Article

ON THE ROLE OF MATRIX-WEIGHTS ELEMENTS IN CONSENSUS ALGORITHMS FOR MULTI-AGENT SYSTEMS

Joshua Ogbebor [1,‡] **and Xiangyu Meng** [1,‡,*]

[1] Division of Electrical and Computer Engineering, School of Electrical Engineering and Computer Science, Louisiana State University; jogbeb1@lsu.edu; xmeng5@lsu.edu
[*] Correspondence:xmeng5@lsu.edu
‡ These authors contributed equally to this work.

Abstract: This paper extends the concept of weighted graphs to matrix weighted graphs. The consensus algorithms dictate that all agents reach consensus when the weighted graph is connected. However, it is not always the case for matrix weighted graphs. The conditions leading to different types of consensus have been extensively analysed based on the properties of matrix-weighted Laplacians and graph theoretic methods. However, in practice, there is concern on how to pick matrix-weights to achieve some desired consensus, or how the change of elements in matrix weights affects the consensus algorithm. By selecting the elements in the matrix weights, different clusters may be possible. In this paper, we map the roles of the elements of the matrix weights in the systems consensus algorithm. We explore the choice of matrix weights to achieve different types of consensus and clustering. Our results are demonstrated on a network of three agents where each agent has three states.

Keywords: matrix-weighted graphs; multi-agent systems; clustered consensus; global consensus

1. Introduction

Multi-agent systems (MASs) consists of multiple autonomous agents[1], which can be used to solve problems that are difficult or impossible for an individual agent or a monolithic system. Communication and interaction between individual agents are fundamental characteristics of MASs, which allow agents to achieve a global objective despite of having access to only local neighbourhood information [2], or edge information [3,4]. Having multiple agents could speed up a system's operation by providing a method for parallel computation, offering advantages in terms of extensibility and flexibility compared to single agent systems [5]. Applications of MASs can be found in cloud computing and social networks [6].

Consensus is one of the fundamental problems in multi-agent coordination, where agents interact with their neighbors according to a local protocol to ensure that a common value in terms of the state components, is agreed upon globally, by all the agents. This sweeping form of consensus is generally referred to as a global consensus. A clustered consensus is also possible, where some agents agree on some values, different on the consensus value of some other agents [7]. MAS is widely studied using the graph theory, in which the vertices and edges represent agents and the inter-agent links, respectively. Conventionally, the inter-agent links have been modelled by scalar weights [8,9]. The consensus algorithm is widely applied in the agent level [10–12]. Matrix-weights can be used to capture the complexity in the state level for MASs. Particularly, the consensus of all agents in corresponding states may be affected by other states. There are a number of works in the literature that extend the conventional scalar weighted graph to the matrix weighted graph for the consensus problem. One of such is [13,14] where the conditions for achieving consensus in matrix-weighted Discrete and continuous-time consensus algorithms are presented using the properties of the graph Laplacian. In [14], discrete-time matrix-weighted consensus is studied over undirected and connected graphs considering symmetric matrix weights and a special case of non-symmetric matrix-weights that can achieve consensus control. Lyapunov stability theory for discrete-time systems is also employed to show the system's convergence to consensus. In the same way, in [13], matrix-weighted consensus algorithm is studied with fixed undirected graphs and a necessary and sufficient condition for exponentially reaching a global average consensus and clustered consensus is presented based on the null-space of the matrix-weighted Laplacian. In particular, a global average consensus can be achieved if and only if the nullspace of the matrix-weighted Laplacian is spanned by the consensus space. Furthermore, an iterative algorithm to determine clusters in the network is provided, based on the concept of positive trees, which is determined using the positive definiteness of the matrix weights along a path. Two clusters are merged together if they satisfy some algebraic conditions on their connections. This merging is carried on iteratively, gradually reducing the number of clusters in the graph until no two clusters can be further merged.

When the agent-to-agent link is weighted by positive semi-definite (PSD) matrices, it can be difficult to tell what form of consensus will be present. Clustered consensus can happen even when the graph is connected in the agent level, due to the existence of PSD matrix weights. In a case, when all the elements of each of the matrix-weights are set to one, for a complete graph on five vertices, there will be no form of consensus for the agents in any of the states.

In this paper, we study how the choice of matrix weights for the inter-agent links affect the presence of clusters in a matrix-weighted graph. A linear algebraic approach is used in [15,16] to examine the properties of the Laplacian matrix and the necessary and sufficient conditions are given for the existence of consensus. We adopt a matrix-element-mapping approach to study how the matrix-weights alter the consensus dynamics. Our approach contrast the matrix-based approach to the conventional scalar weighted graphs which helps view the network of each state, and offers hints on the choice of the matrix weights.

2. Problem Formulation

2.1. Agent Dynamics

The dynamics of agent v_i, $i = 1, \ldots, n$ are modeled by a single integrator given by

$$\dot{x}_i(t) = u_i(t), \tag{1}$$

where $x_i(t) \in \mathbb{R}^m$ and $u_i(t) \in \mathbb{R}^m$ denote the states and the inputs of agent v_i, respectively, $i = 1, \ldots, n$. Let us define $x_i^{[k]}$ as the kth entry of the column vector x_i for $i = 1, \ldots, n$ and $k = 1, \ldots, m$.

2.2. Graph Theory

The communication topology of the n agents is modeled by an *n-partite* graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$ [17]. The vertex set $\mathcal{V}$ is partitioned into n classes $\mathcal{V} = \{v_1, \ldots, v_n\}$, where each class represents an agent. Every partition class contains exactly m vertices, which represent the m-dimensional state of an agent. Every edge in $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ has its ends in different classes: vertices in the same partition class are not adjacent. We assume that every two vertices from different partition classes are adjacent. Therefore, the communication topology is modeled by a *complete n-partite* graph; we abbreviate this to $\mathcal{K}_m^n$. The matrix weight

$$\mathcal{W} = \{W_{ij} | i = 1, \ldots, n, j = 1, \ldots, n\}$$

models the connections between every two different classes, where $W_{ij} \succeq 0 \in \mathbb{R}^{m \times m}$ shows the connections of the vertices between classes v_i and v_j. We use $v_j^{[l]} v_i^{[k]}$ to represent an edge from the lth vertex in class v_j to the kth vertex in class v_i, which means that $x_i^{[k]}$ of agent i is affected by $x_j^{[l]}$ of agent j. The strength of this connection is characterized by $[W_{ij}]_{kl} = w_{kl}^{ij} > 0$, the (k,l) entry of W_{ij}. If $w_{kl}^{ij} = 0$, $x_j^{[l]}$ of agent j has no influence on $x_i^{[k]}$ of agent i. If there is at least one entry in W_{ij} in which $w_{kl}^{ij} > 0$, we say agent v_j is a neighbor of agent v_i and $v_j v_i \in \mathcal{E}$. The set of neighbors of agent v_i is defined as $\mathcal{N}_i \triangleq \{v_j \in \mathcal{V} : v_j v_i \in \mathcal{E}\}$. An induced subgraph $\mathcal{G}^{[k]}$ of $\mathcal{G}$ with the vertex set

$$\mathcal{V}^{[k]} = \left\{v_1^{[k]}, \ldots, v_n^{[k]}\right\}$$

is called the *k-state* graph. A path in $\mathcal{G}^{[k]}$ is given by a sequence of distinct vertices in $\mathcal{V}^{[k]}$ connected by an edge in $\mathcal{E}$. If the induced subgraph $\mathcal{G}^{[k]}$ contains a spanning arborescence, then we say the k-state graph is *connected*.

The degree or valency of a given vertex $d(v_i) \in \mathbb{R}^{m \times m}$ is defined as

$$d(v_i) \triangleq \textstyle\sum_{v_j \in \mathcal{N}_i} W_{ij}$$

that is, the sum of weights on the link from each neighbouring node for all states of agent v_i.

We can define the $nm \times nm$ degree matrix of $\mathcal{G}$

$$\Delta_{\mathcal{W}}(\mathcal{G}) = \mathbf{blkdiag}\{d(v_i), i = 1, \ldots, N\}, \tag{2}$$

as the block diagonal matrix of the degrees of all agents in $\mathcal{G}$. The $nm \times nm$ adjacency matrix $A_{\mathcal{W}}(\mathcal{G})$ can be defined as a block matrix, where the (i,j) block is an m-by-m matrix with

$$[A_{\mathcal{W}}]_{ij} = \begin{cases} W_{ij} & \text{if } v_j v_i \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The weighted Laplacian matrix of the graph, $\mathcal{L}_{\mathcal{W}}(\mathcal{G}) \in \mathbb{R}^{nm \times nm}$ is given by

$$\mathcal{L}_{\mathcal{W}} = \Delta_{\mathcal{W}} - A_{\mathcal{W}}. \tag{4}$$

The Laplacian matrix of the induced subgraph $\mathcal{G}^{[k]}$ is denoted by $\mathcal{L}_{\mathcal{W}}^{[k]}$.

Remark 1. *For the special case when $m = 1$, the weight W_{ij} for each edge in $\mathcal{E}$ is a scalar $w_{ij} \in \mathbb{R}_{\geq 0}$.*

Remark 2. *For the special case when $m > 1$ and*

$$W_{ij} = w_{ij} I_m$$

with $I_m \in \mathbb{R}^{m \times m}$ being the identity matrix, $\Delta_W(\mathcal{G})$, the $nm \times nm$ degree matrix of $\mathcal{G}$, is the diagonal matrix of the degree of all the vertices in $\mathcal{G}$ given by

$$\Delta_W(\mathcal{G}) = \Delta_w \otimes I_m \tag{5}$$

where

$$\Delta_w = \operatorname{diag}\left\{\textstyle\sum_{v_j \in \mathcal{N}_i} w_{ij}, i = 1, \ldots, n\right\},$$

and the $nm \times nm$ adjacency matrix $A_W(\mathcal{G})$ is defined as

$$A_W(\mathcal{G}) = A_w \otimes I_m \tag{6}$$

where

$$[A_w]_{ij} = \begin{cases} w_{ij} & \text{if } v_j v_i \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

The matrix-weighted Laplacian of the graph, $\mathcal{L}_W(\mathcal{G}) \in \mathbb{R}^{nm \times nm}$ *is given by*

$$\mathcal{L}_W(\mathcal{G}) = \Delta_W - A_W = \mathcal{L}_w \otimes I_m, \tag{8}$$

where $\mathcal{L}_w = \Delta_w - A_w$.

2.3. Cluster Consensus

Consensus and clustering is defined based on the similarity of the agents state values. Clustering configuration across the sates is also defined.

Definition 1. *If the states* $x_i^{[k]}$ *and* $x_j^{[k]}$ *satisfy the condition*

$$\lim_{t\to\infty} \left\| x_i^{[k]}(t) - x_j^{[k]}(t) \right\| = 0,$$

then we say they belong to the same cluster. All k-clusters $\mathcal{C}^k = \{\mathcal{C}_1^k, \ldots, \mathcal{C}_{p(k)}^k\}$ *form a partition of the kth entry of all states, where* $p(k)$ *is the number of clusters of the kth entry of all states. All agents are said to reach k-cluster consensus (KCC).*

It is easy to verify that all k-clusters satisfy the conditions

$$\bigcup_{i=1}^{p(k)} \mathcal{C}_i^k = \left\{ x_1^{[k]}, \ldots, x_n^{[k]} \right\}$$

and

$$\mathcal{C}_p^k \bigcap \mathcal{C}_q^k = \emptyset$$

for any $\mathcal{C}_p^k \in \mathcal{C}^k, \mathcal{C}_q^k \in \mathcal{C}^k$, and $\mathcal{C}_p^k \neq \mathcal{C}_q^k$.

Definition 2. *If* $p(k) = 1$ *for some* $k \in \{1, \ldots, m\}$ *then we say the kth entry of all states reach k-global consensus (KGC). When* $p(k) = 1$ *for all* $k = 1, \ldots, m$*, then we say all states reach a global consensus (GC).*

Note that for different entries i and j of all states, the i-clusters $\mathcal{C}^i$ and the j-clusters $\mathcal{C}^j$ may not be the same, that is,

$$\mathcal{C}^i \neq \mathcal{C}^j$$

for some $i, j \in \{1, \ldots, m\}$. In other words, the k-cluster consensus may not be uniform across all entries of all states. However, when the clusters are uniform for all entries of all states, we have the following definition.

Definition 3. *If the clusters of all entries satisfying*

$$\mathcal{C}^i = \mathcal{C}^j$$

for all $i \neq j$*,* $i = 1, \ldots, m$*, and* $j = 1, \ldots, m$*, then we say all states reach a global cluster consensus (GCC).*

doi:10.20944/preprints202108.0455.v1

2.4. Objectives

Our main aim in this paper is to examine how the elements of the matrix-weights $\mathcal{W}$ could be chosen, so as to achieve the desired consensus (*KCC*, *GCC*, *KGC*, *GC*).

3. Results

3.1. Consensus Control Design

The consensus control protocol for the n agents is given by

$$u_i(t) = \sum_{v_j \in \mathcal{N}_i} W_{ij}(x_j(t) - x_i(t)). \tag{9}$$

where $W_{ij} \in \mathbb{R}^{m\times m}$ is the weight matrix and $[W_{ij}]_{kl} = w_{kl}^{ij}$ for $k = 1, \dots, m$, and $l = 1, \dots, m$. This control law can be written in a compact matrix form as

$$u(t) = -\mathcal{L}_{\mathcal{W}}(\mathcal{G})x(t), \tag{10}$$

where

$$\begin{aligned} u(t) &= \begin{bmatrix} u_1 \cdots u_i \cdots u_n \end{bmatrix}^T, \quad u_i = \begin{bmatrix} u_i^{[1]} \, \cdots \, u_i^{[m]} \end{bmatrix}^T \\ x(t) &= \begin{bmatrix} x_1 \cdots x_j \cdots x_n \end{bmatrix}^T, \quad x_j = \begin{bmatrix} x_j^{[1]} \, \cdots \, x_j^{[m]} \end{bmatrix}^T \end{aligned}$$

Remark 3. *For the case when $m = 1$, the consensus control law becomes:*

$$u_i(t) = \sum_{v_j \in \mathcal{N}_i} w_{ij}(x_j(t) - x_i(t)),$$

and it can be written in a compact matrix form as:

$$u(t) = -\mathcal{L}_w(\mathcal{G})x(t).$$

Remark 4. *For the case when $m > 1$ and*

$$W_{ij} = w_{ij} I_m,$$

the consensus control law becomes:

$$u_i(t) = \sum_{v_j \in \mathcal{N}_i} w_{ij} I_m (x_j(t) - x_i(t)),$$

and it can be written in a compact matrix form as:

$$u(t) = -\mathcal{L}_W(\mathcal{G})x(t) = -(\mathcal{L}_w(\mathcal{G}) \otimes I_m)x(t).$$

In order to observe the evolution of $x_i^{[k]}$, we need to study the dynamics of $x_i^{[k]}$, which obeys

$$\dot{x}_i^{[k]} = u_i^{[k]}(t).$$

Here

doi:10.20944/preprints202108.0455.v1

$$\begin{aligned} u_i^{[k]}(t) &= \sum_{v_j \in \mathcal{N}_i} \sum_{l=1}^{m} w_{kl}^{ij}(x_j^{[l]}(t) - x_i^{[l]}(t)) \\ &= \sum_{v_j \in \mathcal{N}_i} w_{kk}^{ij}(x_j^{[k]}(t) - x_i^{[k]}(t)) + \sum_{v_j \in \mathcal{N}_i} \sum_{l=1, l \neq k}^{m} w_{kl}^{ij}(x_j^{[l]}(t) - x_i^{[l]}(t)), \end{aligned} \tag{11}$$

is the kth entry of the control input of agent v_i defined in (9). The coefficient w_{kl}^{ij} is the element on the kth row and the lth column of the matrix weight W_{ij}. When $k \neq l$, w_{kl}^{ij} is an off-diagonal element on the kth row of the matrix weight W_{ij}, and it qualifies the effect of the lth states of the neighbors of agent i on its kth state. When $k = l$, w_{kk}^{ij} is a diagonal element of the matrix weight W_{ij}, and it qualifies the effect the kth states of the neighbors of agent v_i on its kth state.

3.2. Non-diagonal Matrix-Weights

When the matrix weight W_{ij} is a non-diagonal matrix, we examine two possible scenarios.

3.2.1. Control law depends on other states in the same cluster

If for any j and any $l \neq k$, the pair $(x_j^{[l]}, x_i^{[l]})$ belongs to the same cluster, then the control law in (11) approaches

$$u_i^{[k]}(t) \rightarrow \sum_{v_j \in \mathcal{N}_i} w_{kk}^{ij}(x_j^{[k]}(t) - x_i^{[k]}(t)) \tag{12}$$

as $t \rightarrow \infty$.

Theorem 1. *Assume that (12) holds for all $i = 1, \ldots, n$.*

1. *If the k-state graph is connected, the kth states of all agents reach k-global consensus (KGC).*
2. *If the k-state graph is not connected, the kth states of all agents reach k-cluster consensus (KCC). Moreover, the number of clusters of the states is determined by the number of clusters of the k-state graph.*

3.2.2. Control law depends on other states in different clusters

In this case, the evolution of all states follows

$$\dot{x}(t) = u(t)$$

where $u(t)$ is given in (10). By solving the above differential equation, we have

$$x(t) = e^{-t\mathcal{L}_{\mathcal{W}}}x(0). \tag{13}$$

The choice of the weight matrix $\mathcal{W}$ will affect the rate of convergence of the consensus algorithm of the states of all agents. Since the Laplacian matrix $\mathcal{L}_{\mathcal{W}}$ is positive semi-definite, all states will converge to a constant value eventually.

Theorem 2. *Assume that a state k depends on some of agent of state l in y different clusters, and that the k state is not dependent on any other state. The following will hold:*

1. *There will at least be y clusters for the corresponding agent states in the graph causing a k-cluster consensus (KCC).*

2. *If any agent state, $x_j^{[k]}$ that satisfies ((12) is connected to only one other node, $x_i^{[k]}$ that depends on some agent from state l, then $x_j^{[k]}, x_i^{[k]} \in \mathcal{C}_p^k$.*
3. *If any agent state, $x_j^{[k]}$ that satisfies ((12) is connected to more than one node that depends on some agent from state l, then $x_j^{[k]}$ will be in a different cluster.*

3.3. *Diagonal Matrix Weights*

The absence of off-diagonal elements in the matrix-weights W_{ij} guarantees that there is no cross-entry state dependence, that is, $w_{kl}^{ij} = 0$ when $k \neq l$. The control law reduces to:

$$u_i^{[k]}(t) = \sum_{v_j \in \mathcal{N}_i} w_{kk}^{ij}(x_j^{[k]}(t) - x_i^{[k]}(t)).$$

Here we discuss two different cases based on positive definite or positive semi-definite of matrices W_{ij}.

3.3.1. Positive Definite Matrix Weights

When diagonal positive definite (PD) matrices constitute the matrix-weights in the weight set $\mathcal{W} = \{W_{ij} : v_j v_i \in \mathcal{E}\}$, it implies that $w_{kk}^{ij} \neq 0$ for all $k = 1, \ldots, m$, and for all $v_j v_i \in \mathcal{E}$.

Define

$$x^{[k]} = \begin{bmatrix} x_1^{[k]} & \ldots & x_n^{[k]} \end{bmatrix}^T$$

as the vector containing the k-state of all agents. Its dynamics follow

$$\dot{x}^{[k]}(t) = -\mathcal{L}_{\mathcal{W}}^{[k]} x^{[k]}(t) \tag{14}$$

The solution of the above differential equation is given by

$$x^{[k]}(t) = e^{-t\mathcal{L}_{\mathcal{W}}^{[k]}} x^{[k]}(0). \tag{15}$$

The choice of w_{kk}^{ij} will affect the rate of convergence of the consensus algorithm of the k-state of all agents.

Consider the case when the matrix weights are scalars multiplying of the identity matrix. We know that for any square matrices $A_n \in \mathbb{R}^{n \times n}$ with $\lambda_1, ..., \lambda_n$, and $B_m \in \mathbb{R}^{m \times m}$ with $\mu_1, ..., \mu_m$ that the eigenvalues of $A_n \otimes B_m$ are $\lambda_i \mu_j, i = 1, \ldots, n, j = 1, \ldots, m$ [18]. Hence, the eigenvalues of $\mathcal{L}_{\mathcal{W}} = \mathcal{L}_w(\mathcal{G}) \otimes I_m$ are the eigenvalues of $\mathcal{L}_w(\mathcal{G})$ repeated m times, and $\mathcal{L}_{\mathcal{W}}^{[k]} = \mathcal{L}_w$ for all $k = 1, \ldots, m$. Then (15) becomes

$$x^{[k]}(t) = e^{-\mathcal{L}_w t} x^{[k]}(0),$$

which implies *GC*.

Theorem 3. *Assume that all the matrix-weights are diagonal PD matrices.*

1. *If the k-state graph is connected, there will be a global consensus (GC) across the m states of all agents.*
2. *If the k-state graph is not connected, there will be a global cluster consensus (GCC) across the m states such that $\mathcal{C}^i = \mathcal{C}^j$ for all $i \neq j$, $i = 1, \ldots, m$, and $j = 1, \ldots, m$. Moreover, the number of clusters of the states is determined by the number of clusters of the k-state graph.*

3.3.2. Positive Semi-definite Matrix Weights

When diagonal positive semi-definite (PSD) matrices constitute the matrix-weights in the weight set $\mathcal{W} = \{W_{ij} : v_j v_i \in \mathcal{E}\}$, it implies that for any $W_{ij} \neq 0$, there exist k and

l such that $w_{kk}^{ij} = 0$ and $w_{ll}^{ij} \neq 0$. In this case, the edge $v_j^{[k]} v_i^{[k]}$ does not exist for the kth states between agent v_i and agent v_j, which leads to $\mathcal{C}^k \neq \mathcal{C}^l$ in general.

Theorem 4. *Assume that the matrix weights are diagonal PSD matrices, then for any state k, the following holds.*

1. *If the k-state graph is connected, there will be k-global consensus (KGC).*
2. *If the k-state graph is not connected, then there is a k-cluster consensus (KCC). Moreover, the number of clusters of the states is determined by the number of clusters of the k-state graph.*

Hence, by the choice of the diagonal entries of the matrix-weights as zero or non-zero, the graph structure of the states can be made different for different states.

4. Simulations

To demonstrate the effects of the matrix weights on the consensus algorithm, we choose a network of three agents, where each agent has three states. The initial conditions are given by

$$x_1(0) = [4 \quad 25 \quad 30]^T, \; x_2(0) = [13 \quad 2 \quad 12]^T, \; x_3(0) = [21 \quad 18 \quad 10]^T.$$

4.1. Non-diagonal Matrix-Weights

4.1.1. Control law dependent on other state values in the same cluster

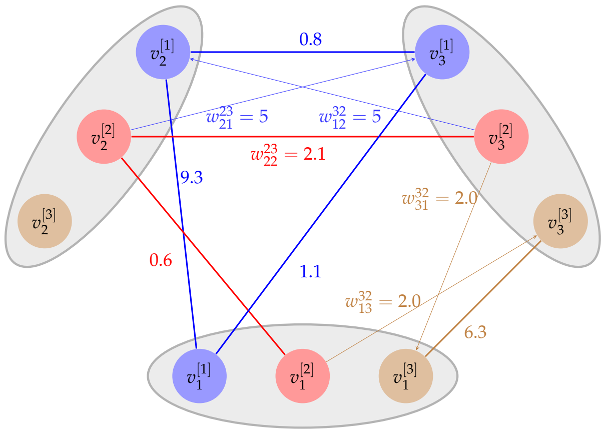

Figure 1. Graph topology with non-diagonal matrix weights: 1*GC*, 2*CC* and 3*CC*

For the graph topology shown in Figure 1, the state trajectory is presented in Figure 2. It can be seen from Figure 1 that the 2-state graph is connected, and is not dependent on any other state. Therefore, there is a 2-global consensus (2*GC*): $\mathcal{C}_1^2 = \{x_1^{[2]}, x_2^{[2]}, x_3^{[2]}\}$. The 1-state graph is connected but is dependent on agent states from cluster $\mathcal{C}_1^2$. According to Theorem 1, there is a 1-global consensus (1*GC*) since all 2-states belong to the same cluster. For 3-state, the 3-state graph is disconnected and is dependent on 2-state values from the cluster $\mathcal{C}_1^2$. Therefore, $\{x_1^{[3]}, x_3^{[3]}\} \in \mathcal{C}_1^3$ form a first cluster and $\{x_2^{[3]}\} \in \mathcal{C}_2^3$ forms another cluster based on Theorem 1. Hence, there is a 3-cluster consensus (3CC).

4.1.2. Control law dependent on other state values in different clusters

The graph topology shown in Figure 3 is obtained from Figure 1 by setting $w_{22}^{32} = 0$, and $w_{33}^{32} = 3$. The state trajectory is presented in Figure 4. It can be seen from Figure 3, the 2-state graph is disconnected and do not depend on any other state values. Thus for the 2-state, there are now two clusters $\mathcal{C}_1^2 = \{x_1^{[2]}, x_2^{[2]}\}$, and $\mathcal{C}_2^2 = \{x_3^{[2]}\}$. For the

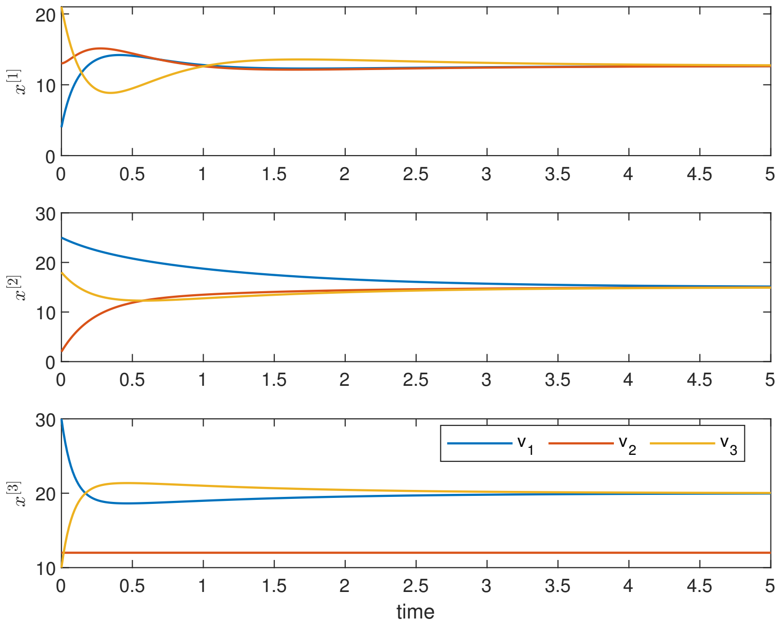

Figure 2. State trajectory for non-diagonal matrix weighted graph: 1GC, 2CC and 3CC

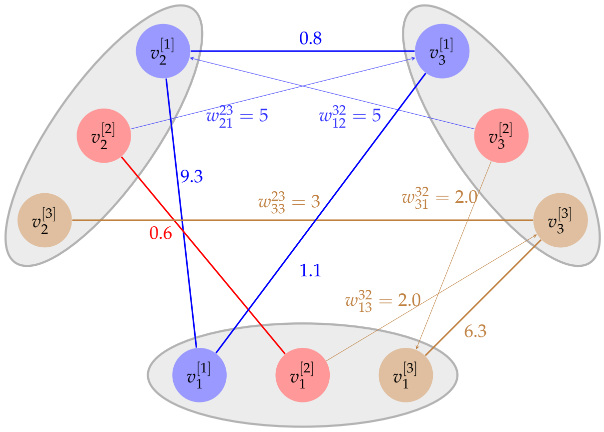

Figure 3. State graphs for non-diagonal matrix weights: 1CC, 2CC and 3CC

1-state, though the graph is connected, 1-state global consensus is not guaranteed since it depends on agent states from different clusters in $\mathcal{C}^2$. There will be at least 2 clusters since the control algorithm $u_2^{[1]} = f(c_{21})$ and $u_3^{[1]} = f(c_{31}) \quad c_{21} \neq c_{31}$ (Theorem 2). Moreso, since $x_1^{[1]}$ is linked to agents $x_2^{[1]} \in \mathcal{C}_1^1$ and $x_3^{[1]} \in \mathcal{C}_2^1$, it will be in a different cluster: $x_1^{[1]} \in \mathcal{C}_3^1$ (Theorem 2).

The same reasoning applies to the 3-state graph. The graph is connected and depends on agent states from different clusters in $\mathcal{C}^2$. There will be at least 2 clusters since the control algorithm $u_1^{[3]} = f(c_{13})$ and $u_3^{[3]} = f(c_{33}) \quad c_{13} \neq c_{33}$ (Theorem 2). Since $u_2^{[3]}$ satisfies (12) and is connected to only $v_3^{[3]}$, there are two clusters in $\mathcal{C}^3$ (Theorem 2).

4.2. Diagonal Matrix-Weights

The consensus algorithm under the graph with all matrix weights set to be scalar multiplying the identity matrix is simulated and the state trajectory is plotted in Figure 5, where the matrix weights are chosen as

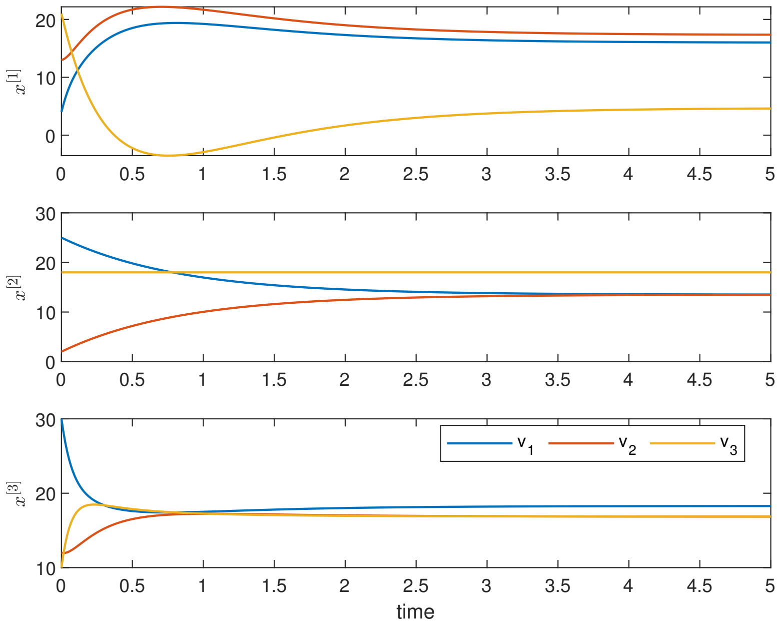

Figure 4. State trajectory for non-diagonal matrix weighted graph: $1CC$, $2CC$ and $3CC$

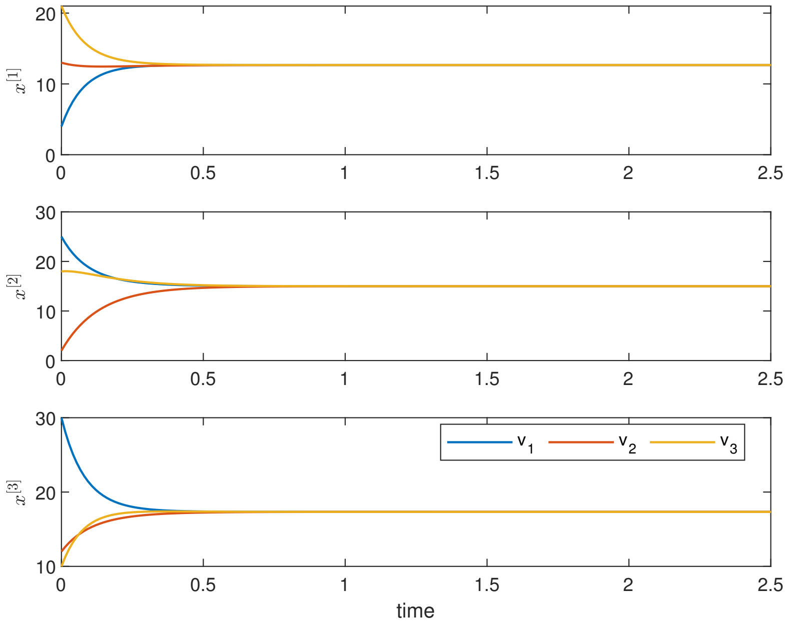

Figure 5. State trajectory for diagonal matrix weighted graph: weights multiple of the identity matrix (GC with similar convergence properties across the states).

$$W_{12} = 3I_{3\times3}, \quad W_{32} = 1.9I_{3\times3}, \quad W_{13} = 5I_{3\times3}. \tag{16}$$

The graph structure per state is the same for all states with similar convergence properties when the matrix-weights are scalars multiplying the identity matrices. When the matrix weights are instead chosen to be general PD diagonal matrices as

doi:10.20944/preprints202108.0455.v1

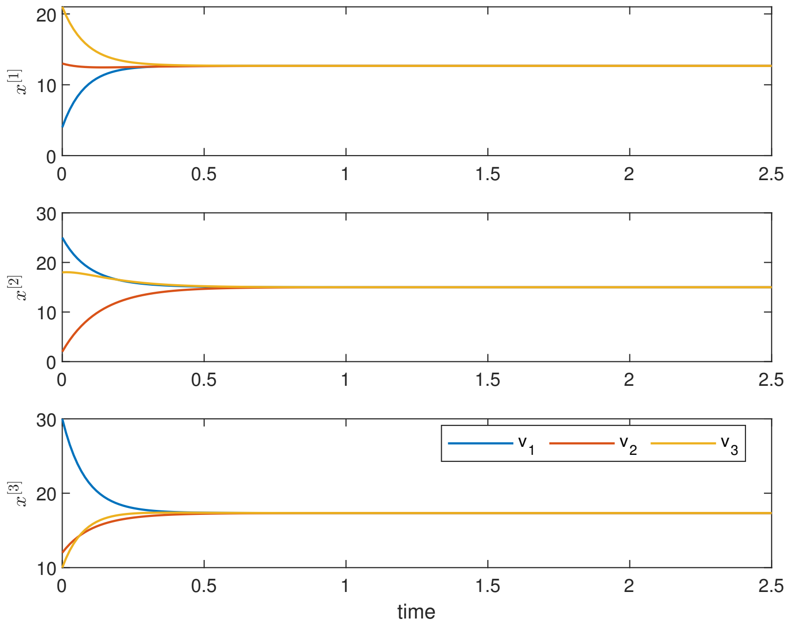

Figure 6. State trajectory for general diagonal PD matrix weighted graph (GC with different convergence properties across the states).

$$W_{12} = \mathrm{diag}\{1.3, 4.0, 6.1\},\ W_{32} = \mathrm{diag}\{0.8, 2.1, 3.1\},\ W_{13} = \mathrm{diag}\{8.1, 1.4, 3.1\},$$

the consensus algorithm is simulated and the state trajectory is shown in Figure 5. The convergence properties are different from state to state. The choice of w_{kk}^{ij} will affect the rate of convergence of the consensus algorithm of the kth state according to (15). In all cases, a global consensus (GC) is achieved based on Theorem 3.

When the matrix weights are diagonal PSD matrices chosen as

$$W_{12} = \mathrm{diag}\{1.3, 4.0, 0\},\ W_{32} = \mathrm{diag}\{0, 0, 0\},\ W_{13} = \mathrm{diag}\{8.1, 0, 3.1\}.$$

the consensus algorithm is simulated. The state trajectory is shown in Figure 7. Since the 1-state graph is connected due to $w_{11}^{12} \neq 0$ and $w_{11}^{13} \neq 0$, there is a 1-global consensus (1GC). For 2-states, there is a 2-cluster consensus (2CC) because $w_{22}^{13} = w_{22}^{32} = 0$. There are no links between $\{x_1^{[2]}, x_2^{[2]}\} \in \mathcal{C}_1^2$ and $\{x_3^{[2]}\} \in \mathcal{C}_2^2$. The same analogy applies for the 3-cluster consensus (3CC) in 3-states: $\{x_1^{[3]}, x_3^{[3]}\} \in \mathcal{C}_1^3$ and $\{x_2^{[3]}\} \in \mathcal{C}_2^3$ because $w_{33}^{12} = w_{33}^{32} = 0$.

5. Discussion

Matrix weights allow for coupling the consensus of a state to agent values in another state. This inter-dependencies may get complex if for instance each agent state in a system is dependent on every other state. However, when there are no much complexities, the nature of the consensus can be determined based on the number of connected components of the state graph and the clusters of the agent states linked to it. Further work may be needed to extend the results to be able to predict the number of clusters when there are complex inter-dependencies among the states in the system.

doi:10.20944/preprints202108.0455.v1

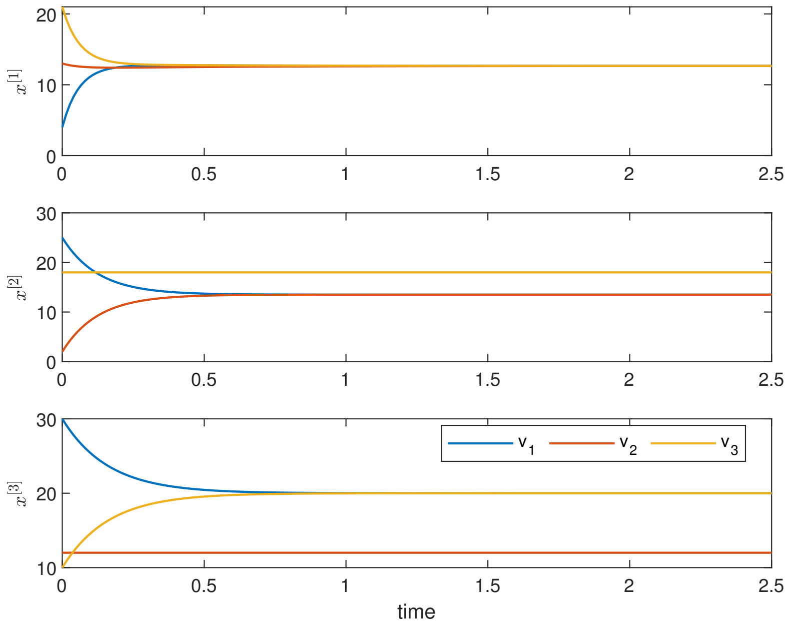

Figure 7. State trajectory for general diagonal PSD matrix weighted graph (KGC and KCC).

6. Conclusions

In this paper, the role of each element in the matrix-weights of a multi-agent system have been studied under two classifications: diagonal and non-diagonal elements. Analyses has been carried for determining the number of clusters per state. The results have further been demonstrated via simulation of a multi-agent system consisting of three agents.

Author Contributions: ‘Conceptualization, J.0.; methodology, J.O and X.M.; software, J.O. and X.M.; validation, X.M. and J.O.; formal analysis, X.M and J.O.; writing—original draft preparation, J.O.; writing—review and editing, X.M.; visualization, X.M.; supervision, X.M.; project administration, X.M.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MAS	Multi-agent Systems
PD	Positive definite
PSD	Positive semi-definite
GC	Global consensus
CC	Clustered consensus

References

1. Wooldridge, M. *An introduction to multiagent systems*; John wiley & sons, 2009.
2. Mesbahi, M.; Egerstedt, M. *Graph theoretic methods in multiagent networks*; Vol. 33, Princeton University Press, 2010.
3. Zhao, H.; Meng, X.; Wu, S. Distributed edge-based event-triggered coordination control for multi-agent systems. *Automatica* **2021**, *132*, 109797.
4. Shi, C.X.; Yang, G.H.; Li, X.J. Data-Based Fault-Tolerant Consensus Control for Uncertain Multiagent Systems Via Weighted Edge Dynamics. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **2019**, *49*, 2548–2558.
5. Zhu, B.; Xie, L.; Han, D.; Meng, X.; Teo, R. A survey on recent progress in control of swarm systems. *Science China Information Sciences* **2017**, *60*, 1–24.

6. Dorri, A.; Kanhere, S.S.; Jurdak, R. Multi-Agent Systems: A Survey. *IEEE Access* **2018**, *6*, 28573–28593.
7. Qin, J.; Ma, Q.; Shi, Y.; Wang, L. Recent advances in consensus of multi-agent systems: A brief survey. *IEEE Transactions on Industrial Electronics* **2016**, *64*, 4972–4983.
8. Meng, X.; Chen, T. Event based agreement protocols for multi-agent networks. *Automatica* **2013**, *49*, 2125–2132.
9. Meng, X.; Xie, L.; Soh, Y.C.; Nowzari, C.; Pappas, G.J. Periodic event-triggered average consensus over directed graphs. 2015 54th IEEE Conference on Decision and Control (CDC), 2015, pp. 4151–4156.
10. Meng, X.; Xie, L.; Soh, Y.C. Asynchronous periodic event-triggered consensus for multi-agent systems. *Automatica* **2017**, *84*, 214–220.
11. Li, Y.; Tan, C. A survey of the consensus for multi-agent systems. *Systems Science & Control Engineering* **2019**, *7*, 468–482.
12. Chen, Y.; Lu, J.; Yu, X.; Hill, D.J. Multi-agent systems with dynamical topologies: Consensus and applications. *IEEE circuits and systems magazine* **2013**, *13*, 21–34.
13. Trinh, M.H.; Van Nguyen, C.; Lim, Y.H.; Ahn, H.S. Matrix-weighted consensus and its applications. *Automatica* **2018**, *89*, 415–419.
14. Van Tran, Q.; Trinh, M.H.; Ahn, H.S. Discrete-Time Matrix-Weighted Consensus. *arXiv preprint arXiv:2006.15530* **2020**.
15. Su, H.; Chen, J.; Yang, Y.; Rong, Z. The Bipartite Consensus for Multi-Agent Systems With Matrix-Weight-Based Signed Network. *IEEE Transactions on Circuits and Systems II: Express Briefs* **2020**, *67*, 2019–2023.
16. Liu, B.; Su, H.; Wu, L.; Shen, X. Controllability for multi-agent systems with matrix-weight-based signed network. *Applied Mathematics and Computation* **2021**, *411*, 126520.
17. Diestel, R. *Graph Theory*, 4 ed.; 2010.
18. Laub, A.J. *Matrix Analysis for Scientists and Engineers*; 2005.