Semi-Autonomous Consensus: Network Measures and Adaptive Trees

Airlie Chapman, *Student Member, IEEE*, and Mehran Mesbahi, *Senior Member, IEEE*

***Abstract*—Examining the effectiveness of control in networked systems is a thriving research area. Autonomous systems that can be intermittently influenced (controlled) by external agents find applications ranging from machine calibration to satellite control. We refer to this class of networks as *semi-autonomous*. If the semi-autonomous agents' interaction dynamics are consensus-based, we dub this subclass as *semi-autonomous consensus*, which is the focus of the paper. Within such a subclass, we consider the dynamics of networked agents in the context of performance (friendly influence) and security (unfriendly influence). Our approach to appraise a semi-autonomous consensus network is to expose the network to fundamental test signals, namely white noise and an impulse, and use the resultant system response to quantify network performance and security. Traditionally, input-output properties are varied by altering the dynamics of the network agents. We instead adopt *topological* methods for this task, designing five protocols for tree graphs that rewire the network topology, leaving the network agents' dynamics untouched. In pursuit of this objective, four adaptive protocols are introduced to either increase or decrease the mean tracking and variance damping measures, respectively. Finally, a proposed fifth hybrid protocol is shown to have a guaranteed performance for *both* measures using a game-theoretic formalism.**

***Index Terms*—Adaptive networks, consensus protocol, coordinated control over networks, graph theory, network security, semi-autonomous networks.**

I. Introduction

CONSENSUS-based systems provide effective means of distributed information-sharing and control for networked, multi-agent systems in settings such as multi-vehicle control, formation control, swarming, and distributed estimation; see for example, [2]–[6]. One of the appeals of consensus algorithms is their ability to operate *distributively* and *autonomously* over simple *trusting* agents. This has the added benefit that external (control) agents, perceived as *native* agents, can seamlessly attach to the network and steer it in particular directions. These additional agents, ignoring consensus rules, will *influence* the system dynamics compared to the *unforced* networked system resulting in scenarios such as leader-follower [3], [6], and drift correction [7]. The detriment is that this same approach can be adopted by malicious infiltrating agents. We refer to consensus-based systems, with friendly and/or unfriendly attached nodes, as *semi-autonomous consensus* networks. Although the convergence properties of consensus algorithms has been extensively studied, examining the network input-output properties in a controlled setting, and their interpretation, is in its infancy—studied in such recent works as [8]–[10].

For a semi-autonomous consensus network exposed to either (or both) friendly and unfriendly agents, it is necessary to reason about either (or both) *performance* and *security*. *Performance* (friendly external agents) in the traditional undirected consensus is a well studied problem with a general favoritism for the second smallest eigenvalue of the graph Laplacian as a metric to quantify the convergence rate [6], [11], though interest has also been shown with other network measures, for example, the largest eigenvalue of the graph Laplacian [12]. These metrics prove less attractive in a semi-autonomous consensus setting where convergence rates can vary dramatically based on where in the network external agents have attached. An alternative is to examine worst, best, or average case convergence of the directed network formed by treating external agents as native agents [1], [6]. Network design to improve some of these measures are explored in [12]–[14].

In regard to *security* (unfriendly external agents), most modern day semi-autonomous networks rely on access security to the network which is unsuited for a trusting semi-autonomous consensus setting. An alternative to generate a secure network is intrusion detection[1] coupled with either inter-agent security through each agent's dynamics or intra-agent security via the network topology. The former includes implementation of disturbance rejection or agent disabling techniques, e.g., noise canceling systems and power grid "brown outs." The latter involves global or local network rewiring, e.g., TCP network re-routing. This adaptive topology approach for security as well as performance is the main focus of the present work.

Network performance and security via adaptive topology (intra-agent security) is a largely unexplored area within the semi-autonomous consensus setting. We are particularly motivated by scenarios where an adaptive topology is the *only* security response available, for example, when the agents dynamics and the underlying interaction protocol are assumed to be fixed or expensive to alter. Representative instances of such systems include: networks with hardwired dynamics and interactions, for example; due to safety and requirements on performance; and systems with physically and biologically motivated dynamics

Manuscript received October 12, 2010; revised June 27, 2011; accepted January 30, 2012. Date of publication June 20, 2012; date of current version December 17, 2012. A preliminary version of this work, dealing with the adaptive protocols for trees, was presented at the 49th IEEE Conference on Decision and Control, Atlanta, GA, 2010. This work was supported by AFOSR grant FA9550-09-1-0091 and National Science Foundation (NSF) grant CMMI-0856737. Recommended by Associate Editor J. Cortes.

The authors are with the Department of Aeronautics and Astronautics, University of Washington, Seattle, WA 98195 USA (e-mail: airliec@uw.edu, mesbahi@uw.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2012.2205429

[1] Techniques for intrusion or fault detection on consensus-type networks include those based on reachability analysis [15], and the more popular unknown-input observers [9], [16], [17].

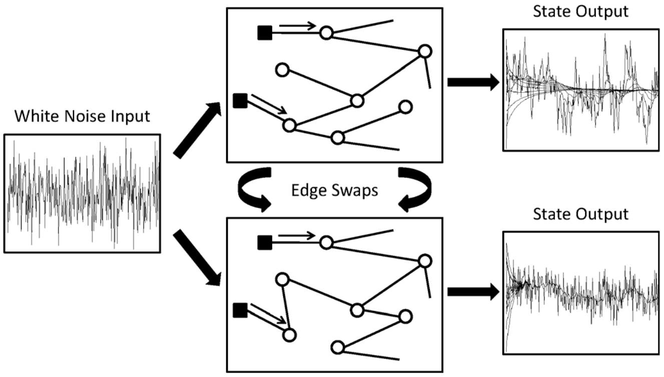


Fig. 1. Illustration of the problem setup: External agents (squares) inject white noise into the network manipulating the state output of the native agents (circles). A protocol performs edge swaps to alter the state output, specifically the network's mean tracking and variance damping measures.

and interactions, e.g., diffusion-driven self assembly and bio-inspired robotic networks. Furthermore, in the case where the intent of external agents may only be known probabilistically, network rewiring presents a security option that is less dramatic than altering agent dynamics and interactions. Characteristics such as the consensus value of an uninfluenced diffusion network are invariant under network rewiring which is not generally the case when the agents' dynamics and interactions are altered due to a security and performance criteria—an attractive property in the event that the intent of the external agent had been misdiagnosed.

The main difficulty for analyzing networks where both friendly and unfriendly agents can attach is that features that are conducive to security are not generally favorable for performance. Our work provides metrics for both performance and security, and discusses associated topological features that can be used to design performant and secure networks. We also propose protocols that rewire the network topology in order to exploit these topological features.

Approach: In this paper, we examine the performance and security of a network in response to an external agent injecting a *test signal*, namely a white Gaussian signal or an impulse, into the network. The performance and security of the network is measured in terms of the subsequent mean and variance of the agents' state; we refer to these metrics as the *mean tracking* measure and *variance damping* measure. Both measures are used to propose five decentralized protocols for tree graphs that adaptively aim to improve or degrade either the performance or the security of the network by undertaking local edge swaps. Fig. 1 illustrates these concepts where the graph topology is used to vary the output characteristics of the network.

Two motivating applications for the present work and the utility of the metrics as they relate to each are:

- Distributed state estimation—where the local estimate at each node reaches the global network-level estimate via consensus, used in scenarios such as drift correction and time synchronization [18], [19]. Consider now external agents that do not accept information exchanges from other agents and deliver instead a Gaussian white noise with unit intensity to their neighboring agents. The external agents' disregard of the consensus dynamics may be due to superior sensing compared to native agents, malfunctions, or malicious intent. Viewing the difference between a states' value and the external agents' as the state error, the mean tracking measure is the expected quadratic performance of this state error. The variance damping measure, on the other hand, is the expected nodes' error variance. The intent of an external agent can only be known probabilistically, and as such, security in the system is left to the less intrusive adaptive topological methods, leaving the agents and their interaction dynamics unchanged.
- Flocking—where $x_i \in \mathbb{R}^m$ (e.g., $m = 2, 3$) is the velocity of agent i, $\dot{x}_i = u_i$, and u_i is dependent on the relative velocities of neighboring agents, used in scenarios such as UAV flocking and fish swarming [6], [11]. A node can then be considered as an external agent guiding the flock by ignoring consensus with either friendly or malicious intent. The ease with which the flock tracks this agent while the agent holds its velocity constant can be gauged using the mean tracking measure while if the agent undertakes a sudden impulse-like maneuver, the damping of its state's propagation through the network can be quantified by the variance damping measure. The underlying interaction dynamics are fixed due to the nature of the onboard relative sensors. The agent dynamics are fixed to guarantee predesignated performance characteristics such as bounds on interagent distances. Subsequently, improving performance and security of the network can only rely on methods that utilize an adaptive topology.

To clarify the contributions of this paper, it is worthwhile to compare our results and approach with similar works in literature. Designing topologies to optimize for certain metrics has been addressed in [13] for maximizing the second smallest eigenvalue of the graph Laplacian, in [12] for optimizing the network $\mathcal{H}_2$ performance, and in [14] for maximizing the largest eigenvalue of the graph Laplacian, each using optimization techniques over *weighted* graphs. Our problem of edge swaps considered in this paper in an *optimization setting*, would require NP-hard mixed-integer programming. We have thus opted for a game theoretic formalism to quantify network performance and security. The protocol's effectiveness is qualified using the sub-optimality properties of the Nash equilibria by modeling the external-native agent dynamics as a non-cooperative game [7], [20]. The simplest form of adaptive network security is the removal of those nodes in the network connected to infiltrators [15], [21]. Using percolation theory, Callaway *et al.* [22] illustrated this to be a potentially disruptive remedy as it can cause the network to become disconnected even for highly dense graphs which subsequently provides an attack vector that an infiltrator could exploit, e.g., by falsely tagging trustworthy agents as untrustworthy. Tyson *et al.* [23] has discussed intuitive methods for network reconfiguration in order to improve resilience, specifically using thresholding methods to decide when to alter the topology. Security techniques that involve adapting the agent dynamics to compensate for k infiltrators has been addressed for $2k + 1$ connected graphs for Byzantine faults and $k + 1$ connected graphs for general faults in [17], [24].

The main contributions of this paper are threefold. First, a pair of network measures are proposed. The mean tracking measure is the average quadratic performance measure of the error in response to the test signal and is linked to the network structure via an electrical network analogy. The variance damping measure, on the other hand, is the expected mean square error of the states, which can be calculated using the controllability gramian and can also be related to the network structure using an electrical network analogy. Secondly, four protocols are developed to optimize the network topology with respect to the proposed measures. These protocols each locally rewire the network topology, using edge swaps between neighboring nodes, to favorably increase or decrease each measure respectively, but not concurrently, thus improving global performance with respect to friendly or unfriendly attached agents. Finally, we formulate the metrics in terms of the effective resistance of an electrical network and, in so doing, illustrate the coupling between mean tracking and variance damping measures. This has motivated the development of a hybrid protocol using a game theoretic formalism that provides a balance between input rejection with respect to the mean and variance measures, particularly useful in security scenarios. All protocols perform edge swaps (rewiring) which can be executed in parallel, asynchronously, and require only local agent information of the network structure. The protocols are applied to two motivating applications, namely, time synchronization and UAV flocking.

The paper is organized as follows. Section II contains the problem formulation and relevant background. The mean tracking measure is examined in Section III and its relationship to the effective resistance is established and subsequently used to design two protocols for increasing and decreasing the mean tracking measure. A similar treatment of the variance damping measure is presented in Section IV. Section V presents a more versatile protocol that provides guarantees on both measures analyzed using game theoretic techniques. We conclude the paper with a few remarks in Section VI.

II. Background and Model

We provide a brief background on constructs that will be used in this paper, including abbreviated descriptions on graphs and the consensus protocol for its unforced and forced versions. First we introduce the notation: $\|\cdot\|$ denotes the Euclidean norm; $\mathbf{tr}(\cdot)$ denotes the trace of a matrix; $|\cdot|$ denotes the cardinality of a set; and $\lambda_i(M)$ denotes the ith smallest eigenvalue of the symmetric matrix M.

An undirected graph $\mathcal{G} = (V, E)$ is defined by a node set V with cardinality n, i.e., the number of nodes in the graph, and an edge set E comprised of pairs of distinct nodes, where nodes v_i and v_j are adjacent if $\{v_i, v_j\} \in E \subseteq [V]^2$.[2] A special family of graphs of particular interest in our subsequent discussions is the set of tree graphs, denoted by $\mathcal{T}$, comprised of connected graphs without cycles. Within this family are the path graph $\mathcal{P}$, where $\{v_i, v_{i+1}\} \in E$ for $i = 1, \ldots, n-1$, and the star graph $\mathcal{S}$, where $\{v_1, v_i\} \in E$ for $i = 2, \ldots, n$.

We denote the set of nodes adjacent to v_i as $\mathcal{N}(v_i)$ and the minimum path length, induced by the graph $\mathcal{G}$, between nodes v_i and v_j as $d(v_i, v_j)$. The degree δ_i of node v_i is the number of its adjacent nodes. The degree matrix $\Delta(\mathcal{G}) \in \mathbb{R}^{n \times n}$ is a diagonal matrix with δ_i as its ith diagonal entry. The adjacency matrix is an $n \times n$ symmetric matrix with $[\mathcal{A}(\mathcal{G})]_{ij} = 1$ when $\{v_i, v_j\} \in E$ and $[\mathcal{A}(\mathcal{G})]_{ij} = 0$ otherwise. The combinatorial Laplacian is defined as $L(\mathcal{G}) = \Delta(\mathcal{G}) - \mathcal{A}(\mathcal{G}) \in \mathbb{R}^{n \times n}$ which is a (symmetric) positive semi-definite matrix. A subset of our results will be concerned with the spectrum of the graph Laplacian. This spectrum is assumed to be ordered as $0 = \lambda_1(\mathcal{G}) \leq \lambda_2(\mathcal{G}) \leq \ldots \leq \lambda_n(\mathcal{G})$, where, for brevity, we have used $\lambda_i(\mathcal{G})$ instead of $\lambda_i(L(\mathcal{G}))$.

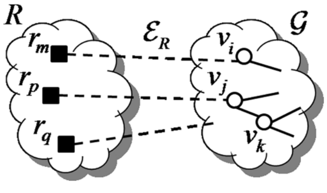


Fig. 2. Example of the notation used for semi-autonomous consensus.

Now consider $x_i(t) \in \mathbb{R}$ to be the ith node's (or for our case agent's) state at time t. The continuous-time consensus protocol is defined as $\dot{x}_i(t) = \sum_{\{i,j\} \in E}(x_j(t) - x_i(t))$. In a compact form with $x(t) \in \mathbb{R}^n$, the corresponding collective dynamics is represented as $\dot{x}(t) = -L(\mathcal{G})x(t)$, with $L(\mathcal{G})$ being the Laplacian of the underlying interaction topology [2].

We next introduce a model of *influenced* consensus associated with a pair $\mathcal{R} = (R, \mathcal{E}_R)$, where R is the r element external agent set and $\mathcal{E}_R \subseteq R \times V$ is the set of edges used by the external agents to inject signals into the network. It is assumed that each external agent $r_j \in R$ is attached to exactly one node $v_i \in V$ along one of the r edges $\{r_j, v_i\} \in \mathcal{E}_R$ and subsequently delivers a signal $u_j(t) \in \mathbb{R}$. Fig. 2 provides a graphical representation of the notation and setup.

The resulting influenced system now assumes the form

$$\dot{x}(t) = A(\mathcal{G}, \mathcal{R})x(t) + B(\mathcal{R})u(t) \qquad (1)$$

where $B(\mathcal{R}) \in \mathbb{R}^{n \times r}$ with $[B(\mathcal{R})]_{ij} = 1$ when $\{r_j, v_i\} \in \mathcal{E}_R$ and $[B(\mathcal{R})]_{ij} = 0$ otherwise, and

$$A(\mathcal{G}, \mathcal{R}) = -\left(L(\mathcal{G}) + M(\mathcal{R})\right) \in \mathbb{R}^{n \times n} \qquad (2)$$

where $M(\mathcal{R}) = B(\mathcal{R})B(\mathcal{R})^T \in \mathbb{R}^{n \times n}$. We also introduce a special type of single-agent control as $\mathcal{R}^i$ where $R = \{r_1\}$ and $\mathcal{E}_R = \{r_1, v_i\}$. Further, the set of agents v_i such that $\{r_j, v_i\} \in \mathcal{E}_R$ for some r_j will be denoted by $\pi(\mathcal{E}_R)$; this is the set of native agents that directly connect to external agents.

We recognize $A(\mathcal{G}, \mathcal{R})$ in (2) as the Dirichlet matrix, or grounded Laplacian [19], [25]. The spectrum of $A(\mathcal{G}, \mathcal{R})$ relates closely to the spectrum of $L(\mathcal{G})$. In this way, the structure of the underlying graph are related to the dynamics of model (1). An auxiliary observation on the Dirichlet matrix, to be used subsequently, is the following.

Proposition 2.1: [1] The matrix $A(\mathcal{G}, \mathcal{R})$ of model (1) is negative definite (and thus invertible) if the original graph is connected.

We approach the network performance and security problem from two fronts; first via the cost to the network for its agents to track a constant signal—dubbed *mean tracking* measure (discussed in Section III) and secondly, as the cost to the network to dampen a noisy external agent's signal—dubbed *variance damping* measure (discussed in Section IV). The following two sections will focus on these measures.

Remark 2.2: A popular model for network intrusion or faults is to consider native agents as those ignoring consensus instead of attaching external agents [8], [9], [17]. This model can be

[2] The notation $[V]^2$ refers to the set of two-element unordered subsets of V.

adapted by considering the subgraph of behaving agents as $\mathcal{G}$ and each edge between the misbehaving agents and $\mathcal{G}$ as $\{r_j, v_i\} \in \mathcal{E}_R$ corresponding, in our model, to a behaving agent v_i at one end and an external agent $r_j \in R$ at the other. The presented analysis is therefore applicable to both models. As this paper has a particular focus on tree graphs it is worthwhile mentioning that if the original graph is a tree then $\mathcal{G}$ in (1) will be the union of tree graphs. The converse is not true; the original graph need not be a tree if $\mathcal{G}$ in (1) is a tree.

III. Mean Tracking Measure

The mean tracking measure is a metric for the effectiveness of a network to track a constant external agents' signal $u_c \in \mathbb{R}$. This metric is equally applicable to the network's performance in regard to tracking the mean u_c of the external agents' noisy signal, e.g., Gaussian noise with mean u_c; hence the metric's name *mean* tracking. We derive the mean tracking measure as the cost incurred by external agents to steer the mean of the agents' state to u_c over an infinite horizon. In order to quantify the performance and security of the network to resist the influence of external agents injecting noisy signals, the following two observations are in order: (i) the dynamics of the state mean is captured by model (1) where u_i for all external agents i is replaced by the mean of external agents' signal u_c, (ii) when the underlying graph is connected, all agents' state converge in the mean to u_c. The last statement is a direct consequence of Proposition 2.1. More specifically, noting that $\mathbf{1} = [1, \dots, 1]^T$, $\mathbf{1}_x \in \mathbb{R}^n$, $\mathbf{1}_u \in \mathbb{R}^r$, $u(t) \equiv \mathbf{1}_x u_c$ and $A(\mathcal{G}, \mathcal{R})^{-1} B \mathbf{1}_u = -\mathbf{1}_x$, the quadratic performance cost of the mean, with coordinate change $\tilde{x}(t) = \mathbb{E}(x(t)) - u_c \mathbf{1}_x$, where $[\mathbb{E}(x(t))]_i$ is the expected value of the variable $x_i(t)$ at time t, can be derived as,[3]

$$\begin{aligned}
2\int_0^{t_f} & \tilde{x}(t)^T \tilde{x}(t) dt \\
&= \int_0^{t_f} \tilde{x}(t)^T x(t) + x(t)^T \tilde{x}(t) - \tilde{x}(t)^T \mathbf{1}_x u_c - u_c \mathbf{1}_x^T \tilde{x}(t) dt \\
&= \int_0^{t_f} \tilde{x}(t)^T x(t) + x(t)^T \tilde{x}(t) + \tilde{x}(t)^T A^{-1} B \mathbf{1}_u u_c \\
&\qquad + u_c \mathbf{1}_u^T B^T A^{-1} \tilde{x}(t) dt \\
&= \int_0^{t_f} \left(Ax(t) + B\mathbf{1}_u u_c\right)^T A^{-1} \tilde{x}(t) \\
&\qquad + \tilde{x}(t)^T A^{-1} \left(Ax(t) + B\mathbf{1}_u u_c\right) dt \\
&= \int_0^{t_f} \dot{x}(t)^T A^{-1} \tilde{x}(t) + \tilde{x}(t)^T A^{-1} \dot{x}(t) dt \\
&= \int_0^{t_f} \frac{d}{dt} \tilde{x}(t)^T A^{-1} \tilde{x}(t) dt \\
&= \tilde{x}(t_f)^T A^{-1} \tilde{x}(t_f) - \tilde{x}(0)^T A^{-1} \tilde{x}(0)
\end{aligned}$$

where we have used A instead of $A(\mathcal{G}, \mathcal{R})$ for brevity.

In order to parametrize the performance and security of the network for a specific set $\mathcal{R}$, let us define the accumulative state mean over the length of time the input is applied t_f as

$$\begin{aligned}
&J_\mu(\mathcal{G}, \mathcal{R}, t_f) \\
&\quad = \mathbb{E}_{\|\tilde{x}(0)\|=1} \left(2 \int_0^{t_f} \tilde{x}(t)^T \tilde{x}(t) dt \right) \\
&\quad = \mathbb{E}_{\|\tilde{x}(0)\|=1} \left(\tilde{x}(t_f)^T A^{-1} \tilde{x}(t_f) - \tilde{x}(0)^T A^{-1} \tilde{x}(0) \right) \\
&\quad = \mathbb{E}_{\|\tilde{x}(0)\|=1} \mathbf{tr} \left(\tilde{x}(0) \tilde{x}(0)^T \left(\left(e^{At_f}\right)^T A^{-1} e^{At_f} - A^{-1} \right) \right) \\
&\quad = \mathbb{E}_{\|\tilde{x}(0)\|^2=n} \mathbf{tr} \left(\frac{1}{\sqrt{n}} \tilde{x}(0) \frac{1}{\sqrt{n}} \tilde{x}(0)^T (e^{At_f} A^{-1} e^{At_f} - A^{-1}) \right) \\
&\quad = \frac{1}{n} \mathbf{tr} \left(\left(\mathbb{E}_{\|\tilde{x}(0)\|^2=n} \tilde{x}(0) \tilde{x}(0)^T \right) (e^{At_f} A^{-1} e^{At_f} - A^{-1}) \right) \\
&\quad = \frac{1}{n} \mathbf{tr} \left((e^{2At_f} - I) A^{-1} \right) \\
&\quad = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\lambda_i(-A)} \left(1 - e^{-2\lambda_i(-A) t_f} \right)
\end{aligned}$$

where $\mathbb{E}_{\|\tilde{x}(0)\|=1}(\cdot)$ denotes the expected value over all initial conditions satisfying $\|\tilde{x}(0)\| = c$, for constant c satisfying $\|\tilde{x}(0)\| = 1$.

It is assumed that the native agents in the network do not know the value of t_f; as such, in the remaining parts of our paper we assume that t_f is large, justifying the use of $J_\mu(\mathcal{G}, \mathcal{R}, \infty)$ as the mean tracking measure. In fact for brevity, we denote $J_\mu(\mathcal{G}, \mathcal{R}, \infty)$ as $J_\mu(\mathcal{G}, \mathcal{R})$, which we now formally define.

Definition 3.1: The *mean tracking* measure of a network is the average quadratic performance cost incurred by external agents to steer the mean state of the network to their mean value, over an infinite horizon, and is equal to

$$J_\mu(\mathcal{G}, \mathcal{R}) = \frac{1}{n} \mathbf{tr} \left(-A(\mathcal{G}, \mathcal{R})^{-1} \right). \tag{3}$$

Remark 3.2: We briefly note the connection between the more familiar best case convergence rate of the grounded dynamics, the average convergence rate of the grounded dynamics and the mean tracking measure, i.e., the minimum nonzero eigenvalue of the Laplacian, the average of the eigenvalues of the Laplacian and the measure $J_\mu(\mathcal{G}, \mathcal{R})$ (3). Consider fusing all external agents to form a node v_{n+1} and adding it to our graph $\mathcal{G}$, connecting v_{n+1} to the network through "directed" edges from v_{n+1} to each node in $\pi(\mathcal{E}_R)$; we call this new "directed" graph $\tilde{\mathcal{G}}$.[4] Then $\mathcal{G}$ and $\tilde{\mathcal{G}}$ have the property $\lambda_2(\tilde{\mathcal{G}}) \leq J_\mu(\mathcal{G}, \mathcal{R})^{-1} \leq (1/n) \sum_{i=1}^{n+1} \lambda_i(\tilde{\mathcal{G}})$, where $\lambda_i(\tilde{\mathcal{G}})$ refers to the ith eigenvalue of the in-degree Laplacian of $\tilde{\mathcal{G}}$, $L(\tilde{\mathcal{G}})$.[5] The following section will provide more insights into the mean tracking measure.

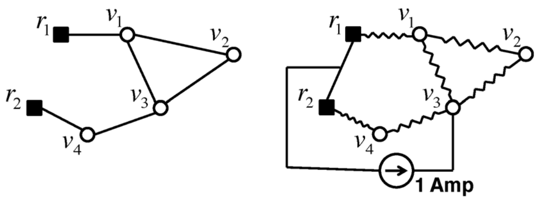
A. Analysis of Mean Tracking Measure

It has previously been established that the diagonal of the matrix $-A(\mathcal{G}, \mathcal{R})^{-1}$, has a resistive electrical network interpretation [19]. In this setup, the agents V and R, defined in Section II,

[3] The scaling by 2 is cosmetic.

[4] For a survey of directed graphs we refer the reader to [6].

[5] All eigenvalues of $L(\tilde{\mathcal{G}})$ are real, $\lambda_1(\tilde{\mathcal{G}}) = 0$ and $\lambda_{i+1}(\tilde{\mathcal{G}}) = \lambda_i(A(\mathcal{G}, \mathcal{R}))$ for $i = 1, \dots, n$.



(a) (b)

$$A(\mathcal{G},\mathcal{R}) = \begin{bmatrix} -3 & 1 & 1 & 0 \\ 1 & -2 & 1 & 0 \\ 1 & 1 & -3 & 1 \\ 0 & 0 & 1 & -2 \end{bmatrix}, \; B(\mathcal{R}) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Fig. 3. (a) Network graph with external (control) agents r_1 and r_2 attached to agents v_1 and v_4 respectively, leading to an altered Laplacian $A(\mathcal{G},\mathcal{R})$ and input matrix $B(\mathcal{R})$ of model (1). (b) Equivalent electrical network. The potential difference $V_{v_3} - V_{\mathcal{R}}$ is the effective resistance between v_3 and common resistor node $\{r_1, r_2\}$.

represent respectively, connection points between resistors corresponding to the edges E and $\mathcal{E}_R$. In addition, all connection points corresponding to the set R are electrically shorted. The effective resistance between two connection points in an electrical network is defined as the voltage drop between the two points, when a 1 Amp current source is connected across the two points. Then, the ith diagonal element of $-A(\mathcal{G},\mathcal{R})^{-1}$ is the effective resistance $E_{\text{eff}}(v_i)$ between the common shorted external agents R and v_i. An example of the equivalent electrical network is displayed in Fig. 3. The implication is that

$$J_\mu(\mathcal{G},\mathcal{R}) = \frac{1}{n}\sum_{i=1}^{n} E_{\text{eff}}(v_i). \tag{4}$$

Tree graphs are often adopted for agent-to-agent communication topologies as they minimize edge (communication) costs while maintaining connectivity. Using (4), we introduce some properties of $J_\mu(\mathcal{G},\mathcal{R})$ (3) specific to trees.

Let us first define the special set of agents that lie on any of the shortest paths between agents in $\mathcal{R}$ as the *main path* agents, designated by the set $\mathcal{M}$. This is a unique set for a given pair $(\mathcal{G},\mathcal{R})$. Moreover for all $v_i \notin \mathcal{M}$, there exists a unique $v_j \in \mathcal{M}$ that has a shorter minimum path to v_i than any other agent in $\mathcal{M}$; we define this agent as $\Gamma(v_i)$, i.e., $\Gamma(v_i)$ is the closest agent to v_i that is a member of the main path. Therefore, for tree graphs we can state the following.

Lemma 3.3: [Mean Tracking Measure for Trees] For the n-agent connected tree $\mathcal{T}$, the mean tracking measure is

$$J_\mu(\mathcal{T},\mathcal{R}) = \frac{1}{n}\left(\sum_{v_i\in\mathcal{M}} E_{\text{eff}}(v_i) + \sum_{v_i\notin\mathcal{M}} \left[E_{\text{eff}}\left(\Gamma(v_i)\right) + d\left(v_i,\Gamma(v_i)\right)\right]\right).$$

Proof: If $v_i \notin \mathcal{M}$ then the equivalent electrical network involving v_i can be simplified into a resistor representing $E_{\text{eff}}(\Gamma(v_i))$ ohms in series with $d(v_i,\Gamma(v_i)) \times 1\ \text{ohm}$ resistors. The result then follows from (4). ■

There is an intuitive link between the *centrality* of an agent in a network and its influence on the network's dynamics. This correlation becomes apparent for tree graphs in the following.

Corollary 3.4: [Single-External Mean Tracking Measure]: For the n-agent connected tree $\mathcal{T}$, the mean tracking measure of the network for a single external agent attached to any agent $v_i \in V$ is

$$J_\mu(\mathcal{T},\mathcal{R}^i) = \frac{1}{n}\left(\sum_{j=1}^{n} d(v_i,v_j) + n\right).$$

Proof: The proof follows from Lemma 3.3 with $\{v_i\} = \mathcal{M}$ and $E_{\text{eff}}(v_i) = 1$. ■

Corollary 3.4 has a few immediate ramifications. Consider the single external agent as a native node v_{n+1} of the graph $\mathcal{T}$ forming the new graph $\widetilde{\mathcal{T}}$ with $n+1$ nodes. The mean tracking measure $J_\mu(\mathcal{T},\mathcal{R}^i)$ is then equal to the closeness centrality measure of node v_{n+1},[6], i.e.

$$J_\mu(\mathcal{T},\mathcal{R}^i) = \frac{1}{n}\sum_{j=1}^{n} d(v_{n+1},v_j) = c(v_{n+1},\widetilde{\mathcal{T}}).$$

Further, as

$$J_\mu(\mathcal{T},\mathcal{R}^i) = \frac{n-1}{n}c(v_i,\mathcal{T}) + 1$$

the *most influential node* to attach in a tree graph under the measure J_μ (3) is the one with the largest closeness centrality measure.

Corollary 3.5: [Single-External Mean Tracking Measure Bounds] For the n-agent connected tree $\mathcal{T}$, the mean tracking measure of the network for a single external agent attached to any agent $v_i \in V$ is bounded as $2 - (1/n) \le J_\mu(\mathcal{T},\mathcal{R}^i) \le (1/2)(n+1)$.

Proof: Over all trees, the central node of the star graph has the smallest accumulative distance of $n-1$ to all other nodes and an end node of the path graph has the largest accumulative distance of $\sum_{i=1}^{n-1} i$ to all other nodes. The statement of the corollary follows from these two observations. ■

Proposition 3.6: [Multi-External Mean Tracking Measure Bounds] For the n-agent connected tree $\mathcal{T}$, the mean tracking measure for r external agents attached to any set of agents in V is bounded above by a tree graph with all main path nodes satisfying $v_i \in \pi(\mathcal{E}_R)$, in which case

$$J_\mu(\mathcal{T},\mathcal{R}) \le \frac{1}{2n}\left((n-r)^2 + 3(n-r) + r + \frac{2}{r+1}\right).$$

Proof: From the effective resistance interpretation of $J_\mu(\mathcal{T},\mathcal{R})$ in (4), we note that adding resistors in series generates a higher resistance than adding them in parallel. Therefore, $\arg\max_{(\mathcal{T},\mathcal{R})} J_\mu(\mathcal{T},\mathcal{R})$ is a tree $\mathcal{T}$ and the influence set $\mathcal{R}$, where $\mathcal{M} = \pi(\mathcal{E}_R)$, as adding an agent to the main path places resistors in parallel rather than the alternative which places them in series. Denote this family of graphs with $\mathcal{M} = \pi(\mathcal{E}_R)$ as $\mathcal{H}$. Furthermore from Lemma 3.3, the largest accumulative distance of these nodes v_i will correspond to a path connected to the highest effective resistance node of the main path subgraph. Now, the main path subgraph of a tree in $\mathcal{H}$ with the largest effective resistance sum, is the star graph as the equivalent

[6] Closeness centrality $c(v_i,\mathcal{G})$ is the mean of the shortest path lengths between node v_i and other nodes in the graph $\mathcal{G}$.

electrical network has the least number of parallel resistors. Applying resistor rules, we thus obtain

$$\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{H}} \sum_{v_i\in\mathcal{M}} E_{\text{eff}}(v_i) = \frac{r^2+r+2}{2(r+1)}. \tag{5}$$

Similarly, the largest $E_{\text{eff}}(v_j)$ for any single node $v_j \in \mathcal{M}$ of a tree in $\mathcal{H}$ corresponds to the main path subgraph, and thus

$$\begin{aligned}\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{H}, v_j\in\mathcal{M}} E_{\text{eff}}(v_j) &= \frac{F_{2r-1}}{F_{2r}}\\ &= \frac{\phi^{2r-1}-(-1/\phi)^{2r-1}}{\phi^{2r}-(-1/\phi)^{2r}} \le 1\end{aligned}$$

where F_i is the ith Fibonacci number and φ is the golden ratio.[7] The largest effective resistance sum over trees in $\mathcal{H}$ of a non-main path subgraph can now be formed from a path attached to an end node of the main path subgraph, i.e.

$$\begin{aligned}&\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{H}} \sum_{v_i\notin\mathcal{M}} E_{\text{eff}}(v_i)\\ &= \sum_{i=1}^{n-r}\left[\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{H}, v_j\in\mathcal{M}} E_{\text{eff}}(v_j)+1\right]\\ &\le \sum_{i=1}^{n-r}(1+i) = (n-r)^2+3(n-r).\end{aligned} \tag{6}$$

Using bounds (5) and (6) combined with (4) we have

$$\begin{aligned}&J_\mu(\mathcal{T},\mathcal{R})\\ &\le \frac{1}{n}\left(\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{H}} \sum_{v_i\in\mathcal{M}} E_{\text{eff}}(v_i) + \max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{H}} \sum_{v_i\notin\mathcal{M}} E_{\text{eff}}(v_i)\right)\\ &= \frac{1}{2n}\left((n-r)^2+3(n-r)+r+\frac{2}{r+1}\right).\end{aligned}$$

■

B. Adaptive Protocols to Improve or Degrade the Mean Tracking Measure for Trees

We now propose a pair of protocols applicable to tree graphs that locally trade edges, e.g., communication links, between adjacent agents with the objective of deterring or encouraging the influence of external agents attached to the network, feeding in a *constant mean signal*. We consider a scenario where agents in $\pi(\mathcal{E}_R)$ broadcast acknowledgment signals informing the network that they are being favorably or unfavorably influenced. Consequently all agents within the graph are aware of the "local" directions of the external agents, and more specifically, their neighboring agents that are closer to these external agents. We denote by $\mathcal{I}(v_i)$ the set of all agents that are neighbors of v_i and lie on the shortest path between v_i and any $r_j \in R$. Formally

$$\mathcal{I}(v_i) := \{v \in \mathcal{N}(v_i) \mid \exists r \in R,\ d(v,r) < d(v_i,r)\}.$$

We emphasize that we assume that the external agents in set R are solely composed of friendly or unfriendly agents and agents are able to distinguish between the external agents' intent.[8]

The following lemma describes **Protocol 1** which can be executed by an arbitrary agent v_i and requires the knowledge of $\mathcal{I}(v_i)$ and $\mathcal{N}(v_i)$; hence the protocol operates on "local" information. In the following, we denote edge removal and addition by the set notation "$-/+$."

Protocol 1 Increased mean tracking measure edge swap

```
for all Agent $v_i$ do
    if $\exists v_j, v_k \in \mathcal{N}(v_i)$, $v_j \neq v_k$ and $v_j, v_k \notin \mathcal{I}(v_i)$ then
        $E \rightarrow E - \{v_i, v_j\} + \{v_j, v_k\}$
    end if
end for
```

Lemma 3.7: [Edge Swap for Improved Mean Tracking Measure] Under **Protocol 1**, $J_\mu(\mathcal{T},\mathcal{R})$ (3) is strictly increasing.

Proof: If $v_m \in \mathcal{M}$ then for all $v_l \in \mathcal{N}(v_m)$ we have $v_m \in \mathcal{I}(v_l)$. Therefore in the context of **Protocol 1**, $v_j, v_k \notin \mathcal{M}$. Thus from Lemma 3.3, before the edge swap, we have $E_{\text{eff}}(v_j) = E_{\text{eff}}(v_k) = E_{\text{eff}}(v_i) + 1$, and after the edge swap $E_{\text{eff}}(v_j) = E_{\text{eff}}(v_i) + 2$. Any agent v_p such that v_j lies on the shortest path between v_p and any agent in $\mathcal{M}$, will increase its effective resistance by 1 after the edge swap. Since the effective resistance strictly increases or stays the same for all agents following the edge swap, $J_\mu(\mathcal{T},\mathcal{R})$ increases. ■

Some attractions of **Protocol 1** is that it can be executed concurrently or in a random agent order, guarantees that $J_\mu(\mathcal{T},\mathcal{R})$ increases, and maintains a connected tree at each iteration. This is attained without the knowledge of the global network topology.

When all agents adopt **Protocol 1**, trees with a single attached external agent will eventually evolve to a graph with the greatest $J_\mu(\mathcal{T},\mathcal{R}^1) = (n+1)/2$, namely a path graph with the external agent at one end. Trees with multiple external agents will acquire a path-like appearance with the main path unaffected by the protocol's edge swaps.

Protocol 1 was applied to a random tree graph on 40 agents with a single external agent connected to v_1. The path graph with the external agent attached to its end node was achieved after 100 edge swaps. A sample of the intermediate graphs, the mean tracking measure over all iterations, and the evolution of the state mean are displayed in Figs. 4 and 5. The network measure $J_\mu(\mathcal{T},\mathcal{R}^1)$ increased for each edge swap and no more edge swaps were possible when the tree became a path graph with $J_\mu(\mathcal{T},\mathcal{R}^1) = 20.5$.

A complementary **Protocol 2** that aims to decrease $J_\mu(\mathcal{T},\mathcal{R})$ can also be obtained from Lemma 3.7. Under this protocol the graph converges to a star-like graph, while preserving the structure of the main path. The protocol was run on a 40 agent random tree graph with 3 external agents. The original and final graphs, achieved after 21 edge swaps, are displayed in Fig. 6.

[7] The derivation follows from that of the infinite connected resistor network and the recursive definition of the golden ratio $\phi = (1/2)(1+\sqrt{5})$. One has $F_i = (1/\sqrt{5})(\phi^i - (-1/\phi)^i)$.

[8] For unfavorable detection an algorithm such as those proposed in [9], [15]–[17] can be used.

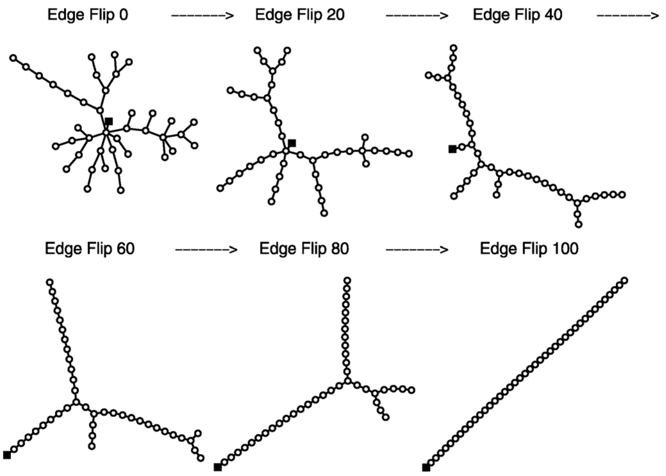
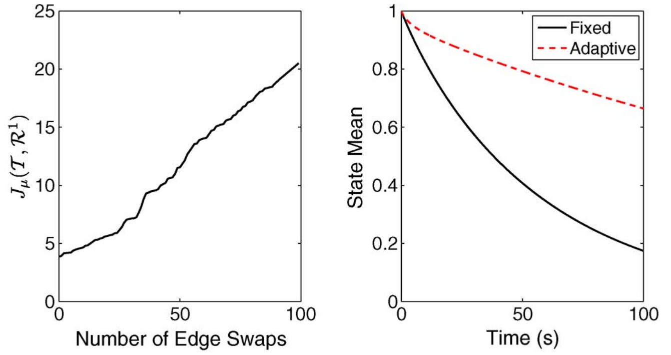
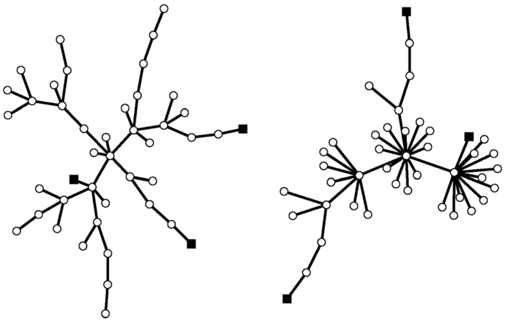


Fig. 4. Selected iterations of an adaptive tree graph running **Protocol 1** with an external agent attached (square).



(a) (b)

Fig. 5. (a) Mean tracking measure and (b) state mean for the fixed and adaptive tree graphs over time for the 40 agent random tree graph in Fig. 4 running **Protocol 1**.

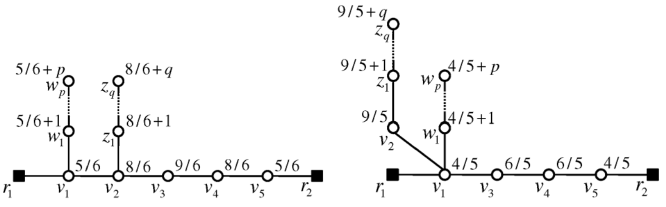


(a) (b)

Fig. 6. (a) Original and (b) final tree graphs with three external agents attached (squares) after applying **Protocol 2**.

Protocol 2 Decreased mean tracking measure edge swap

```
for all Agent $v_i$ do
    if $v_k = \mathcal{I}(v_i)$, $\exists v_j \in \mathcal{N}(v_i)$ and $v_j \neq v_k$ then
        $E \rightarrow E - \{v_i, v_j\} + \{v_j, v_k\}$
    end if
end for
```



(a) (b)

Fig. 7. Tree graphs, (a) $\mathcal{T}_1$ and (b) $\mathcal{T}_4$, with two attached external agents $\{r_1, r_2\}$. The effective resistance $E_{\text{eff}}(v_i)$ appears adjacent to each agent. The variables p and q are the lengths of the paths connected to agent v_1 and v_2, respectively.

Remark 3.8: For agent v_i having access to the "local" information provided by $\mathcal{I}(v_i)$ and $\mathcal{N}(v_i)$, Lemma 3.7 describes the *only* edge swaps protocol that guarantee $J_\mu(\mathcal{T},\mathcal{R})$ increases and a connected tree is maintained. Let us illustrate this by examining edge swap protocols *not* covered by Lemma 3.7; for these edge swap cases v_j and/or v_k can be main path agents, i.e., swaps involving $v_j \in \mathcal{I}(v_i)$ and/or $v_k \in \mathcal{I}(v_i)$. Consider the tree graph $\mathcal{T}_1 = (V, E_1)$ displayed in Fig. 7(a). We note that

$$J_\mu(\mathcal{T}_1,\mathcal{R}^{1,5}) = \frac{1}{n}\left(\frac{35}{6}+\frac{5}{6}p+\frac{8}{6}q+\frac{p}{2}(p+1)+\frac{q}{2}(q+1)\right)$$

where p and q are the lengths of the paths incident on agents v_1 and v_2, respectively.

Let us consider the potential edge swaps available to agent $v_2 \in \mathcal{M}$. Locally, agent v_2 is aware that $\mathcal{N}(v_2) = \{z_1, v_1, v_3\}$ and $\mathcal{I}(v_2) = \{v_1, v_3\}$. The potential edge swaps cases available are:

1) One neighbor on and one off the main path, e.g., swap $E_2 = E_1 - \{v_2, z_1\} + \{z_1, v_1\}$, forming $\mathcal{T}_2 = (V, E_2)$ and $E_3 = E_1 - \{v_2, z_1\} + \{z_1, v_3\}$ forming $\mathcal{T}_3 = (V, E_3)$.
2) Both neighbors on the main path, e.g., swap $E_4 = E_1 - \{v_2, v_3\} + \{v_3, v_1\}$ forming $\mathcal{T}_4 = (V, E_4)$.

Under Case 1, we have

$$J_\mu(\mathcal{T}_2,\mathcal{R}^{1,5}) = \frac{1}{n}\left(\frac{35}{6}+\frac{5}{6}p+\frac{9}{6}q+\frac{p}{2}(p+1)+\frac{q}{2}(q+1)\right),$$
$$J_\mu(\mathcal{T}_3,\mathcal{R}^{1,5}) = \frac{1}{n}\left(\frac{35}{6}+\frac{5}{6}p+\frac{5}{6}q+\frac{p}{2}(p+1)+\frac{q}{2}(q+1)\right)$$

and $J_\mu(\mathcal{T}_3,\mathcal{R}^{1,5}) < J_\mu(\mathcal{T}_1,\mathcal{R}^{1,5}) < J_\mu(\mathcal{T}_2,\mathcal{R}^{1,5})$.

As under "local" information v_1 and v_3 are indiscernible, Case 1 does not guarantee that $J_\mu(\mathcal{T},\mathcal{R})$ is increasing or decreasing.

In the meantime, under Case 2, we are led to graph $\mathcal{T}_4$ as displayed in Fig. 7(b), with

$$J_\mu(\mathcal{T}_4,\mathcal{R}^{1,5}) = \frac{1}{n}\left(\frac{29}{5}+\frac{4}{5}p+\frac{9}{5}q+\frac{p}{2}(p+1)+\frac{q}{2}(q+1)\right)$$

and $J_\mu(\mathcal{T}_1,\mathcal{R}^{1,5}) - J_\mu(\mathcal{T}_4,\mathcal{R}^{1,5}) = (1/30n)(1+p-14q)$. Thus

$$\begin{array}{ll} J_\mu(\mathcal{T}_1,\mathcal{R}^{1,5}) > J_\mu(\mathcal{T}_4,\mathcal{R}^{1,5}) & \text{if } p > 14q - 1 \\ \qquad\qquad\quad \leq J_\mu(\mathcal{T}_4,\mathcal{R}^{1,5}) & \text{otherwise.} \end{array}$$

Under only "local" information, the relative magnitudes of p and q cannot be discerned so no monotonicity guarantees may be assumed.

A by-product of this remark is that a strictly increasing local-knowledge protocol cannot guarantee the tree graph with the largest $J_{\mu}(\mathcal{T},\mathcal{R})$ (3) for $r > 1$ external agents.

C. Example: Clock Synchronization

Clock synchronization is often necessary in many distributed systems, improving the consistency of data and the correctness of algorithms. Precise time synchronization is needed for distributed applications such as sensor data fusion, scheduling, localization, coordinated actuation and power-saving duty cycling. Motivated by the work of [7], we assembled the following experiment.

Consensus on clock time was run on 100 decentralized computer terminals (agents) communicating over a tree network. Because time consensus can only correct for differential errors between terminals and not absolute errors without a reference, friendly external agents periodically connect to the network and deliver the constant correction for the absolute bias in the system. Upon connection, the friendly external agents initiate a friendly flag which is passed through the network, providing the local direction of the friendly agents and initiating **Protocol 2**. The network adapts under this protocol to promote convergence to the correct absolute clock time. On disconnection, the agents initiate a disconnect flag.

Similarly, we introduce a malicious external agent that attempts to drive the system to a false absolute time. Upon connection, the neighbors of the external agents send out a distress signal triggering the network to initiate **Protocol 1** so as to deter the false convergence of the network. It is assumed that the friendly agents on discovery of a malicious external agent will clear the network of these foreign agents and trigger the termination of **Protocol 1** before commencing delivery of the correction signal again. In other words, we assume friendly and malicious agents would not be concurrently connected to the network.

To examine the performance of the protocols, equal access time was provided for both friendly and malicious external agents, specifically alternating 100 s intervals for 5000 s. This switching interval is long enough for transients to settle and is appropriate for the application of these protocols. The network was initialized as a random tree with all agents at the time offset of 0 s (the correct offset is -1 s). The set $\pi(\mathcal{E}_R)$ of agents connected to 3 external agents is randomly selected at each new 10 s interval. The friendly and unfriendly external agents deliver time offsets of -1 s and 1 s, respectively. The average of the constant values, i.e., 0 s, would be expected for the mean offset without the application of the protocols. In the meantime, the protocols are able to favor the friendly agent, bringing the average offset to -0.26 s. Clock offset means are displayed for the first 1000 s for the fixed and adaptive trees in Fig. 8.

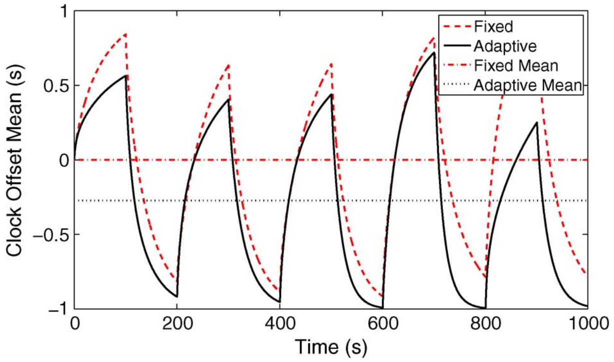


Fig. 8. Clock offset mean of the fixed and adaptive tree graphs (running **Protocols 1** and **2**) and the corresponding mean state. Friendly and unfriendly agents alternate delivering -1 s and 1 s offsets, respectively for 100 s intervals.

IV. Variance Damping Measure

It can be the case that the mean is not of central interest and that adjustment of the variance of the states may be more desirable. Further, motivated by devious intrusion type techniques that may employ pulse-like control to avoid triangulation, the energy of the states from a unit impulse input is another potentially desirable indicator for network performance and security. With this in mind, the controllability gramian, defined as $P =: \int_0^{\infty} e^{A\tau}BB^Te^{A^T\tau}d\tau$ for the system $\dot{x}(t) = Ax(t) + Bu(t)$, proves to be particularly suitable for such an analysis. We will focus on, $\mathbf{tr}(P)$ as: (a) the average variance of the agents' state is

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left(z_i^2(t)\right) = \frac{1}{n}\mathbf{tr}\left(\mathbb{E}\left[z(t)z^T(t)\right]\right) = \frac{1}{n}\mathbf{tr}(P)$$

as $t \to \infty$, where $z(t) = x(t) - u(t)$ and u is a zero mean Gaussian with covariance I, and (b) the energy of the states at the output from a unit impulse input u when $x(0) = 0$ is

$$\int\limits_0^{\infty} x(t)^T x(t)dt = \mathbf{tr}(P).$$

We note that the controllability gramian for (1) will be dependent on $\mathcal{G}$ and $\mathcal{R}$ and henceforth is denoted by $P(\mathcal{G},\mathcal{R})$. The *variance damping* measure is a metric quantifying the network's susceptibility to white noise from external agents.

Definition 4.1: The *variance damping* measure of a network is defined as[9]

$$J_{\sigma}(\mathcal{G},\mathcal{R}) = \frac{2}{n}\mathbf{tr}\left(P(\mathcal{G},\mathcal{R})\right). \tag{7}$$

The following section will provide more insights into the variance damping measure (7).

A. Analysis of Variance Damping Measure

Directly from the definition of the controllability gramian one has

$$\begin{aligned} J_{\sigma}(\mathcal{G},\mathcal{R}) &= \frac{2}{n}\mathbf{tr}\left(P(\mathcal{G},\mathcal{R})\right) \\ &= \frac{2}{n}\mathbf{tr}\left(\int\limits_0^{\infty} e^{A(\mathcal{G},\mathcal{R})\tau}B(\mathcal{R})B(\mathcal{R})^T e^{A(\mathcal{G},\mathcal{R})^T\tau}d\tau\right) \\ &= \frac{2}{n}\mathbf{tr}\left(M(\mathcal{R})\int\limits_0^{\infty} e^{2A(\mathcal{G},\mathcal{R})\tau}d\tau\right) \\ &= -\frac{1}{n}\mathbf{tr}\left(M(\mathcal{R})A(\mathcal{G},\mathcal{R})^{-1}\right). \end{aligned} \tag{8}$$

[9] The scaling by $2/n$ is cosmetic.

Lemma 4.2: [General Variance Damping Measure] For a connected graph $\mathcal{G}$, the variance damping measure is

$$J_\sigma(\mathcal{G}, \mathcal{R}) = \frac{1}{n} \sum_{v_i \in \pi(\mathcal{E}_R)} E_{\text{eff}}(v_i). \tag{9}$$

Proof: We note that $M(\mathcal{R})$ is a diagonal matrix with $[M(\mathcal{R})]_{ii} = 1$ if $v_i \in \pi(\mathcal{E}_R)$ and $[M(\mathcal{R})]_{ii} = 0$, otherwise. Therefore

$$\left[M(\mathcal{R})A(\mathcal{G},\mathcal{R})^{-1}\right]_{ii} = \begin{cases} \left[A(\mathcal{G},\mathcal{R})^{-1}\right]_{ii} & \text{if } v_i \in \pi(\mathcal{E}_R) \\ 0 & \text{otherwise.} \end{cases}$$

The statement of the lemma now follows. ■

Corollary 4.3: [Single-External Variance Damping Measure] For a connected graph and the influence model (1) with one external agent

$$J_\sigma(\mathcal{G}, \mathcal{R}^i) = \frac{1}{n}.$$

Proof: The equivalent effective resistance between v_i and r_1 with $\{v_i\} = \pi(\mathcal{E}_R)$ is $E_{\text{eff}}(v_i) = 1$ as there is only one resistor link between v_i and r_1. The statement of the corollary now follows. ■

Remark 4.4: The implication of Corollary 4.3 is that on average, a single-external agent attached to an n-agent connected network has the same reduction in average variance to white noise and energy dissipation from an impulse input regardless of the structure of the network and where the external agent is attached.

Proposition 4.5: [Multiple-External Variance Damping Measure]: For connected graphs and the influence model (1) with r external agents, the variance damping measure is bounded below by a graph with all main path nodes satisfying $v_i \in \pi(\mathcal{E}_R)$, in which case

$$J_\sigma(\mathcal{G}, \mathcal{R}) \geq \frac{r}{\sqrt{5}n}.$$

Proof: By Rayleigh's Monotonicity Principle[10] the minimum effective resistance will occur when the main path is only composed of the r agents $\pi(\mathcal{E}_R)$. Of these r agent graphs, the path graph with the most resistors in parallel will have the smallest effective resistance and therefore the smallest value of $J_\sigma(\mathcal{G},\mathcal{R})$. The eigenvalues of the Laplacian of an r-node path graph are $\lambda_{r+1-i}(\mathcal{P}) = 2 + 2\cos(\pi i/r)$, for $i = 1, \ldots, r$ [27]. For $\tilde{\mathcal{R}}$ corresponding to an external agent attached to every agent in $\mathcal{P}$, from (2) and $M(\tilde{\mathcal{R}}) = I$, it follows that $\lambda_{r+1-i}(-A(\mathcal{P},\tilde{\mathcal{R}})) = \lambda_{r+1-i}(\mathcal{P}) + 1$. Hence from (8), we conclude that

$$\begin{aligned} J_\sigma(\mathcal{G},\mathcal{R}) &\geq \frac{1}{n}\mathbf{tr}\left(-A(\mathcal{P},\tilde{\mathcal{R}})^{-1}\right) \\ &= \frac{1}{n}\sum_{i=1}^{r} \frac{1}{3 + 2\cos\frac{\pi i}{r}} \geq \frac{r}{\sqrt{5}n}. \end{aligned}$$

■

B. Adaptive Protocols to Improve or Degrade the Variance Damping Measure for Trees

We now propose another protocol for tree graphs with the objective of reducing the state variance due to external agents attached to the network and feeding in Gaussian white noise with covariance I, i.e., decreasing the variance damping measure (7). Again the protocol involves local edge trades executed concurrently and/or in a random agent order, guarantees that $\mathbf{tr}(P(\mathcal{T},\mathcal{R}))$ decreases, and maintains a connected tree at each iteration. A complementary protocol to increase the variance damping measure is also proposed.

We note that for a connected tree graph $\mathcal{T}$, $J_\sigma(\mathcal{T},\mathcal{R})$ is only dependent on $d(r_i, r_j)$ for all $\{r_i, r_j\}$ pairs in the set R (as defined in Section II), and so only dependent on the main path with agent set $\mathcal{M}$ (as defined in Section III-A).

Lemma 4.6: [Edge Swap for Decreased Variance Damping Measure] Under **Protocol 3**, $J_\sigma(\mathcal{T},\mathcal{R})$ (7) monotonically decreases.

Protocol 3 Decreased variance damping measure edge swap

```
for all Agent $v_i \notin \pi(\mathcal{E}_R)$ do
    if $\{v_j, v_k\} = \mathcal{I}(v_i)$ where $v_j \neq v_k$ then
        $E \to E - \{v_i, v_j\} + \{v_j, v_k\}$
    end if
end for
```

Proof: Firstly, when $|\mathcal{I}(v_i)| = 2$, $v_i \in \mathcal{M}$. As v_j and v_k are closer to an external agent than the main path agent v_i, one has $v_j, v_k \in \mathcal{M}$. The edge swap involves removing v_i from $\mathcal{M}$, so the effect is to reduce the resistance of an edge within the electrical network representing this subgraph. By Rayleigh's Monotonicity Law, the sum $\sum_{\{v_i,r_j\}\in\mathcal{E}_R} E_{\text{eff}}(v_i)$ will not increase and the lemma follows. ■

Single-external agent trees will remain unaffected by **Protocol 3**. For double-external agent trees, the main path will degenerate to $\{v_i, v_j\} = \mathcal{M}$, where $\mathcal{E}_R = \{\{v_i, r_1\}, \{v_j, r_2\}\}$.

Protocol 3 was run on a 40 agent random tree with 3 external agents injecting zero mean white noise to the network. The original and final graphs, the variance damping measure, and a sample output comparison between the fixed and adaptive networks (running **Protocol 3**) are displayed in Figs. 9 and 10.

A complementary energy amplification **Protocol 4**, that aims to increase $\mathbf{tr}(P(\mathcal{T},\mathcal{R}))$, can also be obtained from Lemma 3.7. This protocol is suitable for impulse detection as a larger $J_\sigma(\mathcal{T},\mathcal{R})$ produces higher output energy.

Protocol 4 Increased variance damping measure edge swap

```
for all Agent $v_i$ do
    if $|\mathcal{I}(v_i)| > 1$ and $\exists v_j, v_k \in \mathcal{N}(v_i)$, $v_j \in \mathcal{I}(v_i)$ and
    $v_k \notin \mathcal{I}(v_i)$ then
        $E \to E - \{v_i, v_j\} + \{v_j, v_k\}$
    end if
end for
```

[10] Rayleigh's Monotonicity Law states that if the edge resistance in an electrical network is decreased, then the effective resistance between any two agents in the network can only decrease [26].

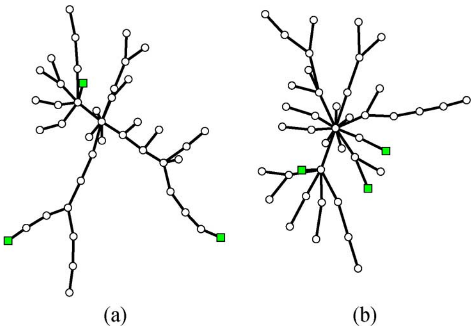


Fig. 9. (a) Original and (b) final tree graphs with three external agents attached (squares) after applying **Protocol 3**.

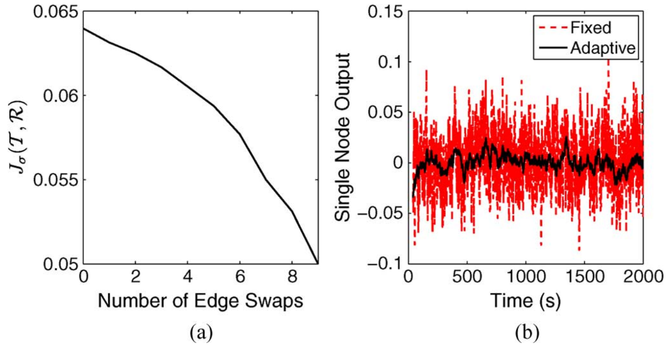


Fig. 10. (a) Variance damping measure and (b) one of the state node's output for the fixed and adaptive tree graphs over time for the 40 agent random tree graph in Fig. 9 exposed to 3 external agents running **Protocol 3**.

Remark 4.7: For the case where $|\mathcal{I}(v_i)| > 2$, an edge swap has the effect of reducing v_i's degree and elongates the main path subgraph. Rayleigh's Monotonicity Law cannot be applied in this scenario as no "resistance" is being removed from the main path. Similar to Remark 3.8, these edge swaps do not guarantee that $J_\sigma(\mathcal{T},\mathcal{R})$ (7) is monotonically decreasing. Therefore, the proposed protocols are the best "local" information edge swapping protocols and no guarantees can be made that the "local" information protocol will converge to the best "global" information edge swap solution.

Remark 4.8: We previously remarked that **Protocols 1** and **2** do not alter the main path. Consequently, by Lemma 4.2, the quantity $J_\sigma(\mathcal{T},\mathcal{R})$ is conserved throughout these protocols so that, although the mean tracking measure is altered, the variance damping measure remains the same. The converse is not true as **Protocols 3** and **4** involve manipulations of the main path and, as mentioned in Remark 3.8, this can arbitrarily vary $J_\mu(\mathcal{T},\mathcal{R})$ (3). Generally speaking as $J_\sigma(\mathcal{T},\mathcal{R})$ increases under **Protocol 4** the graph elongates and so $J_\mu(\mathcal{T},\mathcal{R})$ tends to increase. Similarly, as $J_\sigma(\mathcal{T},\mathcal{R})$ decreases under **Protocol 3** the graph compresses and so $J_\mu(\mathcal{T},\mathcal{R})$ tends to decrease. This trend is starkly apparent when the two metrics are requoted in terms of the effective resistance, i.e., by rearranging (4) and (9), we note that

$$J_\mu(\mathcal{G},\mathcal{R}) = J_\sigma(\mathcal{G},\mathcal{R}) + \frac{1}{n}\sum_{v_i \notin \pi(\mathcal{E}_R)} E_{\text{eff}}(v_i). \tag{10}$$

Remark 4.9: We can requote the two metrics in terms of the error signal $z(t) = x(t) - u(t)$ and using the stochastic interpretation of $J_\sigma(\mathcal{G},\mathcal{R})$ [28] as

$$J_\mu(\mathcal{G},\mathcal{R}) = \mathbb{E}_{\|\tilde{x}(0)\|=1}\left(2\int_0^\infty \mathbb{E}\left(z(t)\right)^T \mathbb{E}\left(z(t)\right)dt\right), \quad \text{and}$$

$$J_\sigma(\mathcal{G},\mathcal{R}) = \mathbb{E}\left(\lim_{T\to\infty}\frac{1}{nT}\int_{-T}^{T} z(t)^T z(t)dt\right).$$

The two metrics can also be interpreted to characterize different components of the output signal. The metric $J_\mu(\mathcal{G},\mathcal{R})$ with respect to the mean is mainly influenced by the initial deviations of $z(t)$, or in other words, the *transient* response. On the other hand, $J_\sigma(\mathcal{G},\mathcal{R})$ with respect to the variance is more sensitive to long term fluctuations or *steady state* response.

Remark 4.10: The adaptive **Protocols 1–4** and a subset of our results are specific to tree graphs. As a preliminary extension to more general connected graphs we consider any spanning tree $\mathcal{T}$ of a connected graph $\mathcal{G}$. In terms of our electrical resistance analogy, the resistor network $\mathcal{T}$ is formed by removing resistors from $\mathcal{G}$. Applying Rayleigh's Monotonicity Principle leads to $J_\mu(\mathcal{G},\mathcal{R}) \le J_\mu(\mathcal{T},\mathcal{R})$ and $J_\sigma(\mathcal{G},\mathcal{R}) \le J_\sigma(\mathcal{T},\mathcal{R})$, i.e., both metrics on the graph are bounded above by the corresponding measures on its spanning trees.

C. Example: UAV Flocking Gust Correction

UAV flocking involves the distribution of tasks, normally performed by one central aerial vehicle, to many smaller vehicles which act cooperatively. One of the costs of such an architecture is increased susceptibility to external disturbances and intrusions.

In this example we consider a UAV flock of 40 agents with subgroups of agents exposed to wind gust disturbances. The network of UAVs is assumed to be running consensus on velocity and is initialized with all agents at hover, i.e., $x_i = 0$ for all i. The network configuration is assumed to be chosen to minimize relative sensing costs which are required to maintain velocity consensus, leading to a tree network $\widetilde{\mathcal{T}}$. Each agent is assumed to be equipped with an accelerometer to sense for wind gusts. The procedure when a gust is detected is for the affected agent to break from the consensus protocol and reacquire the last update state value. In response, similar to the time synchronization example (Section III-C), the non-affected agents rewires the network topology[11] following **Protocol 3** with the objective of minimizing the amplification of the gust impulse on the UAV network. After 10 s the affected agents then return to their normal consensus dynamics and all agents reverse the rewiring process to again achieve the graph $\widetilde{\mathcal{T}}$. It is assumed that the flock is sufficiently spread such that a proximity appropriate subgroup of only 1–2 agents experience a wind gust with a period of 20 s that increases the affected agents' velocity to 20 m/s.

As mentioned in Remark 2.2, the *external* agent model is appropriate to the misbehaving *native* node scenario as well. A comparison of the system state mean when running **Protocol 3**

[11] Edges directly attached to the affected agents can not be wired as it is assumed that affected agents upon reacquiring their normal operation, will expect to utilize these edges.

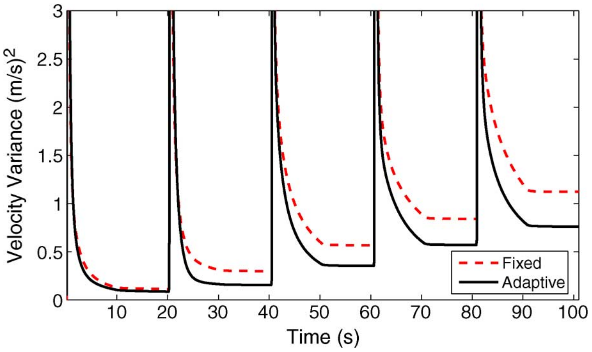


Fig. 11. Average velocity variance over time of a fixed and adaptive (running **Protocol 3**) 40-agent UAV flock with 1–2 agents exposed to wind gusts of 20 m/s every 20 s.

(adaptive topology) and one with a static topology is shown in Fig. 11. Over 100 s, the average velocity variance for the model running **Protocol 3** is 68% of the static tree model.

V. Game Theoretic Adaptive Protocol

Protocols 1–4 in Section III and Section IV possess guarantees on increasing (or decreasing) either the mean tracking or variance damping measures of the network. The weakness of these protocols is that they tend to converge to graphs associated with a local minimum (or maximum) of either $J_\mu(\mathcal{T},\mathcal{R})$ or $J_\sigma(\mathcal{T},\mathcal{R})$, with potentially sub-optimal performance. Furthermore the protocols cannot be applied concurrently, e.g., for security applications where poor tracking of the mean (high $J_\mu(\mathcal{T},\mathcal{R})$) and good noise damping (low $J_\sigma(\mathcal{T},\mathcal{R})$) is favorable. We now present a protocol that exhibits these attributes, i.e., the final graphs are within guaranteed bounds of the optimal network over all graphs for maximizing $J_\mu(\mathcal{T},\mathcal{R})$ and minimizing $J_\sigma(\mathcal{T},\mathcal{R})$, but the protocol no longer possess strictly increasing $J_\mu(\mathcal{T},\mathcal{R})$ and decreasing $J_\sigma(\mathcal{T},\mathcal{R})$. We will present the protocol and use a game theoretic formalism to bound the protocol's performance.

In the following, our game theoretic objective is to increase $J_\mu(\mathcal{T},\mathcal{R})$ and decrease $J_\sigma(\mathcal{T},\mathcal{R})$; in terms of effective resistance, the aim is to increase the final term in (10) while keeping $J_\sigma(\mathcal{G},\mathcal{R})$ small. This produces a graph that both damps the external agents' effect on the system's state mean and variance.

The proposed **Protocol 5** concurrently applies **Protocols 1** and **3** with a slight adaption to the latter, specifically, relaxing the condition $\{v_j, v_k\} = \mathcal{I}(v_i)$ to $v_j, v_k \in \mathcal{I}(v_i)$. This adaption guarantees that the main path subgraph will converge to a graph of only native agents where the external agents *directly* attach, i.e., $v_i \in \pi(\mathcal{E}_R)$. The remaining nodes in the graph, in the meantime, will form paths connected to an agent in $\pi(\mathcal{E}_R)$. There are many graphs and external agents pairs $(\mathcal{T},\mathcal{R})$ that satisfy these properties; we call the set of such pairs the *acquirable* set $\mathcal{A}$. In fact, the specific "equilibrium" that **Protocol 5** converges to will depend on the initial graph structure and the sequence of edge swaps prescribed by the protocol. It turns out that the convergence of the protocol falls under a special class of repeated games called **potential games** [29], and as such, exhibits certain sub-optimality guarantees that will be explored further.

Protocol 5 Increased mean tracking and decreased variance damping measure edge swap

```
for all Agent $v_i$ do
    if $\exists v_j, v_k \in \mathcal{N}(v_i)$, $v_j \neq v_k$, and [$(v_j, v_k \notin \mathcal{I}(v_i))$ or $(v_i \notin \pi(\mathcal{E}_R)$ and $v_j, v_k \in \mathcal{I}(v_i))$] then
        $E \rightarrow E - \{v_i, v_j\} + \{v_j, v_k\}$
    end if
end for
```

A. Game Theoretic Analysis

Game theory supplies tools to assess the optimality properties of equilibria reached following local decisions. Two metrics are generally used for this purpose; the price of stability which is the ratio between the "best" equilibria obtained from local decisions and the global optimum, and the price of anarchy which is the ratio of the "worst" equilibria obtained from local decisions and the global optimum. For our case, these metrics will capture the success of our local protocol with respect to the mean tracking and variance damping measures.

First, we need to establish that the protocol indeed converges to some equilibrium; for this task we use the concept of a potential game. A potential function Φ is a function that maps a strategy vector (a vector of each agent's edge swap) $S = (S_1, S_2, \ldots, S_n)$ to a real number. The implementation of a strategy on graph $\mathcal{T}$ will alter it to produce a graph $\mathcal{T}(S)$. A protocol leads to a potential game if for $S_i' \neq S_i$ as an alternate strategy (edge swap) for agent i, the local cost benefit to the agent $u_i(S') - u_i(S)$ is mirrored by the change in the potential, i.e.,[12]

$$\operatorname{sgn}\left(\Phi(S) - \Phi(S')\right) = \operatorname{sgn}\left(u_i(S') - u_i(S)\right) \tag{11}$$

where $S' = (S_1, \ldots, S_i', \ldots, S_n)$. Consider now the potential function

$$\Phi\left(\mathcal{T}(S), \mathcal{R}\right) = -\sum_{i=1}^{n} d\left(v_i, \Gamma(v_i)\right)$$

where $\Gamma(v_i)$ is defined in Section III-A. Therefore if the local cost of agent v_i is

$$u_i\left(\mathcal{T}(S), \mathcal{R}\right) = d\left(v_i, \Gamma(v_i)\right) \tag{12}$$

then the condition (11) is met. Since **Protocol 5** satisfies (12), it can be considered as a potential game.[13] An important consequence of this observation is that **Protocol 5** will always converge to an equilibrium [29].

We can now find the price of stability and anarchy with respect to the maximization of measures $J_\mu(\mathcal{T},\mathcal{R})$ and $1/J_\sigma(\mathcal{T},\mathcal{R})$ under **Protocol 5**.

Proposition 5.1: Under **Protocol 5**, for $J_\mu(\mathcal{T},\mathcal{R})$ the price of stability is equal to 1 and the price of anarchy is less than or equal to r.

[12] The signum function is represented by $\operatorname{sgn}(\cdot)$.

[13] This approach is similar to other network game problems [6], [29].

Proof: As the graph corresponding to the smallest $J_\mu(\mathcal{T},\mathcal{R})$ (3) is in the acquirable set $\mathcal{A}$ (by Proposition 3.6), the price of stability is equal to 1.

From Proposition 3.6 the maximum $J_\mu(\mathcal{T},\mathcal{R})$ is bounded as

$$\max_{(\mathcal{T},\mathcal{R})} J_\mu(\mathcal{T},\mathcal{R}) \le \frac{1}{2n}\left((n-r)^2 + 3(n-r) + r + \frac{2}{r+1}\right).$$

An acquirable graph with the smallest $J_\mu(\mathcal{T},\mathcal{R})$ corresponds to a network with the main path subgraph as a path $\mathcal{P}$ (by Proposition 4.5) with

$$\min_{(\mathcal{T},\mathcal{R}),v_j\in\mathcal{M}} E_{\text{eff}}(v_j) \ge \frac{1}{\sqrt{5}}.$$

The equilibrium graph in $\mathcal{A}$ corresponding to the smallest $J_\mu(\mathcal{T},\mathcal{R})$ compared to the tree from $\arg\max_{(\mathcal{T},\mathcal{R})} J_\mu(\mathcal{T},\mathcal{R})$ will have $\lfloor (n-r)/r \rfloor = (n-r)/r$ agents attached as a path to each of the main path agents.[14] Applying Lemma 3.3 leads to the inequality

$$\begin{aligned}
&\min_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{A}} J_\mu(\mathcal{T},\mathcal{R})\\
&\quad\ge \frac{1}{n}\left[\frac{r}{\sqrt{5}} + r\sum_{i=1}^{\frac{(n-r)}{r}}\left(\frac{1}{\sqrt{5}} + i\right)\right]\\
&\quad= \frac{1}{2nr}\left((n-r)^2 + \left(\frac{2}{\sqrt{5}}+1\right)r(n-r) + \frac{2}{\sqrt{5}}r^2\right).
\end{aligned}$$

For $r = 1$, the protocol always acquires the optimal equilibrium of a path graph with an external agent attached to one end node, so for this case the price of anarchy is equal to 1. For $1 < r \le n$, on the other hand

$$\begin{aligned}
\text{Price of anarchy} &= \frac{\max_{(\mathcal{T},\mathcal{R})} J_\mu(\mathcal{T},\mathcal{R})}{\min_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{A}} J_\mu(\mathcal{T},\mathcal{R})}\\
&\le r\frac{(n-r)^2+3(n-r)+r+\frac{2}{r+1}}{(n-r)^2+\left(\frac{2}{\sqrt{5}}+1\right)r(n-r)+\frac{2}{\sqrt{5}}r^2}\\
&< r
\end{aligned}$$

thus proving the proposition. ■

Proposition 5.2: Under **Protocol 5**, for $J_\sigma(\mathcal{T},\mathcal{R})$ (7) the price of stability is equal to 1 and the price of anarchy is less than $11\sqrt{5}/20 \approx 1.23$.

Proof: As the graph corresponding to the maximum $J_\sigma(\mathcal{T},\mathcal{R})$ is in $\mathcal{A}$ (by Proposition 4.5), the price of stability is equal to 1.

From Proposition 4.5, the optimal graph for $J_\sigma(\mathcal{T},\mathcal{R})$ corresponds to a network with the main path subgraph as a path $\mathcal{P}$ with

$$\min_{(\mathcal{T},\mathcal{R})} J_\sigma(\mathcal{T},\mathcal{R}) \ge \frac{r}{\sqrt{5}n}.$$

From (5), the equilibrium graph in $\mathcal{A}$ with the largest value of $J_\sigma(\mathcal{T},\mathcal{R})$ is associated with the main path subgraph as a star $\mathcal{S}$ with

$$\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{A}} J_\sigma(\mathcal{T},\mathcal{R}) = \frac{r^2+r+2}{2n(r+1)}.$$

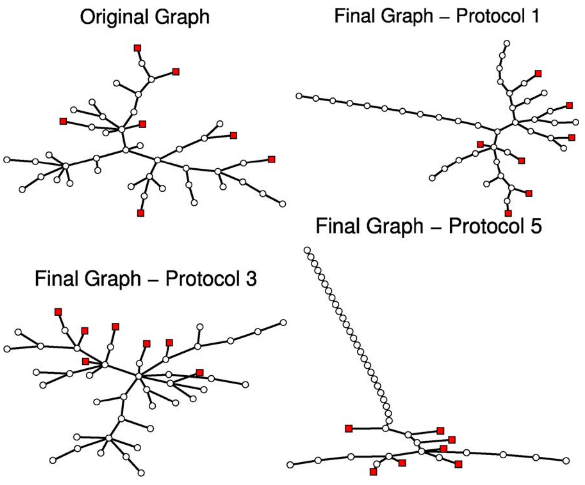
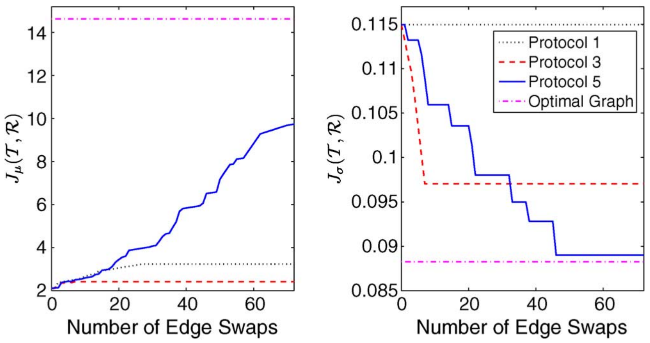


Fig. 12. Original and final tree graphs with seven external agents attached (squares) after applying **Protocols 1**, **3** and **5**.



(a) (b)

Fig. 13. (a) $J_\mu(\mathcal{T},\mathcal{R})$ and (b) $J_\sigma(\mathcal{T},\mathcal{R})$ after each edge swap from **Protocols 1**, **3** and **5** applied to the original graph in Fig. 12 as well as the optimal tree graphs with 40 nodes and 7 external agents.

For $r = 1, 2, 3$, the protocol always acquires the optimal equilibrium corresponding to the main path subgraph as a path $\mathcal{P}$ so for this case the price of anarchy is equal to 1. For $3 < r \le n$, on the other hand

$$\begin{aligned}
\text{Price of anarchy} &= \frac{\max_{(\mathcal{T},\mathcal{R})\subseteq\mathcal{A}} J_\sigma(\mathcal{T},\mathcal{R})}{\min_{(\mathcal{T},\mathcal{R})} J_\sigma(\mathcal{T},\mathcal{R})}\\
&< \frac{\sqrt{5}}{2}\frac{r^2+r+2}{r^2+r} < \frac{11\sqrt{5}}{20}
\end{aligned}$$

thus proving the proposition. ■

Protocol 5 was applied to a 40 node tree graph with 7 external agents attached. For comparison, **Protocol 1** (increasing mean tracking measure) and **Protocol 3** (decreasing variance damping measure) were applied to the same graph. The original and final graphs for each protocol appear in Fig. 12 while the metrics $J_\mu(\mathcal{T},\mathcal{R})$ and $J_\sigma(\mathcal{T},\mathcal{R})$ for each protocol, as compared with the optimal trees for $J_\mu(\mathcal{T},\mathcal{R})$ and $J_\sigma(\mathcal{T},\mathcal{R})$, are displayed in Fig. 13. We note that **Protocol 5** outperforms **Protocols 1** and **3**. The ratio of the optimal to the final equilibrium under **Protocol 5** was less than 1.51 for $J_\mu(\mathcal{T},\mathcal{R})$ and 1.08 for $1/J_\sigma(\mathcal{T},\mathcal{R})$, agreeing with the game-theoretic bounds stated in Propositions 5.1 and 5.2.

[14] $\lfloor x \rfloor$ is defined as the 'floor' of x.

VI. Conclusion

The aim of the present work is to propose a system-theoretic approach to examine the notion of semi-autonomy. In particular, the paper presents a class of consensus-type networks under the influence of external agents and proposes metrics for quantifying the network's ability, via its topology, to promote or resist the influence of external agents. Four decentralized protocols were proposed for tree graphs to vary the mean tracking and variance damping measures within the network. The protocols were applied to time synchronization and UAV flocking applications.

The proposed metrics were then analyzed and an effective resistance analogy was established by modeling the interconnection as a resistive network. The effective resistance interpretation provided a method to compare the two metrics and illustrated their relationship. The challenge of presenting a protocol that increased one metric while decreasing the other was addressed for tree graphs with a hybrid protocol and analyzed using a game theoretic approach. The extension of these protocols to more general networks will be discussed in a subsequent work. Finally, we encourage future research into performance and security of networks exploiting *both* topological and agent dynamic features of the network.

Acknowledgment

The authors would like to thank the Associate Editor and anonymous reviewers for their constructive comments.

References

[1] A. Chapman and M. Mesbahi, "Semi-autonomous networks: Network resilience and adaptive trees," in *Proc. 49th IEEE Conf. Decision Control*, 2010, no. 2, pp. 7473–7478.

[2] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[3] H. G. Tanner, G. J. Pappas, and V. Kumar, "Leader-to-formation stability," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 443–455, 2004.

[4] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.

[5] Y. Hatano and M. Mesbahi, "Agreement over random networks," *IEEE Trans. Autom. Control*, vol. 50, no. 11, pp. 1867–1872, Nov. 2005.

[6] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ: Princeton Univ. Press, 2010.

[7] S. Graham and P. R. Kumar, "Time in general-purpose control systems: The control time protocol and an experimental evaluation," in *Proc. 43rd IEEE Conf. Decision Control*, 2004, pp. 4004–4009.

[8] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt, "Controllability of multi-agent systems from a graph-theoretic perspective," *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 162–186, 2009.

[9] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious agents—Part I: Attacking the network," in *Proc. Amer. Control Conf.*, 2008, pp. 1350–1355.

[10] A. Chapman and M. Mesbahi, "System theoretic aspects of influenced consensus: Single input case," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1505–1511, Jun. 2012.

[11] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Autom. Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.

[12] D. Zelazo and M. Mesbahi, "Edge agreement: Graph-theoretic performance bounds and passivity analysis," *IEEE Trans. Autom. Control*, vol. 56, no. 3, pp. 544–555, Mar. 2011.

[13] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Proc. 45th IEEE Conf. Decision Control*, 2006, pp. 6605–6611.

[14] Y. Wan, S. Roy, and A. Saberi, "Network design problems for controlling virus spread," in *Proc. 46th IEEE Conf. Decision Control*, 2007, pp. 3925–3932.

[15] A. Fagiolini, G. Valenti, L. Pallottino, G. Dini, and A. Bicchi, "Decentralized intrusion detection for secure cooperative multi-agent systems," in *Proc. 46th IEEE Conf. Decision Control*, 2007, pp. 1553–1558.

[16] F. Pasqualetti, A. Bicchi, and F. Bullo, "Distributed intrusion detection for secure consensus computations," in *Proc. 46th IEEE Conf. Decision Control*, 2007, pp. 5594–5599.

[17] F. Pasqualetti, A. Bicchi, and F. Bullo, "On the security of linear consensus networks," in *Proc. 48th IEEE Conf. Decision Control*, 2009, pp. 4894–4901.

[18] A. Giridhar and P. R. Kumar, "Distributed clock synchronization over wireless networks: Algorithms and analysis," in *Proc. 45th IEEE Conf. Decision Control*, 2006, pp. 4915–4920.

[19] P. Barooah and J. P. Hespanha, "Graph effective resistance and distributed control: Spectral properties and applications," in *Proc. 45th IEEE Conf. Decision Control*, 2006, pp. 3479–3485.

[20] M. Mavronicolas, V. G. Papadopoulou, A. Philippou, and P. G. Spirakis, "A graph-theoretic network security game," *Int. J. Autonomous Adaptive Commun. Syst.*, vol. 1, no. 4, p. 390, 2008.

[21] S. D. Antonio, S. P. Romano, S. Simpson, and P. Smith, "A semi-autonomic framework for intrusion tolerance in heterogeneous networks," in *Proc. 3rd Int. Workshop Self-Organizing Syst.*, 2008, pp. 230–241.

[22] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, "Network robustness and fragility: Percolation on random graphs," *Phys. Rev. Lett.*, vol. 85, no. 25, p. 4, 2000.

[23] G. Tyson, A. T. Lindsay, S. Simpson, and D. Hutchison, "Improving wireless sensor network resilience with the INTERSECTION framework," in *Proc. 2nd Int. Conf. Mobile Lightweight Wireless Syst.*, 2010, pp. 415–426.

[24] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in thepresence of malicious agents—Part II: Overcoming malicious behavior," in *Proc. Amer. Control Conf.*, 2008, pp. 1356–1361.

[25] S. Salsa, *Partial Differential Equations in Action: From Modelling to Theory*. New York: Springer, 2008.

[26] B. Bollobás, *Modern Graph Theory*. New York: Springer, 1998.

[27] M. Petrovic and I. Gutman, "The path is the tree with smallest greatest Laplacian eigenvalue," *Kragujevac J. Math.*, vol. 24, pp. 67–70, 2002.

[28] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*. West Sussex, U.K.: Wiley, 2005.

[29] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, *Algorithmic Game Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

Airlie Chapman (S'09) received the B.S. degree in aeronautical (space) engineering and the M.S. degree in engineering research from the University of Sydney, Sydney, Australia, in 2006 and 2008, respectively, and is currently pursuing the Ph.D. degree in the Aeronautics and Astronautics Department, University of Washington, Seattle.

Her research interests are networked dynamic systems and graph theory with applications to aerospace systems and network security.

Mehran Mesbahi (SM'11) received the Ph.D. degree from the University of Southern California, Los Angeles, in 1996.

He was a member of the Guidance, Navigation, and Analysis Group, JPL, from 1996 to 2000, and an Assistant Professor with the Aerospace Engineering and Mechanics Department, University of Minnesota, Minneapolis, from 2000 to 2002. He is currently a Professor with the Aeronautics and Astronautics Department, University of Washington, Seattle. His research interests are distributed and networked aerospace systems, systems and control theory, and engineering applications of optimization and combinatorics.

Dr. Mesbahi received the NSF CAREER Award in 2001, NASA Space Act Award in 2004, the UW Distinguished Teaching Award in 2005, and the UW College of Engineering Innovator Award for Teaching in 2008.