Submitted to CDC 2016

Adaptation and Synchronization over a Network: Asymptotic Error Convergence and Pinning

Travis E. Gibson

***Abstract*—The synchronization of unknown systems is studied for both undirected and directed graphs. In the undirected setting it is shown that consensus can be achieved without an external consensus protocol (that is, without a high gain linear error input usually written as a function of the graph laplacian and the states in the network), but solely through local adaptive feedback. In the directed case several different scenarios are addressed. An emphasis is placed on analyzing the simplest possible control design to achieve the goal of consensus. This breaks from the pinning adaptive control literature where the most general case is usually addressed, inadvertently obscuring what is, and what is not needed to achieve stability. Also breaking from the literature in the area of *Distributed Adaptive Control with Synchronization* (DACS) we do not assume a-priori knowledge of a uniform bound on the plant state.**

I. Introduction

Synchronization is now studied rigorously by many researchers in the control community. Any type of control that existed before, now has a distributed or network based result that looks to exploit neighbor information to achieve consensus. Adaptive control is no different. Even as early as the 80's *Decentralized Adaptive Control* (DAC) was being studied [7, 15, 17, 23]. In these works each agent in the network has its own reference model dynamics to follow and in addition to adapting to the reference model, the controller rejects unknown disturbances from neighboring agents. Synchronization is not a goal in those works and thus is not directly related to the contribution in this paper.

In this work the problem of synchronization is studied, where only a subset of the agents have access to the reference model. The closest manifestation to our work is that of *Distributed Adaptive Control with Synchronization* (DACS) [3,29] and *Distributed Adaptive Control with Adaptive Synchronization* (DACAS) [4]. In the DACS paradigm adaptation is incorporated so as to overcome uncertainty in the local dynamics while a linear non-adaptive synchronization input is given to each agent. In the specific DACS strategies just referenced it is worth noting that a pinning trajectory is used as a reference [6, 11]. The pinning trajectory need only be shared with one of the agents. In [29] the asymptotic convergence properties are addressed under the assumption of persistence of excitation and the underlying graphs are symmetric. In [3] it is worth noting that it is a-priori assumed that the regressor vector is bounded, which in our case implies boundedness of the plant state. Also, in [3] the compact set that the model following error is proved to converge to is proportional to the upper bound on the matching condition. Thus, the results in [3] are local and asymptotic error convergence is not possible even without disturbances or nonlinearities.

There are three main contributions in this work. First, for the simple case of symmetric graphs it shown how consensus can be achieved solely through adaptation, i.e. *without* the use of a synchronizing input from nearest neighbor errors. The second contribution of this work is a derivation of asymptotic error convergence in DACS without the a-priori assumption of bounded state trajectories. This is achieved through the appropriate waiting of the adaptive parameter error in our Lyapunov functions and through the exploitation of the existence of a specific diagonal solution to the Lyapunov equation of interest that balances the underlying graph. This diagonal solution can easily be shown to exist by using the *Perron-Frobenius* (PF) Theorem, and has been proved several times in the literature, but the fact that these diagonal solutions are a graph balancing, *as well*, has not been fully exploited in the past. The final contribution of this work comes via the fact that our adaptive laws are not a function of the diagonal solutions to the Lyapunov equation (which here is a function of the graph) and thus our solutions are truly local.

The paper is organized as follows. In Section II notation is covered. In Section III the general problem is presented. In Section IV analysis is performed for undirected graphs. Section V contains analysis for directed graphs. Section VI contains simulation results, and finally Section VII closes with a discussion.

II. Notation

Real numbers are denoted as $\mathbb{R}$, $n\times m$ matrices in the reals are denoted as $\mathbb{R}^{n\times m}$, positive diagonal square matrices as $\mathbb{D}^n_{>0}$ and non-negative square diagonal matrices as $\mathbb{D}^n_{\geq 0}$. The following shorthand is used for vectors of identical values $\mathbf{1} \triangleq [1,\ 1,\ \ldots,\ 1]^\mathsf{T}$ and similarly $\mathbf{0} \triangleq [0,\ 0,\ \ldots,\ 0]^\mathsf{T}$ where the dimension will be obvious from the algebra.. One special linear subspace of $\mathbb{R}^n$ that we will encounter frequently contains those elements $\mathbf{1}c$ where $c \in \mathbb{R}$, this special 1-dimensional linear subspace is denoted $\mathbb{R}^n_{\mathbf{1}}$. The Hadamard product is denoted $\circ$. Throughout this work $(\cdot)^\mathsf{T}$ is the transpose operator, $\|\cdot\|$ is the euclidean or induced euclidean norm, and $|\cdot|$ is the cardinality of a set.

A digraph is defined by the double $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{i\}_{i=1}^n$ is the vertex set and the directed edges are defined by the ordered pairs $(i,j) \in \mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. An element $(i,j) \in \mathcal{E}$ if and only if there is a directed edge from vertex i to vertex j. A useful algebraic component when discussing

T. E. Gibson is with Harvard Medical School and the Channing Division of Network Medicine, Department of Medicine, Brigham and Women's Hospital, Boston MA, 02115, email: (travis.gibson@channing.harvard.edu).

graphs is the adjacency matrix $\mathcal{A}(\mathcal{G})$, whose components are defined as follows $[\mathcal{A}]_{ij} = 1$ if $(v_j, v_i) \in \mathcal{E}$ and $[\mathcal{A}]_{ij} = 0$, otherwise. $\mathcal{L}(\mathcal{G}) \triangleq \mathcal{D}(\mathcal{G}) - \mathcal{A}(\mathcal{G})$ is the in-degree *laplacian* of $\mathcal{G}$, and $\mathcal{D}(\mathcal{G})$ a diagonal matrix with each $[\mathcal{D}]_{ii}$ equal to the in-degree of node i.

The following convention will loosely be followed. Vectors are lower case and matrices are uppercase. Local variables will be denoted in italic, x, y, z or X, Y, Z, global variables in bold $\mathbf{x}, \mathbf{y}, \mathbf{z}$ or $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$, graph properties in uppercase calligraphic, $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$. Familiarity with results in the seminal papers [5, 16, 21, 24], the monographs on algebraic graph theory [2, 10], and the following text on nonnegative matrices [1, 19] is assumed. Some familiarity with the adaptive control texts [14, 20] is also assumed.

III. Problem Statement

For the problem under consideration there are n agents, Σ_i where $i \in \{1, 2, \ldots, n\}$, on a graph $\mathcal{G}$, where each agent is associated with a vertex and the edges on the graph illustrate the communication topology. State information is exchanged per the graph topology so as to update a local adaptive controller. For the case of symmetric graphs synchronization will be proved *without a pinning consensus input, but entirely through adaptation*. Similar in spirit to pinning control, only a subset of the nodes will have access to the reference model. For directed graphs we only show stability in the presence of a linear consensus input.

The dynamics for each agent are defined as

$$\Sigma_i : \quad \dot{x}_i(t) = a_i x_i(t) + u_i(t), \quad i \in \mathcal{V}(\mathcal{G}) \tag{1}$$

where $x_i, u_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ and $a_i \in \mathbb{R}$. For the first problem we tackle the following adaptive control input will be used[1]

$$u_i(t) = \hat{k}_i(t)x(t) + \hat{r}(t), \tag{2}$$

where $\hat{k}_i, \hat{r}_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$, whose update laws are to be defined shortly. The goal is to design the update laws for $\hat{k}$ and $\hat{r}$ so that each x_i will follow the scalar reference model

$$\dot{x}_{\mathrm{m}}(t) = a_{\mathrm{m}} x_{\mathrm{m}}(t) + r \tag{3}$$

where $x_{\mathrm{m}} : \mathbb{R}_{\geq 0} \to \mathbb{R}$, $a_{\mathrm{m}} < 0$, and r is a constant. The explicit time dependance of signals will be suppressed from this point forwards, except for emphasis. In this construction the the upright roman letter "m" is used in the subscript of the reference model so that it is not confused with a specific $m \in \mathcal{V}$.

As a preliminary step let us write the plant dynamics in a form that is more amicable to adaptive control law construction. First let us define the matching gain

$$k_i \triangleq a_{\mathrm{m}} - a_i, \tag{4}$$

thus if $\hat{k}_i = k_i$, the closed loop plant Jacobian will match the reference model Jacobian. Now the dynamics in (1) and (2), with the definition in (4), can be written as

$$\Sigma_i : \quad \dot{x}_i = a_{\mathrm{m}} x_i + r + \tilde{k}_i x_i + \tilde{r}, \quad i \in \mathcal{V}(\mathcal{G}) \tag{5}$$

[1]The controller and update laws will be modified as necessary as we discuss slightly different problems.

where feedback gain error $\tilde{k}_i \triangleq \hat{k}_i - k_i$ and reference input error $\tilde{r} \triangleq \hat{r} - r$. The seasoned adaptive control theorist will recognize these standard definitions. Before moving onto the graphical issues and how we can construct a local learning law without global knowledge of the reference model state, let us write the plant in (5) and the reference model in (3) in the following compact form

$$\dot{\mathbf{x}} = \mathbf{A}_{\mathrm{m}}\mathbf{x} + \mathbf{r} + \tilde{\mathbf{K}}\mathbf{x} + \tilde{\mathbf{r}} \tag{6}$$

$$\dot{\mathbf{x}}_{\mathrm{m}} = \mathbf{A}_{\mathrm{m}}\mathbf{x}_{\mathrm{m}} + \mathbf{r} \tag{7}$$

where $\mathbf{x} = [x_1,\ x_2,\ \ldots,\ x_n]^\mathsf{T}$, $\mathbf{x}_{\mathrm{m}} = \mathbf{1}x_{\mathrm{m}}$, $\mathbf{A}_{\mathrm{m}} = a_{\mathrm{m}} I_{n\times n}$, $\mathbf{r} = \mathbf{1}r$, $\hat{\mathbf{r}} = [\hat{r}_1,\ \hat{r}_2,\ \ldots,\ \hat{r}_n]^\mathsf{T}$, $\tilde{\mathbf{r}} = \hat{\mathbf{r}} - \mathbf{r}$, $\hat{\mathbf{k}} = [\hat{k}_1,\ \hat{k}_2,\ \ldots,\ \hat{k}_n]^\mathsf{T}$, $\mathbf{k} = [k_1,\ k_2,\ \ldots,\ k_n]^\mathsf{T}$, $\tilde{\mathbf{k}} = \hat{\mathbf{k}} - \mathbf{k}$ and finally $\tilde{\mathbf{K}} = \mathsf{diag}(\tilde{\mathbf{k}})$ with $\mathbf{K}$ and $\mathbf{K}^*$ similarly defined. Using the compact form in (6) and (7) the global error $\mathbf{e} = \mathbf{x} - \mathbf{x}_{\mathrm{m}}$ satsifies

$$\dot{\mathbf{e}} = \mathbf{A}_{\mathrm{m}}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \tilde{\mathbf{r}}.$$

We are now ready to discuss the network that the agents communicate over and the relationship between the network and the reference model.

As with most synchronization protocols we wish to define an error that is locally computable. most often in linear consensus the following error is used $\mathcal{L}\mathbf{x}$. In addition to the local uncertain agents however, we must also inject the reference model state information into the system. Let $\mathcal{T} \subset \mathcal{V}$ be the set of all *target nodes*. The target nodes are all those nodes that receive information from the reference model. The target nodes will incorporate the reference model state through the matrix $\mathcal{M} \in \mathbb{D}^n_{\geq 0}$ which is defined as follows, $[\mathcal{M}]_{ii} = 1$ if $i \in \mathcal{T}$ and $[\mathcal{M}]_{ii} = 0$ otherwise.

A local error $\mathbf{e}_\beta$ incorporating the reference model is then constructed as follows

$$\mathbf{e}_\beta = \mathcal{L}\mathbf{x} + \mathcal{M}(\mathbf{x} - \mathbf{x}_{\mathrm{m}}). \tag{8}$$

Noting that $\mathbf{x}_{\mathrm{m}} \in \mathbb{R}^n_{\mathbf{1}}$ it follows that $\mathcal{L}\mathbf{x}_{\mathrm{m}} = \mathbf{0}$, and thus the error above can be written in the slightly more compact form

$$\mathbf{e}_\beta = \mathcal{B}\mathbf{e} \tag{9}$$

$$\mathcal{B} \triangleq \mathcal{L} + \mathcal{M}. \tag{10}$$

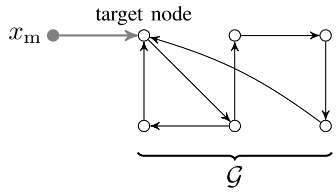
While $\mathbf{e}$ is the global error, not all agents have information regarding the reference model and thus it is not directly computable. The error $\mathbf{e}_\beta$ on the other hand is locally computable. The update law for the adaptive parameters is then given by[2]

$$\dot{\hat{\mathbf{k}}} = -\mathbf{x} \circ \mathbf{e}_\beta \tag{11}$$

$$\dot{\hat{\mathbf{r}}} = -\mathbf{e}_\beta. \tag{12}$$

[2]Recall that $\circ$ denotes the Hadamard product.

Example 1. So as to illustrate the role of $\mathcal{L}$ and $\mathcal{M}$, consider the following example



where the vertices in the graph $\mathcal{G}$ are denoted with open circles as $\circ$, and the reference model node, which is outside the graph, denoted as $\bullet$. If the single target node in this example is vertex 1, then $[\mathcal{M}]_{11} = 1$ and $[\mathcal{M}]_{ii} = 0$ for all $i \in \{j\}_{j=2}^{n}$.

We now wish to make three of our assumptions, some of which have already been stated, explicit.

Assumption 1. The digraph $\mathcal{G}$ is strongly connected without self loops or redundant edges.

Assumption 2. The reference model interacts with at least one vertex in the graph, $|\mathcal{T}| \geq 1$.

Assumption 3. The signal r in the reference model dynamics (3) is a constant, and thus $\mathbf{r}$ by implication.

We note that Assumption 1 can be relaxed, for some of our analysis, to scenarios where the graph is only connected, but with the added constraint that the reference model must then share information with a root of the graph (or a minimum root set). If Assumption 3 is relaxed to general time varying signals then asymptotic agreement in the model following error is impossible, unless $r(t)$ is globally known. We will assume r is known globally in one section so that we can directly address state feedback gain adaptation without extra pieces in the controller complicating our analysis. When we analyze the special case for r globally known, we will thus replace Assumption 3 with the following.

Assumption 3′. The signal r in the reference model is bounded and globally known.

Indeed if $r(t)$ is unknown and not a constant, then the model following error could be shown to asymptotically converge to a compact set proportional to the product of the bound on the signal and the bound on the derivative of the signal.

IV. Symmetric Graphs

We now address the stability of the controller presented in the previous section for symmetric graphs.

Theorem 1. *Given the dynamics in* (6)*, the reference model in* (7)*, the error dynamics in* (9)*, and the update laws in* (11) *and* (12)*, communicating over a symmetric graph $\mathcal{G}$ with the target nodes denoted by $\mathcal{M}$ and satisfying Assumptions 1-3, all states are uniformly bounded and* $\lim_{t\to\infty} \mathbf{e}(t) = \mathbf{0}$.

Proof. Consider the Lyapunov candidate

$$V(\mathbf{e}_\beta, \tilde{\mathbf{k}}, \tilde{\mathbf{r}}) = \mathbf{e}_\beta^\mathsf{T} \mathcal{B}^{-1} \mathbf{e}_\beta + \tilde{\mathbf{k}}^\mathsf{T} \tilde{\mathbf{k}} + \tilde{\mathbf{r}}^\mathsf{T} \tilde{\mathbf{r}}.$$

Differentiating along the system dynamics it follows that

$$\begin{aligned}\dot{V} = &\ \mathbf{e}_\beta^\mathsf{T}(\mathbf{A}_{\mathrm{m}}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \tilde{\mathbf{r}}) + (\mathbf{A}_{\mathrm{m}}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \tilde{\mathbf{r}})^\mathsf{T}\mathbf{e}_\beta \\ &+ \dot{\tilde{\mathbf{k}}}^\mathsf{T}\tilde{\mathbf{k}} + \tilde{\mathbf{k}}^\mathsf{T}\dot{\tilde{\mathbf{k}}} + \dot{\tilde{\mathbf{r}}}^\mathsf{T}\tilde{\mathbf{r}} + \tilde{\mathbf{r}}^\mathsf{T}\dot{\tilde{\mathbf{r}}}\end{aligned}$$

Recalling that $\mathbf{A}_{\mathrm{m}} = a_{\mathrm{m}} I_{n\times n}$, defined just below (7), and the definition of $\mathbf{e}_\beta$ in (10), it follows that

$$\dot{V} = 2a_{\mathrm{m}}\mathbf{e}_\beta^\mathsf{T}\mathcal{B}^{-1}\mathbf{e}_\beta + 2\mathbf{e}_\beta^\mathsf{T}\tilde{\mathbf{K}}\mathbf{x} + 2\mathbf{e}_\beta^\mathsf{T}\tilde{\mathbf{r}} + 2\dot{\tilde{\mathbf{k}}}^\mathsf{T}\tilde{\mathbf{k}} + 2\dot{\tilde{\mathbf{r}}}^\mathsf{T}\tilde{\mathbf{r}}$$

Using the update laws in (11) and (12) it follows that

$$\dot{V} = 2a_{\mathrm{m}}\mathbf{e}_\beta^\mathsf{T}\mathcal{B}^{-1}\mathbf{e}_\beta.$$

Given that $a_m < 0$ it follows that all parameters are uniformly bounded, the entire system is stable and applying Barbalet Lemma it follows that $\mathbf{e}$ tends to zero as t goes to infinity. □

Remark 1. The Lyapunov function is equivalent to $V(\mathbf{e}, \tilde{\mathbf{k}}, \tilde{\mathbf{r}}) = \mathbf{e}^\mathsf{T}\mathcal{B}\mathbf{e} + \tilde{\mathbf{k}}^\mathsf{T}\tilde{\mathbf{k}} + \tilde{\mathbf{r}}^\mathsf{T}\tilde{\mathbf{r}}$.

We only derived stability here for undirected graphs to show that synchronization could be achieved solely through adaptation. We note that a nonlinear map $\mathbf{f}(\mathbf{x})$ in place of $\mathbf{A}_{\mathrm{m}}\mathbf{x}$ can easily be incorporated into the undirected graph case, as has been shown in the literature [25]. One needs however to add a linear consensus expression of the form $c\mathbf{e}_\beta$, where c is negative, to the plant input. Then under the assumption that $\mathbf{f}$ is Lipschitz continuos and c is sufficiently negative, stability can be shown.

V. General Graphs

In this section we will first review the concept of graph balancing. Then, we will present a proof of consensus when the reference input r is globally known, followed by analysis when r is unknown to all agents.

A. Node Balancing Weights and Balanced Graphs

Definition 1. A diagonal matrix D is an *output balancing* of the graph $\mathcal{G}$ if $\mathbf{1}^\mathsf{T} D\mathcal{L}(\mathcal{G}) = \mathbf{0}^\mathsf{T}$.

The following lemma appears frequently in the text. It can be derived directly from the PF Theorem, but one of the first to explicitly use this lemma in a constructive fashion with regard to synchronization on directed graphs was Chai Wah Wu in the following three papers all appearing in 2005 [26–28]. Interestingly, Wu was tackling some of the very same problems that were being addressed in [21], but where the graphs were not a-priori assumed to be balanced.

Lemma 1. *There always exists an output balancing $D \in \mathbb{D}^n_{>0}$ for the in-degree laplacian $\mathcal{L}(\mathcal{G})$ of a strongly connected graph $\mathcal{G}$.*

Corollary 1. *If $D \in \mathbb{D}^n_{>0}$ is an output balancing of the graph $\mathcal{G}$ then $\mathcal{L}(\mathcal{G})^\mathsf{T} D + D\mathcal{L}(\mathcal{G}) \succeq 0$ with a single simple eigenvalue at* 0*, and with the corresponding eigenvector in $\mathbb{R}^n_{\mathbf{1}}$.*

A proof of Lemma 1 can be found in the Appendix. The corollary follows from [21, Theorem 7]. Per our construction the original graphs $\mathcal{G}$ are unweighted, however the above lemma also applies when the initial graph has arbitrary positive weights as well.

Lemma 2. *If $D \in \mathbb{D}^n_{>0}$ is an output balancing of a strongly connected graph $\mathcal{G}$ and $\mathcal{C} \in \mathbb{D}^n_{\geq 0}$ such that at least one of the diagonal elements is nonzero, then it follows that $\mathcal{L}(\mathcal{G})^\mathsf{T} D + D\mathcal{L}(\mathcal{G}) + D\mathcal{C} \succ 0$.*

Proof. From Corollary 1 it follows that there exists an output balancing matrix $D \in \mathbb{D}^n_{>0}$ such that $x^\mathsf{T}(\mathcal{L}(\mathcal{G})^\mathsf{T} D + D\mathcal{L}(\mathcal{G}))x > 0 \quad \forall x \notin \mathbb{R}^n_{\mathbf{1}}$. Given that $D \in \mathbb{D}^n_{>0}$ and at least one of the diagonal elements in $\mathcal{C}$ is nonzero, it follows that $x^\mathsf{T} D\mathcal{C}x > 0 \quad \forall x \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$. Combining these two facts it follows that $\mathcal{L}(\mathcal{G})^\mathsf{T} D + D\mathcal{L}(\mathcal{G}) + D\mathcal{C}$ is positive definite. □

Corollary 1 and Lemma 2 illustrate the fact that the diagonal solutions to the Lyapunov equation, which exist for M- and Metzler matrices, can in fact be chosen to coincide with the output balancing of the underlying graph. For details on diagonal solutions to the Lyapunov equation see the classic text [1, Chapter 6 Theorem 2.3, H_{24}], and [12, 13].

B. DACS with Global Knowledge of $r(t)$

We are now ready to discuss the stability of adaptive synchronization on general connected graphs with at least a single node targeted by the reference model. For symmetric graphs we did not implement pinning control, but in this section we will use an explicit synchronizing input following the pinning control methodology. Similar to [3] we will use the Schur Complement to prove stability, but here our block structures are different than those in [3] and we will not be assuming a-priori that the plant states are bounded. The control input in this section takes the form

$$u_i = \hat{k}_i x + r + c \sum_{j \in \mathcal{N}(i)} (x_i - x_j) + c(x_i - x_\mathrm{m})_{i \in \mathcal{T}}. \quad (13)$$

Comparing (13) to (2) we now use r in the input and thus we no longer need the estimate $\hat{r}$. In addition we have added the linear consensus input which combines local errors and the reference model trajectory if the node is a target node. Also, we now have a tuning parameter c which controls the strength of the error feedback. Writing (13) more compactly we have

$$\mathbf{u} = \hat{\mathbf{K}}\mathbf{x} + \mathbf{r} + \mathbf{C}\mathbf{e}_\beta \quad (14)$$

where $\mathbf{C} = cI_{n\times n}$. The update for the adaptive parameter $\hat{\mathbf{k}}$ is now defined via a projection algorithm as [18, 22]

$$\dot{\hat{\mathbf{k}}} = \mathsf{proj}_\infty(-\mathbf{x} \circ \mathbf{e}_\beta, \hat{\mathbf{k}}, k_{\max}) \quad (15)$$

where it is assumed implicitly that an upper bound on k_i for all i is known, which we will call $k_{\max}$, then it follows that $|\hat{k}_i| \leq k_{\max}$ for all i by the projection based update law. Essentially projection is needed here because the coupling strength c is chosen to overwhelm the feedback gain error.

With the control input in (14), the plant and the error dynamics can be written compactly as

$$\dot{\mathbf{x}} = \mathbf{A}_\mathrm{m}\mathbf{x} + \tilde{\mathbf{K}}\mathbf{x} + \mathbf{r} + \mathbf{C}\mathbf{e}_\beta \quad (16)$$

$$\dot{\mathbf{e}} = \mathbf{A}_\mathrm{m}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \mathbf{C}\mathbf{e}_\beta. \quad (17)$$

The following definitions will be used in the subsequent theorem.

$$\begin{aligned} Q_1 &\triangleq (\mathcal{L}^\mathsf{T} + 2\mathcal{M})D + D(\mathcal{L} + 2\mathcal{M}) \\ \lambda_2(\mathcal{L}, D) &\triangleq \min_{i, \lambda_i \neq 0} \lambda_i(\mathcal{L}^\mathsf{T} D + D\mathcal{L}) \\ \lambda_{\min}(Q) &\triangleq \min_i \lambda_i(Q) \\ d_{\max} &\triangleq \max_i d_i, \quad \text{and} \quad d_{\min} \triangleq \min_i d_i. \end{aligned} \quad (18)$$

Stability will now be shown for the dynamics in this subsection when the error feedback consensus tuning parameter c is less than or equal to the the negative scalar

$$c^* \triangleq -\frac{4 d_{\max} k_{\max}}{|a_m| d_{\min} \lambda_2(\mathcal{L}, D)}. \quad (19)$$

Theorem 2. *For the dynamics in* (16)*, the reference model in* (7)*, the error dynamics in* (17)*, and the update law in* (15)*, communicating over a directed graph $\mathcal{G}$ with the target nodes denoted by $\mathcal{M}$ and satisfying assumptions 1, 2 and 3′, if $c \leq c^*$, then all states are uniformly bounded and* $\lim_{t\to\infty} \mathbf{e}(t) = \mathbf{0}$.

Arguments for the proof will be given in two parts. First analysis is performed for when $x \notin \mathbb{R}^n_{\mathbf{1}}$ and then for when $x \in \mathbb{R}^n_{\mathbf{1}}$. Many mathematical operations that follow will exploit the fact that D and $\mathcal{M}$ commute as well as the fact that D is an output balancing.

1) The State Space $x \notin \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$: Consider the Lyapunov candidate[3]

$$V(\mathbf{e}) = \mathbf{e}^\mathsf{T} D\mathbf{e} \quad (20)$$

Differentiating along the system dynamics it follows that

$$\dot{V} = \mathbf{e}^\mathsf{T} D(\mathbf{A}_\mathrm{m}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \mathbf{C}\mathbf{e}_\beta) + (\mathbf{A}_\mathrm{m}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \mathbf{C}\mathbf{e}_\beta)^\mathsf{T} D\mathbf{e}$$

Expanding $\mathbf{e}_\beta$ and recalling that $\mathbf{A}_\mathrm{m} = a_\mathrm{m} I_{n\times n}$ it follows that

$$\begin{aligned} \dot{V} = {} & 2a_m \mathbf{e}^\mathsf{T} D\mathbf{e} + \mathbf{e}^\mathsf{T} D\tilde{\mathbf{K}}\mathbf{x} + \mathbf{x}^\mathsf{T}\tilde{\mathbf{K}}D\mathbf{e} \\ & + c\mathbf{e}^\mathsf{T}((\mathcal{L}^\mathsf{T} + \mathcal{M})D + D(\mathcal{L} + \mathcal{M}))\mathbf{e}. \end{aligned} \quad (21)$$

Dividing the expression $\mathbf{e}^\mathsf{T}((\mathcal{L}^\mathsf{T} + \mathcal{M})D + D(\mathcal{L} + \mathcal{M}))\mathbf{e}$ into two equal parts, and exploiting the fact that D is an output balancing it follows that

$$\begin{aligned} & \mathbf{e}^\mathsf{T}((\mathcal{L}^\mathsf{T} + \mathcal{M})D + D(\mathcal{L} + \mathcal{M}))\mathbf{e} = \\ & \quad \tfrac{1}{2}\mathbf{e}^\mathsf{T}((\mathcal{L}^\mathsf{T} + 2\mathcal{M})D + D(\mathcal{L} + 2\mathcal{M}))\mathbf{e} \\ & \quad + \tfrac{1}{2}\mathbf{x}^\mathsf{T}(\mathcal{L}^\mathsf{T} D + D\mathcal{L})\mathbf{x} \end{aligned} \quad (22)$$

Rearranging terms, recalling that $\mathbf{A}_\mathrm{m} = a_\mathrm{m} I_{n\times n}$, using the definition of Q_1 in (18) and the definition of $\mathbf{e}_\beta$ in (9) it follows that

$$\dot{V} = \begin{bmatrix} \mathbf{e}^\mathsf{T} & \mathbf{x}^\mathsf{T} \end{bmatrix} \begin{bmatrix} 2a_\mathrm{m} D + \frac{c}{2} Q_1 & D\tilde{\mathbf{K}} \\ \tilde{\mathbf{K}}D & \frac{c}{2}(\mathcal{L}^T D + D\mathcal{L}) \end{bmatrix} \begin{bmatrix} \mathbf{e} \\ \mathbf{x} \end{bmatrix} \quad (23)$$

[3] This may seem strange, as we do not have quadratic cost in terms of $\tilde{\mathbf{k}}$. Indeed this term could be included as well. We explicitly do not include it here, just to show that in general our proof does not exploit the fact that learning is occurring. This continues to be an issue in DACS over directed graphs. See Remark 2 for the same analysis with the adaptive parameter costs explicitly included in the Lyapunov function.

when $\mathbf{x} \notin \mathbb{R}^n_{\mathbf{1}}$. When $\mathbf{x} = \mathbf{0}$ it follows that $\dot{V} = \mathbf{e}^\mathsf{T}(2a_\mathrm{m} + \frac{c}{2}Q_1)\mathbf{e}$. Given the fact that $c \leq c^*$ where c^* is defined in (19) (and recall that $a_\mathrm{m} < 0$ as well) it follows by the Schur Complement that $\dot{V} < 0$ when $\mathbf{x} \notin \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ unless $\mathbf{e} = \mathbf{0}$.

2) The state space $\mathbf{x} \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$: If D is a graph balancing then $\mathbf{x} \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ and the expression in (23) would become

$$\dot{V} = \begin{bmatrix}\mathbf{e}^\mathsf{T} & \mathbf{x}^\mathsf{T}\end{bmatrix}\begin{bmatrix}2a_\mathrm{m}D + \frac{c}{2}Q_1 & D\tilde{\mathbf{K}} \\ \tilde{\mathbf{K}}D & \mathbf{0}_{n\times n}\end{bmatrix}\begin{bmatrix}\mathbf{e} \\ \mathbf{x}\end{bmatrix}.$$

We will now show however that x can not remain in $\mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ for any finite amount of time. We will first investigate the adaptive gains and then the plant states themselves. If $x \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ then from the definition of the Laplacian and our update law it follows that

$$\begin{aligned} \dot{k}_i &= 0, \quad i \notin \mathcal{T} \\ \dot{k}_j &= -x_j(x_j - x_\mathrm{m}), \quad j \in \mathcal{T} \end{aligned} \tag{24}$$

Let t_1 be a time such that $\mathbf{x}(t_1) \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$, and if $\mathbf{x}(t) \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ for a nonzero amount of time $t - t_1 \neq 0$ with $t \geq t_1$ then

$$\dot{x}_i(t) = a_\mathrm{m}x_i(t) + r + x_i(t)\int_{t_1}^{t} 0 \,\mathrm{d}\tau$$

$$\dot{x}_j(t) = a_\mathrm{m}x_j(t) + r + x_j(t)\int_{t_1}^{t} -x_j(\tau)e_j(\tau)\,\mathrm{d}\tau + ce_j(t)$$

for $i \notin \mathcal{T}$ and $j \in \mathcal{T}$. Letting $\kappa = \tilde{k}_i(t_1)$ which remains constant for all $i \notin \mathcal{T}$ and noting that $x_i = x_j$ necessarily when $\mathbf{x} \in \mathbb{R}^n_{\mathbf{1}}$ it follows that in order for $\mathbf{x}$ to remain in this special subspace it necessarily follows that

$$x_i(t)\kappa = -x_i(t)\int_{t_1}^{t} x_i(\tau)e_i(\tau)\,\mathrm{d}\tau + ce_i(t), \tag{25}$$

note that all subscripts are now i. The expression in (25) only holds when $e_i = 0$, see Appendix B, and thus $\mathbf{x}$ can not remain in the special linear subspace $\mathbb{R}^n_{\mathbf{1}}$ unless $\mathbf{e} = 0$. □

Remark 2. Note that we did not include the parameter error $\tilde{\mathbf{k}}$ in the Lyapunov candidate. It was not necessary to illustrate stability. Due to the fact that a projection algorithm is used $\mathbf{k}$, and thus $\tilde{\mathbf{k}}$ in return, are bounded. If indeed our Lyapunov candidate was

$$V(\mathbf{e}, \tilde{\mathbf{k}}) = \mathbf{e}^\mathsf{T}D\mathbf{e} + \tilde{\mathbf{k}}^\mathsf{T}D\tilde{\mathbf{k}} \tag{26}$$

then the derivative of $\dot{V}$, after substitution of the update law, would have the extra terms

$$-\mathbf{e}^\mathsf{T}(\mathcal{L} + \mathcal{M})^\mathsf{T}D\tilde{\mathbf{K}}\mathbf{x} - \mathbf{x}^\mathsf{T}\tilde{\mathbf{K}}D(\mathcal{L} + \mathcal{M})\mathbf{e}$$

on the right hand side. Exploiting the fact that D is an output balancing, the above expression is equivalent to

$$-\mathbf{x}^\mathsf{T}\mathcal{L}^\mathsf{T}D\tilde{\mathbf{K}}\mathbf{x} - \mathbf{x}^\mathsf{T}\tilde{\mathbf{K}}D\mathcal{L}\mathbf{x} - \mathbf{e}^\mathsf{T}\mathcal{M}^\mathsf{T}D\tilde{\mathbf{K}}\mathbf{x} - \mathbf{x}^\mathsf{T}\tilde{\mathbf{K}}D\mathcal{M}\mathbf{e}.$$

Therefore, the 2×2 block matrix in (23) would be replaced with

$$\begin{bmatrix}2a_\mathrm{m}D + \frac{c}{2}Q_1 & D(I_{n\times n} - \mathcal{M})\tilde{\mathbf{K}} \\ \tilde{\mathbf{K}}D(I_{n\times n} - \mathcal{M}) & \frac{c}{2}(\mathcal{L}^TD + D\mathcal{L}) - (\mathcal{L}^TD\tilde{\mathbf{K}} + \tilde{\mathbf{K}}D\mathcal{L})\end{bmatrix}$$

which for c sufficiently negative and $\mathbf{x}$ not in $\mathbb{R}^n_{\mathbf{1}} \setminus 0$ is negative definite.

C. DACS with unknown constant input r

In this subsection we will not assume global knowledge of r and thus the input is now given as

$$\mathbf{u} = \hat{\mathbf{K}}\mathbf{x} + \hat{\mathbf{r}} + \mathbf{C}\mathbf{e}_\beta$$

where the term $\hat{\mathbf{r}}$ appears in place of $\mathbf{r}$ in (14). The update law for $\hat{\mathbf{k}}$ remains the same as in the previous section and the update term for $\hat{\mathbf{r}}$ is defined as

$$\dot{\hat{\mathbf{r}}} = -\mathbf{e}_\beta + c\mathcal{L}\hat{\mathbf{r}}. \tag{27}$$

The term $c\mathcal{L}\hat{\mathbf{r}}$ illustrates the fact that agents share their local estimates $\hat{r}_i$ with their neighbors. For ease of analysis we scale the term $\mathcal{L}\hat{\mathbf{r}}$ with the same c that scales the error consensus feedback in the input. The dynamics in this section can be compactly written as

$$\dot{\mathbf{x}} = \mathbf{A}_\mathrm{m}\mathbf{x} + \tilde{\mathbf{K}}\mathbf{x} + \mathbf{r} + \tilde{\mathbf{r}} + \mathbf{C}\mathbf{e}_\beta \tag{28}$$

$$\dot{\mathbf{e}} = \mathbf{A}_\mathrm{m}\mathbf{e} + \tilde{\mathbf{K}}\mathbf{x} + \tilde{\mathbf{r}} + \mathbf{C}\mathbf{e}_\beta. \tag{29}$$

Theorem 3. *For the dynamics in* (28)*, the reference model in* (7)*, the error dynamics in* (29)*, and the update laws in* (15) *and* (27)*, communicating over a directed graph $\mathcal{G}$ with the target nodes denoted by $\mathcal{M}$ and satisfying assumptions 1, 2 and 3, if c is sufficiently negative, then all states are uniformly bounded and* $\lim_{t\to\infty}\mathbf{e}(t) = \mathbf{0}$.

Proof. Consider the Lyapunov candidate

$$V(\mathbf{e}, \tilde{\mathbf{r}}) = \mathbf{e}^\mathsf{T}D\mathbf{e} + \tilde{\mathbf{r}}^\mathsf{T}D\tilde{\mathbf{r}}. \tag{30}$$

The time derivative of $\mathbf{e}^\mathsf{T}D\mathbf{e}$ will now introduce two new terms in the derivative of the Lyapunov function that were not present in (23)

$$\mathbf{e}^\mathsf{T}D\tilde{\mathbf{r}} + \tilde{\mathbf{r}}D\mathbf{e}. \tag{31}$$

These two terms appear because of the presence of $\tilde{\mathbf{r}}$ in (29). The time derivative of $\tilde{\mathbf{r}}^\mathsf{T}D\tilde{\mathbf{r}}$ with the update law in (27) will introduce the following four terms in the Lyapunov derivative that are not present in (23)

$$-\mathbf{e}^\mathsf{T}\mathcal{B}^\mathsf{T}D\tilde{\mathbf{r}} - \tilde{\mathbf{r}}^\mathsf{T}D\mathcal{B}\mathbf{e} + c\tilde{\mathbf{r}}^\mathsf{T}D\mathcal{L}\hat{\mathbf{r}} + c\hat{\mathbf{r}}^\mathsf{T}\mathcal{L}^\mathsf{T}D\tilde{\mathbf{r}} \tag{32}$$

Given that $\mathbf{r}$ by definition is in $\mathbb{R}^n_{\mathbf{1}}$ it follows that (32) is equivalent to

$$-\mathbf{e}^\mathsf{T}\mathcal{B}^\mathsf{T}D\tilde{\mathbf{r}} - \tilde{\mathbf{r}}^\mathsf{T}D\mathcal{B}\mathbf{e} + c\tilde{\mathbf{r}}^\mathsf{T}D\mathcal{L}\tilde{\mathbf{r}} + c\tilde{\mathbf{r}}^\mathsf{T}\mathcal{L}^\mathsf{T}D\tilde{\mathbf{r}} \tag{33}$$

where we have replaced $\hat{\mathbf{r}}$ in the last two expressions with $\tilde{\mathbf{r}}$. Adding the terms in (31) and (33) to the derivative of $\dot{V}$ in (23) results in the following for the derivative of V in (30)

$$\dot{V} = \mathbf{z}^\mathsf{T}Q_2\mathbf{z}$$

where $\mathbf{z}^T = [\mathbf{e}^\mathsf{T},\, \mathbf{x}^\mathsf{T},\, \tilde{\mathbf{r}}]^\mathsf{T}$ and

$$Q_2 = \begin{bmatrix}2a_\mathrm{m}D + \frac{c}{2}Q_1 & D\tilde{\mathbf{K}} & (I_{n\times n} - \mathcal{B})^\mathsf{T}D \\ \tilde{\mathbf{K}}D & \frac{c}{2}(\mathcal{L}^TD + D\mathcal{L}) & \mathbf{0}_{n\times n} \\ D(I_{n\times n} - \mathcal{B}) & \mathbf{0}_{n\times n} & c(\mathcal{L}^TD + D\mathcal{L})\end{bmatrix}.$$

For c sufficiently negative and $\mathbf{x} \notin \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ it follows that $\dot{V}$ is negative definite. The analysis for the case when $\mathbf{x} \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ is not presented here, but similar steps to those carried out in the proof of Theorem 2 can be carried out here as well. □

VI. Simulation Examples

We now present simulation results for the control design presented in Sections V-C. The parameters for the plant were constructed as follows. The graph is a directed cycle with $|\mathcal{V}| = 10$ and a single target node. The reference model Jacobian $a_{\mathrm{m}} = -1$ and $r = 1$. The values a_i are drawn from a uniform distribution with values in $[0, 1]$, and thus all plants are initially unstable. The initial conditions of the plant $x_i(0)$ and $x_{\mathrm{m}}(0)$ are drawn from a normal distribution with mean 0 and variance 1. All the adaptive parameters start with an initial condition of 0. For the first simulation $c = -1$ and for the second $c = 0$. Note that while we proved stability for the controller in Theorem 3 when c is sufficiently negative, we did not prove stability of the controller when we do not have the linear consensus input or when the estimates of $\hat{r}_i$ are not shared locally. We did however prove stability for $c = 0$ when the graph is undirected in Theorem 1, but as already stated, these simulation results are for a directed graph. We will discuss this more later.

From Figure 1 we can see that with $c = -1$ the system is stable and $\mathbf{e}$ asymptotically approaches $\mathbf{0}$. An interesting phenomenon occurs however starting at about 5 seconds where there is a small burst in the error that lasts for about 10 seconds. This phenomenon has been observed elsewhere in the adaptive literature, see for instance [8, 9]. The phenomenon can be explained as follows. The use of error feedback into the plant can retard adaptation. So for instance, it is possible for the model following error to be small, because of the linear consensus input, while the local state feedback combined with the open-loop plant Jacobian $a_i + \hat{k}_i$ can be positive, [9, Section II].

From Figure 2 we can see that the system is stable even without the linear consensus input. While we were able to prove stability in the case of undirected graphs, we were not able to prove stability for directed graphs with $c = 0$. Future research will look to close this theoretical gap.

VII. Conclusions

In this work it was shown that synchronization can be achieved for our plants solely through adaptation and without a linear consensus input when the graph is undirected. We also explored DACS and presented proofs that did not require a-priori knowledge of a plant state bound. Our analysis exploited the existence of output balancing weights for strongly connected graphs and the adaptive laws were local without any global network knowledge. A proof of Theorem 1 for directed graphs is an important open problem. We close by conjecturing that indeed Theorem 1 holds even for directed graphs.

References

[1] Abraham Berman and Robert J Plemmons, *Nonnegative matrices*, SIAM, 1994.

[2] Norman Biggs, *Algebraic graph theory*, Cambridge university press, 1993.

[3] Abhijit Das and Frank L Lewis, *Distributed adaptive control for synchronization of unknown nonlinear networked systems*, Automatica **46** (2010), no. 12, 2014–2021.

[4] Pietro DeLellis, Mario di Bernardo, and Franco Garofalo, *Adaptive pinning control of networks of circuits and systems in lur'e form*, Circuits and Systems I: Regular Papers, IEEE Transactions on **60** (2013), no. 11, 3033–3042.

[5] J.A. Fax and R.M. Murray, *Information flow and cooperative control of vehicle formations*, Automatic Control, IEEE Transactions on **49** (2004Sept), no. 9, 1465–1476.

[6] Hu Gang and Qu Zhilin, *Controlling spatiotemporal chaos in coupled map lattice systems*, Physical Review Letters **72** (1994), no. 1, 68.

[7] Donald T Gavel and Dragoslav D Šiljak, *Decentralized adaptive control: structural conditions for stability*, Automatic Control, IEEE Transactions on **34** (1989), no. 4, 413–426.

[8] Travis E. Gibson, *Closed-loop reference model adaptive control: with application to very flexible aircraft*, Ph.D. Thesis, 2014.

[9] Travis E. Gibson, *Adaptation and Synchronization over a Network: Stabilization without a Reference Model*, Submitted to CDC (2016), available at http://www.mit.edu/~tgibson/papers/gibson_cdc16_stabilization.pdf.

[10] Christopher David Godsil, Gordon Royle, and CD Godsil, *Algebraic graph theory*, Vol. 207, Springer New York, 2001.

[11] RO Grigoriev, MC Cross, and HG Schuster, *Pinning control of spatiotemporal chaos*, Physical Review Letters **79** (1997), no. 15, 2795.

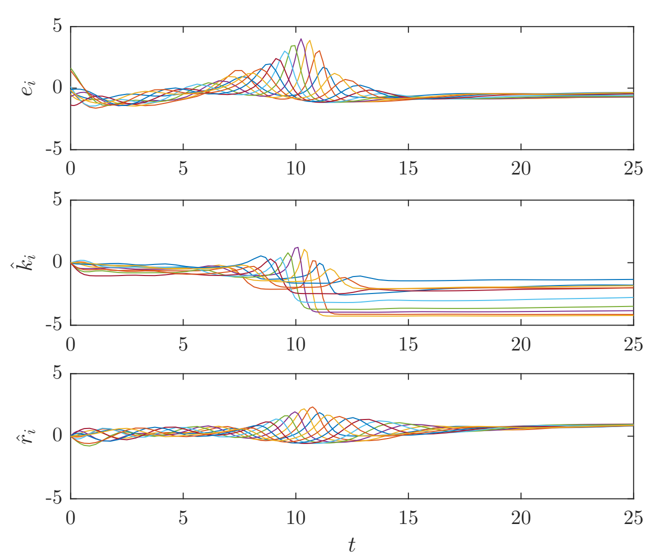


Fig. 1. Trajectories for the plant and controller from Section V-C, $c = -1$.

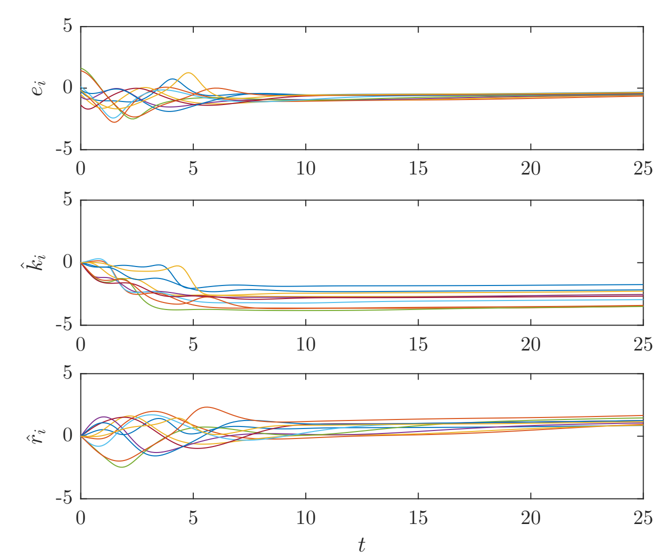


Fig. 2. Trajectories for the plant and controller from Section V-C, $c = 0$.

[12] Daniel Hershkowitz and Hans Schneider, *Lyapunov diagonal semistability of real h-matrices*, Linear algebra and its applications **71** (1985), 119–149.
[13] Daniel Hershkowitz and Hans Schneider, *Scalings of vector spaces and the uniqueness of lyapunov scaling factors*, Linear and Multilinear Algebra **17** (1985), no. 3-4, 203–226.
[14] P.A. Ioannou and J. Sun, *Robust adaptive control*, Dover, 2013.
[15] Petros A Ioannou, *Decentralized adaptive control of interconnected systems*, Automatic Control, IEEE Transactions on **31** (1986), no. 4, 291–298.
[16] Ali Jadbabaie, Jie Lin, et al., *Coordination of groups of mobile autonomous agents using nearest neighbor rules*, Automatic Control, IEEE Transactions on **48** (2003), no. 6, 988–1001.
[17] Sandeep Jain and Farshad Khorrami, *Decentralized adaptive control of a class of large-scale interconnected nonlinear systems*, Automatic Control, IEEE Transactions on **42** (1997), no. 2, 136–154.
[18] E. Lavretsky, T. E. Gibson, and A. M. Annaswamy, *Projection operator in adaptive systems*, arXiv:1112.4232 (2011).
[19] H. Minc, *Nonnegative matrices*, Wiley Series in Discrete Mathematics and Optimization, 1988.
[20] K. S. Narendra and A. M. Annaswamy, *Stable adaptive systems*, Dover, 2005.
[21] Reza Olfati-Saber and Richard M Murray, *Consensus problems in networks of agents with switching topology and time-delays*, Automatic Control, IEEE Transactions on **49** (2004), no. 9, 1520–1533.
[22] J. Pomet and L. Praly, *Adaptive nonlinear regulation: Estimation from the lyapunov equation*, IEEE Trans. Automat. Contr. **37** (1992June), no. 6.
[23] Jeffrey T Spooner and Kevin M Passino, *Decentralized adaptive control of nonlinear systems using radial basis neural networks*, Automatic Control, IEEE Transactions on **44** (1999), no. 11, 2050–2057.
[24] John N Tsitsiklis, *Problems in decentralized decision making and computation*, Ph.D. Thesis, 1984.
[25] LFR Turci, P De Lellis, EEN Macau, M Di Bernardo, and MMR Simões, *Adaptive pinning control: A review of the fully decentralized strategy and its extensions*, The European Physical Journal Special Topics **223** (2014), no. 13, 2649–2664.
[26] Chai Wah Wu, *Algebraic connectivity of directed graphs*, Linear and Multilinear Algebra **53** (2005), no. 3, 203–223.
[27] Chai Wah Wu, *On rayleigh–ritz ratios of a generalized laplacian matrix of directed graphs*, Linear algebra and its applications **402** (2005), 207–227.
[28] Chai Wah Wu, *Synchronization in networks of nonlinear dynamical systems coupled via a directed graph*, Nonlinearity **18** (2005), no. 3, 1057.
[29] Hui Yu and Xiaohua Xia, *Adaptive consensus of multi-agents in networks with jointly connected topologies*, Automatica **48** (2012), no. 8, 1783–1790.

APPENDIX A
PROOFS OF TECHNICAL LEMMAS NOT CONTAINED IN THE MAIN TEXT

Proof of Lemma 1. Let $h \in \mathbb{R}_{>0}$ be the maximum in-degree over all n nodes in the network, then the matrix $F = hI_{n\times n} - \mathcal{L}$ is a non-negative matrix, and thus F^T is non-negative as well. Also, given that the graph is strongly connected it follows that there exists a unique largest eigenvalue of 0 for $-\mathcal{L}$ with algebraic multiplicity 1. Therefore the PF eigenvalue of F^T is h. The PF Theorem also states that the eigenvector $p \in \mathbb{R}^n$, $F^\mathsf{T} p = hp$, is positive. Subtracting $cI_{n\times n}$ from both sides it follows that $-\mathcal{L}^\mathsf{T} p = \mathbf{0}$. Therefore, the left eigenvector for the eigenvalue 0 of $\mathcal{L}$ is p. If we define a weighted graph as the triple $\tilde{\mathcal{G}} = (\mathcal{V}, \mathcal{E}, \tilde{\mathcal{A}})$ where the weighted adjacency matrix is defined as $\tilde{\mathcal{A}} = \mathsf{diag}(p)\mathcal{A}(\mathcal{G})$, then it follows that the Laplacian of the weighted graph, defined as $\tilde{\mathcal{L}} = \mathsf{diag}(p)(\mathcal{D}(\mathcal{G}) - \mathcal{A}(\mathcal{G}))$ satisfies the following two properties $\mathbf{1}^\mathsf{T}\tilde{\mathcal{L}} = \mathbf{0}^\mathsf{T}$ and $\tilde{\mathcal{L}}\mathbf{1} = \mathbf{0}$. From [21, Theorem 6] it follows that $\tilde{\mathcal{G}}$ is therefore balanced. Setting $D = \mathsf{diag}(p)$ completes the proof. $\square$

APPENDIX B
DISCUSSION REGARDING THE SECOND PART OF THE PROOF OF THEOREM 2

In this section we will continue the discussion from V-B2 picking up just after Equation (25) where $i \notin \mathcal{T}$. Given that $x_i \neq 0$ when $\mathbf{x} \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ the expression in (25) can be divided by x_i, resulting in

$$\kappa = -\int_{t_1}^{t} x_i(\tau)e_i(\tau)\,\mathrm{d}\tau + ce_i(t)x_i^{-1},$$

Differentiating and suppressing the explicit time dependencies, we have

$$0 = -x_i e_i + c\dot{e}_i x_i x_i^{-2} - ce_i\dot{x}_i x_i^{-2}.$$

Multiplying both sides by x_i^2 we have

$$0 = -x_i^3 e_i + c\dot{e}_i x_i - ce_i\dot{x}_i.$$

Substituting the dynamics for x_i and noting that $\tilde{k} = \kappa$ is a constant and $\mathbf{x} \in \mathbb{R}^n_{\mathbf{1}} \setminus \mathbf{0}$ it follows that

$$0 = -x_i^3 e_i + c\dot{e}_i x_i - ce_i(a_\mathrm{m} x_i + \kappa x_i + r).$$

Dividing by cx_i and moving $\dot{i}$ to the lefthand side, it follows that

$$\dot{e}_i = \frac{1}{c}x_i^2 e_i + e_i\left(a_\mathrm{m} + \kappa + \frac{r}{x_i}\right).$$

Taking $e_i a_\mathrm{m}$ out of the parenthesis and expanding the term $\kappa e_i = \kappa x_i - x_\mathrm{m}$

$$\dot{e}_i = a_\mathrm{m} e_i + \kappa x_i + \frac{1}{c}x_i^2 e_i - \kappa x_\mathrm{m} + \frac{re_i}{x_i}. \tag{34}$$

From the fact that $\mathbf{x}$ and thus $\mathbf{e}$ are necessarily in the special linear subspace $\mathbb{R}^n_{\mathbf{1}}$ it follows that $x_i = x_j$ and $e_i = e_j$ for all i, j. Therefore, the dynamics in (34) must also satisfy the error dynamics for the agents $i \notin \mathcal{T}$ and thus e_i must also satisfy

$$\dot{e}_i = a_\mathrm{m} e_i + \kappa x_i. \tag{35}$$

Equating (34) and (35) it follows that

$$\frac{1}{c}x_i^2 e_i - \kappa x_\mathrm{m} + \frac{re_i}{x_i} = 0$$

which is not a solution to the dynamics. Therefore, in order for $\mathbf{x}$ to remain in the special linear subspace of interest, $\mathbf{e}$ must remain zero over that same period of time, and only then can the two equations above (25) (for the dynamics $\dot{x}_i$ and $\dot{x}_j$) remain equal.