

Performance and Design of Consensus on Matrix-Weighted and Time Scaled Graphs

Dillon R. Foight, *Student Member, IEEE*, Mathias Hudoba de Badyn, *Member, IEEE*,
and Mehran Mesbahi, *Fellow, IEEE*

Abstract—In this paper, we consider the \mathcal{H}_2 -norm of networked systems with multi-time scale consensus dynamics and vector-valued agent states. This allows us to explore how measurement and process noise affect consensus on matrix-weighted graphs by examining edge-state consensus. In particular, we highlight an interesting case where the influences of the weighting and scaling on the \mathcal{H}_2 norm can be separated in the design problem. We then consider optimization algorithms for updating the time scale parameters and matrix weights in order to minimize network response to injected noise. Finally, we present an application to formation control for multi-vehicle systems.

I. INTRODUCTION

DYNAMICAL systems operating over networks appear in many natural and cyber-physical systems. A popular model of such dynamic processes is *consensus*, which has been used for a variety of control and estimation applications, such as multi-agent systems [2]–[4], robotics, [5], [6], and distributed Kalman filtering [7]. A natural question is how the underlying network topology affects the behavior of the dynamics operating over the network, motivated by the fact that notions of performance and control can be directly related to graph theoretic properties of the network. Of particular interest for this work is the \mathcal{H}_2 system norm, which for networked dynamical systems can be interpreted as a measure of how input energy is attenuated over the network, or how noise drives deviations from the natural consensus state [8].

In light of these interpretations, several works have characterized the \mathcal{H}_2 performance for consensus networks. In [9], [10], the performance of leader-follower networks is considered, and algorithms for rewiring and reweighting the network

for optimal noise rejection are discussed. Similarly, [11]–[13] have utilized the \mathcal{H}_2 -norm as a measure of coherence in networks and considered problems such as local feedback laws and leader selection to promote coherence. Most relevant to the present contribution, the works [14], [15] investigated the impact of cycles on the \mathcal{H}_2 performance of noise-driven consensus networks. Examining networks under noise inputs is especially important for network security and resiliency [16], which motivates considering the minimization of the \mathcal{H}_2 system norm in order to promote resilience to external noise inputs (which could be adversarial in nature).

A common simplifying assumption within the literature is that the agent dynamics are identical single or double integrators. An extension of this simplified model is to consider the case where individual agents' states evolve at differing rates, motivated by similar formulations in areas such as electrical networks [17] and power networks with generator inertia [18]. The analysis of multi-scale problems has historically offered techniques for formal description and controller synthesis for complex systems [19]; analysis of a multi-scale consensus model can increase the applicability of the consensus protocol to a wider range of real-world systems. As such, there is a growing body of literature addressing the complications that arise from the integrating multiple time scales into consensus, starting with the discussion of the consensus value for multi-rate integrators in [20]. Issues such as convergence [21], stability [22], [23], controller design [24], [25], formation control [26], as well as single-influenced consensus performance [27] have since been addressed for such multi-scale networks. Considerations of optimizing the time scale parameters have been rare, possibly in part due to complications such as bilinear matrix equations that can naturally arise in multi-time scale formulations [22]. An advantage of the noise driven edge consensus formulation presented here is that, under specific assumptions, the \mathcal{H}_2 performance is characterized in a form that allows for streamlined optimization formulations.

Additionally, most of the existing literature on control and performance analysis for consensus consider the case where each node in the network has a *scalar* state, or can be decomposed as such. Recently, several extensions to the case of *vector*-valued node states have been proposed; such consensus protocols are characterized by having graphs with *matrix-valued edge weights*. System-theoretic properties of matrix-valued weighted consensus networks, as well as applications to bearing-constrained formation control have been considered

Manuscript received November 22, 2019; revised March 18, 2020 and May 12, 2020; accepted May 26, 2020. This work was supported by the National Science Foundation (grant number DGE- 1762114), the National Sciences and Engineering Research Council of Canada (funding reference number CGSD2-502554-2017), the U.S. Army Research Laboratory and the U.S. Army Research Office (contract number W911NF-13-1-0340), and the U.S. Air Force Office of Scientific Research (grant number FA9550-16-1-0022). (D. Foight and M. Hudoba de Badyn are co-first authors.) (Corresponding author: D. Foight.) A preliminary version of this work appears in the Proceedings of the 58th IEEE Conference on Decision and Control [1].

The authors are with the William E. Boeing Department of Aeronautics and Astronautics at the University of Washington, Seattle, WA 98195, USA. MHdB is now with the Automatic Control Laboratory at ETH Zürich, 8092 Zürich, Switzerland. e-mails: {dfoight, hudomath, mesbahi}@uw.edu.

in [28]. Matrix weighted graphs have also been utilized in applications such as estimation [29], [30], spacecraft formation control [31], the control of coupled oscillators [32], and opinion dynamics [33], [34].

In this paper, we consider design problems networks using the \mathcal{H}_2 system norm as a metric of network's resiliency. In particular, we examine consensus on matrix-weighted and time scaled networks, with process and measurement noise. Drawing from the work in [14], we transform the general consensus problem to one over the edge states, and consider \mathcal{H}_2 -optimal design of the agent time scales and matrix edge weights. The main contributions of the paper are the formulation of the consensus problem for matrix-weighted networks with time scales, a transformation yielding the dynamics of the edge states for this consensus problem, a method for separating the contributions of edge weighting and node time scales on the \mathcal{H}_2 performance, design problems for time scale and edge weight assignment, and an application to formation control.

The paper is organized as follows. In §II, we outline the necessary notation and graph theory used in the paper, and introduce the problem setup in §III. The main results are divided into the formulation of the \mathcal{H}_2 performance metric in §IV-A, followed by design problems for time scale and edge weights in §IV-C and §IV-B. We apply the distributed design problems to a flocking model on second-order consensus in §V. For readability, we relegate some technical proofs to the Appendix.

II. MATHEMATICAL PRELIMINARIES

Here, we provide a brief overview of the notation and terminology used throughout the paper, as well as relevant graph theoretic concepts. Column vectors are denoted as $x \in \mathbb{R}^n$. Special vectors include the vector of all ones (zeros), $\mathbf{1}$ ($\mathbf{0}$), the vector of diagonal elements in a matrix, $\text{diag}(M)$, and Euclidean basis vectors, $e_i \in \mathbb{R}^n$, where the i denotes the index of the non-zero element. Matrices will be denoted as $M \in \mathbb{R}^{m \times n}$. The $(k \times k)$ identity matrix will be denoted by I_k . The set of positive-(semi) definite matrices will be denoted by \mathcal{S}_{++}^n (\mathcal{S}_+^n). The Kronecker product of two matrices A, B is denoted by $A \otimes B$, and the operation $\text{Blkdiag}(A_1, \dots, A_n)$ yields the matrix which has the matrices A_1, \dots, A_n on its diagonal. For matrices A and B , $A \preceq B$ implies $B - A \in \mathcal{S}_+$. Time-dependent quantities will be denoted as $x(t)$.

This paper considers dynamics governed by the interconnections of multi-rate, single integrator agents over connected, matrix-weighted communication graphs. In this formulation, we can consider a graph object defined by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where \mathcal{V} is the set of agents (nodes), and \mathcal{E} is the set of edges. Associated with the graph are \mathcal{W} , a set of matrix edge weights, and \mathcal{T} , a set of time scaling factors for agents' states.

Individual agents states are vector-valued, $x \in \mathbb{R}^k$, and each agent will be indexed by subscripts, e.g. $\nu_i \in \mathcal{V}$ to represent the i -th agent where $1 \leq i \leq |\mathcal{V}|$. If $(i, j) \in \mathcal{E}$, the i -th and j -th agents are connected by an edge ($i \sim j$), and they are referred to as adjacent agents. For a given agent, i , $N(i) = \{j \mid i \sim j \forall j \in \mathcal{V}\}$ denotes the neighbors of i , and $\deg(\nu_i) = |N(i)|$ denotes the unweighted degree of i . The k values comprising

an agent's state will be referred to as *substates*, and the j -th substate of the i -th agent is denoted as $x_{i,j}$.

As a consequence of considering vector-valued agent states, edges between agents are matrix-valued, which allows for a notion of dynamical coupling between neighboring agent states. Such matrix-valued weights will be denoted $W_e \in \mathcal{S}_{++}^k$, and so $\mathcal{W} = \{W_e \mid e \in \mathcal{E}\}$. The *weight matrix* \mathbf{W} is a $k|\mathcal{E}| \times k|\mathcal{E}|$ blockwise diagonal matrix containing the weights W_{ij} of each edge e . As in graphs with scalar agent states, the edge set can be ordered by a mapping, $\kappa(\cdot)$, such that $l = \kappa(ij)$ if and only if $(i, j) \in \mathcal{E}$. By this mapping, we can denote the weight on edge $\kappa(ij)$ by W_l or W_{ij} , interchangeably. Furthermore, we also assume that each individual substate of each agent can operate on an independent time scale. Thus, for each node i and set of corresponding time scales $T_i = \{\epsilon_{i,1}, \dots, \epsilon_{i,k}\} \in \mathcal{T}$, we associate the *time scale matrix* $E_i = \text{diag}(\epsilon_{i,1}, \dots, \epsilon_{i,k}) \in \mathcal{S}_{++}^k$. Note that the positive-definiteness of E_i is equivalent to requiring that each $\epsilon_{i,k} > 0$.

The *incidence matrix* $D(\mathcal{G})$ is a $|\mathcal{N}| \times |\mathcal{E}|$ matrix with rows and columns indexed by the nodes and edges of \mathcal{G} , respectively. For each edge $l := (i, j)$, where i is the tail and j is the head, $D(\mathcal{G})_{il} = 1$ and $D(\mathcal{G})_{jl} = -1$, and we denote the edge vector for edge (i, j) by \mathbf{a}_{ij} (column of $D(\mathcal{G})$). If \mathcal{G} is undirected, by convention we write that $D(\mathcal{G})_{il} = 1$ and $D(\mathcal{G})_{jl} = -1$ for $i > j$. For the formulation of matrix-valued weights, we define $\mathbf{D}(\mathcal{G}) \triangleq D(\mathcal{G}) \otimes I_k$ and $\mathbf{a}_{ij} \triangleq \mathbf{a}_{ij} \otimes I_k$. The *weighted graph Laplacian* \mathcal{L}_w of an undirected graph \mathcal{G} can be defined thusly as $\mathcal{L}_w \triangleq \mathbf{D}(\mathcal{G})\mathbf{W}\mathbf{D}(\mathcal{G})^T = \sum_{ij \in \mathcal{E}} \mathbf{a}_{ij} W_{ij} \mathbf{a}_{ij}^T$. Equivalently, it can be defined blockwise with the $k \times k$ block whose rows are associated with the i th node and whose columns are associated with the j th node given by $\sum_{j \in N_i} W_{ij}$ if $i = j$, $-W_{ij}$ if $(i, j) \in \mathcal{E}$, and $\mathbf{0}_{k \times k}$ if $(i, j) \notin \mathcal{E}$.

III. NETWORK MODELS

In this section, we will describe a general formulation for consensus over a communication network with positive-definite edge weighting and agent time scaling, with a model for measurement and process noise. The scaled consensus problem is derived from considering a group of n multi-rate integrators [20], with zero-mean Gaussian process noise, $\omega_i(t)$ such that $\mathbb{E}[\omega_i(t)\omega_i(t)^T] = \Omega_i$, $\forall i \in \mathcal{N}$,

$$\begin{bmatrix} \epsilon_{i,1}\dot{x}_{i,1} \\ \vdots \\ \epsilon_{i,k}\dot{x}_{i,k} \end{bmatrix} = E_i \dot{x}_i(t) = u_i(t) + \omega_i(t), \quad (1)$$

where x_i is the vector state of the i -th agent, E_i is $\text{diag}[\epsilon_{i,1}, \dots, \epsilon_{i,k}]$, and u_i is the control input.

Suppose that communication between agents i and j is corrupted by zero-mean Gaussian noise v_{ij} , and let v_i denote the sum of all noise inputted into agent i from the connections to its neighbors in $N(i)$. Without loss of generality, we can assume that the covariance of v_i is given by $\mathbb{E}[v_i(t)v_i(t)^T] = \Gamma_i$. A weighted, decentralized feedback controller, with noise, that seeks to bring agents into consensus is given by,

$$\begin{aligned} u_i(t) &= \sum_{j \in N(i)} [\mathbf{W}_{ij}(x_j(t) - x_i(t)) + v_{ij}(t)] \\ \mathbf{u}(t) &= -\mathbf{D}(\mathcal{G})\mathbf{W}\mathbf{D}(\mathcal{G})^T \mathbf{x}(t) + \mathbf{D}(\mathcal{G})\mathbf{v}(t), \end{aligned} \quad (2)$$

where \mathbf{W} is the block-diagonal matrix of edge weights with properties detailed in §II. The vector \mathbf{u} is the stacked vector of control vectors u_i , and \mathbf{v} is the stacked vector of all measurement noises. Applying (2) to the system (1) with appropriate dimensions gives the general, time scaled and matrix weighted consensus problem with process and measurement noise,

$$\begin{aligned}\dot{\mathbf{x}}(t) &= -\mathbf{E}^{-1}\mathcal{L}_w(\mathcal{G})\mathbf{x}(t) + \begin{bmatrix} \mathbf{E}^{-1} & -\mathbf{E}^{-1}\mathbf{D}(\mathcal{G}) \end{bmatrix} \begin{bmatrix} \omega(t) \\ \mathbf{v}(t) \end{bmatrix} \\ \mathbf{z}(t) &= \mathbf{D}(\mathcal{G})^T \mathbf{x}(t)\end{aligned}\quad (3)$$

where $\mathcal{L}_w(\mathcal{G})$ is the weighted Laplacian matrix, and $\mathbf{E} = \text{Blkdiag}(E_1, \dots, E_n)$ is the full time scale matrix of \mathcal{G} . Here, we have introduced an output $\mathbf{z}(t)$ which is used to monitor the network performance which captures the differences between the node states as they evolve.

As noted by [14], [27] for scalar-valued node states over a connected graph, the zero eigenvalue (corresponding to the consensus subspace) of the Laplacian matrix precludes reasoning about the \mathcal{H}_2 performance of (3). Under matrix weighting, the zero eigenvalue will have algebraic multiplicity k (corresponding to the consensus subspace of each layer of substates) [28], so as in [14] we will appeal to a similarity transformation that separates out the zero eigenvalues. We define this transformation in the following theorem.

Theorem 1: The scaled and edge-weighted graph Laplacian for a connected graph with time scale matrix \mathbf{E} and weight matrix \mathbf{W} , given by $\mathcal{L}_{w,s} = \mathbf{E}^{-1}\mathbf{D}(\mathcal{G})\mathbf{W}\mathbf{D}(\mathcal{G})^T$, is similar to

$$\begin{bmatrix} \mathcal{L}_{e,s}\mathbf{R}\mathbf{W}\mathbf{R}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix},$$

where $\mathcal{L}_{e,s} = \mathbf{D}(\mathcal{G}_\tau)^T \mathbf{E}^{-1} \mathbf{D}(\mathcal{G}_\tau)$ is the edge Laplacian for a spanning tree \mathcal{G}_τ which is symmetrically “weighted” by the time scaling parameters, and $\mathbf{R}(\mathcal{G}) = \begin{bmatrix} \mathbf{I} & \mathbf{T}_\tau^c \end{bmatrix} = \mathbf{R} \otimes \mathbf{I}_k$, where \mathbf{R} is the basis of the cut space of \mathcal{G} as defined as in [14], with $\mathbf{T}_\tau^c = (\mathbf{D}(\mathcal{G}_\tau)^T \mathbf{D}(\mathcal{G}_\tau))^{-1} \mathbf{D}(\mathcal{G}_\tau)^T \mathbf{D}(\mathcal{G}_c)$. Here, the τ and c subscripts on \mathcal{G} denote the incidence matrices for a spanning tree and the complementary edges in \mathcal{G} , respectively.

Proof: First, we use a lemma, proven in the Appendix.

Lemma 1: The following hold: $\mathbf{T}_\tau^c = \mathbf{T}_\tau^c \otimes \mathbf{I}_k$, $\mathbf{D} = \mathbf{D}_\tau \mathbf{R}$.

Following [1], [14], we define a similarity transformation,

$$\begin{aligned}S_v(\mathcal{G}) &= \begin{bmatrix} \mathbf{E}^{-1}\mathbf{D}(\mathcal{G}_\tau) & (\mathbf{D}(\mathcal{G}_\tau)^T \mathbf{E}^{-1} \mathbf{D}(\mathcal{G}_\tau))^{-1} & \mathbf{1} \end{bmatrix} \\ S_v(\mathcal{G})^{-1} &= \begin{bmatrix} \mathbf{D}(\mathcal{G}_\tau)^T \\ \Xi^{-1} \mathbf{F} \end{bmatrix}, \\ \mathbf{1} &= \mathbf{1}_n \otimes \mathbf{I}_k, \mathbf{F} = \begin{bmatrix} E_1 & \dots & E_n \end{bmatrix} \\ \Xi &= \text{diag}(\{\epsilon_{s,i}\}_{i=1}^k), \epsilon_{s,i} = \sum_{j=1}^n \epsilon_{j,i}.\end{aligned}$$

Note that $\epsilon_{s,i}$ is the sum of all the time scale parameters of the i -th substate over all nodes. We establish that this transformation is well-defined in the Appendix.

Lemma 2: The similarity transforms are well-defined, in that $S_v^{-1} S_v = \mathbf{I}$.

Next, denoting for brevity $\mathbf{D}_\tau := \mathbf{D}(\mathcal{G}_\tau)$, $\mathbf{D} := \mathbf{D}(\mathcal{G})$, etc, and noting that $\mathbf{D}^T \mathbf{1} = (\mathbf{D}^T \mathbf{1}) \otimes \mathbf{I}_k = \mathbf{0}$, we conclude,

$$\begin{aligned}S_v^{-1} \mathcal{L}_{w,s}(\mathcal{G}) S_v &= \\ &= \begin{bmatrix} \mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D} \mathbf{W} \mathbf{D}^T \mathbf{E}^{-1} \mathbf{D}_\tau & (\mathbf{D}_\tau \mathbf{E}^{-1} \mathbf{D}_\tau)^{-1} & \mathbf{A} \\ \Xi^{-1} \mathbf{F} \mathbf{E}^{-1} \mathbf{D} \mathbf{W} \mathbf{D}^T \mathbf{E}^{-1} \mathbf{D}_\tau & (\mathbf{D}_\tau \mathbf{E}^{-1} \mathbf{D}_\tau)^{-1} & \mathbf{B} \end{bmatrix} \\ &\quad (\text{where } \mathbf{A} = \mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D} \mathbf{W} \mathbf{D}^T \mathbf{1}, \mathbf{B} = \Xi^{-1} \mathbf{F} \mathbf{E}^{-1} \mathbf{D} \mathbf{D}^T \mathbf{1}) \\ &= \begin{bmatrix} \mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D}_\tau \mathbf{R} \mathbf{W} \mathbf{R}^T \mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D}_\tau & (\mathbf{D}_\tau \mathbf{E}^{-1} \mathbf{D}_\tau)^{-1} & \mathbf{0} \\ \Xi^{-1} \mathbf{1}^T \mathbf{D}_\tau \mathbf{R} \mathbf{W} \mathbf{R}^T & & \mathbf{0} \end{bmatrix} \\ &= \begin{bmatrix} \mathcal{L}_{e,s} \mathbf{R} \mathbf{W} \mathbf{R}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.\end{aligned}$$

By noting that $S_v \mathbf{x}_e(t) = \mathbf{x}(t)$, the scaled, matrix weighted consensus model with noise (1) is equivalent to,

$$\begin{aligned}\dot{\mathbf{x}}_e(t) &= \begin{bmatrix} -\mathcal{L}_{e,s}(\mathcal{G}_\tau) \mathbf{R}(\mathcal{G}) \mathbf{W} \mathbf{R}(\mathcal{G})^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{x}_e(t) \\ &\quad + \begin{bmatrix} \mathbf{D}_\tau^T \mathbf{E}^{-1} & -\mathcal{L}_{e,s}(\mathcal{G}_\tau) \mathbf{R}(\mathcal{G}) \\ \Xi^{-1} \mathbf{1}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \omega(t) \\ \mathbf{v}(t) \end{bmatrix} \\ \mathbf{z}(t) &= \begin{bmatrix} \mathbf{R}(\mathcal{G}) & \mathbf{0} \end{bmatrix} \mathbf{x}_e(t).\end{aligned}\quad (4)$$

We can see that the form of (4) naturally suggests a partitioning of the edge state variable into a set of states in the spanning tree and those in the consensus space ($\text{span}(\mathbf{1} \otimes \mathbf{I}_k)$), $\mathbf{x}_e(t) = [\mathbf{x}_\tau(t) \quad \mathbf{x}_1(t)]$. The resulting dynamics for the spanning tree states is taken from (4) as,

$$\Sigma_\tau := \begin{cases} \dot{\mathbf{x}}_\tau(t) = -\mathcal{L}_{e,s}(\mathcal{G}_\tau) \mathbf{R}(\mathcal{G}) \mathbf{W} \mathbf{R}(\mathcal{G})^T \mathbf{x}_\tau(t) \\ \quad + \mathbf{D}_\tau^T \mathbf{E}^{-1} \Omega \hat{\omega} - \mathcal{L}_{e,s}(\mathcal{G}_\tau) \mathbf{R}(\mathcal{G}) \Gamma \hat{\mathbf{v}} \\ \mathbf{z}(t) = \mathbf{R}(\mathcal{G})^T \mathbf{x}_\tau(t), \end{cases} \quad (5)$$

where $\hat{\mathbf{v}}$ and $\hat{\omega}$ are normalized noise signals, $\Omega = \mathbb{E}[\omega(t)\omega(t)^T]$, and $\Gamma = \mathbb{E}[\mathbf{v}(t)\mathbf{v}(t)^T]$. An important note is that the chosen output for (3) results in the output of (5) containing information of the cycle states due to the fact that the cycle states are linear combinations of the tree states.

The \mathcal{H}_2 performance of (5) is given by $\text{tr}(R^T X^* R)$, where X^* is the positive-definite solution to the Lyapunov equation,

$$\begin{aligned}-\mathcal{L}_{e,s}^T \mathbf{R} \mathbf{W} \mathbf{R}^T X - X \mathbf{R} \mathbf{W} \mathbf{R}^T \mathcal{L}_{e,s}^T + \mathbf{D}_\tau^T \mathbf{E}^{-1} \Omega \Omega^T \mathbf{E}^{-1} \mathbf{D}_\tau + \\ \mathcal{L}_{e,s}^T \mathbf{R} \Gamma \Gamma^T \mathbf{R}^T \mathcal{L}_{e,s}^T = 0.\end{aligned}\quad (6)$$

In general, the addition of the matrix weighting and scaling precludes a closed form solution to (6) (which is desirable to find X 's dependence on \mathbf{E}, \mathbf{W}), and numeric results yield a nontrivial mixing of weights and scaling parameters in the entries of X . However, in the following section we will outline a case when analytic solutions to (6) exist, providing insights for design of edge weights and scaling parameters for optimal performance.

IV. \mathcal{H}_2 PERFORMANCE AND DESIGN PROBLEMS

A. \mathcal{H}_2 Performance

In this section, we discuss the \mathcal{H}_2 performance for the models of edge consensus in the cases of nodes with time scales, and matrix weighted edges. Specifically, we identify an interesting case where explicit solutions to (6) can be found by

an appropriate choice of noise covariances, and then discuss how the general solution can be approximated by this choice.

By inspection of (6), we can note that by selecting the covariance matrices $\Omega = \sigma_\omega \mathbf{E}^{1/2}$ and $\Gamma = \sigma_v \mathbf{W}^{1/2}$, we can find an analytic solution for (6) in line with those in [1], [14]. With this choice, (6) has the solution,

$$X^* = \frac{1}{2} (\sigma_\omega^2 (\mathbf{RWR}^T)^{-1} + \sigma_v^2 \mathcal{L}_{e,s}^T). \quad (7)$$

This solution is of particular interest because the edge and node weightings are separated in their effect on the \mathcal{H}_2 performance, save for the placement of the σ_ω and σ_v parameters (that is, the effective covariance parameter of the process noise is a “node” parameter, but multiplies the term containing the edge weighting in (7), and vice versa). Note that while this choice of Ω and Γ allows for the analytic solution (7), this solution is merely a proxy for the true performance of the system; for it to be useful for analysis, the error induced by the choice of covariances needs to be assessed.

Consider the scenario in which we have some given covariance matrices, Ω_T and Γ_T , which when used to solve (6) yield the “true” performance of the system. These matrices may or may not be known to us; in either case a relevant endeavor is the quantification of the error incurred by estimating the true performance by (7). The method we will use to quantify this error depends on the positive-definite ordering of performances given by covariance matrices. The following lemma, proven in the Appendix, provides this ordering.

Lemma 3 (Ordering of Performance): For covariance matrices satisfying, $\underline{\Omega} \preceq \Omega \preceq \bar{\Omega}$ and $\underline{\Gamma} \preceq \Gamma \preceq \bar{\Gamma}$, then denoting the solution to (6) using $(\underline{\Omega}, \underline{\Gamma})$; (Ω, Γ) ; $(\bar{\Omega}, \bar{\Gamma})$ as \underline{X} ; X ; \bar{X} , we have, $\underline{X} \preceq X \preceq \bar{X}$. From this we can conclude,

$$\text{tr}(R^T \underline{X} R) \leq \text{tr}(R^T X R) \leq \text{tr}(R^T \bar{X} R).$$

We can employ Lemma 3 to place a bound on the potential error of making the assumption that gives (7). Observe that it is possible to choose multiplicative factors such that,

$$\underline{\alpha} \mathbf{E}^{1/2} \preceq \Omega_T \preceq \bar{\alpha} \mathbf{E}^{1/2} \\ \underline{\beta} \mathbf{W}^{1/2} \preceq \Gamma_T \preceq \bar{\beta} \mathbf{W}^{1/2}.$$

From Lemma 3, we know that the true performance will lie within the performances calculated using $(\underline{\alpha} \mathbf{E}^{1/2}, \underline{\beta} \mathbf{W}^{1/2})$ and $(\bar{\alpha} \mathbf{E}^{1/2}, \bar{\beta} \mathbf{W}^{1/2})$. Thus, taking the difference between the maximum performance and the minimal performance gives the worst case error of (7), which is given by,

$$\text{tr}(R^T \bar{X}^* R) - \text{tr}(R^T \underline{X}^* R) = \frac{1}{2} (\bar{\alpha} - \underline{\alpha}) \sigma_\omega^2 \text{tr}((\mathbf{RWR}^T)^{-1}) \\ + \frac{1}{2} (\bar{\beta} - \underline{\beta}) \sigma_v^2 \text{tr}(\mathcal{L}_{e,s}^T).$$

From this result, we can see that the relative error will be determined by the relative sizes of the multiplicative factors, which raises the question of how the factors can be found or chosen. From the definition of the necessary ordering, it is of course sufficient that,

$$\bar{\alpha} = \frac{\lambda_{\max}(\Omega_T)}{\epsilon_{\max}^{1/2}}; \quad \underline{\alpha} = \frac{\lambda_{\min}(\Omega_T)}{\epsilon_{\min}^{1/2}} \\ \bar{\beta} = \frac{\lambda_{\max}(\Gamma_T)}{\lambda_{\max}(\mathbf{W})^{1/2}}; \quad \underline{\beta} = \frac{\lambda_{\min}(\Gamma_T)}{\lambda_{\min}(\mathbf{W})^{1/2}}. \quad (8)$$

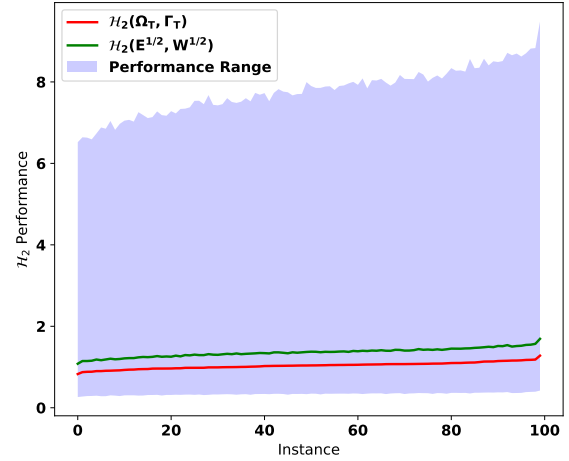


Fig. 1: Numerically calculated \mathcal{H}_2 performance for random graphs ($n = 20, k = 2$) with random covariances ($\mathcal{H}_2(\Omega_T, \Gamma_T)$) and the solution to (7) ($\mathcal{H}_2(\mathbf{E}^{1/2}, \mathbf{W}^{1/2})$). The blue shaded region represents the possible range in performance, calculated using the parameters found via optimization problems given by (9).

The sufficiency, as opposed to necessity, of the parameters given in (8) results in conservative performance bounds. The bounds can be tightened by solving for the minimal/maximal parameters via a simple (convex) optimization problem of the form (for $\bar{\alpha}$),

$$\min_{\alpha} \alpha \\ \text{s.t. } \Omega_T - \alpha \mathbf{E}^{1/2} \preceq 0. \quad (9)$$

The above problem can be modified to give the minimal/maximal values of $\bar{\beta}$ and $\underline{\alpha}/\underline{\beta}$. While (9) gives an obvious advantage over (8), it does require complete information about the covariances, whereas (8) requires only the spectral bounds of the true covariances. In either case, however, it can be seen that the worst case error is helped when the true covariances and the assumed covariances have similar spectral bounds, that is, if the minimum and maximum eigenvalues of Ω_T/Γ_T are approximately equal to those of $\sigma_\omega \mathbf{E}^{1/2}/\sigma_v \mathbf{W}^{1/2}$, the necessary multiplicative factors will be approximately unity and the possible discrepancy small (within a factor of $\simeq 5$ -10). We can see this numerically for random graphs on $n = 20$ nodes with $k = 2$ in Figure 1, where the true covariances, edge weights, and node time scales were randomly generated then scaled to align maximum/minimum eigenvalues. Numerical results over a range of n and k suggest that as the number of substates increases, the bounds (calculated via (9)) become more conservative.

While the results of Lemma 3 (along with (8) or (9)) give the ability to assess whether or not (7) is an acceptable proxy for the true performance, there are no explicit guarantees that the bounds are tight enough for all applications. In cases where there is large discrepancy between the spectral bounds of the true covariances and the time scale or edge weight matrices, for example, (7) may not be useful for performance estimation or optimization. For the cases where it is an appropriate proxy,

however, we proceed with time scale and edge weight design to promote minimal \mathcal{H}_2 performance, starting with a remark pertaining to the special case of tree graphs.

Remark 1: When the underlying graph topology is a tree, $\mathbf{R} = \mathbf{I}$, and (7) simplifies to,

$$\mathbf{X}^* = \frac{1}{2} (\sigma_\omega^2 \mathbf{W}^{-1} + \sigma_v^2 \mathcal{L}_{e,s}^\tau).$$

Furthermore, in this case $\mathcal{H}_2(\Sigma_\tau) = \text{tr}(\mathbf{X}^*)$. A closed form solution for the performance in this case is given in the following lemma.

Lemma 4: For a tree graph, the \mathcal{H}_2 norm of the Σ_τ (5) system is given by,

$$\begin{aligned} \mathcal{H}_2(\Sigma_\tau) &= \frac{1}{2} \text{tr} (\sigma_\omega^2 \mathbf{W}^{-1} + \sigma_v^2 \mathcal{L}_{e,s}^\tau) \\ &= \frac{1}{2} \left(\sigma_\omega^2 \sum_{j=1}^{n-1} \text{tr} (W_j^{-1}) + \sigma_v^2 \sum_{l=1}^k \sum_{i=1}^n \frac{\deg(\nu_i)}{\epsilon_{i,l}} \right), \end{aligned}$$

where $\deg(\nu_i)$ is the unweighted degree of agent ν_i , and j is the index over the edges.

Proof: First consider the weight term. \mathbf{W} is block-diagonal, so $\mathbf{W}^{-1} = \text{Blkdiag}(W_1^{-1}, \dots, W_{n-1}^{-1})$. Thus, its trace is the sum of the traces of the edge weight matrix inverses. Now, consider a single layer of substates, denoted by l . For the second term, consider one of the diagonal elements of $\mathcal{L}_{e,s}$,

$$[L_{e,s}^\tau]_{(ql)(ql)} = a_q^T E_l^{-1} a_q = \epsilon_{i,l}^{-1} + \epsilon_{j,l}^{-1},$$

where a_q is the edge vector corresponding to the edge between nodes i and j , that is, $q = \kappa(ij)$. Now consider a single node, ν_i . In the sum over all edges of the graph, $\epsilon_{i,l}^{-1}$ will appear once for every edge that connects ν_i to its neighbors, which is the unweighted degree of ν_i . Considering all other nodes gives the result for the second term. Finally, the preceding argument holds for all the sub-state layers, which gives the sum over all sub-states. ■

From this result, we can see that there exists a trade off between the time scale parameters and the topology (in this case, the degree distribution) which determines the overall performance of the network. Also, we can contrast the influence of time scale parameters and edge weights in this case. For a given distribution of scaling parameters and edge weights, changing the assignment of edge weights does not affect the \mathcal{H}_2 performance contribution from a given sub-state layer. However, the assignment of scaling parameters can have a significant effect on the performance of the network, which is in line with the similar results in the context of single-input influenced consensus [27].

We can also see that the performance contribution from the time scale parameters in the matrix weighted case is identical to considering a network with k disconnected layers, where each layer has its own time scale distribution. Thus, the evaluation of the time scale assignment in the matrix weighted case is effectively the same as considering assignment in the scalar case. With this in mind, we direct interested readers to [1] for an example of this assignment in action.

B. Gradient Updates on Edge Weights

In the previous section, we saw that one could separate the contributions of the time scales and the edge weights on the \mathcal{H}_2 norm. We now present a design problem for optimizing the edge weight term of Equation (7). Consider Problem (P1),

$$\begin{aligned} \min_{\{W_i\}_{i=1}^{|\mathcal{E}|}} & \text{tr} (\mathbf{R}^T (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} \mathbf{R}) + \frac{h}{2} \sum_{e \in \mathcal{E}} \text{tr} [\mathbf{W}_e^T \mathbf{W}_e]^2 \\ \text{s.t. } & W_{\min} \preceq W_e \preceq W_{\max}, \forall e \in \mathcal{E} \\ & W = \text{blkdiag}(W_i). \end{aligned} \quad (\text{P1})$$

We include a regularization term in the cost function to avoid the trivial solution of completely disconnecting the graph, as well as upper/lower bounds on the matrix weights. A gradient update for solving Problem (P1) is derived in Proposition 1, and proven in the Appendix.

Proposition 1 (Gradient Update for Edge Weights): The gradient of the cost function with respect to the edge weight W_e in Problem (P1) is given by

$$\nabla_{W_e} f[H] = -\text{deblk}_e^k [Q^T Q] + h W_e,$$

where $\text{deblk}_e^k [Q^T Q]$ is the e th $k \times k$ diagonal block of $Q^T Q$, and Q is given by

$$\begin{aligned} Q &= \mathbf{R}^T (\mathbf{R} \mathbf{W}_H^c \mathbf{R}^T)^{-1} \mathbf{R} \\ \mathbf{W}_H^c &= \text{blk}_c^k(H) + \sum_{l \in \mathcal{E} \setminus c} \text{blk}_l^k(W_l), \end{aligned}$$

where

$$\text{blk}_c^k(H) = \mathbf{e}_c H \mathbf{e}_c^T = (e_c \otimes I_k) H (e_c^T \otimes I_k)$$

denotes the $kn \times kn$ matrix with H on the c th $k \times k$ diagonal block, with zeros otherwise.

A gradient update scheme for solving Problem (P1) is therefore

$$\begin{aligned} W_e^{k+1} &= W_e^k - \frac{1}{h\sqrt{k}} \nabla_{W_e} f[H] \\ &= W_e^k - \frac{1}{h\sqrt{k}} (h W_e - \text{deblk}_e^k [Q^T Q]). \end{aligned}$$

C. Decentralized Time Scale Assignment

We saw previously in the definition of (P1) that a regularization term was included to prevent the trivial solution of disconnecting the graph. In the optimization of the time scale term of Equation (7), this trivial solution takes the form of all agents/substates adopting the slowest time scale parameter possible. Thus, consider (P2),

$$\begin{aligned} \min_{\epsilon_{1,1}^{-1}, \dots, \epsilon_{n,k}^{-1}} & \frac{1}{2} \text{tr} (\mathbf{R}^T \mathcal{L}_{e,s}^\tau \mathbf{R}) + \frac{h}{2} \sum_{i=1}^n \sum_{j=1}^k \epsilon_{i,j}^\tau \\ \text{s.t. } & \epsilon_{\max}^{-1} \leq \epsilon_{i,j}^{-1} \leq \epsilon_{\min}^{-1} \quad \forall i \in \mathcal{N}, j \in [k]. \end{aligned} \quad (\text{P2})$$

This is a minimization of the time scale portion of the separated \mathcal{H}_2 performance. A regularization term $2^{-1} h \sum_{i=1}^n \sum_{j=1}^k \epsilon_{i,j}^\tau$ penalizes large time scales for all nodes and their substates assuming positive, integer r . In the following proposition (proven in the Appendix), we show an

analytic solution for the optimal time scale assignment which minimizes the \mathcal{H}_2 performance.

Proposition 2 (Analytic Optimal Time Scale Assignment): Consider (P2). Let the region defined by the box constraints on $1/\epsilon_{i,j}$ be denoted by \mathcal{C} . Then, the minimizing assignment of time scale parameters is given by,

$$\epsilon_{i,j}^* = \text{Proj}_{\mathcal{C}} \left[\left(\frac{\deg(\nu_i)}{hr} \right)^{\frac{1}{r+1}} \right].$$

Remark 2: The assignment rule in Proposition 2 is decentralized, as the optimal assignment value depends only on the (unweighted) degree of the i -th node and the parameters h and r , which are locally known to the i -th node without global knowledge of the network topology.

From this result we can see that for a class of regularization terms, the optimal time scale assignment is again driven by the degree distribution, which is in-line with the previous results. It is conceivable to consider using this result with online signal identification to locally adjust time scales in response to adversarial noise entering the system.

V. EXAMPLE: FLOCKING VIA SECOND-ORDER CONSENSUS

Flocking is a behaviour exhibited by certain multi-agent systems that are coordinating their motion into a cohesive formation, for example birds or stampeding buffalo. A consensus-type algorithm can be proposed that allows a system of n agents to agree on their velocity vector while maintaining a separation from their neighbours [9].

A. Matrix-Valued Double Integrator Consensus

In the case of vector-valued states, we can write the dynamics as

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{E}\ddot{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & I \\ -\mathcal{L}_w & -\mathcal{L}_w \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ I & -\mathbf{D}_G \end{bmatrix} \begin{bmatrix} \omega \\ v \end{bmatrix}.$$

Theorem 2: The double-integrator edge consensus model is given by

$$\begin{bmatrix} \dot{\mathbf{x}}_e \\ \ddot{\mathbf{x}}_e \end{bmatrix} = \begin{bmatrix} \mathbf{0} & 0 & I \\ -\mathcal{L}_e \mathbf{R} \mathbf{W} \mathbf{R}^T & \mathbf{0} & -\mathcal{L}_e \mathbf{R} \mathbf{W} \mathbf{R}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_e \\ \dot{\mathbf{x}}_e \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{D}_\tau^T \mathbf{E}^{-1} & -\mathcal{L}_e \mathbf{R} \end{bmatrix} \begin{bmatrix} \omega \\ v \end{bmatrix},$$

and so the double-integrator consensus on the edge states of the chosen spanning tree τ is given by

$$\begin{bmatrix} \dot{\mathbf{x}}_\tau \\ \ddot{\mathbf{x}}_\tau \end{bmatrix} = \begin{bmatrix} \mathbf{0} & I \\ -\mathcal{L}_e \mathbf{R} \mathbf{W} \mathbf{R}^T & -\mathcal{L}_e \mathbf{R} \mathbf{W} \mathbf{R}^T \end{bmatrix} \begin{bmatrix} \mathbf{x}_\tau \\ \dot{\mathbf{x}}_\tau \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{D}_\tau^T & -\mathcal{L}_e \mathbf{R} \end{bmatrix} \begin{bmatrix} \omega \\ v \end{bmatrix}. \quad (10)$$

Proof: Applying the coordinate transform $\mathbf{x}_e = S_v \mathbf{x}$ and following a similar calculation as in Theorem 1 yields the result. ■

We can also explicitly compute the form of the \mathcal{H}_2 norm for the matrix-weighted double-integrator consensus.

Theorem 3: The controllability gramian for the time scaled double-integrator consensus is given by

$$\mathbf{X}^* = \frac{1}{2} \begin{bmatrix} \mathbf{X}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{X}_2 \end{bmatrix}$$

$$\mathbf{X}_1 = \sigma_w^2 \left[(\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} \mathcal{L}_e^{-1} (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} \right] + \sigma_v^2 (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1}$$

$$\mathbf{X}_2 = \sigma_w^2 (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} + \sigma_v^2 \mathcal{L}_e.$$

Furthermore, the blocks $\mathbf{X}_1, \mathbf{X}_2$ of \mathbf{X}^* correspond to the position and velocity states, respectively. Hence, one can consider the \mathcal{H}_2 performance of the position and velocity states separately or aggregately by examining the \mathcal{H}_2 norms

$$\mathcal{H}_2^{(1)} = \frac{1}{2} \text{tr} (\mathbf{R}^T \mathbf{X}_1 \mathbf{R}), \quad \mathcal{H}_2^{(2)} = \frac{1}{2} \text{tr} (\mathbf{R}^T \mathbf{X}_2 \mathbf{R})$$

$$\mathcal{H}_2^{(1)} = \frac{1}{2} \text{tr} (\mathbf{R}^T (\mathbf{X}_1 + \mathbf{X}_2) \mathbf{R}).$$

Proof: From the dynamics (10), the controllability gramian is given by the positive semi-definite solution to the Lyapunov equation

$$\mathbf{A} \mathbf{X}^* + \mathbf{X}^* \mathbf{A}^T + \mathbf{B} \mathbf{B}^T = \mathbf{0}, \quad (11)$$

where \mathbf{X}^*, \mathbf{A} , and $\mathbf{B} \mathbf{B}^T$ are given by,

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & I \\ -\mathcal{L}_e \mathbf{R} \mathbf{W} \mathbf{R}^T & -\mathcal{L}_e \mathbf{R} \mathbf{W} \mathbf{R}^T \end{bmatrix},$$

$$\mathbf{B} \mathbf{B}^T = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_w^2 \mathcal{L}_e + \sigma_v^2 \mathcal{L}_e \mathbf{R} \mathbf{R}^T \mathcal{L}_e \end{bmatrix},$$

$$\mathbf{X}^* = \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_3 \\ \mathbf{X}_3 & \mathbf{X}_2 \end{bmatrix}.$$

Solving Equation (11) yields

$$\mathbf{X}_3 = \mathbf{0}, \quad \mathbf{X}_2 = \sigma_w^2 (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} + \sigma_v^2 \mathcal{L}_e$$

$$\mathbf{X}_2 = \mathbf{X}_1 \mathbf{R} \mathbf{W} \mathbf{R}^T \mathcal{L}_e,$$

and solving for \mathbf{X}_1 in the last display yields

$$\mathbf{X}_1 = \sigma_w^2 \left[(\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} \mathcal{L}_e^{-1} (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1} \right] + \sigma_v^2 (\mathbf{R} \mathbf{W} \mathbf{R}^T)^{-1}.$$

To measure the position, velocity, or both states for consideration in the \mathcal{H}_2 norm, one can choose the observation matrices

$$\begin{bmatrix} \mathbf{R}^T & \mathbf{0} \end{bmatrix}, \quad \begin{bmatrix} \mathbf{0} & \mathbf{R}^T \end{bmatrix}, \quad \begin{bmatrix} \mathbf{R}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{R}^T \end{bmatrix},$$

respectively. ■

B. Numerical Example

The weight update scheme applied to the second-order consensus problem was implemented numerically. The task assigned to the agents was to use the second-order consensus protocol to achieve the formation shown in Figure 2 – a formation assigned by sampling discrete points on a 2D spiral.

Consensus on the formation is achieved by the second-order protocol with a constant signal \mathbf{d}_i specifying the position in the formation:

$$\ddot{\mathbf{x}}_i = -\sum_{j \in \mathcal{N}(i)} \mathbf{W}_{ij} (\mathbf{x}_i - \mathbf{x}_j - \mathbf{d}_i) - \sum_{j \in \mathcal{N}(i)} \mathbf{W}_{ij} (\dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j). \quad (12)$$

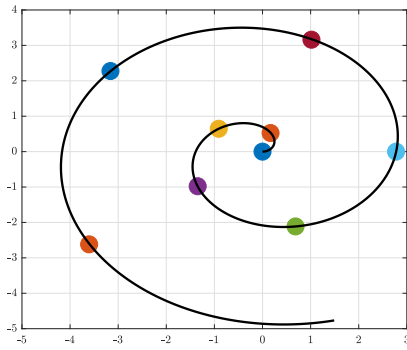


Fig. 2: Relative formation of the agents, defined by discrete points on a spiral.

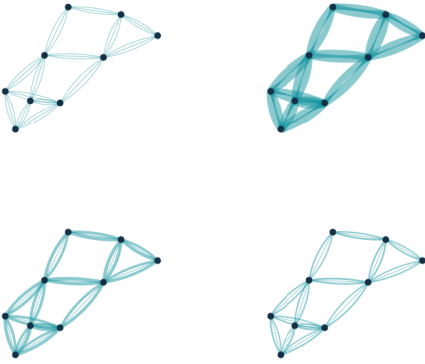


Fig. 3: Visualization of edge weights over iterations $k = 1$ (top left), 2 (top right), 3 (bottom left) & 4 (bottom right). Each of the 3 independent parameters of the 2×2 matrix-valued weight is visualized in a multigraph.

Since this is a constant signal, the \mathcal{H}_2 performance of the edge states remains the same as in the previous section.

The initial selection of weights was chosen at random using the generator

$$W_{\{i,j\}} = \alpha \left[\frac{G_{\{i,j\}} + G_{\{i,j\}}^T}{2} + 2I_2 \right], \quad (13)$$

where $G_{\{i,j\}}$ is a 2×2 matrix with entries distributed according to a zero-mean standard Gaussian, and $\alpha = 0.3$ was chosen arbitrarily to yield a suboptimal initial selection of weights. The upper and lower bounds on the weights were chosen with the same generator in Equation (13), but with $\alpha_l = 0.05$ and $\alpha_u = 10$. The initial time scale parameters were taken to be identically unity. The penalty parameter was chosen as $h = 0.01$.

The gradient descent algorithm from Proposition 1 converges quickly, and intermediate graph weights are visualized in Figure 3. None of the optimal edge weights saturated the upper and lower bounds in this setup. The minimizing time scale assignment was calculated using Proposition 2.

These weights and scaling parameters were then used in a simulation of the dynamics in (12) over a time span of 30 seconds. At $10 \leq t \leq 20$, the formation is subject to a ‘gust’ of noise on the nodes and edges with covariance $\sigma_w^2 I$

and $\sigma_v^2 \mathbf{W}$, with $\sigma_v = \sigma_w = 5$. Simulations were performed with no updates (NUD), updates to edge weights (WUD) or time scales (TUD) during the wind gusts, and with both updates (BUD) during gusts. The edge states for the x and y directions for these four cases are shown in Figure 4, and the variance away from the consensus value $x_e(t_f)$, $\text{Var}[x_e(t)] := [x_e(t) - x_e(t_f)]^2$, is shown in Figure 5. The updated weights and scales outperform their initial, suboptimal values.

VI. CONCLUSION

In this paper we have developed a framework for investigating noise-driven consensus on matrix-weighted and time scaled graphs, with the aim of minimizing the \mathcal{H}_2 performance of such systems. We identified a convenient choice of noise covariances that allows for the separation of the performance contributions from the edge weighting and time scales. This allowed us to investigate the optimal assignment of time scale parameters and edge weights to promote network resilience. Finally, we applied these results to a flocking example, where we observed that applying a time scale assignment and edge weight update in response to an injection of noise results in less perturbation of the agent states compared to the non-updated case, or optimizing over only the edges or only the time scales.

In real-world networked dynamical problems adversarial noise will likely not be applied to all nodes within the network as was taken to be the case in this work. Thus, a useful extension of this work is to investigate what results can be found when the noise injection is limited to a subset of nodes, or in the matrix weighted case, a subset of substates. Furthermore, interesting problems such as leader selection in noise driven consensus networks have previously been investigated, so these areas are also potential directions for future work by extending those related topics to the time scaled and matrix weighted case.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] D. R. Foight, M. Hudoba de Badyn, and M. Mesbahi, “Time Scale Design for Network Resilience,” in *Proc. 58th IEEE Conference on Decision and Control*, Nice, France, 2019, pp. 1–8.
- [2] Y. Chen, J. Lu, X. Yu, and D. J. Hill, “Multi-agent systems with dynamical topologies: Consensus and applications,” *IEEE Circuits and Systems Magazine*, vol. 13, no. 3, pp. 21–34, 2013.
- [3] R. Olfati-Saber and R. M. Murray, “Consensus protocols for networks of dynamic agents,” *Proc. of the American Control Conference*, vol. 2, pp. 951–956, 2003.
- [4] H. G. Tanner, G. J. Pappas, and V. Kumar, “Leader-to-formation stability,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 443–455, 2004.
- [5] M. A. Joordens and M. Jamshidi, “Underwater swarm robotics consensus control,” in *Proc. IEEE International Conference on Systems, Man and Cybernetics*, no. October, San Antonio, USA, 2009, pp. 3163–3168.

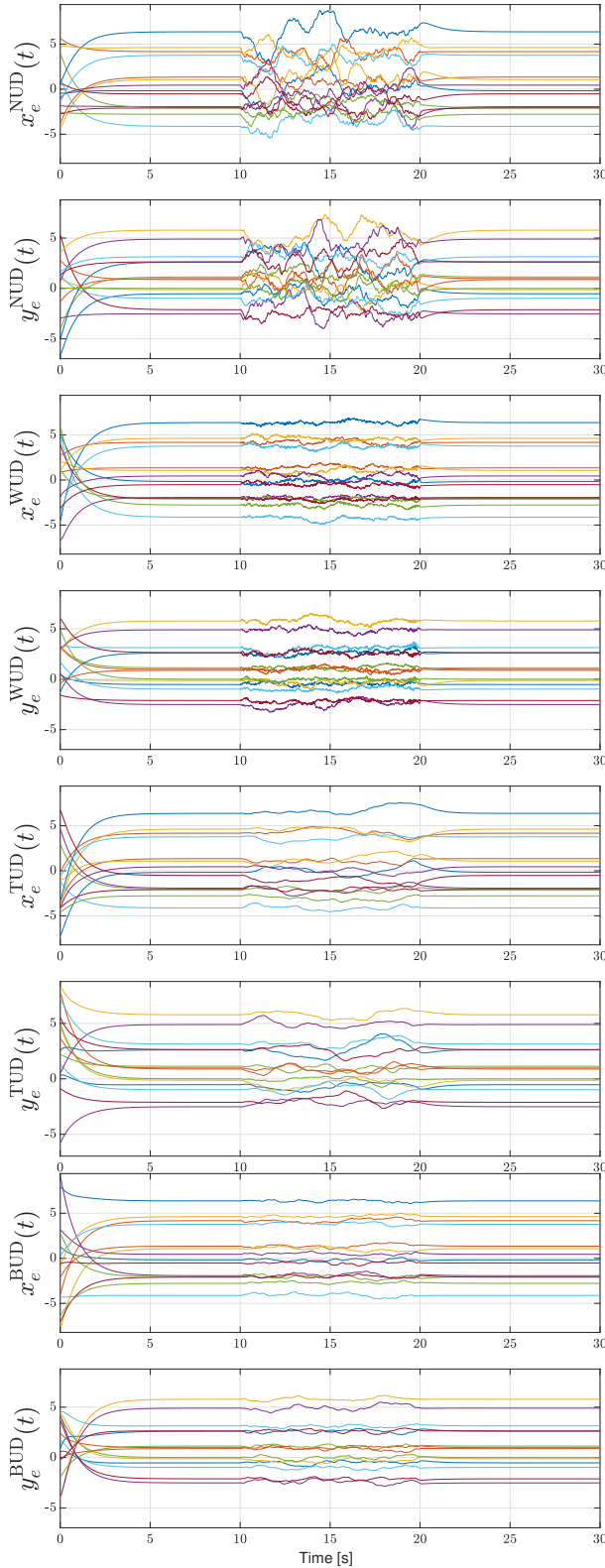


Fig. 4: Edge states in x, y directions over time subjected to gust at $10 \leq t \leq 20$; cases from top to bottom are with no updates (NUD), weight updates (WUD), time scale update (TUD), and both updates (BUD).

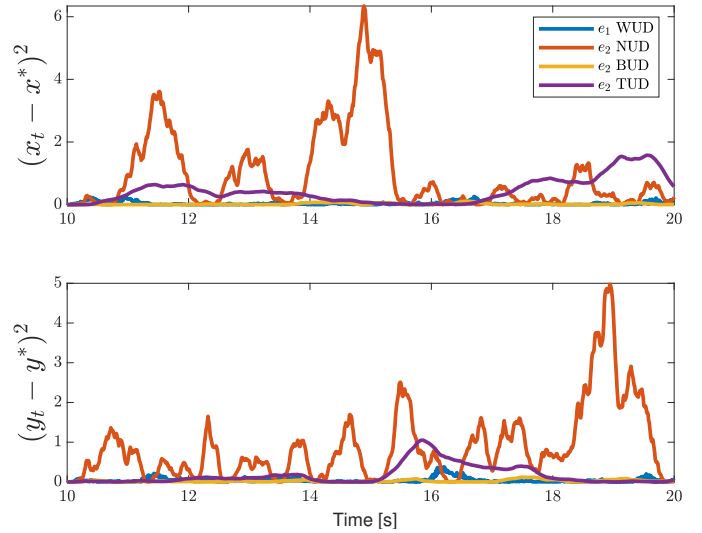


Fig. 5: Variance from consensus for a representative edge $[x_e(t) - x_e(t_f)]^2$ for the case with the weight update (WUD), time scale update (TUD), both updates (BUD), and with no updates (NUD).

- [6] M. Hudoba de Badyn, U. Eren, B. Açkmeşe, and M. Mesbahi, "Optimal mass transport and kernel density estimation for state-dependent networked dynamic systems," in *Proc. 57th IEEE Conference on Decision and Control*, Miami Beach, USA, 2018.
- [7] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference*, Seville, Spain, 2005, pp. 8179–8184.
- [8] M. Siami and N. Motee, "Fundamental limits and tradeoffs on disturbance propagation in linear dynamical networks," *IEEE Transaction on Automatic Control*, vol. 61, no. 12, pp. 4055–4062, 2016.
- [9] A. Chapman, E. Schoof, and M. Mesbahi, "Online Adaptive Network Design for Disturbance Rejection," in *Principles of Cyber-Physical Systems*. Cambridge University Press, 2015.
- [10] M. Hudoba de Badyn and M. Mesbahi, "H2 performance of series-parallel networks: A compositional perspective," *IEEE Transactions on Automatic Control*, pp. 1–8, 2020.
- [11] B. Bamieh, M. R. Jovanović, P. Mitra, and S. Patterson, "Coherence in Large-Scale Networks: Dimension-Dependent Limitations of Local Feedback," *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2235–2249, 2012.
- [12] S. Patterson and B. Bamieh, "Leader selection for optimal network coherence," in *49th IEEE Conference on Decision and Control (CDC)*. IEEE, 2010, pp. 2692–2697.
- [13] —, "Consensus and coherence in fractal networks," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 4, pp. 338–348, 2014.
- [14] D. Zelazo and M. Mesbahi, "Edge agreement: Graph-theoretic performance bounds and passivity analysis," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 544–555, 2011.
- [15] D. Zelazo, S. Schuler, and F. Allgöwer, "Performance and design of cycles in consensus networks," *Systems and Control Letters*, vol. 62, no. 1, pp. 85–96, 2013.
- [16] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE transactions on automatic control*, vol. 58, no. 11, pp. 2715–2729, 2013.
- [17] F. Dörfler, J. W. Simpson-Porco, and F. Bullo, "Electric networks and algebraic graph theory: Models, properties and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 997–1005, 2018.
- [18] A. Chakraborty and P. P. Khargonekar, "Introduction to wide-area control of power systems," in *2013 American Control Conference*. IEEE, 2013, pp. 6758–6770.
- [19] P. Kokotović, H. K. Khalil, and J. O'Reilly, *Singular Perturbation Methods in Control: Analysis and Design*. Philadelphia: Society for Industrial and Applied Mathematics, 1999.
- [20] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks

of agents with switching topology and time-delays,” *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

- [21] F. Pedrocchi, M. Rebollo, C. Carrascosa, and A. Palomares, “Convergence of weighted-average consensus for undirected graphs,” *International Journal of Complex Systems in Science*, vol. 4, no. 1, pp. 13–16, 2014.
- [22] A. Chapman and M. Mesbahi, “Multiple Time-Scales in Network-of-Networks,” in *Proc. American Control Conference*, Boston, USA, 2016, pp. 5563–5568.
- [23] A. Awad, A. Chapman, E. Schoof, A. Narang-Siddharth, and M. Mesbahi, “Time-scale separation in networks: state-dependent graphs and consensus tracking,” *IEEE Transactions on Control of Network Systems*, vol. 5870, no. c, pp. 1–1, 2018.
- [24] J. B. Rejeb, I.-C. Morărescu, and J. Daafouz, “Synchronization in networks of linear singularly perturbed systems,” in *Proc. American Control Conference*. Boston, USA: American Automatic Control Council (AACC), 2016, pp. 4293–4298.
- [25] —, “Control design with guaranteed cost for synchronization in networks of linear singularly perturbed systems,” *Automatica*, vol. 91, pp. 89–97, 2018.
- [26] M. Deghat, Z. Sun, D. Nešić, and C. Manzie, “Singularly perturbed algorithms for velocity consensus and shape control of single integrator multi-agent systems,” *IFAC-PapersOnLine*, vol. 51, no. 23, pp. 200–205, 2018.
- [27] D. R. Foight and M. Mesbahi, “Influenced Consensus for Multi-Scale Networks,” in *Proc. American Control Conference*, Philadelphia, USA, 2019.
- [28] M. H. Trinh and H.-S. Ahn, “Theory and applications of matrix-weighted consensus,” *arXiv preprint arXiv: 1703.00129v3*, no. 1, pp. 1–21, 2017.
- [29] P. Barooah and J. P. Hespanha, “Estimation from relative measurements: Electrical analogy and large graphs,” *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2181–2193, 2008.
- [30] B. H. Lee and H. S. Ahn, “Distributed formation control via global orientation estimation,” *Automatica*, vol. 73, pp. 125–129, 2016.
- [31] J. L. Ramirez, M. Pavone, E. Frazzoli, and D. W. Miller, “Distributed control of spacecraft formation via cyclic pursuit: Theory and experiments,” *Proc. American Control Conference*, pp. 4811–4817, 2019.
- [32] S. E. Tuna, “Synchronization under matrix-weighted Laplacian,” *Automatica*, vol. 73, pp. 76–81, 2016.
- [33] N. E. Friedkin, A. V. Proskurnikov, R. Tempo, and S. E. Parsegov, “Network science on belief system dynamics under logic constraints,” *Science*, vol. 354, no. 6310, pp. 321–326, 2016.
- [34] S. E. Parsegov, A. V. Proskurnikov, R. Tempo, and N. E. Friedkin, “Novel multidimensional models of opinion dynamics in social networks,” *IEEE Transactions on Automatic Control*, vol. 62, no. 5, pp. 2270–2285, 2017.
- [35] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control, Analysis and design*. John Wiley & Sons, 2001.

APPENDIX

Here, we prove Lemmas 1-3, and Propositions 1 & 2.

Lemma 1. *Proof:* We can compute:

$$\begin{aligned} \mathbf{T}_\tau^c &= (\mathbf{D}_\tau^T \mathbf{D}_\tau)^{-1} \mathbf{D}_\tau^T \mathbf{D}_c \\ &= \left((\mathbf{D}_\tau^T \mathbf{D}_\tau)^{-1} \mathbf{D}_\tau^T \mathbf{D}_c \right) \otimes \mathbf{I}_k = \mathbf{T}_\tau^C \otimes \mathbf{I}_k, \end{aligned}$$

and similarly, $\mathbf{D} = \mathbf{D} \otimes \mathbf{I}_k = (\mathbf{D}_\tau \otimes \mathbf{I}_k) (\mathbf{R} \otimes \mathbf{I}_k) = \mathbf{D}_\tau \mathbf{R}$. ■

Lemma 2. *Proof:* Denote the product $S_v^{-1} S_v$ in block form:

$$S_v^{-1} S_v = \begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{bmatrix}.$$

First, note that

$$\mathbf{F} \mathbf{E}^{-1} = [\mathbf{E}_1 \quad \cdots \quad \mathbf{E}_n] \mathbf{E}^{-1} = [\mathbf{I} \quad \cdots \quad \mathbf{I}] = \mathbf{1}_n^T \otimes \mathbf{I}_k.$$

Then, we can compute each term:

$$\begin{aligned} \mathcal{A} &= \mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D}_\tau (\mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D}_\tau)^{-1} = \mathbf{I} \\ \mathcal{B} &= \mathbf{D}_\tau^T \mathbf{1} = (\mathbf{D}_\tau^T \mathbf{1}_n) \otimes \mathbf{I}_k = \mathbf{0} \otimes \mathbf{I}_k = \mathbf{0}. \\ \mathcal{C} &= \Xi^{-1} \mathbf{F} \mathbf{E}^{-1} \mathbf{D}_\tau (\mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D}_\tau)^{-1} \\ &= \Xi^{-1} (\mathbf{1}_n^T \mathbf{D}_\tau \otimes \mathbf{I}_k) (\mathbf{D}_\tau^T \mathbf{E}^{-1} \mathbf{D}_\tau)^{-1} = \mathbf{0}. \end{aligned}$$

Lastly, we have

$$\begin{aligned} \mathcal{D} &= \Xi^{-1} \mathbf{F} \mathbf{1} \\ &= \begin{bmatrix} \epsilon_{s,1}^{-1} & & \\ & \ddots & \\ & & \epsilon_{s,k}^{-1} \end{bmatrix} [\mathbf{E}_1 \quad \cdots \quad \mathbf{E}_n] (\mathbf{1}_n \otimes \mathbf{I}_k) \\ &= \begin{bmatrix} \epsilon_{s,1}^{-1} & & \\ & \ddots & \\ & & \epsilon_{s,k}^{-1} \end{bmatrix} \begin{bmatrix} \sum_{j=1}^n \epsilon_{j,1} & & \\ & \ddots & \\ & & \sum_{j=1}^n \epsilon_{j,k} \end{bmatrix} = \mathbf{I}_k. \end{aligned}$$

Lemma 3. *Proof:* For simplicity, we adopt the following notation the state and input matrix of (5), $\mathbf{A} := -\mathcal{L}_{e,s} \mathbf{R} \mathbf{W} \mathbf{R}^T$, and $\mathbf{B} := \mathbf{B}_\tau \Omega \Omega^T \mathbf{B}_\tau^T + \mathbf{B}_c \Gamma \Gamma^T \mathbf{B}_c^T$. Due to the stability of $\mathcal{L}(\mathcal{G})$, we know the solution to (6) is positive definite, and can be written as in integral form [35],

$$\mathbf{X} = \int_0^\infty e^{\mathbf{A}t} \mathbf{B}_\tau \Omega \Omega^T \mathbf{B}_\tau^T e^{\mathbf{A}^T t} dt + \int_0^\infty e^{\mathbf{A}t} \mathbf{B}_c \Gamma \Gamma^T \mathbf{B}_c^T e^{\mathbf{A}^T t} dt$$

Consider either term in the above solution, and generalize as,

$$\mathbf{X}_p = \int_0^\infty e^{\mathbf{A}t} \mathbf{B}_p \mathbf{Z} \mathbf{Z}^T \mathbf{B}_p^T e^{\mathbf{A}^T t} dt$$

where \mathbf{Z} and \mathbf{B}_p are placeholders for a covariance matrix and input matrix, respectively. From this, consider the quadratic form with any u ,

$$u^T \mathbf{X}_p u = \int_0^\infty \| \mathbf{Z}^T \mathbf{B}_p^T e^{\mathbf{A}^T t} u \|_2^2 dt$$

This form is usually employed to show the positive definiteness of \mathbf{X}_p based on the stability of \mathbf{A} , but here it can be used to order \mathbf{X}_p 's based on the ordering of \mathbf{Z} :

$$\begin{aligned} u^T (\mathbf{X}_p - \bar{\mathbf{X}}_p) u &= \int_0^\infty \| \mathbf{Z}^T \mathbf{B}_p^T e^{\mathbf{A}^T t} u \|_2^2 - \| \bar{\mathbf{Z}}^T \mathbf{B}_p^T e^{\mathbf{A}^T t} u \|_2^2 dt \\ &\leq \int_0^\infty (\| \mathbf{Z}^T \|_2^2 - \| \bar{\mathbf{Z}}^T \|_2^2) \| \mathbf{B}_p^T e^{\mathbf{A}^T t} u \|_2^2 dt \end{aligned}$$

From $\mathbf{Z} \preceq \bar{\mathbf{Z}}$, it follows that $\| \mathbf{Z} \|_2^2 \leq \| \bar{\mathbf{Z}} \|_2^2$. Thus, this integral is always negative, which then implies $\mathbf{X}_p - \bar{\mathbf{X}}_p \preceq 0 \Rightarrow \mathbf{X}_p \succeq \bar{\mathbf{X}}_p$. By identical argument,

$$u^T (\underline{\mathbf{X}}_p - \mathbf{X}_p) u \leq \int_0^\infty (\| \underline{\mathbf{Z}}^T \|_2^2 - \| \mathbf{Z}^T \|_2^2) \| \mathbf{B}_p^T e^{\mathbf{A}^T t} u \|_2^2 dt$$

implies that $\underline{\mathbf{X}}_p \preceq \mathbf{X}_p$. This shows the ordering for the solutions of (6) for ordered covariances.

Now, consider the ordering of the performance, which is given by $\text{tr}(\mathbf{R} \mathbf{X} \mathbf{R}^T)$. From the ordering and positive definiteness of $\underline{\mathbf{X}}, \mathbf{X}, \bar{\mathbf{X}}$, we have, $u^T \underline{\mathbf{X}} u \leq u^T \mathbf{X} u \leq u^T \bar{\mathbf{X}} u$, for all u . Letting $u = \mathbf{R}x$ gives,

$$x^T \mathbf{R}^T \underline{\mathbf{X}} \mathbf{R} x \leq x^T \mathbf{R}^T \mathbf{X} \mathbf{R} x \leq x^T \mathbf{R}^T \bar{\mathbf{X}} \mathbf{R} x.$$

Now, note that the trace of a matrix M can be written as $\sum_i e_i^T M e_i$. Taking $M = R^T X R$ and $x = e_i$ in the above equation gives the desired ordering on the performances. ■

Proposition 2. *Proof:*

We first identify the gradient of the cost function with respect to W_c , the weight on the c th edge in \mathcal{E} . To this end, consider the functions

$$\begin{aligned} f(X) &= \mathbf{R} \left[\mathbf{e}_c X \mathbf{e}_c^T + \sum_{l \in \mathcal{E} \setminus c} \mathbf{e}_l W_l \mathbf{e}_l^T \right] \mathbf{R}^T \\ &= \mathbf{R} \left[\text{blk}_c^k(X) + \sum_{l \in \mathcal{E} \setminus c} \text{blk}_c^k(W_l) \right] \mathbf{R}^T \\ g(X) &= X^{-1}, \quad h(X) = \text{tr} [\mathbf{R}^T X \mathbf{R}], \end{aligned}$$

where

$$\text{blk}_l^k(W_l) = \mathbf{e}_l W_l \mathbf{e}_l^T = (e_l \otimes I_k) W_l (e_l^T \otimes I_k)$$

denotes the $kn \times kn$ matrix with W_l on the l th $k \times k$ diagonal block, with zeros otherwise. Then, the cost function with W_c as the argument, is given by

$$\text{tr} [\mathbf{R}^T (\mathbf{R} W \mathbf{R}^T)^{-1} \mathbf{R}] = (h \circ g \circ f)(W_c). \quad (14)$$

These functions have differentials

$$\begin{aligned} df_X[H] &= \mathbf{R} [\text{blk}_c^k(H)] \mathbf{R}^T, \quad dg_X[H] = -X^{-1} H X^{-1} \\ dh_X[H] &= \text{tr} [\mathbf{R}^T H \mathbf{R}]. \end{aligned}$$

By the chain rule, we have that

$$\begin{aligned} d(g \circ f)_X[H] &= -Y^{-1} \mathbf{R} \text{blk}_c^k(H) \mathbf{R}^T Y^{-1} \\ Y &\triangleq f[H] = \mathbf{R} \left[\text{blk}_c^k(H) + \sum_{l \in \mathcal{E} \setminus c} \text{blk}_l^k(W_l) \right] \mathbf{R}^T \\ &\triangleq \mathbf{R} W_H^c \mathbf{R}^T, \end{aligned}$$

and so

$$d(g \circ f)_X[H] = -(\mathbf{R} W_H^c \mathbf{R}^T)^{-1} \mathbf{R} \text{blk}_c^k(H) \mathbf{R}^T (\mathbf{R} W_H^c \mathbf{R}^T)^{-1}$$

$$d(h \circ g \circ f)_X[H] = -\text{tr} [Q \text{blk}_c^k(H) Q^T], \quad Q^T \triangleq \mathbf{R}^T (\mathbf{R} W_H^c \mathbf{R}^T)^{-1} \mathbf{R}.$$

Hence, we can write

$$\begin{aligned} d(h \circ g \circ f)_X[H] &= -\text{tr} [Q \text{blk}_c^k(H) Q^T] \\ &= -\text{tr} [Q \mathbf{e}_c H \mathbf{e}_c^T Q^T] = \langle -\mathbf{e}_c^T Q^T Q \mathbf{e}_c, H \rangle, \end{aligned}$$

and so the gradient of (14) with respect to the c th weight W_c is identified as the c th $k \times k$ diagonal block of $-Q^T Q$. ■

Proposition 2. *Proof:* Consider the cost function (denoted by $f(\epsilon_{i,j}^{-1})$) without the box constraint, and note that the \mathcal{H}_2 portion can be rewritten as a double sum of the same form as the regularization term,

$$\text{tr} (\mathbf{R}^T \mathcal{L}_{e,s}^T \mathbf{R}) = \text{tr} (\mathbf{E}^{-1} \mathcal{L}) = \sum_{i=1}^n \sum_{j=1}^k \epsilon_{i,j}^{-1} \deg(\nu_i),$$

resulting in,

$$f(\epsilon_{i,j}^{-1}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^k \epsilon_{i,j}^{-1} \deg(\nu_i) + \frac{h}{2} \sum_{i=1}^n \sum_{j=1}^k (\epsilon_{i,j}^{-1})^{-r}.$$

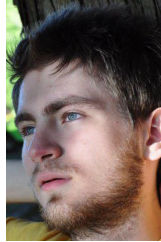
Minimizing this cost alone can be achieved by setting its gradient equal to zero,

$$\frac{\partial f}{\partial \epsilon_{i,j}^{-1}} = \frac{\deg(\nu_i)}{2} - \frac{hr}{2} (\epsilon_{i,j}^{-1})^{-(r+1)} = 0,$$

implying that, $\epsilon_i^* = (\deg(\nu_i)/hr)^{1/r+1}$. Projecting this result onto the constraint set gives the result. ■



Dillon Foight received a B.S. degree in space physics (astrophysics) from Embry-Riddle Aeronautical University, Prescott, AZ, USA in 2009, and worked as a science mission planner for the Chandra Space Telescope until 2015. He is currently a graduate student in the William E. Boeing Department of Aeronautics & Astronautics at the University of Washington, Seattle, WA, USA. As a member of the Robotics, Aerospace, and Information Networks (RAIN) Lab, he focuses on analysis, control, and influence of networks featuring multi-time scale behavior. He is the recipient of the National Science Foundation Graduate Research Fellowship (2017).



Mathias Hudoba de Badyn is a postdoctoral scholar in the Automatic Control Laboratory at the Swiss Federal Institute of Technology in Zürich. He received his Ph.D. degree in the William E. Boeing Department of Aeronautics and Astronautics, and an M.Sc. degree in the Department of Mathematics in 2019 at the University of Washington. In 2014, he graduated from the University of British Columbia with a BSc in Combined Honours in Physics and Mathematics. He held an NSERC PGS-D (CGS-D

offered) from 2017-2019, and a University of Washington College of Engineering Dean's Fellowship from 2014-2015. His research interests include the analysis and control of networked dynamical systems, with applications to autonomous air and space multi-vehicle systems.



Mehran Mesbahi is a Professor of Aeronautics & Astronautics and an Adjunct Professor of Mathematics and Electrical Engineering at the University of Washington. He received his Ph.D. from USC in 1996. He was a member of the Guidance, Navigation, and Analysis group at Jet Propulsion Laboratory from 1996-2000 and an Assistant Professor of Aerospace Engineering and Mechanics at the University of Minnesota from 2000-2002. He is currently the Director of the Robotics, Aerospace, and Information Networks (RAIN) Laboratory and the Executive Director of the Joint Center for Aerospace Technology Innovation. His research interests are distributed and networked systems, autonomous aerospace and robotic systems, and the intersection between data science, networks, autonomy, and control theory, with multi-disciplinary applications.