in Fig. A.1 corresponds to the decomposition of the cooperative control loop in Fig. 2 proposed by Massioni and Verhaegen (2009). For any diagonalisable Laplacian there exists an eigenvalue decomposition $Z^{-1} L Z = \Lambda$ with $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ and

$$0 = \lambda_1(L) < \lambda_2(L) \leq \dots \leq \lambda_N(L).$$

The right and left eigenvectors of L are denoted by $Z = [p_1 \ p_2 \ \dots \ p_N]$ and $(Z^{-1})^T = [q_1 \ q_2 \ \dots \ q_N]$, respectively. The transformation proposed by Massioni and Verhaegen (2009) is not only a similarity transformation, but also the following signal transformation

$$\tilde{\xi}(t) = Z_{(n_p+n_k)}^{-1} \xi(t), \quad \tilde{y}(t) = Z^{-1} y(t), \quad \tilde{r}(t) = Z^{-1} r(t).$$

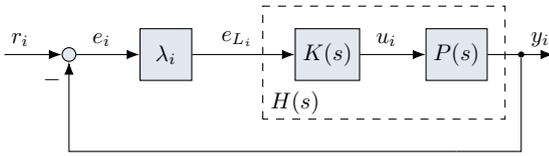


Fig. A.1. Modal subsystems parameterised by λ_i

Using this the state space model of the modal subsystems is

$$\begin{aligned} \dot{\tilde{\xi}}_i(t) &= \begin{bmatrix} A_P - \lambda_i B_P D_K C_P & B_P D_k \\ -\lambda_i B_K C_P & A_K \end{bmatrix} \tilde{\xi}_i(t) + \lambda_i \begin{bmatrix} B_P D_K \\ B_K \end{bmatrix} \tilde{r}_i(t), \\ &= \Gamma_i \tilde{\xi}_i(t) + \Sigma_i \tilde{r}_i(t), \end{aligned}$$

$$\tilde{y}_i(t) = C_P \tilde{\xi}_i(t)$$

with $\tilde{\xi}_i(t) \in \mathbb{R}^{n_p+n_k}$ and $\tilde{r}_i(t), \tilde{y}_i(t) \in \mathbb{R}^{n_y}$. The solution of the first-order differential equations consists of a homogeneous and a particular solution, that is

$$\tilde{\xi}_i(t) = \exp(\Gamma_i t) \tilde{\xi}_i(0) + \int_0^t \exp(\Gamma_i(t-\tau)) \Sigma_i \tilde{r}_i(\tau) d\tau.$$

Since we assumed that the formation reference r is constant for all agents and consequently $\tilde{r}_i(t) = \tilde{r}_i$ the above equation simplifies to

$$\tilde{\xi}_i(t) = \exp(\Gamma_i t) \tilde{\xi}_i(0) + \lambda_i \int_0^t \exp(\Gamma_i(t-\tau)) d\tau \begin{bmatrix} B_P D_k \\ B_k \end{bmatrix} \tilde{r}_i.$$

Based on the fact that λ_1 is a distinct eigenvalue at zero, the plant state of the first agent only consists of a homogeneous part

$$\tilde{\xi}_1(t) = \exp(\Gamma_1 t) \tilde{\xi}_1(0),$$

and therefore the position states of the other agents

$$\tilde{\xi}_i(t) = \exp(\Gamma_i t) \tilde{\xi}_i(0) - \lambda_i \Gamma_i^{-1} [I - \exp(\Gamma_i t)] \begin{bmatrix} B_P D_k \\ B_k \end{bmatrix} \tilde{r}_i$$

with $i \in \{2, \dots, N\}$. It is assumed that the controller is designed in such a way that the closed-loop system is marginally stable. This means that Γ_1 has one eigenvalue at zero since A_P has one integrator and A_K is stable and that Γ_j for $j \in \{2, \dots, N\}$ have all eigenvalues in the open left half plane. With the help of the eigenvalue decomposition of $\Gamma_i = X_i M_i X_i^{-1}$ and $X_i X_i^{-1} = I$ the steady state values of the agent states are

$$\lim_{t \rightarrow \infty} \tilde{\xi}_i(t) = \begin{cases} e_1 e_1^T \tilde{\xi}_1(0), & i = 1, \\ -\lambda_i \Gamma_i^{-1} \begin{bmatrix} B_P D_k \\ B_k \end{bmatrix} \tilde{r}_i, & i \in \{2, \dots, N\}, \end{cases}$$

where e_i is the i -th unit vector. ■

Proof. (Lemma 2.) For $i \in \{2, \dots, N\}$, let

$$\Gamma_i^{-1} \begin{bmatrix} B_P D_k \\ B_k \end{bmatrix} = -\frac{1}{\lambda_i} \begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix},$$

where $\psi \in \mathbb{R}^{n_p+n_k}$. Multiplying from the left with Γ_i yields

$$\begin{bmatrix} B_P D_k \\ B_k \end{bmatrix} = -\frac{1}{\lambda_j} \Gamma_j \psi.$$

Now we assume that the state space model of the plant was given in controller canonical form. The left-hand side of the above equation is a column vector, therefore

$$\psi = \frac{1}{c_1} e_1,$$

where c_1 is the first entry of C_P . Then the steady state value is

$$\lim_{t \rightarrow \infty} \tilde{\xi}_i(t) = \begin{cases} e_1 e_1^T \tilde{\xi}_1(0), & i = 1, \\ \frac{\tilde{r}_i}{c_1} e_1, & i \in \{2, \dots, N\}. \end{cases}$$

Transforming the signals back and considering only the position states of each agent yields

$$\begin{aligned} \lim_{t \rightarrow \infty} \xi^{\text{pos}}(t) &= \lim_{t \rightarrow \infty} Z \tilde{\xi}^{\text{pos}}(t) \\ &= p_1 q_1^T \xi^{\text{pos}}(0) + (I - p_1 q_1^T) r \end{aligned}$$

using the fact that $\sum_{i=1}^N p_i q_i^T = Z Z^{-1} = I$ and

$$\lim_{t \rightarrow \infty} \tilde{\xi}^{\text{pos}}(t) = [q_1^T \xi^{\text{pos}}(0) \quad q_2^T r \quad \dots \quad q_N^T r]^T.$$

Proof. (Theorem 2.) With the help of Lemma 1 and 2 it is easy to show that the position states converge to the average of the initial positions. The steady state values of the first (19) and second process (19) are

$$\lim_{t \rightarrow \infty} y_{g_i}(t) = \frac{1}{N} \mathbf{1} y_{g_i}(0) = \frac{1}{N} \mathbf{1}^T R^{-1} S \mathbf{1},$$

$$\lim_{t \rightarrow \infty} y_{h_i}^{\text{pos}}(t) = \frac{\xi^{\text{pos}}(0)}{\mathbf{1}^T R^{-1} S \mathbf{1}} + r_F + \frac{\mathbf{1}^T r}{\mathbf{1}^T R^{-1} S \mathbf{1}},$$

respectively. Then the steady state value of the position values yields

$$\begin{aligned} \lim_{t \rightarrow \infty} y_i^{\text{pos}}(t) &= y_{g_i}^{\text{pos}} + r_i - r_{F_i} y_{g_i}^{\text{pos}}, \\ &= \frac{\xi^{\text{pos}}(0)}{N} + r_i + \frac{\mathbf{1}^T r}{N}. \end{aligned}$$

■

■