International Journal of Control

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tcon20

Controlling consensus in networks with symmetries

Francesco Lo Iudice, Anna Di Meglio, Fabio Della Rossa & Francesco Sorrentino

To cite this article: Francesco Lo Iudice, Anna Di Meglio, Fabio Della Rossa & Francesco Sorrentino (2021): Controlling consensus in networks with symmetries, International Journal of Control, DOI: 10.1080/00207179.2021.1946157

To link to this article: https://doi.org/10.1080/00207179.2021.1946157

Published online: 09 Jul 2021.

INTERNATIONAL JOURNAL OF CONTROL
https://doi.org/10.1080/00207179.2021.1946157

Taylor & Francis
Taylor & Francis Group

Controlling consensus in networks with symmetries

Francesco Lo Iudice [a], Anna Di Meglio[a], Fabio Della Rossa[b] and Francesco Sorrentino[c]

[a]Department of Information Technology and Electrical Engineering, University of Naples Federico II, Naples, Italy; [b]Department of Electronics, Information, and Bioengineering, Politecnico of Milan, Milan, Italy; [c]Department of Mechanical Engineering, University of New Mexico, Albuquerque, NM, USA

ABSTRACT

We study networks with linear dynamics where the presence of symmetries of the pair (A, B), induces a partition of the network nodes in clusters and the matrix A is not restricted to be in Laplacian form. For these networks, an invariant *group consensus subspace* can be defined, in which the nodes in the same cluster evolve along the same trajectory in time. We prove that the network dynamics is uncontrollable in directions orthogonal to this subspace. Under the assumption that the dynamics parallel to this subspace is controllable, we design optimal controllers that drive the group consensus dynamics towards a desired state. Then, we consider the problem of selecting additional control inputs that stabilise the group consensus subspace and obtain bounds on the minimum number of additional inputs and driver nodes needed to this end. Altogether, our results indicate that it is possible to *independently design* the control actions along and transverse to the group consensus subspace.

ARTICLE HISTORY
Received 19 October 2020
Accepted 14 June 2021

KEYWORDS
Controllability of complex networks; control of complex networks; networks with symmetries; consensus; multi-agent systems

1. Introduction

The number of real-word systems modelled as complex networks is ever increasing, and ranges from natural (Sethi et al., 2009; Van Vreeswijk & Sompolinsky, 1996), technological, Stegink et al. (2016) and Yu et al. (2012) and social systems (De Lellis et al., 2018; Proskurnikov et al., 2015) to epidemic spreading (Gatto et al., 2020). The ultimate goal of being able to arbitrarily affect the behaviour of these systems has spurred researchers across different scientific communities to investigate the controllability properties of linear complex networks (Lo Iudice et al., 2019; Pasqualetti et al., 2014; Yuan et al., 2013). In this framework, several works (Liu et al., 2011; Lo Iudice et al., 2015) have revisited the classical tools of structural controllability (Lin, 1974) from the viewpoint that in order to control complex networks, controllability must be guaranteed by a proper selection of the set of nodes, the *drivers*, in which control signals are injected. If the selection of the driver nodes ensures structural controllability, then the network will also be controllable in Kalman's sense for all possible edge weights but for a set of Lebesgue measure zero. Among the combinations of edge weights inside this set, there are those that induce the emergence of symmetries (Chapman & Mesbahi, 2014, 2015) or equitable partitions (Gambuzza & Frasca, 2019) in the network graph. In the presence of symmetries, there exist permutations of the network nodes that leave the graph unchanged, and these symmetries induce a partition of the network in clusters. On the other hand, an equitable partition (Godsil, 1997) clusters the network nodes such that the sum of the incoming edges in any node of the same cluster from nodes in any cluster is the same. While symmetries and equitable partitions cause loss of controllability (Aguilar & Gharesifard, 2017), they also induce the emergence of group consensus and cluster synchronisation(Blaha et al., 2019; Klickstein et al., 2019; Pecora et al., 2014), i.e. solutions in which the state of each node in the same cluster is the same. It has been shown that many real world systems such as power grids (Della Rossa et al., 2020), yeast protein networks, or social systems (see Table 1 in MacArthur et al., 2008) share this degeneracy. Notably, these systems usually exhibit collective behaviour, and thus can achieve cluster consensus or synchronisation in the presence of symmetries.

In this work, we focus on developing tools allowing to enforce group consensus in networks with symmetries. Group consensus arises when some members, or agents, of the network reach consensus with each other, but not necessarily with the other members of the network (Klickstein et al., 2019; Qin et al., 2016; Qin & Yu, 2013; Sorrentino et al., 2020; Yu & Wang, 2009, 2010, 2012).

Group consensus is desirable whenever the apparently conflicting control goals of achieving different consensus values in groups of agents and of maintaining the network topology connected arise. Think for instance of a feet of autonomous vehicles attempting to rendezvous in groups at different times while maintaining the communication topology connected for safety. Differently from the existing literature (Chapman & Mesbahi, 2014, 2015), here we cope with the general case in which the network dynamics is not constrained to be in Laplacian form. The assumption that the network connectivity is in Laplacian form is prevalent in the existing literature. However, this

CONTACT Francesco Lo Iudice francesco.loiudice2@unina.it Department of Information Technology and Electrical Engineering, University of Naples Federico II, Naples 80125, Italy

assumption may be relaxed in applications in various fields, such as opinion dynamics, social interaction systems, and election campaign strategies (Acemoglu & Ozdaglar, 2011; Galam, 2004; Hołyst et al., 2001), for which the connectivity between agents may not be necessarily in Laplacian form, or may include antagonistic interactions (Altafini & Lini, 2014). Under these more general conditions, we show that loss of controllability and emergence of group consensus are different sides of the same coin. Both are due to the presence of symmetry-induced invariant subspaces that are smaller than the entire state space. While the dynamics orthogonal to the group consensus subspace is not controllable, it is possible that the dynamics along this subspace can be controlled. If this is the case, control of the consensus solution can be achieved by designing controllers on a reduced network, whose nodes correspond to clusters of nodes of the original network, yielding a substantial computational advantage in the control design.

Stabilisability of the dynamics orthogonal to the group consensus subspace is a necessary requirement to achieve group consensus. When the network dynamics is not described by a Laplacian matrix (which is the case considered in this paper), stabilisability is not guaranteed. This indicates the need for enforcing specific stabilisation conditions in order to obtain convergence of the dynamics on the group consensus subspace. In this paper we show how to add inputs to the network with the specific goal to stabilise the group consensus subspace. Moreover, we show how to independently design the control action on the group consensus subspace and the stabilising action transverse to the subspace. We also give bounds on the number of independent control inputs and on the number of nodes where these inputs are injected to achieve stabilisability of the group consensus subspace.

2. Mathematical preliminaries and network dynamics

We denote by $\mathcal{G}(\mathcal{V},\mathcal{E})$ an undirected graph with $\mathcal{V}=\{v_i,\ i=1,\dots,N\}$ the set of N nodes, and $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$ the set of edges defining the interconnections among the nodes. The symmetric matrix $A\in\mathbb{R}^{N\times N}$ is the adjacency matrix of the graph, that is, a matrix whose elements are $A_{ij}=A_{ji}\neq 0$ if $(i,j)\in\mathcal{E}$ and $A_{ij}=A_{ji}=0$ otherwise. A permutation $\pi(\mathcal{V})=\widetilde{\mathcal{V}}$ is an automorphism (or symmetry) of $\mathcal{G}$ if (i) $\mathcal{V}=\widetilde{\mathcal{V}}$, i.e. π does not add or remove nodes, and (ii) $(i,j)\in\mathcal{E}$, then $(\pi(i),\pi(j))\in\mathcal{E}$. The set of automorphisms of a graph with adjacency matrix A, with the operation composition, is the automorphism group, $aut(\mathcal{G}(A))$. Any permutation of this group can be represented by a permutation matrix P that commutes with A, i.e. such that $PA=AP$. The set of all automorphisms in the group will only permute certain subsets of nodes (the *orbits* or *clusters*) among each other. For any two nodes in the same orbit there exists a permutation that maps them into each other. Moreover, the *coarsest orbital partition* is defined as the partition of the nodes corresponding to the orbits of the automorphism group. Given a partition Π of the set $\mathcal{V}$ of the network nodes $\mathcal{V}$ into s subsets $\{S_1,S_2,\dots S_s\}$, such that $\cup_{i=1}^{s}S_i=\mathcal{V}$, $S_i\cap S_j=\emptyset$ for $i\neq j$, we can introduce the $N\times s$ indicator matrix E^{Π}, such that $E^{\Pi}_{ij}=1$ if node i belongs to S_j and $E^{\Pi}_{ij}=0$ otherwise. A simple example of a network with symmetries is reported in the left panel of Figure 1. The automorphism group of the network is composed of four elements, $aut(\mathcal{G}(A))=\{\pi_0,\pi_1,\pi_2,\pi_1\pi_2\}$, where

- π_0 is the trivial permutation ($\pi_0(i)=i,\ i=1,\dots,8$);
- π_1 is the horizontal symmetry of the network ($\pi_1(1)=2,\pi_1(2)=1,\ \pi_1(3)=4,\ \pi_1(4)=3,\ \pi_1(5)=5,\ \pi_1(6)=6,\ \pi_1(7)=8,\pi_1(8)=7$);
- π_2 is the vertical symmetry of the network ($\pi_1(1)=3,\pi_1(2)=4,\ \pi_1(3)=1,\ \pi_1(4)=2,\ \pi_1(5)=6,\ \pi_1(6)=5,\ \pi_1(7)=7,\pi_1(8)=8$);

The axes of π_1 and π_2 are the dashed lines in Figure 1. The automorphism group partitions the network nodes into three subsets ($S_1=\{1,2,3,4\}$, $S_2=\{5,6\}$, and $S_3=\{7,8\}$).

We consider a linear dynamical network described by

$$\dot{x}=Ax+Bu. \tag{1}$$

where $x\in\mathcal{X}=\mathbb{R}^N$ is the stack vector of the states of the N network nodes and u is the stack vector of the M input signals injected in the network. Consistently, the $N\times N$ symmetric matrix A defines the network topology, while the $N\times M$ matrix B describes the way in which the M input signals affect the network dynamics. Namely, if the j-th input is injected in the i-th node then $B_{ij}=1$, while $B_{ij}=0$ otherwise.

3. Controllability properties of networks with symmetries

In this section, we will show how the presence of symmetries in the controlled network (1) affects controllability.

Lemma 3.1: *The subset of automorphisms of $\mathcal{G}(A)$ given by the set of matrices $\mathcal{P}:=\{P_i:\ P_iA=AP_i$ and $P_iB=B\}$ forms a subgroup of $aut(\mathcal{G}(A))$.*

Proof: For the set $\mathcal{P}$ to be a subgroup, the following four properties must be true:

(i) $P_i(P_jP_k)=(P_iP_j)P_k\ \forall(P_i,P_j,P_k)\in\mathcal{P}$;
(ii) $P_i\in\mathcal{P}$ is non-singular $\forall\ i$;
(iii) $I\in\mathcal{P}$;
(iv) given any two matrices $P_i\in\mathcal{P}$ and $P_j\in\mathcal{P}$, then $P_iP_j\in\mathcal{P}$.

Proving that the matrices in $\mathcal{P}$ satisfy property (i) and (ii) is trivial as (i) is true for any three square matrices with the same dimensions $(P_i,P_j,P_k)\in\mathcal{P}$ regardless of whether these are, or are not, in $\mathcal{P}$, while (ii) is true as permutation matrices are not singular. Moreover, (iii) holds as $IA=AI=A$, and $IB=B$. Moreover, property (iv) is proved as

$$(P_iP_j)A=P_i(P_jA)=P_i(AP_j)=AP_iP_j=A(P_iP_j)$$

from which we see that $P_iP_jA=AP_jP_i$ for all $(P_i,P_j)\in\mathcal{P}$. The proof is finally completed by noting that, as from our hypotheses $P_jB=P_iB=B$ for all $(P_i,P_j)\in\mathcal{P}$, it follows that $P_iP_jB=P_iB=B$. ■

We will denote as $aut(\mathcal{G}(A,B))$ the group represented by the permutation matrices P such that $PA-AP=0$ and $PB-B=0$.

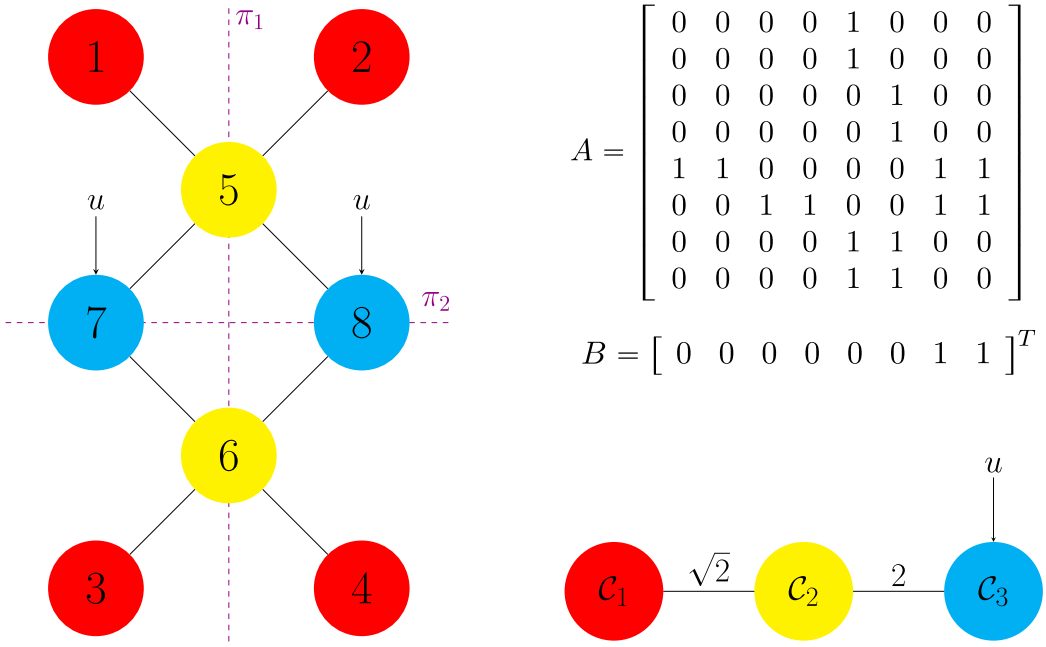

Figure 1. A simple 8 node network, with edge weights all equal to one. The coarsest orbital partition of the network shown in the figure has three clusters $\mathcal{C}_1 = \{5, 6\}$, $\mathcal{C}_2 = \{1, 2, 3, 4\}$, and $\mathcal{C}_3 = \{7, 8\}$.

Similarly to $aut(\mathcal{G}(A))$, $aut(\mathcal{G}(A, B))$ partitions the set of network nodes into orbits or clusters, where an orbit is a subset of symmetric nodes. Hence, we can define the coarsest orbital partition Π into clusters corresponding to the orbits of the automorphism group $aut(\mathcal{G}(A, B))$, $C_1, C_2, \ldots, C_K$, such that $\cup_{i=1}^{K} C_i = \mathcal{V}$, and $C_i \cap C_j = 0$ for $i \neq j$. We will use the indicator matrix E^{Π} to keep track of the orbit to which each node belongs.

Lemma 3.2: *Each orbit of the coarsest partition* Π *induced by* $aut(\mathcal{G}(A, B))$ *is a subset of an orbit of the coarsest partition induced by* $aut(\mathcal{G}(A))$.

Proof: The thesis follows from the observation that if two (or more) nodes are permuted by a permutation matrix P in $aut(\mathcal{G}(A, B))$ and thus belong to the same orbit, then they also belong to the same orbit of the coarsest orbital partition induced by $aut(\mathcal{G}(A))$, as the same matrix P also belongs to $aut(\mathcal{G}(A))$. ■

Theorem 3.1: *If there exists a permutation matrix* $P \neq I$ *such that* $PA - AP = 0$ *and* $PB - B = 0$, *then*

(i) *the set of states* $\mathcal{X}_{or} := \{x : x_i = x_l \ \forall\, i, l \in \mathcal{C}_j,\ \forall j\} \subset \mathcal{X}$, *is an invariant subspace of the matrix A, i.e.* $\forall x \in \mathcal{X}_{or}$, $Ax \in \mathcal{X}_{or}$;
(ii) *if* $x_i = x_l$ *then* $\dot{x}_i = \dot{x}_l$ *for all* $(i, l) \in \mathcal{C}_j$ *and for all j.*

Proof: Let us start by showing that if there exists a permutation matrix P such that $PA = AP$ and $PB = B$, then the network state x and the permuted state vector $y := Px$ share the same dynamics. Indeed, by left multiplying both sides of Equation (1) by P we get

$$P\dot{x} = PAx + PBu.$$

Then, as $PA = AP$ and $PB = B$, we get

$$\dot{y} = Ay + Bu.$$

Moreover, as there always exists a permutation matrix $P \in aut(\mathcal{G}(A, B))$ that maps any two nodes belonging to the same clusters into each other (Klickstein et al., 2019), then statement (ii) follows, i.e.nodes in the same clusters share the same dynamics, and thus if $x_i = x_l$ for all i and l in the same cluster, then also $\dot{x}_i = \dot{x}_l$. Moreover, this also ensures that the subspace made of all the points of the state-space such that $x_i = x_l\ \forall i, l \in \mathcal{C}_j$ and $\forall j = 1, \ldots, K$ is A-invariant (statement (i)). ■

Theorem 3.1 establishes the existence of the group consensus subspace $\mathcal{X}_{or}$ for network (1). Hence, to tackle consensus control problems, it is useful to introduce a transformation that allows us to separate the dynamics along $\mathcal{X}_{or}$ from that orthogonal to $\mathcal{X}_{or}$ itself. This task is accomplished by the Irreducible Representation (IRR) of the symmetry group through a transformation in a new coordinate system (Pecora et al., 2014) $z_{or} = T_{or}x$. The transformation matrix

$$T_{or} = \begin{bmatrix} T^{\parallel} \\ T^{\perp} \end{bmatrix} \in \mathbb{R}^{N \times N} \tag{2}$$

is orthogonal, and the elements of the block $T^{\parallel} \in \mathbb{R}^{K \times N}$ are such that

$$T^{\parallel}_{ij} = \sqrt{|\mathcal{C}_i|}^{-1} \tag{3}$$

if node j is in cluster i and 0 otherwise. Note that the K rows of the matrix $T^{\parallel}$ are a basis of $\mathcal{X}_{or}$ while the rows of the matrix $T^{\perp} \in \mathbb{R}^{(N-K) \times N}$ are a basis of the orthogonal complement to the group consensus subspace. Notably, each of the rows of the matrix $T^{\perp}$, say the j-th, can be associated to a single cluster say $\mathcal{C}_i$. Namely, each element $T^{\perp}_{jl}$ is non-zero only if node l belongs to the cluster $\mathcal{C}_i$. Consistently, the dynamic matrix $\tilde{A} = T_{or} A T_{or}^{T}$

has the following structure:

$$\tilde{A} = T_{or} A T_{or}^T = \begin{bmatrix} A_\parallel & 0 \\ 0 & A_\perp \end{bmatrix}. \tag{4}$$

From Equation (4), we see that the IRR decouples the dynamics along the consensus subspace governed by the block $A_\parallel$ from that orthogonal to the group consensus subspace governed by the block $A_\perp$. In this new coordinate system, the dynamics of network (1) can be rewritten as

$$\dot{z}_{or} = \tilde{A} z_{or} + \tilde{B} u, \tag{5}$$

and

$$\tilde{B} = T_{or} B = \begin{bmatrix} B_\parallel \\ B_\perp \end{bmatrix}. \tag{6}$$

Indeed, the pair $(A_\parallel, B_\parallel)$, which we will denote as the *quotient pair*, determines the controllability properties of the dynamics along the subspace $\mathcal{X}_{or}$ and thus our ability to control the consensus state, while the pair $(A_\perp, B_\perp)$ determines our ability to stabilise $\mathcal{X}_{or}$. We are interested in studying the controllability properties of the two pairs $(A_\parallel, B_\parallel)$ and $(A_\perp, B_\perp)$. Before doing so, we will present a few more details on this representation. First of all, let us point out that the block $T_\parallel$ of the matrix T_{or} is such that $T_\parallel = E^{\Pi^\dagger}$, where $E^\Pi \in \mathbb{R}^{N\times K}$ is the indicator matrix corresponding to the coarsest orbital partition Π. Consistently, the state of the quotient network, the network associated to pair $(A_\parallel, B_\parallel)$, can be computed as

$$z^\parallel = {E^\Pi}^\dagger x \in \mathbb{R}^K$$

and thus we have that $A_\parallel = {E^\Pi}^\dagger A E^\Pi$ and $B_\parallel = {E^\Pi}^\dagger B$.

Now, we are ready to give the following theorem.

Theorem 3.2: *If there exists a matrix $P \neq I$ such that $PA = AP$ and $PB = B$, then $\mathcal{X}_{or}$, the invariant subspace of the matrix A associated to the cluster consensus solution, encompasses the controllable subspace.*

Proof: To prove the statement we must show that if $PB = B$, $\mathcal{X}_{or}$ encompasses the range of B. Indeed, if $PB = B$, then B is such that $b_{il} = b_{jl}$ for all l and for all i, j in the same cluster, due to the fact that left-multiplying a vector by the matrix P only permutes the elements associated to nodes of the same cluster. Hence, all the columns of B and thus its range, are encompassed in $\mathcal{X}_{or}$. As the controllable subspace is defined as the smallest A-invariant subspace encompassing the range of B, the thesis follows. ■

Corollary 3.1: $B_\perp = \mathbf{0}_{(N-K)\times M}$.

Proof: The statement is a direct consequence of the statement of Theorem 3.2 and of the definition of $B_\perp$. ■

4. Controlling group consensus

In Section 3 we have established some controllability limitations of networks with symmetries. Here, we show how to operate within these limitations in order to control group consensus.

Corollary 4.1: *Consider a graph $\mathcal{G}(A, B)$ with coarsest orbital partition Π. If the pair $(A_\parallel, B_\parallel)$ is controllable, then for any cost function $J(u(t))$ the optimal control problem*

$$\min_u \quad \int_0^{t_f} J(u(t))\,\mathrm{d}t \tag{7a}$$

$$\text{s.t.} \quad \dot{x} = Ax + Bu \tag{7b}$$

$$x(0) = x_0 \tag{7c}$$

$$x(t_f) = x_f \tag{7d}$$

admits solution $u^*(t) := \operatorname{argmin} \int_0^{t_f} J(u(t))\,\mathrm{d}t$ if and only if x_0 and x_f are such that $T_{or} x_0 = [z_0^\parallel\ 0]^T$ and $T_{or} x_f = [z_f^\parallel\ 0]^T$, i.e. $z_0^\perp = z_f^\perp = 0$. Moreover, if $T_{or} x_f = [z_f^\parallel\ 0]$, then $u^* = u^{**}$, where u^{**} is the solution of the following optimal control problem

$$\min_u \quad \int_0^{t_f} J(u(t))\,\mathrm{d}t \tag{8a}$$

$$\text{s.t.} \quad \dot{z}^\parallel = A_\parallel z^\parallel + B_\parallel u \tag{8b}$$

$$z^\parallel(0) = T^\parallel x_0 \tag{8c}$$

$$z^\parallel(t_f) = T_\parallel x_f. \tag{8d}$$

Proof: From Theorem 3.2, if x_f is such that $z_f^\perp \neq 0$ then x_f is not reachable, while if x_0 is such that $z_0^\perp \neq 0$ then x_f is not reachable from x_0. Hence, in both cases problem (7a) is not feasible. On the other hand, if x_0 and x_f are such that $z_0^\perp = z_f^\perp = 0$, then both x_0 and x_f belong to $\mathcal{X}_{or}$, which we know coincides with the controllable subspace from Theorem 3.2 and from the hypotheses. Then, reaching $z_f^\parallel$ is equivalent to reaching the point x_f. Hence, to prove our thesis, we are left with showing that $u^* = u^{**}$. We will do so by showing that problems (7a) and (8a) share the same decision variables, cost function, and constraints. Indeed, the decision variables are the same by definition, as well as the cost function as input signals are not affected by equivalent transformations. Finally, to prove that problems (7a) and (8a) share the same constraints, let us show that by left multiplying both sides of Equations (7b)–(7d), we obtain Equations (8b)–(8d) together with a set of Equations that are always verified independently of u. Indeed from the hypotheses this is true for Equations (7c) (7d), as left multiplying both by the matrix T_{or} we obtain Equations (8c) and (8d) together with two sets of $N-K$ equations of the type $0 = 0$. Finally, from Equations (4), (6), and Corollary 3.1 we know that left-multiplying Equation (7b) by T_{or} yields

$$\dot{z}^\parallel = A_\parallel z^\parallel + B_\parallel u \tag{9a}$$

$$\dot{z}^\perp = A_\perp z^\perp. \tag{9b}$$

As $z^\perp(0) = 0$, from Equation (9b) we have that $z^\perp(t) = 0$ for all t, and thus Equation (9a), which coincides with Equation (8b), captures completely the dynamics in Equation (7b) independently of u. Hence, problem (7a) and the reduced order problem in (8a) share the same decision variables, cost function, and constraints which implies that $u^* = u^{**}$. ■

Remark 4.1: Note that Corollary 4.1 provides an approach to design an input to control group consensus. A viable alternative is to solve

$$\min_u \int_0^{t_f} J(u(t))\,\mathrm{d}t \tag{10a}$$

$$\text{s.t.} \quad \dot{x} = Ax + Bu \tag{10b}$$

$$y = {E^{\Pi}}^T x \tag{10c}$$

$$x(0) = x_0 \in \mathcal{X}_{or} \tag{10d}$$

$$y(t_f) = y_f. \tag{10e}$$

with E^{Π} being the indicator matrix corresponding to the partition Π of the network nodes, and

$$\frac{y_i}{|\mathcal{C}_i|}$$

being the consensus value for all the nodes of the cluster $\mathcal{C}_i$.

Remark 4.2: Corollary 4.1 provides an approach to control the consensus solution. The stability properties of the group consensus subspace are determined by the eigenvalues of the block $A_\perp$ of the matrix $\tilde{A}$ in Equation (4). However this solution is not stabilisable, as the dynamics orthogonal to the group consensus subspace are uncontrollable (see Theorem 3.2).

Motivated by the considerations in Remark 4.2 which are consistent with the statement of Theorem 3 in Alemzadeh et al. (2017) coping with the special case of signed graphs relevant to model opinion dynamics (Altafini & Lini, 2014), we now tackle the problem of selecting a set of nodes in which additional inputs must be injected to stabilisable $\mathcal{X}_{or}$. To do so, we leverage the following conditions from Hautus (1970).

Definition 4.1: Given a pair (A, B) an eigenvalue λ_i of A is controllable if and only if $\exists j$ such that $v_i^T b_j \neq 0$, for any eigenvector v_i associated to λ_i.

Theorem 4.1 ((Hautus, 1970)): *A dynamical system defined by the pair (A, B) is stabilisable if and only if every unstable eigenvalue of A is controllable.*

We denote by w the W-dimensional vector of the additional inputs and by D the $N \times W$ dimensional matrix indicating the nodes in which these inputs are injected, that is, the drivers. Namely, $D_{ij} \neq 0$ if the jth additional input w_j is injected in the ith network node and 0 otherwise. Considering these additional inputs leads to rewriting the dynamics of the network in Equation (1) as

$$\dot{x} = Ax + Bu + Dw. \tag{11}$$

As a result, applying the transformation T_{or} in Equation (4) to the controlled network in Equation (11) yields

$$\dot{z} = \tilde{A}z + \tilde{B}u + \tilde{D}w, \tag{12}$$

where

$$\tilde{D} = T_{or}D = \begin{bmatrix} D_{\|} \\ D_\perp \end{bmatrix}. \tag{13}$$

We constrain the selection of the matrix D to be such that the input signals w do not affect the dynamics along the group consensus subspace, so to allow independent design of (i) the control action u responsible for controlling the group consensus solution and (ii) the stabilising action w.

To be able to formulate and solve our driver node selection problem, let us relabel the eigenvalues of A so that the first K are also eigenvalues of $A_{\|}$ and the last $(N-K)$ are also eigenvalues of $A_\perp$ (here we just list all the eigenvalues of A regardless of their multiplicity). Note that this is possible from the block diagonal structure of $\tilde{A}$ in Equation (4). After this relabelling, the eigenvectors of A associated with its first K eigenvalues span the group consensus subspace, while the eigenvectors of A associated with the last $(N-K)$ eigenvalues span its orthogonal complement. In particular, the last $(N-K)$ eigenvalues of A determine the stability properties of the group consensus subspace. Moreover, we denote by Ω_i the subspace of the eigenspace of the eigenvalue λ_i of A that is orthogonal to $\mathcal{X}_{or}$ and by μ_i the dimension of Ω_i. Given a vector d, we denote by $proj_{\Omega_i}(d)$ its projection on Ω_i. Finally, we denote by $\Lambda^\perp$ the subset of the eigenvalues of A with non-negative real part that are also eigenvalues of $A_\perp$. Thanks to these preliminary considerations and notation, we can now formulate our driver node selection problem

Problem 4.1: Select a matrix D such that

$$D_{\|} = 0 \tag{14a}$$

$$(A_\perp, D_\perp) \text{ is stabilisable} \tag{14b}$$

Algorithm 1 Driver Node Selection Algorithm

procedure INITIALISATION ($i = 1$, D is the empty matrix, $j = 0$)
  while $i \leq |\Lambda^\perp|$ **do**
    $\Delta_i = \{D_j : proj_{\Omega_i}(D_i) \neq 0 \quad \wedge \quad \nexists D_k : proj_{\Omega_i}(D_k) \parallel proj_{\Omega_i}(D_j)\}$
    $h_i = |\Delta_i|$
    while $j \leq \mu_i - h_i$ **do**
      $j = j + 1$
      Build an N-dimensional vector D_j by solving

$$proj_{\Omega_i}(D_j) \neq 0 \tag{15}$$

$$proj_{\Omega_i}(D_j) \neq proj_{\Omega_i}(D_m)\ \forall m < j \tag{16}$$

$$\sum_{k \in \mathcal{C}_l} D_j(k) = 0\ \forall l \tag{17}$$

      $D = [D\ D_j]$
    end while
    $i = i + 1$
  end while
end procedure

Algorithm 1 prescribes to initialise the matrix D as an empty matrix. Then, for all the eigenvalues in the set $\Lambda^\perp$, we find the number h_i of columns of the matrix D with non-zero and

linearly independent projection on Ω_i, that is, the subspace of the eigenspace associated to λ_i that is orthogonal to the group consensus subspace. Then, we add $\mu_i - h_i$ column vectors to the matrix D each having non-zero and linearly independent projection on Ω_i, thus ensuring, from Definition 4.1 that λ_i is controllable. Thanks to the condition in Equation (29), these $\mu_i - h_i$ added columns will be orthogonal to the group consensus subspace thus ensuring $D_\parallel = 0$. Doing so for all λ_i in $\Lambda^\perp$ ensures the pair $(A_\perp, D_\perp)$ is stabilisable thanks to Theorem 4.1.

Theorem 4.2: *Algorithm 1 solves Problem 4.1.*

Proof: To prove that any matrix selected by Algorithm 1 satisfies condition (14a) it suffices to note that from Equation (13) and the structure of the matrix T_{or} in Equation (2) we have that the ith element of the jth column of $D_\parallel$ is obtained as $\sum_{k\in\mathcal{C}_i} D_j(k)$. Then, Equation (14a) follows directly from Equation (29). On the other hand, note that from Theorem 4.1 and Definition 4.1, to prove that any matrix selected according to Algorithm 1 satisfies (14b) it suffices to show that for each eigenvector, say $v_j^\perp$ of $A_\perp$ associated to an eigenvalue that is encompassed in the set $\Lambda^\perp$ there exists a column $D_l^\perp$ of the matrix $D_\perp$ such that ${v_j^\perp}^T D_l^\perp \neq 0$. In turn, as any μ_i vectors of Ω_i can be chosen as eigenvectors of $A_\perp$, and as the columns of $D_\perp$ are the projection of the columns of D on the orthogonal complement to the group consensus subspace, ensuring that for any $v_j^\perp$ associated to an eigenvalue $\lambda_i \in \Lambda^\perp$ there exists $D_l^\perp$ such that ${v_j^\perp}^T D_l^\perp \neq 0$ is equivalent to ensuring that there exist μ_i columns of D that span Ω_i. As this is ensured by the inner while loop in Algorithm 1 thanks to Equations (27) and (28), the thesis follows. ■

Remark 4.3: Note that while indeed the symmetries of the pair $(A, [B\,D])$, with D selected according to Algorithm 1, are not the same of that of the pair (A, B), this has no effect on the dynamics along the group consensus manifold as from Problem 4.1 and Theorem 4.2 we know that $D_\parallel = 0$. Consistently, as the control signal w is conceived to be a stabilising feedback action, it will vanish asymptotically, and in the absence of perturbations the network dynamics will revert to that in Equation (1).

Corollary 4.2: *The number of independent input signals required to solve Problem 4.1 is lower bounded by*

$$\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|.$$

Proof: Let us start by noting that any vector in Ω_i is an eigenvector of $A_\perp$ associated to λ_i. Hence, for the stabilisability condition in Theorem 4.1 to be verified for the pair $(A_\perp, D_\perp)$, we must have that for all $\lambda_i \in \Lambda^\perp$ there exist $|\Omega_i|$ columns of $D^\perp$, and thus also of D, with non-zero and non-parallel projection on Ω_i. Hence, the pair $(A_\perp, D_\perp)$ can be stabilisable only if the number of columns of D is at least equal to $\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|$ which proves our statement. ■

After giving a bound on the number of input signals required to solve Problem 4.1, let us now give a bound on the number of drivers, i.e. the number of rows of D encompassing at least a non-zero entry, required to solve Problem 4.1. To do so, let us define the operator

$$|\cdot|_\emptyset := \begin{cases} |\cdot| \text{ if } |\cdot| > 0 \\ -1 \text{ otherwise} \end{cases}$$

Corollary 4.3: *The number of rows of the matrix D with at least one non-zero entry required to solve Problem 4.1 is lower bounded by*

$$\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|_\emptyset + 1.$$

Proof: From Corollary 4.2, we know that the number of columns of D required to stabilise $\mathcal{X}_{or}$ is lower bounded by $\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|$. As the projections of these columns on Ω_{i^*}, with $i^* = \text{argmax}_{i:\lambda_i\in\Lambda^\perp}|\Omega_i|$, must be non-zero and non-parallel, then the rank of the matrix D is lower bounded by $\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|$. On the other hand, to ensure the condition in (14a) is fulfilled, each column of D must be parallel to $\mathcal{X}_{or}$ which is true iff the columns of D verify Equation (29), that is, their elements sum to zero. Hence, for the matrix D to be zero column sum and have at least rank $\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|$ it must have at least $\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i|_\emptyset + 1$ rows encompassing non-zero entries thus proving our statement. ■

Corollary 4.3 provides a bound on the number of driver nodes required to solve Problem 4.1. We will now show how to exploit the clusters induced by the network symmetries to give a different bound from that provided in Corollary 4.3. To do so, let us denote by Ω_i^j the subspace of Ω_i that is spanned by vectors e_l, $l = 1, \ldots, |\Omega_i^j|$ such that each element e_{lm} of e_l is non-zero iff node m is encompassed in cluster $\mathcal{C}^j$. Roughly speaking, Ω_i^j is the jth cluster specific subspace of Ω_i. As in general Ω_i cannot be completely spanned by cluster specific vectors, we have that $\Omega_i = \cup_{j=1}^K \Omega_i^j + \tilde{\Omega}_i$, where $\tilde{\Omega}_i$ is thus the subspace of Ω_i that cannot be spanned by cluster specific vectors. Finally let us relabel the network nodes so that node i belongs to $\mathcal{C}_j$ if $|\mathcal{C}_{j-1}| < i \leq |\mathcal{C}_j|$, with $|\mathcal{C}_0| = 0$ as $\mathcal{C}_0$ does not exist. Then, the matrix D can be decomposed in blocks as follows

$$D = \begin{bmatrix} D^1 \\ D^2 \\ \vdots \\ D^K \end{bmatrix} \tag{18}$$

with each D^j having $|\mathcal{C}_j|$ rows.

Corollary 4.4: *The number of rows of the matrix D encompassing non-zero entries required to solve Problem 4.1 is lower bounded by*

$$\sum_{j=1}^K \left(\max_{i:\lambda_i\in\Lambda^\perp} |\Omega_i^j|_\emptyset + 1 \right). \tag{19}$$

Proof: From Theorem 4.1, Definition 4.1, and Equation (14a), we know that to solve Problem 4.1 we need to ensure that each $\lambda_i \in \Lambda^\perp$ is made controllable by a matrix D such that

$\sum_{l\in\mathcal{C}_j} D_{li} = 0 \forall i$. Moreover, from Corollary 4.2 and as Ω_i^j is spanned by cluster specific vectors, it is possible to show that to ensure an eigenvalue $\lambda_i \in \Lambda^\perp$ is controllable we need that at least $|\Omega_i^j|$ columns of the matrix D^j have non-zero and non-parallel projection on Ω_i^j. Hence, these columns must define a matrix D that is both full rank and also zero column sum so to ensure fulfillment of Equation (14a). This implies that stabilising any $\lambda_i \in \Lambda^\perp$ requires that at least $|\Omega_i^j| + 1$ rows of the block D^j encompass a non-zero entry for all j such that $\Omega_i^j \neq \emptyset$. Hence, the total number of rows of the matrix D encompassing a non-zero entry is lower bounded by the quantity in (19). ■

Remark 4.4: The problem of identifying the cluster specific vectors spanning the subspaces Ω_i^j for all i and j can be easily solved using the IRR transformation T_{or}. Indeed, one of the properties of this transformation is to have cluster specific rows that can be linearly combined through the coefficients of the eigenvectors of the corresponding block of $\tilde{A}$ to generate eigenvectors of A. Therefore, each eigenvector of A associated to an eigenvalue λ_i obtained through this procedure either belongs to (i) Ω_i^j if the rows that are combined to obtain them are all associated to the same cluster $\mathcal{C}_j$, or (ii) $\tilde{\Omega}_i$ otherwise.

Remark 4.5: Corollaries 4.2, 4.3, and 4.4 give lower bounds on the number of input signals and on the number of rows of the matrix D encompassing non-zero entries so to achieve stabilisability of the dynamics transverse to $\mathcal{X}_{or}$. Let us remark that as each unstable eigenvalue requires an input signal to be stabilised, then the number of columns of the matrix D is upper bounded by $|\Lambda^\perp|$. Moreover, in order to ensure $D_\parallel = 0$, then the columns of D must sum to zero, the number of rows of the matrix D encompassing non-zero entries is upper bounded by $2|\Lambda^\perp|$. Finally, as Algorithm 1 is constructive, in the sense that it goes through the elements of the set $\Lambda^\perp$ adding an input signal whenever the current eigenvalue has not already been stabilised by the existing columns of the matrix D, its computational complexity grows linearly with the number of unstable eigenvalues of the matrix $A^\perp$.

5. Numerical examples

We consider the $N = 8$ node network in Figure 1. A study of the symmetries of the pair (A, B) shows that there are $K = 3$ orbital clusters, $\mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3 = \mathcal{V}$ and $\mathcal{C}_1 = \{1, 2, 3, 4\}$, $\mathcal{C}_2 = \{5, 6\}$, $\mathcal{C}_3 = \{7, 8\}$. The corresponding indicator matrix is

$$E^{\Pi^T} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \tag{20}$$

We tackle the problem of steering the network state towards the group consensus value $[\mathbf{1}_{1\times 4}\ \mathbf{2}_{1\times 2}\ \mathbf{3}_{1\times 2}]^T$ in $t_f = 5$ seconds. To do so, according to the results in Section 4 we must first decouple the dynamics along and transverse to the group consensus subspace by leveraging the state transformation $z = T_{or}x$ with

$$T_{or} = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & -0.5 & -0.5 \\ 0 & 0 & 0 & 0 \\ \sqrt{2}^{-1} & -\sqrt{2}^{-1} & 0 & 0 \\ 0 & 0 & \sqrt{2}^{-1} & -\sqrt{2}^{-1} \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2}^{-1} & \sqrt{2}^{-1} \\ \sqrt{2}^{-1} & \sqrt{2}^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \sqrt{2}^{-1} & -\sqrt{2}^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2}^{-1} & -\sqrt{2}^{-1} \end{bmatrix}, \tag{21}$$

obtaining

$$A_\parallel = \begin{bmatrix} 0 & 0 & \sqrt{2} \\ 0 & 0 & 2 \\ \sqrt{2} & 2 & 0 \end{bmatrix}, \quad B_\parallel = \begin{bmatrix} 0 \\ \sqrt{2} \\ 0 \end{bmatrix},$$

$$A_\perp = \begin{bmatrix} 0 & -\sqrt{2} & 0 & 0 & 0 \\ -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B_\perp = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{22}$$

Consistently with Corollary 3.1, we obtain that $B_\perp = 0$. Moreover, the pair $(A_\parallel, B_\parallel)$ defines the dynamics of the quotient network, whose three node structure is portrayed in Figure 1. As the reader may easily check, the pair $(A_\parallel, B_\parallel)$ is controllable, and thus to control the dynamics along $\mathcal{X}_{or}$ we pose the following minimum energy control problem:

$$\begin{aligned} \min_u \quad & \frac{1}{2}\int_0^5 u^T(t)u(t)\,\mathrm{d}t \\ \text{s.t.} \quad & \\ & \dot{z}^\parallel = A_\parallel z^\parallel + B_\parallel u \\ & z^\parallel(0) = T^\parallel x_0 \\ & z^\parallel(5) = T^\parallel[\mathbf{1}_{1\times 4}\ \mathbf{2}_{1\times 2}\ \mathbf{3}_{1\times 2}]^T = [2\ \ 3\sqrt{2}\ \ 2\sqrt{2}]^T \end{aligned} \tag{23}$$

where $z^\parallel \in \mathbb{R}^3$ is the state vector of the quotient network.

The solution of this optimal control problem is

$$\begin{aligned} u^{**}(t) &= B_\parallel^T e^{A_\parallel^T(5-t)} W^{-1}(z^\parallel(5) - e^{5A_\parallel} z(0)) \\ &= B_\parallel^T (V_\parallel^T)^{-1} e^{\Lambda_\parallel(5-t)} V_\parallel^T W^{-1}(z^\parallel(5) - V_\parallel^{-1} e^{5\Lambda_\parallel} V_\parallel z(0)) \end{aligned}$$

$$\approx -0.00003e^{\sqrt{6}(5-t)} + 2.54e^{-\sqrt{6}(5-t)} + 0.732 \tag{24}$$

where

$$W(t_0, t_f) = \int_{t_0}^{t_f} e^{A_{\|}(t_f-t)} B_{\|} B_{\|}^T e^{A_{\|}^T(t_f-t)}\, \mathrm{d}t \tag{25}$$

is the reachability gramian of the quotient network. Note that the optimal control input is a linear combination of the three eigenmodes corresponding to the three clusters of the orbital partition Π of $\mathcal{G}(A, B)$. It's worth underlining that, since the consensus subspace is unstable, numerical computation of the optimal control solution is hard due to the positive eigenvalue $\sqrt{6}$. Notably, due to the low dimensionality of the quotient network, the IRR allows us to solve (23) analytically.

Having dealt with controlling the dynamics along the group consensus subspace, we can now turn to stabilising the dynamics transverse to this subspace. To this aim, note that the spectrum of the matrix $A_\perp$ in (22) is composed of the following set of eigenvalues

$$\{-\sqrt{2},\ 0,\ \sqrt{2}\} \tag{26}$$

with the geometric multiplicity of the null eigenvalue being equal to 3, and the other two eigenvalues being simple. Hence, in order to apply Algorithm 1, we must first consider that $\Lambda^\perp = \{\ 0,\ \sqrt{2}\}$, with $\mu_1 = 3$, and $\mu_2 = 1$. Then, setting $i = 1$, and as D is initialised as the empty matrix, then $h_1 = 0$ as Δ is the empty set and we can enter the inner while loop. The three vectors spanning Ω_1 are the last three rows of the matrix T_{or} that brings the system in the IRR-coordinates, namely

$$\begin{bmatrix} \sqrt{2}^{-1} & -\sqrt{2}^{-1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2}^{-1} & -\sqrt{2}^{-1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{2}^{-1} & -\sqrt{2}^{-1} \end{bmatrix}^T$$

and a feasible solution that iteratively solves Equations (27)–(29) is

$$\begin{aligned} D_1 &= \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}^T, \\ D_2 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}^T, \\ D_3 &= \begin{bmatrix} 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T. \end{aligned} \tag{27}$$

Hence, we can turn to $i = 2$ noting that as the vector

$$\begin{bmatrix} -0.35 & -0.35 & 0.35 & 0.35 & -0.50 & 0.50 & 0 & 0 \end{bmatrix}^T$$

is a basis for Ω_2, then $h_2 = 1$ as there already exists a column of D, namely D_1 in Equation (27) with non-zero projection on Ω_2. Hence, as $\mu_2 = 1$, and $|\Lambda^\perp| = 2$, the driver node selection procedure comes to an end. Note that this solution achieves both the bound given in Corollary 4.2 as well as that given in Corollary 4.4 and thus minimises both the number of input signals and the number of driver nodes required to stabilise $\mathcal{X}_{or}$.

Having performed the selection of the matrix D that ensures stabilisability of the pair $(A_\perp, D_\perp)$ we can now turn our attention to designing the stabilising signal w as

$$w = -Gz_\perp$$

with the matrix G being such that the eigenvalues of the matrix $(A_\perp - D_\perp G)$ are all smaller than or equal to $-\sqrt{2}$, the only negative eigenvalue of $A_\perp$ which we do not move. Specifically, we design G so that all the originally non-negative eigenvalues are placed in -2. This selection ensures that the slowest time constant of the transverse dynamics is the one of the only stable eigenvalue we did not touch $(1/\sqrt{2})$. Note that this placement ensures the transverse dynamics become negligible well before the time $t_f = 5$ when the dynamics parallel to the group consensus subspace will converge to the target state $z^{\|}(5)$. The designed control inputs can be now used to steer the network towards the group consensus state $[\mathbf{1}_{1\times 4}\ \mathbf{2}_{1\times 2}\ \mathbf{3}_{1\times 2}]$. In Figure 2 we report the network state evolution (panel a) and the control inputs (panel b). As expected, the optimal control input u^{**} in Equation (24), shown in black in Figure 2(b) is able to steer the nodes in $\mathcal{C}_1$ to 1, the nodes in $\mathcal{C}_2$ to 2 and the nodes in $\mathcal{C}_3$ to 3 at $t_f = 5$. In the meantime, the stabilising control input w makes the transverse clustered synchronous solution stable, ensuring the network state converges on the cluster consensus subspace. Note that as expected, this control action vanishes in time, as shown in Figure 2(b).

Applying Algorithm 1 to the eight node network in Figure 1 yielded a selection of six driver nodes in order to stabilise $\mathcal{X}_{or}$, that is, 75% of the network nodes. We now consider a larger network with $N = 48$ nodes, shown in Figure 3(a), obtained using the algorithm proposed in Klickstein and Sorrentino (2018). We assume that the same input signal u is injected in all the nodes i such that $21 \le i \le 35$. A study of the symmetries of the pair (A, B) for this network shows that there are $K = 3$ orbital clusters with $\mathcal{C}_1 := \{i : i \le 20\}$, $\mathcal{C}_2 := \{i : 21 \le i \le 36\}$, and $\mathcal{C}_3 := \{i : i \ge 37\}$ defining the quotient network in Figure 3(b). Applying the transformation in Equation (4) and computing the eigenvalues of the matrix $A_\perp$ in Equation (5), we find that $|\Lambda^\perp| = 8$ and that $\sum_{i:\lambda_i\in\Lambda^\perp} \mu_i = 19$, that is, the number of eigenvectors associated to the non-stable eigenvalues of $A_\perp$ is 19. Hence, in order to ensure the network in Figure 3(a) achieves group consensus we need to select an additional set of driver nodes defining the matrix D in Equation (11). To do so, we apply Algorithm 1 finding that eight input signals, i.e. a matrix D with eight columns, are sufficient to stabilise the dynamics transverse to $\mathcal{X}_{or}$. Notably, only 11 rows of the matrix D encompass at least one non-zero entry, and thus only 11 driver nodes, roughly 23% of the network nodes, are sufficient to stabilise $\mathcal{X}_{or}$, five of which were already nodes in which the input signal u is injected. In the appendix we give all the details on the driver node selection procedure for this example, showing that the bound in Corollary 4.4 is achieved also for the 48 node network considered here. Figure 3(c), shows the trajectory generated by the joint action of an optimal controller u^{**} which solves the problem

$$\begin{aligned} \min_u \quad & \frac{1}{2}\int_0^5 u^T(t)u(t)\,\mathrm{d}t \\ \text{s.t.} \quad & \\ & \dot{z}^{\|} = A_{\|}z^{\|} + B_{\|}u \\ & z^{\|}(0) = T^{\|}x_0 \\ & z^{\|}(1) = T^{\|}[\mathbf{1}_{1\times 20}\ \mathbf{2}_{1\times 16}\ \mathbf{3}_{1\times 12}]^T = [\sqrt{20}\ \ 8\ \ 3\sqrt{12}]^T \end{aligned} \tag{28}$$

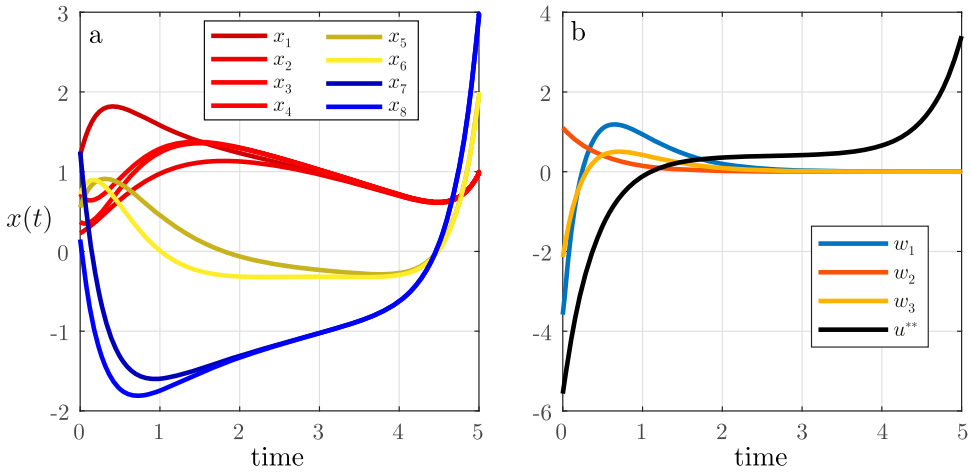

Figure 2. (a) State trajectories of the original network. Those of nodes in cluster $\mathcal{C}_1$ converge to 1, those of nodes in cluster $\mathcal{C}_2$ converge to 2, while those of nodesin cluster C_3 converge to 3. (b) Control inputs.

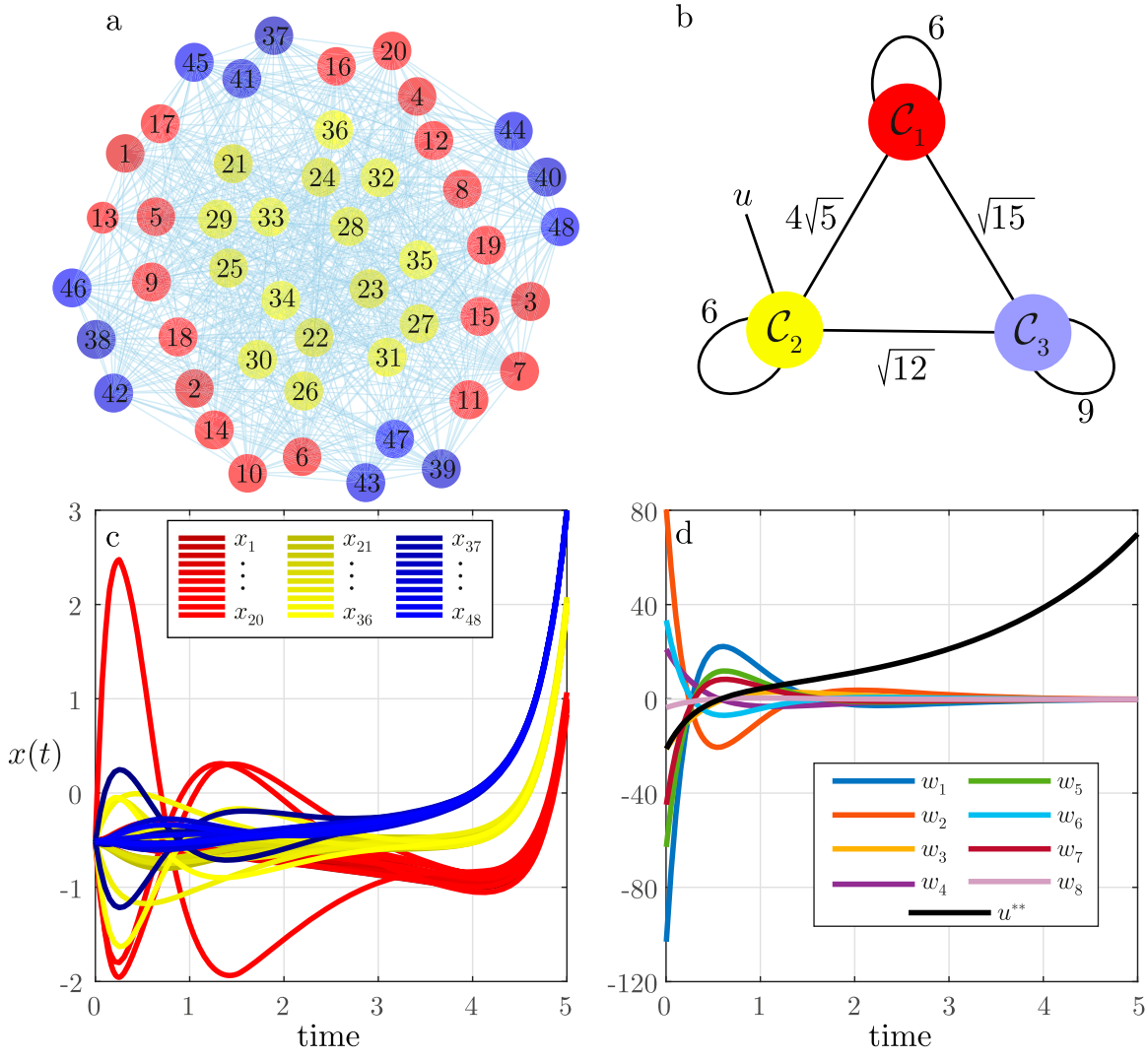

Figure 3. (a) The 48 node random network with 3 orbital cluster and (b) its three node quotient network. (c) Controlled state trajectories of the network nodes driven towards the group consensus state through the joint action of the optimal control input u and of the stabilising action w. (d) Time evolution of the control inputs.

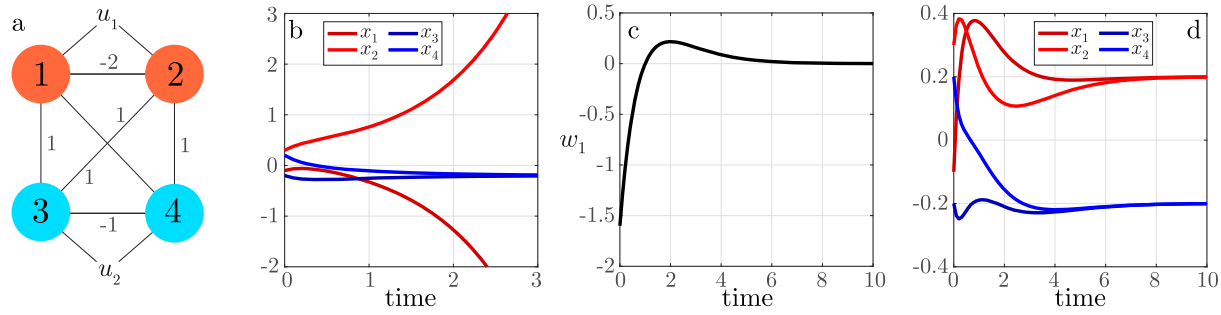

Figure 4. (a) Network topology. (b) free dynamics. (c) control input. (d) controlled dynamics.

and of a stabilising state feedback control action w designed on the pair $(A_\perp, D_\perp)$ which places all the formerly unstable eigenvalues of $A_\perp$ in -10. As can be seen from the figure, group consensus is achieved starting from an initial condition that lies outside $\mathcal{X}_{or}$. Figure 3(d) shows the control inputs u^{**} and $w_i(t)$ $i = 1, \dots, 8$.

In our third and last numerical example, we consider a mathematical paradigm used to model opinion dynamics in real-world social systems in the presence of antagonistic interactions, consensus on signed graphs. In this framework, the dynamics of each node can be described by

$$\dot{x}_i = -\sigma_i x_i + \sum_{j \neq i} a_{ij}(x_j - \text{sign}(a_{ij})x_i) + \sum_{m=1}^{M} b_{im}u_m, \qquad (29)$$

where $\text{sign}(\cdot) = +$ if $\cdot \geq 0$ and $\text{sign}(\cdot) = -$ otherwise. Note that the interpretation of Equation (29) is that any pair of connected agents (i, j) interact cooperatively if $a_{ij} > 0$ and antagonistically otherwise. In the context of opinion dynamics, the (usually constant) exogenous signal u_m can be interpreted as the cognitive bias of each individual with respect to a given topic (Gray et al., 2018) while each element b_{ij} of the matrix B determines which network nodes share the same bias.

Being linear, the dynamics of the N network nodes in Equation (29) can be rewritten as in Equation (1). Note that the coefficients a_{ij}, $j \neq i$ in Equation (29) are the off-diagonal elements of the matrix A, while the weights of the self loops a_{ii} for all i can be computed as

$$a_{ii} = -\sigma_i + \sum_{i \neq j} -\text{sign}(a_{ij})a_{ij}.$$

In our example, we consider the four node graph endowed of non-trivial symmetries described in Figure 4(a) and defined by the pair of matrices

$$A = \begin{bmatrix} -1 & -2 & 1 & 1 \\ -2 & -1 & 1 & 1 \\ 1 & 1 & -2 & -1 \\ 1 & 1 & -1 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}. \qquad (30)$$

The pair (A, B) defines a partition Π of the network nodes in $K = 2$ clusters $\mathcal{C}_1 = \{1, 2\}$ and $\mathcal{C}_2 = \{3, 4\}$. In the field of opinion dynamics, it has been highlighted (Bizyaeva et al., 2020) how symmetries in the network structure model the emergence of clusters of agents sharing the same opinion.

In this simple example, the transformation allowing to describe the network dynamics in the IRR coordinate systems is

$$T = \begin{bmatrix} \sqrt{2}^{-1} & \sqrt{2}^{-1} & 0 & 0 \\ 0 & 0 & \sqrt{2}^{-1} & \sqrt{2}^{-1} \\ \sqrt{2}^{-1} & -\sqrt{2}^{-1} & 0 & 0 \\ 0 & 0 & \sqrt{2}^{-1} & -\sqrt{2}^{-1} \end{bmatrix}$$

yielding

$$A_\parallel = \begin{bmatrix} -3 & 2 \\ 3 & -3 \end{bmatrix}, \quad A_\perp = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

The reader can easily note that while the dynamics along the group consensus subspace is stable, we have that $\Lambda^\perp = \{1\}$. This means that if no additional stabilising control action is applied, then the unstable eigenvalue of the matrix A ensures group consensus is not achieved, i.e. the antagonism between members of the same cluster causes their opinions to diverge, see panel b of Figure 4. Hence, we leverage Algorithm 1 to add a stabilising control action allowing to achieve group consensus. As $\Omega_1 = [1; -1; 0; 0]$, adding a control action in node 1, that is, selecting $D = [1; 0; 0; 0]$ ensures the pair $(A_\perp, D_\perp)$ is stabilisable. Figure 4(c–d) shows the time evolution of the stabilising signal w and the resulting convergence of the controlled trajectories to the group consensus subspace.

6. Conclusions

Motivated by the observation that symmetries induce both loss of controllability and the emergence of group consensus, in this work we studied the controllability properties of networks endowed of symmetries. We found that controllability is lost in directions orthogonal to the group consensus subspace, but it is still possible to control the consensus state either if the network initial condition belongs to the group consensus subspace, or if the subsystem of the dynamics orthogonal to this subspace is asymptotically stable. Moreover, we showed that when the network controllable subspace coincides with the group consensus subspace, we can control consensus by designing control strategies on a lower-dimensional network, the quotient network, thus reducing the computational burden. We also considered the issue of stabilisability of the network dynamics and provided a simple algorithm to place additional control inputs that ensure that the group consensus subspace is stabilisable. By using the

IRR transformation of the network symmetry group, we provided bounds on the minimum number of additional inputs and on the number of driver nodes that are needed to achieve stabilisability. We demonstrated our theoretical analysis through two representative numerical examples and then introduced a third example to show how are results can be applied to the field of opinion dynamics. We envision that our results could be useful towards the design of symmetric network topologies in engineering applications for which the emergence of group consensus is desired.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Francesco Lo Iudice http://orcid.org/0000-0002-7155-2756

References

Acemoglu, D., & Ozdaglar, A. (2011). Opinion dynamics and learning in social networks. *Dynamic Games and Applications*, *1*(1), 3–49. https://doi.org/10.1007/s13235-010-0004-1

Aguilar, C. O., & Gharesifard, B. (2017). Almost equitable partitions and new necessary conditions for network controllability. *Automatica*, *80*(6), 25–31. https://doi.org/10.1016/j.automatica.2017.01.018

Alemzadeh, S., de Badyn, M. H., & Mesbahi, M. (2017). Controllability and stabilizability analysis of signed consensus networks. In *2017 IEEE Conference on Control Technology and Applications (CCTA)* (pp. 55–60). IEEE.

Altafini, C., & Lini, G. (2014). Predictable dynamics of opinion forming for networks with antagonistic interactions. *IEEE Transactions on Automatic Control*, *60*(2), 342–357. https://doi.org/10.1109/TAC.9

Bizyaeva, A., Franci, A., & Leonard, N. E. (2020). A general model of opinion dynamics with tunable sensitivity. arXiv preprint arXiv:2009.04332.

Blaha, K. A., Huang, K., Della Rossa, F., Pecora, L., Hossein-Zadeh, M., & Sorrentino, F. (2019). Cluster synchronization in multilayer networks: A fully analog experiment with L C oscillators with physically dissimilar coupling. *Physical Review Letters*, *122*(1), 014101. https://doi.org/10.1103/PhysRevLett.122.014101

Chapman, A., & Mesbahi, M. (2014). On symmetry and controllability of multi-agent systems. In *53rd IEEE Conference on Decision and Control* (pp. 625–630). IEEE.

Chapman, A., & Mesbahi, M. (2015). State controllability, output controllability and stabilizability of networks: A symmetry perspective. In *2015 54th IEEE Conference on Decision and Control (CDC)* (pp. 4776–4781). IEEE.

De Lellis, P., Di Meglio, A., & Lo Iudice, F. (2018). Overconfident agents and evolving financial networks. *Nonlinear Dynamics*, *92*(1), 33–40. https://doi.org/10.1007/s11071-017-3780-y

Della Rossa, F., Pecora, L., Blaha, K., Shirin, A., Klickstein, I., & Sorrentino, F. (2020). Symmetries and cluster synchronization in multilayer networks. *Nature Communications*, *11*(1), 1–17. https://doi.org/10.1038/s41467-019-13993-7

Galam, S. (2004). Contrarian deterministic effects on opinion dynamics:the hung elections scenario. *Physica A: Statistical Mechanics and Its Applications*, *333*, 453–460. https://doi.org/10.1016/j.physa.2003.10.041

Gambuzza, L. V., & Frasca, M. (2019). A criterion for stability of cluster synchronization in networks with external equitable partitions. *Automatica*, *100*, 212–218. https://doi.org/10.1016/j.automatica.2018.11.026

Gatto, M., Bertuzzo, E., Mari, L., Miccoli, S., Carraro, L., Casagrandi, R., & Rinaldo, A. (2020). Spread and dynamics of the COVID-19 epidemic in Italy: Effects of emergency containment measures. *Proceedings of the National Academy of Sciences*, *117*(9), 10484–10491. https://doi.org/10.1073/pnas.2004978117

Godsil, C. D. (1997). Compact graphs and equitable partitions. *Linear Algebra and Its Applications*, *255*(1–3), 259–266. https://doi.org/10.1016/S0024-3795(97)83595-1

Gray, R., Franci, A., Srivastava, V., & Leonard, N. E. (2018). Multiagent decision-making dynamics inspired by honeybees. *IEEE Transactions on Control of Network Systems*, *5*(2), 793–806. https://doi.org/10.1109/TCNS.2018.2796301

Hautus, M. (1970). Stabilization controllability and observability of linear autonomous systems. In *Indagationes Mathematicae (Proceedings)* (Vol. 73, pp. 448–455). North-Holland.

Hołyst, J. A., Kacperski, K., & Schweitzer, F. (2001). Social impact models of opinion dynamics. In *Annual Reviews of Computational Physicsix* (pp. 253–273). World Scientific.

Klickstein, I., Pecora, L., & Sorrentino, F. (2019). Symmetry induced group consensus. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *29*(7), 073101. https://doi.org/10.1063/1.5098335

Klickstein, I., & Sorrentino, F. (2018). Generating symmetric graphs. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *28*(12), 121102. https://doi.org/10.1063/1.5064375

Lin, C. T. (1974). Structural controllability. *IEEE Transactions on Automatic Control*, *19*(3), 201–208. https://doi.org/10.1109/TAC.1974.1100557

Liu, Y. Y., Slotine, J. J., & Barabási, A. L. (2011). Controllability of complex networks. *Nature*, *473*(7346), 167–173. https://doi.org/10.1038/nature10011

Lo Iudice, F., Garofalo, F., & Sorrentino, F. (2015). Structural permeability of complex networks to control signals. *Nature Communications*, *6*(1), 1–6. https://doi.org/10.1038/ncomms9349

Lo Iudice, F., Sorrentino, F., & Garofalo, F. (2019). On node controllability and observability in complex dynamical networks. *IEEE Control Systems Letters*, *3*(4), 847–852. https://doi.org/10.1109/LCSYS.7782633

MacArthur, B. D., Sánchez-García, R. J., & Anderson, J. W. (2008). Symmetry in complex networks. *Discrete Applied Mathematics*, *156*(18), 3525–3531. https://doi.org/10.1016/j.dam.2008.04.008

Pasqualetti, F., Zampieri, S., & Bullo, F. (2014). Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, *1*(1), 40–52. https://doi.org/10.1109/TCNS.2014.2310254

Pecora, L., Sorrentino, F., Hagerstrom, A., Murphy, T. E., & Roy, R. (2014). Cluster synchronization and isolated desynchronization in complex networks with symmetries. *Nature Communications*, *5*(1), 1–8. https://doi.org/10.1038/ncomms5079

Proskurnikov, A. V., Matveev, A. S., & Cao, M. (2015). Opinion dynamics in social networks with hostile camps: Consensus vs. polarization. *IEEE Transactions on Automatic Control*, *61*(6), 1524–1536. https://doi.org/10.1109/TAC.2015.2471655

Qin, J., Ma, Q., Gao, H., Shi, Y., & Kang, Y. (2016). On group synchronization for interacting clusters of heterogeneous systems. *IEEE Transactions on Cybernetics*, *47*(12), 4122–4133. https://doi.org/10.1109/TCYB.2016.2600753

Qin, J., & Yu, C. (2013). Group consensus of multiple integrator agents under general topology. In *52nd IEEE Conference on Decision and Control* (pp. 2752–2757).

Sethi, A., Eargle, J., Black, A. A., & Luthey-Schulten, Z. (2009). Dynamical networks in tRNA: Protein complexes. *Proceedings of the National Academy of Sciences*, *106*(16), 6620–6625. https://doi.org/10.1073/pnas.0810961106

Sorrentino, F., Pecora, L., & Trajković, L. (2020). Group consensus in multilayer networks. *IEEE Transactions on Network Science and Engineering*, *7*(3), 2016–2026. https://doi.org/10.1109/TNSE.6488902

Stegink, T., De Persis, C., & van der Schaft, A. (2016). A unifying energy-based approach to stability of power grids with market dynamics. *IEEE Transactions on Automatic Control*, *62*(6), 2612–2622. https://doi.org/10.1109/TAC.2016.2613901

Van Vreeswijk, C., & Sompolinsky, H. (1996). Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science (New York, N.Y.)*, *274*(5293), 1724–1726. https://doi.org/10.1126/science.274.5293.1724

Yu, J., & Wang, L. (2009). Group consensus of multi-agent systems with undirected communication graphs. In *2009 7th Asian Control Conference* (pp. 105–110). IEEE.

Yu, J., & Wang, L. (2010). Group consensus in multi-agent systems with switching topologies and communication delays. *Systems & Control Letters*, *59*(6), 340–348. https://doi.org/10.1016/j.sysconle.2010.03.009

Yu, J., & Wang, L. (2012). Group consensus of multi-agent systems with directed information exchange. *International Journal of Systems Science*, *43*(2), 334–348. https://doi.org/10.1080/00207721.2010.496056

Yu, T., Zhou, B., Chan, K., Yuan, Y., Yang, B., & Wu, Q. (2012). R (λ) imitation learning for automatic generation control of interconnected power grids R (λ) imitation learning for automatic generation control of interconnected power grids. *Automatica*, *48*(9), 2130–2136. https://doi.org/10.1016/j.automatica.2012.05.043

Yuan, Z., Zhao, C., Di, Z., Wang, W. X., & Lai, Y. C. (2013). Exact controllability of complex networks. *Nature Communications*, *4*(1), 1–9. https://doi.org/10.1038/ncomms3447

Appendix. Stabilising the cluster consesnsus on the example in Figure 3

The adjacency matrix of the proposed network is

$$A = \begin{bmatrix}
0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&1&0&1\\
1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&0&1&0\\
1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&0&1\\
1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0\\
0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1\\
0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0\\
0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1\\
0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0\\
0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&1&0&1&0&1\\
0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&0&0&1&0&1&0\\
0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&1&0&1&0&1\\
0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&0&0&1&0&1&0\\
0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&1&0&1&0&1\\
0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&0&0&1&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&1&0&1&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&1&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&1&0&1&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&0&1&0&1&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&0&1&0&1\\
1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&1&0&1&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&1&1&1&0\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&1&0&1&1&1\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&1&0&1&1\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&1&1&1&0&1\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&1&1&1&0\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&1&1&1\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&1&1\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&1\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0&0\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0&0\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0&0\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&1&0&0&0&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&1&0&0&0\\
0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&1&1&1&0&0\\
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1\\
0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0\\
0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0\\
0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0\\
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1\\
0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0\\
0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0\\
0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0
\end{bmatrix}$$

$$\left.\begin{matrix}
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0
\end{matrix}\right],$$

$$B^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \end{bmatrix}.$$

Using the algorithm in Pecora et al. (2014), we compute the transformation to the IRR coordinate system of this network, that is

$100\ T_{or} =$

$$
\left[\begin{array}{cccccccccccccccccccccccccccccccccc}
22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 22 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 25 & 25 & 25 & 25 & 25 & 25 & 25 & 25 & 25 & 25 & 25 & 25 & 25 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 22 & -22 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -25 & 25 & -25 & 25 & -25 & 25 & -25 & 25 & -25 & 25 & -25 & 25 & -25 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 35 & 0 & -35 & 0 & 35 & 0 & -35 & 0 & 35 & 0 & -35 & 0 & 35 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & -32 & 0 & 32 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -35 & 0 & 35 & 0 & -35 & 0 & 35 & 0 & -35 & 0 & 35 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
7 & -27 & -24 & 13 & 31 & 7 & -27 & -24 & 13 & 31 & 7 & -27 & -24 & 13 & 31 & 7 & -27 & -24 & 13 & 31 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
31 & 16 & -21 & -29 & 3 & 31 & 16 & -21 & -29 & 3 & 31 & 16 & -21 & -29 & 3 & 31 & 16 & -21 & -29 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-29 & 24 & -17 & 8 & 2 & -12 & 20 & -27 & 31 & -32 & 29 & -24 & 17 & -8 & -2 & 12 & -20 & 27 & -31 & 32 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-12 & 20 & -27 & 31 & -32 & 29 & -24 & 17 & -8 & -2 & 12 & -20 & 27 & -31 & 32 & -29 & 24 & -17 & 8 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 28 & 28 & 5 & -23 & -31 & -14 & 15 & 31 & 22 & -5 & -28 & -28 & -5 & 23 & 31 & 14 & -15 & -31 & -22 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-31 & -14 & 15 & 31 & 22 & -5 & -28 & -28 & -5 & 23 & 31 & 14 & -15 & -31 & -22 & 5 & 28 & 28 & 5 & -23 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-5 & -28 & 22 & 14 & -31 & 5 & 28 & -22 & -14 & 31 & -5 & -28 & 22 & 14 & -31 & 5 & 28 & -22 & -14 & 31 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
31 & -14 & -22 & 28 & 5 & -31 & 14 & 22 & -28 & -5 & 31 & -14 & -22 & 28 & 5 & -31 & 14 & 22 & -28 & -5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -19 & 30 & -30 & 18 & 1 & -19 & 30 & -30 & 18 & 1 & -19 & 30 & -30 & 18 & 1 & -19 & 30 & -30 & 18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-32 & 25 & -9 & -10 & 26 & -32 & 25 & -9 & -10 & 26 & -32 & 25 & -9 & -10 & 26 & -32 & 25 & -9 & -10 & 26 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
29 & -27 & 3 & 24 & -31 & 12 & 16 & -31 & 21 & 7 & -29 & 27 & -3 & -24 & 31 & -12 & -16 & 31 & -21 & -7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-12 & -16 & 31 & -21 & -7 & 29 & -27 & 3 & 24 & -31 & 12 & 16 & -31 & 21 & 7 & -29 & 27 & -3 & -24 & 31 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-28 & -31 & -22 & -5 & 15 & 28 & 31 & 22 & 5 & -15 & -28 & -31 & -22 & -5 & 15 & 28 & 31 & 22 & 5 & -15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-14 & 5 & 23 & 31 & 28 & 14 & -5 & -23 & -31 & -28 & -14 & 5 & 23 & 31 & 28 & 14 & -5 & -23 & -31 & -28 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
32 & 30 & 25 & 17 & 8 & -2 & -11 & -20 & -26 & -31 & -32 & -30 & -25 & -17 & -8 & 2 & 11 & 20 & 26 & 31 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-2 & -11 & -20 & -26 & -31 & -32 & -30 & -25 & -17 & -8 & 2 & 11 & 20 & 26 & 31 & 32 & 30 & 25 & 17 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -29 & 7 & 35 & 20 & -20 & -35 & -7 & 29 & 29 & -7 & -35 & -20 & 20 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -20 & -35 & -7 & 29 & 29 & -7 & -35 & -20 & 20 & 35 & 7 & -29 & -29 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 35 & -29 & 19 & -7 & -7 & 20 & -30 & 35 & -35 & 29 & -19 & 7 & 7 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7 & 20 & -30 & 35 & -35 & 29 & -19 & 7 & 7 & -20 & 30 & -35 & 35 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9 & -28 & 31 & 5 & -34 & 22 & 18 & -35 & 9 & 28 & -31 & -5 & 34 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 34 & -22 & -18 & 35 & -9 & -28 & 31 & 5 & -34 & 22 & 18 & -35 & 9 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -22 & -10 & 3 & 17 & 27 & 34 & 35 & 31 & 22 & 10 & -3 & -17 & -27 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 27 & 34 & 35 & 31 & 22 & 10 & -3 & -17 & -27 & -34 & -35 & -31 & -22 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -30 & 11 & -18 & 34 & 30 & -11 & 18 & -34 & -30 & 11 & -18 & 34 & 30 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 30 & -11 & -37 & 11 & -30 & 11 & 37 & -11 & 30 & -11 & -37 & 11 & -30 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -16 & 16 & -28 & -35 & 16 & -16 & 28 & 35 & -16 & 16 & -28 & -35 & 16 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 21 & 45 & 5 & 6 & -21 & -45 & -5 & -6 & 21 & 45 & 5 & 6 & -21 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right.
$$

$$
\left.\begin{array}{ccccccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
25 & 25 & 25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 29 & 29 & 29 & 29 & 29 & 29 & 29 & 29 & 29 & 29 & 29 & 29 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
25 & -25 & 25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -29 & 29 & -29 & 29 & -29 & 29 & -29 & 29 & -29 & 29 & -29 & 29 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -41 & 0 & 41 & 0 & -41 & 0 & 41 & 0 & -41 & 0 & 41 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-35 & 0 & 35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 41 & 0 & -41 & 0 & 41 & 0 & -41 & 0 & 41 & 0 & -41 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
35 & 7 & -29 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
7 & 35 & 20 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-20 & 30 & -35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-29 & 19 & -7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-22 & -18 & 35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
28 & -31 & -5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-34 & -35 & -31 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-10 & 3 & 17 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-11 & 18 & -34 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
11 & 37 & -11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-16 & 28 & 35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-45 & -5 & -6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 35 & -35 & 0 & 35 & -35 & 0 & 35 & -35 & 0 & 35 & -35 & 0 \\
0 & 0 & 0 & -20 & -20 & 41 & -20 & -20 & 41 & -20 & -20 & 41 & -20 & -20 & 41 \\
0 & 0 & 0 & -35 & 20 & 0 & -20 & 35 & -41 & 35 & -20 & 0 & 20 & -35 & 41 \\
0 & 0 & 0 & -20 & 35 & -41 & 35 & -20 & 0 & 20 & -35 & 41 & -35 & 20 & 0 \\
0 & 0 & 0 & -38 & -32 & 6 & 38 & 32 & -6 & -38 & -32 & 6 & 38 & 32 & -6 \\
0 & 0 & 0 & -15 & 26 & 40 & 15 & -26 & -40 & -15 & 26 & 40 & 15 & -26 & -40 \\
0 & 0 & 0 & -35 & -20 & 0 & 20 & 35 & 41 & 35 & 20 & 0 & -20 & -35 & -41 \\
0 & 0 & 0 & 20 & 35 & 41 & 35 & 20 & 0 & -20 & -35 & -41 & -35 & -20 & 0
\end{array}\right].
$$

Note that each row of the transformation is cluster specific, that is, each row has non-zero entries in the elements corresponding to only one of the clusters. Applying this transformation to our example, we obtain

$$
\tilde{A} = \begin{bmatrix} A_{\parallel} & 0 & 0 & 0 & 0 \\ 0 & A_{\perp}^{1} & 0 & 0 & 0 \\ 0 & 0 & A_{\perp}^{2} & 0 & 0 \\ 0 & 0 & 0 & A_{\perp}^{3} & 0 \\ 0 & 0 & 0 & 0 & A_{\perp}^{4,\ldots,39} \end{bmatrix},
$$

$$
A_{\parallel} = \begin{bmatrix} 6 & 4\sqrt{5} & \sqrt{15} \\ 4\sqrt{5} & 6 & \sqrt{12} \\ \sqrt{15} & \sqrt{12} & 9 \end{bmatrix}, \quad A_{\perp}^{1} = \begin{bmatrix} -2 & -9 & -3.9 \\ -9 & -2 & 3.5 \\ -3.9 & 3.5 & 1 \end{bmatrix},
$$

$$
A_{\perp}^{2} = A_{\perp}^{3} = \begin{bmatrix} -2 & 0 & -3.9 \\ 0 & -2 & -3.5 \\ -3.9 & -3.5 & -1 \end{bmatrix}
$$

$$
A_{\perp}^{4,\ldots,19} = \mathrm{diag}(-2.6, -2.6, -1.5, -1.5, -1.3, -1.3, -0.6, -0.6, -0.4, -0.4, 0.1, 0.1, 1.6, 1.6, 4.7, 4.7)
$$

$$
A_{\perp}^{20,\ldots,31} = \mathrm{diag}(-2.5, -2.5, -1.2, -1.2, -0.3, -0.3, 4, 4, 0, 0, 0, 0)
$$

$$
A_{\perp}^{32,\ldots,39} = \mathrm{diag}(0, 0, -2.7, -2.7, -2, -2, 0.7, 0.7)
$$

where we have highlighted the block structure of the matrix $\tilde{A}$. Note that the first three rows of T_{or} span the cluster consensus subspace $\mathcal{X}_{or}$. Then we have three sets of three rows of the so called *intertwined* symmetry-breaks (Pecora et al., 2014), that define three 3×3 blocks $A^1_\perp, \ldots, A^3_\perp$ of $\tilde{A}$ each governing the dynamics along an A-invariant subspace. Any one of these blocks is generated by three rows of the matrix T_{or} each specific of a different cluster. The eigenvectors of the matrix A generating these three dimensional invariant subspaces have therefore non-zero entries in all their elements (since they involve all the three clusters/all the nodes of the network). The remaining 36 rows of T_{or} define 36 monodimensional blocks of $\tilde{A}$ ($A^4_\perp, \ldots, A^{39}_\perp$), and are therefore themselves eigenvectors of the matrix A. The first 16 are specific of cluster $\mathcal{C}_1$, the next 12 are specific of cluster $\mathcal{C}_2$, and finally the last 8 are specific of cluster $\mathcal{C}_3$.

The transverse non-stable eigenvalues that define $\Lambda^\perp$ are the 16 non-negative monodimensional block of $\tilde{A}$, together with three other positive eigenvalues, one for each fo the 3x3 blocks of $A_\perp$. As a result

$$\Lambda^\perp = \Big[\ \underbrace{9.9}\quad \underbrace{3.7}\quad \underbrace{0.1}\quad \underbrace{1.6}\quad \underbrace{4.7}\quad \underbrace{4}\quad \underbrace{0}\quad \underbrace{0.7}\ \Big]$$

$$\left[\begin{array}{cccccccccccccccccc}
1 & 0 & 2 & 3 & -1 & -1 & 3 & -1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & 2 & -2 & 1 & 3 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -2 & 1 & -3 & 2 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 2 & 0 & 0 & -3 & 3 & 0 & -2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 2 & -1 & -3 & 3 & -2 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & -1 & -3 & 1 & -3 & 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -2 & -2 & -3 & -1 & -3 & -3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 2 & 0 & -3 & -2 & -2 & -2 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 2 & -3 & -1 & -3 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & -3 & 0 & -3 & 2 & -3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -2 & -3 & 1 & -1 & 3 & 1 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 2 & 0 & -2 & 2 & 1 & 3 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 2 & -1 & 3 & 2 & 2 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & 0 & 3 & 3 & 0 & 2 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -2 & 1 & 3 & 3 & -2 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 2 & 0 & 1 & 3 & 1 & -3 & -3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 2 & 2 & 3 & -1 & -3 & 3 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & 3 & 2 & -2 & -2 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -2 & 3 & 1 & -3 & 0 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 2 & 0 & 3 & 0 & -3 & 2 & 3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -3 & -3 & 3 & -2 & 2 & 0 & 0 & 0 & 0 \\
-2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & -2 & 1 & -1 & 2 & 4 & 0 & 0 & 0 & 0 \\
2 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & -1 & -2 & -4 & -3 & 1 & 0 & 0 & 0 & 0 \\
-2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 3 & 1 & -3 & 1 & 0 & 0 & 0 & 0 \\
2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 1 & 3 & -3 & 2 & -2 & 0 & 0 & 0 & 0 \\
-2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 3 & -1 & 1 & -2 & -4 & 0 & 0 & 0 & 0 \\
2 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 3 & 2 & 4 & 3 & -1 & 0 & 0 & 0 & 0 \\
-2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -3 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\
2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 & -3 & 3 & -2 & 2 & 0 & 0 & 0 & 0 \\
-2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & 2 & 1 & -1 & 2 & 4 & 0 & 0 & 0 & 0 \\
2 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & 1 & -2 & -4 & -3 & 1 & 0 & 0 & 0 & 0 \\
-2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & 0 & 3 & 1 & -3 & 1 & 0 & 0 & 0 & 0 \\
2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & 3 & -3 & 2 & -2 & 0 & 0 & 0 & 0 \\
-2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -3 & -1 & 1 & -2 & -4 & 0 & 0 & 0 & 0 \\
2 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -3 & 2 & 4 & 3 & -1 & 0 & 0 & 0 & 0 \\
-2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & -3 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\
\hline
1 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -2 & -3 & 3 \\
-1 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & -2 & -1 & 4 \\
1 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 1 & 4 \\
-1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -2 & 3 & 3 \\
1 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & -2 & 4 & 1 \\
-1 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 4 & -1 \\
1 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -2 & 3 & -3 \\
-1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & -2 & 1 & -4 \\
1 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -1 & -4 \\
-1 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -2 & -3 & -3 \\
1 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & -2 & -4 & -1 \\
-1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -4 & 1
\end{array}\right]$$

where the brackets associate each $\lambda_i \in \Lambda^\perp$ to the eigenvectors obtained according to Remark 4.4 and spanning Ω_i.

We are now ready to apply Algorithm 1 to find the driver nodes needed to stabilise $\mathcal{X}_{or}$.

- when $i = 1$ we consider the eigenspace of the eigenvalue 9.9. Its dimension is 1, so we need at least one control input and two driver nodes to stabilise it. We select nodes 1 and 2 as drivers and thus $D_{1,1} = 1$, $D_{1,2} = -1$, $D_{1,j} = 0$, $j = 3, \ldots, 48$.
- when $i = 2$ we consider the eigenspace of the eigenvalue 3.7. This eigenspace is two-dimensional, so we need another control input and another driver node to stabilise it. We then add a second (independent) column to the matrix D with $D_{2,1} = 1$, $D_{2,4} = -1$, $D_{2,j} = 0$, $j = 2, 3, 5, \ldots, 48$. We then verify that D has now two columns with non-zero and non-parallel projection on the eigenspace associated to the eigenvalue 3.7 by computing the elements

$$\mathcal{D}^2_{1,1} = [1,-1,0][0,-2,2]^T = 2, \quad \mathcal{D}^2_{1,2} = [1,-1,0][2,0,0]^T = 2,$$

$$\mathcal{D}^2_{2,1} = [1,0,-1][0,-2,2]^T = -2, \quad \mathcal{D}^2_{2,2} = [1,0,-1][2,0,0]^T = 2,$$

of the matrix $\mathcal{D}^2$ and then verifying that this matrix is full rank as $\det(\mathcal{D}^2) = 8 \neq 0$.

- when $i = 3$ we consider the eigenspace associated to the eigenvalue 0.1. Its dimension is 2, and the vectors in D have a two dimensional projection on it as the elements

$$\mathcal{D}^3_{1,1} = [1,-1,0][3,2,0]^T = 1, \quad \mathcal{D}^3_{1,2} = [1,-1,0][-1,-2,-3]^T = 1,$$

$$\mathcal{D}^3_{2,1} = [1,0,-1][3,2,0]^T = 3, \quad \mathcal{D}^3_{2,2} = [1,0,-1][-1,-2,-3]^T = 2,$$

define the matrix $\mathcal{D}^3$ that is is full rank as $\det(\mathcal{D}^3) = -1 \neq 0$.
- when $i = 4$ we consider the eigenspace associated to the eigenvalue 1.1. It's dimension is 2, and the vectors in D have a two dimensional projection on it as the elements

$$\mathcal{D}^4_{1,1} = [1,-1,0][-1,1,3]^T = -2, \quad \mathcal{D}^4_{1,2} = [1,-1,0][3,3,0]^T = 0,$$

$$\mathcal{D}^4_{2,1} = [1,0,-1][-1,1,3]^T = -4, \quad \mathcal{D}^4_{2,2} = [1,0,-1][3,3,0]^T = 3,$$

define the matrix $\mathcal{D}^4$ that is is full rank as $\det(\mathcal{D}^4) = -6 \neq 0$.
- when $i = 5$ we consider the eigenspace associated to the eigenvalue 4.7. It's dimension is 2, and the vectors in D have a two dimensional projection on it as the elements

$$\mathcal{D}^5_{1,1} = [1,-1,0][-1,-1,-2]^T = 0, \quad \mathcal{D}^5_{1,2} = [1,-1,0][3,-2,3]^T = 5,$$

$$\mathcal{D}^5_{2,1} = [1,0,-1][-1,-1,-2]^T = 1, \quad \mathcal{D}^5_{2,2} = [1,0,-1][3,-2,3]^T = 0,$$

define the matrix $\mathcal{D}^5$ that is is full rank being $\det(\mathcal{D}^5) = -5 \neq 0$.
- the eigenspaces Ω_i when $i \geq 6$ have 0 components on cluster $\mathcal{C}_1$. As a consequence, we need to select additional drivers from the other clusters in order to stabilise them. In particular, for $i = 6$ we consider the eigenspace associated to the eigenvalue 4. It's dimension is 2, and so we need at least 3 driver nodes in the cluster $\mathcal{C}_2$ in order to have a two dimensional projection on it. We then select $D_{3,21} = 1,\ D_{3,22} = -1,\ D_{3,j} = 0\ j = 1,\dots,20,23,\dots,48$ and $D_{4,21} = 1,\ D_{4,23} = -1,\ D_{4,j} = 0,\ j = 1,\dots,20,22,24,\dots,48$. This achieves our goal as the elements

$$\mathcal{D}^6_{1,1} = [1,-1,0][1,3,3]^T = -2, \quad \mathcal{D}^6_{1,2} = [1,-1,0][-3,-2,-1]^T = 1,$$

$$\mathcal{D}^6_{2,1} = [1,0,-1][1,3,3]^T = -2, \quad \mathcal{D}^6_{2,2} = [1,0,-1][-3,-2,-1]^T = 2,$$

define the matrix $\mathcal{D}^6$ that is is full rank being $\det(\mathcal{D}^6) = -2 \neq 0$.
- when $i = 7$ we consider the eigenspace associated to the eigenvalue 0. It's dimension is 6, but we can treat separately the first 4 eigenvectors, associated to cluster $\mathcal{C}_2$ and thus spanning Ω^2_7, from the other 2 eigenvectors, associated to cluster $\mathcal{C}_3$ and thus spanning Ω^2_7. As $|\Omega^2_7| = 4$, we need to select two additional driver nodes for the matrix D to have four columns with non-zero and non-parallel projection on it. We therefore select nodes 24 and 25 as drivers by adding to D the columns $D_{5,21} = 1,\ D_{5,24} = -1,\ D_{5,j} = 0\ j = 1,\dots,20,22,23,25,\dots,48$ and $D_{6,21} = 1,\ D_{6,25} = -1, D_{6,j} = 0\ j = 1,\dots,20,22,23,24,26,\dots,48$. As the matrix

$$\mathcal{D}^7 = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} -3 & 3 & -2 & 2 \\ 1 & -1 & 2 & 4 \\ -2 & -4 & -3 & 1 \\ 3 & 1 & -3 & 1 \\ 3 & -3 & 2 & -2 \end{bmatrix}$$

$$= \begin{bmatrix} -4 & 4 & -4 & -2 \\ -1 & 7 & 1 & 1 \\ -6 & 2 & 1 & 1 \\ -6 & 6 & -4 & 4 \end{bmatrix}$$

is full rank, then the matrix D has now four columns with non-zero and non-parallel projection on $|\Omega^2_7| = 4$.

Then, we turn our attention to Ω^3_7 noting that $|\Omega^3_7| = 2$. We therefore need to select three additional drivers defined by $D_{7,37} = 1,\ D_{3,38} = -1,\ D_{7,j} = 0,\ j = 1,\dots,36,29,\dots,48$ and $D_{8,37} = 1,\ D_{8,39} = -1, D_{8,j} = 0\ j = 1,\dots,36,38,40,\dots,48$. As the matrix

$$\mathcal{D}'^7 = \begin{bmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 4 & -2 & \\ -4 & -20 & 4 \end{bmatrix} = \begin{bmatrix} 8 & 0 \\ 4 & -6 \end{bmatrix}$$

is full rank being $\det(\mathcal{D}'^7) = -48 \neq 0$, then the matrix D has now two columns with non-zero and non-parallel projection on Ω^3_7.
- Our procedure ends with iteration $i = 8$ in which we consider the eigenspace associated to the eigenvalue 0.7. Note that the matrix D already has two columns with non-zero and non-parallel projection on it, namely D_7 and D_8 as the matrix

$$\det(\mathcal{D}^8) = \det\left(\begin{bmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} -3 & 3 \\ -1 & 4 \\ 1 & 4 \end{bmatrix} \right)$$

$$= \det\left(\begin{bmatrix} -2 & -1 \\ -4 & -1 \end{bmatrix} \right) = -2 \neq 0$$

is full rank. Note that our selection achieved bound on the number of driver nodes given in Corollary 4.4, but not the minimum number of inputs (that are 6, applying Corollary 4.2). This last achievement can easily be obtained replacing D_1 and D_2 with $D_1 + D_7$ and $D_2 + D_8$, and then removing D_7 and D_8 from D.