Online Distributed Optimization on Dynamic Networks

Saghar Hosseini, Airlie Chapman, and Mehran Mesbahi

***Abstract*—This paper presents a distributed optimization scheme over a network of agents in the presence of cost uncertainties and over switching communication topologies. Inspired by recent advances in distributed convex optimization, we propose a distributed algorithm based on a dual sub-gradient averaging. The objective of this algorithm is to minimize a cost function cooperatively. Furthermore, the algorithm changes the weights on the communication links in the network to adapt to varying reliability of neighboring agents. A convergence rate analysis as a function of the underlying network topology is then presented, followed by simulation results for representative classes of sensor networks.**

***Index Terms*—Distributed optimization; adaptive weight selection; online optimization; switching graphs; weighted dual-averaging**

I. Introduction

The past decade has witnessed successful applications of networked systems in areas ranging from environmental monitoring, robotics, target recognition, air traffic control, to industrial and manufacturing automation. By increasing the size and complexity of networked systems, decentralized optimization schemes are desired for reducing data transmission rates and ensuring robustness in the presence of local failures. These methods are particularly relevant when there is a lack of access to centralized information by individual agents.

In recent years, there has grown an extensive literature on distributed convex optimization [1], [2], [3] and the adaptation and monitoring of the underlying network has become of increasing interest [4], [5], [6]. Moreover, agent's communication range or disturbances may cause the underlying network topology to change dynamically. In this direction, a class of distributed sub-gradient algorithms for convex optimization has been developed [7], [8], [9], [10], [11]. In these works, local convex cost functions are assumed to be known while the topology of network is allowed to vary.

In addition to uncertainties in the network's structure, the environment can also affect the corresponding cost functions. For such scenarios, traditional optimization approaches become unsuitable. One approach to improve the robustness of algorithms for convex optimization is via stochastic methods [12], [13], [14], where the probability distribution of uncertain variable is known *a priori*. One such approach has been pursued by Duchi *et al.* [10] who approached this problem using a stochastic sub-gradient method where the distribution of sub-gradients is known *a priori*.

Despite its many successes, stochastic optimization-based methods do not explicitly address the dynamic aspect of the problem in an unknown environment. Online learning is an extension of stochastic optimization where the uncertainty in the system is demonstrated by an *arbitrarily* varying cost function. In particular, at the time the relevant decision is made the cost function is assumed to be unknown, without probabilistic assumptions, to the decision-maker. Such learning algorithms have had a significant impact on modern machine learning [15], [16], [17]. One standard metric to measure the performance of these online algorithms is called *regret*. Regret measures the difference between the incurred cost and the cost of the *best fixed decision in hindsight*. An online algorithm is then declared "good" when its regret is sub-linear.

Distributed online optimization and its applications in multi-agent systems has not been studied at large by the systems and control community. Yan *et al.* in [18] introduced a decentralized online optimization based on a sub-gradient method in which the agents interact over a weighted strongly connected directed graph. Considering an undirected path graph with a fixed-radius neighborhood information structure, Raginsky *et al.* [19] proposed an online algorithm for distributed optimization based on sequential updates, proving a regret bound of $O(\sqrt{T})$. In [20], we proposed an extension to the work of Duchi *et al.* [10] on distributed optimization with convergence rate of $O(\sqrt{T}\log T)$ to an online setting. In addition, an improved regret bound of $O(\sqrt{T})$ has been derived for strongly connected networks, also highlighting the dependence of the regret on the connectivity of the underlying network.

We note that the aforementioned works do not exploit a dynamic weight selection procedure to improve the performance of the corresponding distributed algorithms. In systems and control literature, certain metrics have been used for designing *adaptive* mechanisms for networks based on centralized [21], [22], [23] and distributed [4], [24] strategies. Chapman *et al.* [25] proposed an online distributed algorithm for re-weighting the network edges in order to dampen the effect of external disturbances on the system. Dynamic weight selection is also favorable in the area of sensor networks and distributed estimation due to power and data rate constraints as well as failure modes of the inter-sensor communication links [26], [27], [28].

In this paper, we consider two types of uncertainties in the networked systems corresponding to disturbances in cost

A preliminary version of this work will appear in the *IEEE Conference on Decision and Control, 2013.*

The research of the authors was supported by the ONR grant N00014-12-1-1002 and AFOSR grant FA9550-12-1-0203-DEF. The authors are with the Department of Aeronautics and Astronautics, University of Washington, WA 98105. Emails: {saghar, airliec, mesbahi}@uw.edu.

functions and the network structure. An adaptive algorithm for distributed optimization over fixed networks is proposed and further extended to switching graphs. The main assumption used for implementing this algorithm is that the local cost function and its sub-gradient are observable at each node and can be shared with the neighboring nodes in the network.

The contribution of this paper is threefold. First, we present the Distributed Weighted Dual Averaging (DWDA) algorithm [20] for distributed optimization over networks. A distributed dynamic weight selection method based on an online weighted majority approach [29], [30] is then embedded in the DWDA algorithm allowing the weights on the network's edges to adaptively change in order to optimize the information diffusion in the network. The proposed algorithm is inspired by the Distributed Dual Averaging (DDA) algorithm [10]. Second, DWDA is applied on switching networks capturing the uncertainties in communication links. Third, the DWDA is further extended to Online Distributed Weighted Dual Averaging (Online-DWDA) algorithm which takes into account the uncertainties in cost functions and unavailability of reliable statistics on the noise characteristics. We then proceed to derive regret bounds that highlight the link between the adaptive weight selection and the Online-DWDA algorithm and can thus be used to design networks with good regret performance.

The organization of the paper is as follows. The notation and background on graphs and regret are reviewed in §II. In §III, the formulation for the distributed convex optimization problem over networks is presented. This is then followed by the description for the DWDA algorithm and dynamic weight selection procedure in §IV and §V; the convergence analysis of the proposed algorithm is discussed over switching topologies in §VI. In §VII, the distributed convex optimization problem is extended to the online setting, with applications to networked systems operating in an uncertain environment. The performance of Online-DWDA is subsequently studied using the regret analysis. In §VIII, we examine online distributed estimation over sensor networks, demonstrating the viability of the online approach in distributed estimation. Finally, §IX provides our concluding remarks and future directions for utilizing the online framework for system and control problems.

II. Background and Preliminaries

We provide a brief background on constructs that will be used in this paper. For the column vector $v \in \mathbb{R}^p$, v_i or $[v]_i$ denotes the ith element and e_i denotes the column vector which contains all zero entries except $[e_i]_i = 1$. The vector of all ones is denoted as $\mathbf{1}$. For matrix $M \in \mathbb{R}^{p\times q}$, $[M]_{ij}$, or simply M_{ij}, denotes the element in its ith row and jth column. The family of probability vectors is denoted by Ω and contains all non-negative vectors $\sigma \in [0,1]^n$ such that $\sum \sigma_i = 1$. A row stochastic matrix P is a non-negative matrix with rows in Ω. Moreover, the ergodic coefficient for a stochastic matrix $Q \in \mathbb{R}^{n\times n}$ is given by

$$\tau(Q) = 1 - \min_{i,j\in[n]} \sum_{k=1}^{n} \min\{Q_{ik}, Q_{jk}\}. \quad (1)$$

A time varying matrix is denoted by P^t and a (backward) sequence of time varying matrices is presented by $P^{(t,0)} = P^t P^{t-1}\cdots P^0$. For any positive integer n, the set $\{1,2,...,n\}$ is denoted by $[n]$. The inner product of two vectors θ and w is represented by $\langle\theta,\phi\rangle$. The 2-norm is signified by $||.||_2$, a general norm of vector is denoted as $\|\theta\|$, and its associated dual norm is defined as $||\theta||_* = \sup_{||\phi||=1} \langle\theta,\phi\rangle$. A function $f:\Theta\to\mathbb{R}$, where $\Theta\subseteq\mathbb{R}^m$ for some positive integer m, is called L-Lipschitz continuous with respect to the norm $\|\cdot\|$ if there exists a positive constant L for which

$$|f(\theta)-f(\phi)| \le L\|\theta-\phi\| \;\text{ for all } \theta,\phi\in\Theta. \quad (2)$$

A. Graphs

A succinct way to represent the interactions of dynamic agents, e.g., sensors, over a network is through a graph. A weighted directed graph $\mathcal{G}=(V,E,W)$ is defined by a node set V with cardinality n, the number of nodes in the graph representing the agents in the network, and an edge set E comprising of pairs of nodes which represent the agents' interactions, i.e., agent i affects agent j's dynamics if there is an edge from i to j, denoted as $(i,j)\in E$. In addition, a function $W: E\to\mathbb{R}$ is given that associates a weight $w_{ji}\in W$ to every edge $(i,j)\in E$. Moreover, dist(i,j) denotes the minimum number of edges of any directed path from node i to node j. The time-varying graph topology is presented by $\mathcal{G}^t=(V,E^t,W^t)$ at time step t. The neighborhood of node i is defined as the set $N_i=\{j|(j,i)\in E\}$ and the time-varying neighborhood set is presented by $N_i^t=\{j|(j,i)\in E^t\}$. The adjacency matrix $A(\mathcal{G})$ is a matrix representation of $\mathcal{G}$ with $[A(\mathcal{G})]_{ji}=w_{ji}$ for $(i,j)\in E$ and $[A(\mathcal{G})]_{ji}=0$, otherwise. A graph $\mathcal{G}$ is strongly connected if there exists a directed path between every pair of distinct vertices. For a graph $\mathcal{G}$, d_i is the weighted in-degree of i, defined as $d_i=\sum_{\{j|(j,i)\in E\}} w_{ij}$. In addition, $L(\mathcal{G})=\Delta(\mathcal{G})-A(\mathcal{G})$ is called the graph Laplacian where $\Delta(\mathcal{G})$ is the diagonal matrix of d_i's. Based on the construction of weighted directed graph Laplacian, every graph $\mathcal{G}$ has a right eigenvector of $\mathbf{1}$ associated with eigenvalue $\lambda=0$ [31].

There are many families of graphs that are often used to model networks of practical interest. In this paper, we use path and random graphs for some of our simulations. Specifically, Erdős-Rényi random graphs with edge probability p are constructed by having an edge $(i,j)\in E$ in the graph with probability p for all possible edges. A random tree is a particular realization of a random graph that is minimally connected and a random k-regular graph is a random graph in which $d_i=k$ for all vertices $i\in V$. In the path graph, $(i,j)\in E$ if and only if $|i-j|=1$.

B. Regret

Regret is one measure of performance for learning algorithms. In the online optimization setting, an algorithm is used to generate a sequence of decisions $\{x(t)\}_{t=1}^{T}$. The number of iterations is denoted by T which is unknown to the online player. At each iteration t, after committing to $x(t)$, a previously unknown convex cost function f_t is revealed, and a loss $f_t(x(t))$ is incurred. The goal of the online algorithm is to ensure that the time average of the difference between the total cost and the cost of the best fixed decision $x^* = \text{argmin} \sum_{t=1}^{T} f_t(x)$ is small. The difference between these two costs over $t = 1, 2, ..., T$, iterations is called the regret of the online algorithm, i.e.,

$$R_T(x^*, x) = \sum_{t=1}^{T} \left(f_t(x(t)) - f_t(x^*)\right). \tag{3}$$

An algorithm performs well if its regret is sub-linear as a function of T, i.e. $\lim_{T\to\infty} R_T/T = 0$. This implies that on average, the algorithm performs as well as the best fixed strategy in hindsight independent of the adversary's moves and environmental uncertainties. Further discussion on online algorithms and their regret analysis can be found in [32], [33], [34].

The general definition of regret is presented in (3) for a single decision-making unit. In order to analyze the performance of *distributed online algorithms* two variations of the notion of regret are introduced. First is the regret due to agent i's decision,

$$R_T(x^*, x_i) = \sum_{t=1}^{T} \left(f_t(x_i(t)) - f_t(x^*)\right), \tag{4}$$

which is the cumulative penalty agent i incurs due to its local decisions $\{x_i(t)\}_{t=1}^{T}$ on the global cost sequence $\{f_t\}$. Second is the regret based on the *running average* of the decisions $\{x_i(t)\}_{t=1}^{T}$,

$$R_T(x^*, \widetilde{x}_i) = \sum_{t=1}^{T} \left(f_t(\widetilde{x}_i(t)) - f_t(x^*)\right), \tag{5}$$

where $\widetilde{x}_i(T) = \frac{1}{T}\sum_{t=1}^{T} x_i(t)$.

III. Problem Statement

In this section a distributed decision process is considered in which a large number of agents cooperatively optimize a global objective function over the network denoted by $\mathcal{G} = (V, E, W)$.

The global objective to be minimized is

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) \quad \text{subject to} \quad x \in \chi, \tag{6}$$

where $f_i(x) : \mathbb{R}^d \to \mathbb{R}$ is a convex cost function associated with agent $i \in V$ and $\chi \subseteq \mathbb{R}^d$ is a closed convex set. The global optimization problem will be solved *locally* by each agent i via the local decision variable $x_i \in \chi$.

IV. Weighted Dual Averaging

In order to solve the optimization problem (6), we adapt Nesterov's dual averaging algorithms [35] and our preliminary results on the Distributed Weighted Dual Averaging (DWDA) algorithm [20], that in turn is inspired by [10]. The DWDA algorithm sequentially updates the *local* $x_i(t)$ and a *working* variable $y_i(t)$ for each agent i. The update itself is based on a provided local sub-gradient of the loss $f_i(x_i(t))$ denoted as $g_i(t)$. The centralized form of the dual averaging algorithm appears as a sub-gradient decent method followed by a projection step onto the constraint set χ, specifically,

$$y(t+1) = y(t) + g(t), \tag{7}$$

where $g(t) \in \partial f(x(t))$. Then

$$x(t+1) = \Pi_{\chi}^{\psi}\left(y(t+1), \alpha(t)\right), \tag{8}$$

where $\Pi_{\chi}^{\psi}(\cdot)$ is a regularized projection onto χ, to be formally defined shortly. The Distributed Weighted Dual Averaging (DWDA) algorithm is presented in Algorithm 1. The projection function used in this algorithm is defined as

$$\Pi_{\chi}^{\psi}(y(t), \alpha(t)) = \arg\min_{x\in\chi} \left\{ \langle y(t), x\rangle + \frac{1}{\alpha(t)}\psi(x) \right\}, \tag{9}$$

where $\alpha(t)$ is a non-increasing sequence of positive functions and $\psi(x) : \chi \to \mathbb{R}$ is a proximal function. The standard dual averaging algorithm uses proximal function $\psi(x)$ to avoid undesirable oscillations in the projection step. Without loss of generality, ψ is assumed to be strongly convex with respect to $\|.\|$, $\psi(x) \geq 0$, and $\psi(0) = 0$.

Algorithm 1: Distributed Weighted Dual Averaging (DWDA)

```
for t = 1 to T do
    Evaluate f(t) = {f_i(t); for all i = 1, ..., n}
    foreach Agent i do
        Compute subgradient g_i(t) ∈ ∂f_i(x_i(t))
        y_i(t+1) = Σ_{j∈N(i)} P_ji(t) y_j(t) + g_i(t)
        x_i(t+1) = Π_χ^ψ(y_i(t+1), α(t))
        x̃_i(t+1) = 1/(t+1) Σ_{s=1}^{t+1} x_i(s)
    end
end
```

The distributed algorithm can be considered as an approximated sub-gradient descent. The approximation is attained by an agent via a convex combination of local sub-gradients provided by its neighbors. This operation can be represented compactly as a stochastic matrix $P \in \mathbb{R}^{n\times n}$ which preserves the zero structure of the Laplacian matrix $L(\mathcal{G})$. It is clear that for all agents to have access to each cost function f_i there must be a path from every agent i to every other agent. Consequently, a minimum requirement on the underlying network is that it must be strongly connected. The distributed dynamic weight selection procedure presented in the following section constructs a row stochastic matrix P of the required form that is associated with a weighted directed graph.

V. Distributed Dynamic Weight Selection

In this section, we propose an adaptation scheme for the network weight selection in order to improve the information diffusion in line 5 of Algorithm 1 such that the communication matrix P is a row stochastic matrix with positive diagonal elements. In this proposed distributed algorithm, each agent $i \in [n]$ estimates its loss function via a convex combination of loss functions available to it by its neighboring agents. This convex combination is specified by weights w_{ij}'s on each edge $(j,i) \in E$ and w_{ii}, respectively. The edge re-weighting problem parallels the Weighted Majority (WM) algorithm [29]. The context of the WM algorithm is the presence of $|N_i| + 1$ experts, and the associated cost $h_j(t)$ assigning a loss value to expert $j \in \{N_i, i\}$, where $0 \leq h_j(t) \leq 1$. At each time-step, agent i selects a probability distribution $q(t)$ over the $|N_i| + 1$ experts, i.e., $q(t) \in \mathcal{Q}_i = \{q \in \Omega | q_j = 0 \text{ for } j \notin \{N_i, i\}\}$, in order to minimize $l_{t,i}(q) = \sum_{j \in \{N(i),i\}} q_j(t) h_j(t)$. The regret for agent i for the WM algorithm is then defined as

$$L_T(q^*, q) = \sum_{t=1}^{T} l_{t,i}(q) - \sum_{t=1}^{T} l_{t,i}(q^*), \tag{10}$$

where

$$q^* = \operatorname{argmin}_{q \in \mathcal{Q}_i'} \sum_{t=1}^{T} f_t(q_j)$$

with $\mathcal{Q}_i^{'} = \{q \in \mathcal{Q}_i | q_j \in \{0,1\}\}$, and the best fixed strategy q^* is the best expert $j \in \{N_i, i\}$ in hindsight. Consequently, (10) is of the same form as (3).

The general form of the WM algorithm is presented in [30] as the Online Allocation (OA) algorithm that is applicable to any bounded loss function over general decision and outcome spaces. Based on the OA algorithm, the regret for each agent $i \in [n]$ is bounded as

$$L_T \leq M \left(\sqrt{2T \ln\left(|N_i| + 1\right)} + \ln\left(|N_i| + 1\right) \right), \tag{11}$$

where M is the upper bound on the loss function $h_j(t)$ for all $j \in \{N_i, i\}$. Since the regret in (11) is sub-linear over time, the weight allocation performs as well as the best strategy in hindsight.

A Distributed Online Allocation (DOA) algorithm is proposed based on the OA algorithm where agent $i \in [n]$ specifies the weights w_{ij}'s associated to each edge $(j,i) \in E$ as well as the weight on the self-loop w_{ii}. The DOA algorithm presented in Algorithm 2 is embedded in Algorithm 1 at each iteration.

In the distributed optimization process considered, each agent decides on the weights associated with the information *received* from its neighboring agents. This information is based on the neighbor's local loss function. Intuitively, the algorithm places more weight on the link associated with the neighboring agent that has a higher confidence in its decision. The positive diagonal entries represent the self-confidence of each agent and is updated based on the local loss.

In networks with fixed topology, the communication matrix $P(t)$ preserves its zero structure for all time t. In addition, the

Algorithm 2: Distributed Online Allocation (DOA)

1 **foreach** *Agent* i **do**
2   Choose $\beta \in [0,1]$ and initial weight vector $\mathbf{w}_i(t) = \mathbf{1}$
3   Let $q(t) = \frac{1}{n}\mathbf{w}_i(t)$
4   **for** $t = 1$ **to** T **do**
5   Adversary reveals $f(t) = \{f_j(t);\ \text{for all } j \in \{N(i), i\}\}$
6   Suffer loss $l_{t,i} = \sum_{j \in \{N(i),i\}} q_j(t) f_j(t)$
7   **foreach** *Agent* $j \in [n]$ **do**
8   **if** $j \in \{N(i), i\}$ **then**
9         $w_{ij}(t+1) = w_{ij}(t)\beta^{f_j(t)}$
10   **else**
11         $w_{ij}(t+1) = w_{ij}(t)$
12   **end**
13   **end**
14     $q(t+1) = \frac{\mathbf{w}_i(t+1)}{\sum_{j \in \{N(i),i\}} w_{ij}(t+1)}$
15   **end**
16 **end**

non-zero elements in each row of the communication matrix P is specified by line 14 in Algorithm 2:

$$P_{ij}(t) = \begin{cases} q_j(t) & \text{for } j \in \{N_i, i\} \\ 0 & \text{otherwise} \end{cases}. \tag{12}$$

Since for each agent i, $q(t) \in \mathcal{Q}_i$ is a probability distribution, the communication matrix $P(t)$ will be row stochastic at every time step. The weighted graph Laplacian can then be formed as

$$L(\mathcal{G}(t)) = I - P(t). \tag{13}$$

In addition, note that since the graph is strongly connected, the communication matrix P is 1-irreducible ([36]; Corollary 4) and given positive diagonal elements, it is indecomposable and aperiodic (SIA). In the following section, the DOA algorithm is extended to construct a row stochastic communication matrix $P(t)$ for directed switching graphs with time-varying edge sets.

A. Switching Topologies

The network topology may change dynamically due to disturbances or communication range limitations. In this section we apply the dynamic weight selection procedure discussed in §V to switching topologies. In this paper we assume that the union of directed topologies $\mathcal{G}^{\cup_{i=0}^{\delta-1}} = \bigcup_{i=0}^{\delta-1} \mathcal{G}^i$ over some fixed uniform intervals δ, with $\delta \geq 1$ a positive integer, is strongly connected. We note that the communication matrix of $\mathcal{G}^{\cup_{i=0}^{\delta-1}}$ can be presented as

$$P^{\cup_{i=0}^{\delta-1}} = P^0 + P^1 + \cdots + P^{\delta-1}. \tag{14}$$

Thus, each row of the communication matrix P over switching topologies is specified by line 14 in Algorithm 2 as

$$P_{ij}(t) = \begin{cases} q_j(t) / \left(\sum_{k \in \{N_i^t, i\}} q_k(t) \right) & \text{for } j \in \{N_i^t, i\} \\ 0 & \text{otherwise} \end{cases}, \tag{15}$$

where $q(t) \in Q_i^t = \{q \in \Omega | q_j = 0 \text{ for } j \notin \{N_i^t, i\}\}$ and is a probability distribution. Note that the communication matrix $P(t)$ will be row stochastic at every time step and thus the weighted graph Laplacian is the same as in (13). Since the graph $\mathcal{G}^{\cup_{i=0}^{\delta-1}}$ is strongly connected and $P^{\cup_{i=0}^{\delta-1}}$ has positive diagonal elements, the communication matrix $P^{\cup_{i=0}^{\delta-1}}$ is SIA ([36]; Corollary 4). These properties of communication matrices will be subsequently employed in the convergence analysis of the DWDA algorithm.

VI. Convergence Analysis

Before presenting the convergence analysis of the distributed optimization algorithm, a few preliminary remarks and assumptions are in order. We assume that each convex function f_i is positive and L-Lipschitz with respect to $\|.\|$. Assuming that $\{P^t\}$ is SIA, there exists a vector $\pi \in \Omega$ [37], such that

$$\pi_j = \sum_{i=1}^{n} \pi_i P_{ij}^t \text{ for all } t \in [T], \tag{16}$$

where π_i is referred to as the weighting factor for agent i.

In order to take advantage of the properties of the standard weighted dual averaging in our regret analysis, the sequences $\bar{y}(t)$ and $\bar{g}(t)$ are defined as

$$\bar{y}(t) = \sum_{i=1}^{n} \pi_i y_i(t), \text{ and } \bar{g}(t) = \sum_{i=1}^{n} \pi_i g_i(t), \tag{17}$$

signifying the (network-level) weighted average of dual variables and subgradients in the DWDA algorithm, respectively. Therefore, based on (16) and (17),

$$\begin{aligned}\bar{y}(t+1) &= \sum_{i=1}^{n} \pi_i \left\{ \sum_{j=1}^{n} P_{ij}^t y_j(t) + g_i(t) \right\} \\ &= \sum_{j=1}^{n} y_j(t) \sum_{i=1}^{n} \pi_i P_{ij}^t + \bar{g}(t) \\ &= \sum_{j=1}^{n} y_j(t) \pi_j + \bar{g}(t) \\ &= \bar{y}(t) + \bar{g}(t),\end{aligned} \tag{18}$$

which is analogous to the dual averaging update (7). Thus, the following update rule is introduced which is analogous to the standard dual averaging algorithm projection step (8), where the primal variable is updated as

$$\phi(t+1) = \Pi_{\mathcal{X}}^{\psi}(\bar{y}(t+1), \alpha(t)). \tag{19}$$

The performance analysis of the distributed optimization and adaptive weight selection can now be presented.

The following result by Duchi *et al.* implies that after T iterations of Algorithm 1, each agent's error in the evaluation of total cost is bounded by the error due to Dual-Averaging method.

Theorem 1. *[10] Given the sequences $x_i(t)$ and $y_i(t)$ generated by lines 5 and 6 in* Algorithm 1, *for all $i \in [n]$ with proximal function ψ and $\alpha(t) > 0$, we have*

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} f(x_i(t)) - f(x^*) &\leq \frac{L^2}{2}\sum_{t=1}^{T}\alpha(t-1) + \frac{1}{\alpha(T)}\psi(x^*) \\ &+ \frac{L}{T}\sum_{t=1}^{T}\alpha(t)(\|\bar{y}(t) - y_i(t)\|_* + \frac{2}{n}\sum_{i=1}^{n}\|\bar{y}(t) - y_i(t)\|_*).\end{aligned} \tag{20}$$

The last two terms on the right hand side of (20) represent the error due to the network which is defined as the deviation of local dual variable y_i from the weighted average of dual variables $\bar{y}$ over the network. Lemma 10 in the Appendix imposes an upper bound on the effect of network topology associated with $\|\bar{y}(t) - y_i(t)\|_*$ as

$$\|\bar{y}(t) - y_i(t)\|_* \leq L \sum_{k=0}^{t-2}\sum_{j=1}^{n} \left| P_{ij}^{(t-1,k+1)} - \pi_j \right| + 2L. \tag{21}$$

Inequality (21) highlights the importance of the underlying network topology through the communication matrix P^t and its products. Note that the network effect is analogues to the consensus problems [38]. Therefore we proceed to extend the distributed optimization algorithm to switching graphs in the following subsection and provide a sub-linear convergence rate for the DWDA algorithm.

A. Switching Topologies

In this section we employ the weak ergodicity of inhomogeneous Markov chains to reason about the convergence of the DWDA algorithm. This property implies that the product of stochastic SIA matrices converges exponentially to a rank-one matrix of the form $\mathbf{1}\pi^T$ as $t \to \infty$, where $\pi \in \Omega$. Applying the following result from [39], it thus suffices to show that $P^{(\delta-1,0)}$ is SIA.

Lemma 2. *[39] Let $m \geq 1$ be a positive integer and P^τ be non-negative matrices with positive diagonal elements for $\tau = 0, 1, \ldots, m$. Then, $P^m P^{m-1} \ldots P^0 \geq \mu^m (P^0 + P^1 + \cdots + P^m)$, where $\mu > 0$ is specified by the diagonal elements of matrices P^τ for all $\tau = 0, 1, \ldots, m$.*

From Lemma 2 and (14), $P^{(\delta-1,0)}$ is bounded below by an SIA matrix. Moreover, we note that $P^{(\delta-1,0)}$ is also a stochastic matrix, thus it must be SIA. Therefore, based on the weak ergodicity of inhomogeneous Markov chains, the product

$$P^{(k\delta-1,(k-1)\delta)} \ldots P^{(2\delta-1,\delta)} P^{(\delta-1,0)}$$

converges exponentially to a rank-one matrix of the form $\mathbf{1}\pi^T$ as $t \to \infty$, and based on Theorem 1 of [37], we have

$$\left| P^{(k\delta-1,0)} - \pi_j \right| \leq \gamma^{\left\lfloor \frac{k}{\nu} \right\rfloor}, \tag{22}$$

where

$$\gamma = \max_{\nu \geq 1} \left\{ \tau(P^{(\delta\nu-1,0)}) < 1 \right\}. \tag{23}$$

Note that the maximization is over all realizations of the sequence $P^{(\delta\nu-1,0)}$ and ν is bounded as stated in the following proposition.

Proposition 3. *Consider a set $\mathcal{P}$ of stochastic matrices with positive diagonal elements, representing arbitrarily strongly connected topologies over n nodes, i.e., $P^k \in \mathcal{P}$ for all positive integers k. Then, there exists an integer ν, $1 \leq \nu \leq n-1$, for which if the sequence $Q = P^{(m+\nu-2,m)}$ of matrices in $\mathcal{P}$ is not scrambling, $P^{m+\nu-1}Q$ is scrambling.*

Proof: Let $Q_1 = P^m$ and $Q_2 = P^{m+1}P^m$. Then every entry of Q_2 is represented as

$$[Q_2]_{ij} = \sum_{k=1}^{n} \left[P^{m+1}\right]_{ik} [P^m]_{kj} .$$

Let $E^{\cup_{i=m}^{m+1}}$ represent the edge set of the union of directed graphs associated with P^m and P^{m+1}. Since $[P^m]_{ii} > 0$ for all $i \in [n]$ and integer $m \geq 1$, the entry $[Q_2]_{ij}$ is positive if $(j,i) \in E^{\cup_{i=m}^{m+1}}$ or if there exists a node $k \in [n]$ such that $(j,k) \in E^m$ and $(k,i) \in E^{m+1}$. Thus, the corresponding zero entry of Q_1 that has one of the aforementioned properties will be positive in Q_2. By induction, it follows that the entry of $[Q_\nu]_{ij}$ will be positive if $(j,i) \in E^{\cup_{i=m}^{m+\nu-1}}$, or if there exists a set of nodes $\{k_m, k_{m+1}, \ldots, k_{m+\nu-2}\}$ such that $\{(j,k_m), (k_m,k_{m+1}), ..., (k_{m+\nu-2}, i)\} \in E^m \times E^{m+1} \times \cdots \times E^{m+\nu-1}$. Therefore, for each row i of Q_ν, all entries will be positive when

$$\nu_i = \max_{j\in[n], m\geq 1} \{\mathrm{dist}(j,i) \text{ for } \mathcal{G}^m = (V, E^m, W)\} . \quad (24)$$

Note that the maximization in (24) is over all possible strongly connected graphs with the directed cycle graph representing the worst case with $\nu_i = n-1$. Since every element of any row of the sequence $Q = P^{(m+\nu-2,m)}$ of matrices in $\mathcal{P}$ is positive, the matrix Q is scrambling and thus $1 \leq \nu \leq n-1$. ■

We also note that the fixed topology is a special case of switching graphs with $\delta = 1$ in (14). Moreover, Proposition 11 of the Appendix presents a less conservative bound on ν for fixed topologies.

Now, we can state the following theorem for the rate of convergence of DWDA over switching graphs.

Theorem 4. *Given the sequences $x_i(t)$ and $y_i(t)$ generated by lines* 5 *and* 6 *in* Algorithm 1*, for all $i \in [n]$ with $\psi(x^*) \leq R^2$ and $\alpha(t) = k/\sqrt{t}$, we have*

$$\frac{1}{T}\sum_{t=1}^{T} f(x_i(t)) - f(x^*) \leq \left(\frac{R^2}{k} + kL^2\left(\frac{6n}{1-\gamma} + 6n\delta\nu + 1\right)\right)\frac{1}{\sqrt{T}}, \quad (25)$$

where $\gamma < 1$ is a function of the ergodicity of the communication matrix (see (23)*) while ν is a measure of network connectivity and is bounded by the diameter of the graph $\mathcal{G}^t$ (see also Proposition 3). In addition, $k > 0$ is an arbitrary constant and $\delta \geq 1$ is a positive integer as presented in* (14)*.*

Proof: Based on (21) and (22) we have

$$\|\bar{y}(t) - y_i(t)\|_* \leq nL\sum_{k=1}^{t-1}\gamma^k + nL(\delta\nu - 1) + 2L, \quad (26)$$

and since $\gamma < 1$, (26) is further bounded as

$$\|\bar{y}(t) - y_i(t)\|_* \leq nL\left(\frac{1}{1-\gamma} + \delta\nu - 1\right) + 2L. \quad (27)$$

Therefore, the integral test on $\alpha(t) = k/\sqrt{t}$ provides a bound[1] on the first and last terms in (25) as

$$\frac{1}{T}\sum_{t=1}^{T} f(x_i(t)) - f(x^*) \leq \frac{kL^2}{\sqrt{T}} + \frac{\psi(\theta^\star)}{k\sqrt{T}} + \frac{6kL^2}{\sqrt{T}}\left(\frac{n}{1-\gamma} + n\delta\nu + 1\right).$$

Given $\psi(x^*) \leq R^2$, the statement of the theorem now follows. ■

Theorem 4 states that Algorithm 1 performs "well" as it exhibits a sub-linear convergence rate. It also highlights the importance of the underlying network topology through the parameters γ and ν. In particular, ν corresponds to the diameter of the graph as expressed in Proposition 3 and γ is proportional to the ergodic coefficient $\tau(P^t)$ of the communication matrix P^t as formed by Algorithm 1. The ergodic coefficient bounds the second largest eigenvalue of P^t, $\lambda_2(P^t)$, as $|\lambda_2(P^t)| \leq \tau(P^t) < 1$. Thus, based on (13), $1 - \lambda_2(P^t) = \lambda_{n-1}(\mathcal{G}^t)$ where $\lambda_{n-1}(\mathcal{G}^t)$ is the second smallest eigenvalue of the weighted graph Laplacian $L(\mathcal{G}^t)$ and a well known measure of network connectivity. Consequently, high network connectivity promotes good performance of the proposed algorithm.

In the following section we study the effect of the proposed dynamic weight selection on the network connectivity and the convergence rate (25).

B. Adaptive Weight Selection

In this section, we show that embedding Algorithm 2 within Algorithm 1 improves the network information flow and the speed of convergence in (25). To this end the following result provides a bound on the ergodic coefficient.

Theorem 5. *Suppose the sequence $q_j(t)$ generated by line* 14 *of Algorithm* 2 and *the communication matrices P^t are constructed by* (15)*. Then*

$$\tau(P^{(\delta\nu-1,0)}) \leq 1 - \frac{n}{(\max_{i\in[n], t\in[\delta\nu-1]} |N_i^t| + 1)^{\delta\nu}},$$

where ν is a measure of network connectivity and is bounded by the diameter of the graph $\mathcal{G}^t$ (see also Proposition 3). In addition, $\delta \geq 1$ is a positive integer as presented in (14)*.*

Proof: Based on line 14 of Algorithm 2, we have that for all $k \in \{N_i, i\}^t$,

$$q_k(t) = \frac{\beta^{-\left(\sum_{s=1}^{r_{k,i}(t-1)} f_k(s)\right)}}{\sum_{j\in\{N_i^{t-1}, i\}} \beta^{-\left(\sum_{s=1}^{r_{j,i}(t-1)} f_j(s)\right)}}, \quad (28)$$

[1] Note that $\sum_{t=1}^{T} \frac{k}{\sqrt{t}} \leq 2k\sqrt{T} - k$.

where $r_{ji}(t)$ represents the number of communication rounds through the directed edge (j,i) up to time t. Subsequently (15) and (28) imply

$$P_{ik}^t = \begin{cases} \dfrac{\beta^{-\left(\sum_{s=1}^{r_{k,i}(t-1)} f_k(s)\right)}}{\sum_{j\in\{N_i^t,i\}} \beta^{-\left(\sum_{s=1}^{r_{j,i}(t-1)} f_j(s)\right)}} & (k,i)\in E^t, \\ 0 & \text{otherwise.} \end{cases}$$

Since $\min_{j\in[n]} \sum_{s=1}^{r_{j,i}(t-1)} f_j(s) \le \sum_{s=1}^{r_{k,i}(t-1)} f_k(s)$, we have

$$\beta^{-\left(\sum_{s=1}^{r_{k,i}(t-1)} f_k(s)\right)} \le \beta^{-\left(\min_{j\in[n]} \sum_{s=1}^{r_{j,i}(t-1)} f_j(s)\right)}$$

and one can bound P_{ik}^t from below for all $(k,i)\in E$ as

$$P_{ik}^t \ge \frac{1}{(|N_i^t|+1)}\beta^{-C_k(t-1)}, \tag{29}$$

where $C_k(t) = \sum_{s=1}^{r_{k,i}(t)} f_k(s) - \min_{j\in[n]} \sum_{s=1}^{r_{j,i}(t)} f_j(s)$. Since $C_k(t-1)\ge 0$ for all $k\in[n]$ and $\beta\in[0,1]$, we have

$$P_{ij}^t \ge \frac{1}{(|N_i^t|+1)},$$

for all $t\in[T]$ and subsequently

$$P_{ij}^{(\delta\nu-1,0)} \ge \frac{1}{(\max_{i\in[n],t\in[\delta\nu-1]} |N_i^t|+1)^{\delta\nu}}.$$

Based on (1), the statement of the theorem now follows. ■

Theorem 5 in conjunction with (23) imply

$$\gamma \le 1 - \frac{n}{(\max_{i\in[n],t\in[\delta\nu-1]} |N_i^t|+1)^{\delta\nu}},$$

which proves to be a conservative bound as the DOA algorithm leads to a tighter upper bound capturing the performance of agents. In other words, based on (29), we can show that

$$\gamma \le 1 - \frac{n\beta^{-J_{\delta\nu}}}{(\max_{i\in[n]} |N_i|+1)^{\delta\nu}},$$

where $J_{\delta(n-1)} = C_i(1) + C_j(2) + \ldots + C_k(\delta\nu)$ and $i,j,\ldots,k\in[n]$. In addition, we know that $\beta^{-J_{\delta\nu}} > 1$ and $J_{\delta\nu}$ is an increasing sequence. If the agents are not performing well, $J_{\delta\nu}$ will increase and subsequently $\tau(P^{(\delta\nu,0)})$ will decrease which suggests that the DOA algorithm mitigates the effect of the network topology in (25). Moreover, Theorem 5 implies that the DWDA algorithm performs well for certain types of graphs such as k-regular and expander graphs where the maximum number of neighbors can be bounded.

VII. Online Distributed Optimization

We now consider the effect of uncertainties in the environment on distributed decision processes where the global objective is to minimize

$$f(x) = \frac{1}{n}\sum_{i=1}^n f_{t,i}(x) \quad \text{subject to} \quad x\in\chi, \tag{30}$$

where $f_{t,i}:\mathbb{R}^d\to\mathbb{R}$ is a convex cost function associated with agent $i\in[n]$, assumed to be revealed to the agent only after the agent commits to the decision $x(t)$. In other words, the function $f_{t,i}$ is allowed to change over time in an unpredictable manner due to modeling errors and uncertainties in the environment. The optimization variable $x_i\in\mathbb{R}^d$ belongs to a closed convex set $\chi\subseteq\mathbb{R}^d$ and represents the local decision made by agent i. Furthermore, the online-DWDA scheme is analogous to the DWDA presented in Algorithm 1. The regret analysis is presented in the following result quantifying the performance of the proposed algorithm.

Theorem 6. *Given the sequences $x_i(t)$ and $y_i(t)$ generated by lines 5 and 6 in* Algorithm 1*, for all $i\in[n]$ with $\psi(x^*)\le R^2$ and $\alpha(t)=k/\sqrt{t}$, we have*

$$R_T(x^*,x_i) \le \left(\frac{R^2}{k} + kL^2\left(\frac{6n}{1-\gamma} + 6n\delta\nu + 1\right)\right)\sqrt{T}, \tag{31}$$

where γ is a function of the ergodicity of the communication matrix (see (23)*) while ν is a measure of network connectivity and is bounded by the diameter of the graph $\mathcal{G}^t$ (see also Proposition 11). In* (31)*, $k>0$ is an arbitrary constant and $\delta\ge 1$ is a positive integer at which the union of directed topologies $\mathcal{G}^{\cup_{i=0}^{\delta-1}} = \bigcup_{i=0}^{\delta-1}\mathcal{G}^i$ over some fixed uniform intervals δ is strongly connected.*

Proof: Consider an arbitrary fixed decision $x^*\in\chi$ and a sequence $\phi(t)$ generated by (19). From the L-Lipschitz continuity of $f_{t,i}$'s and the definition of regret in (4), the regret is bounded as

$$R_T(x^*,x_i) \le \sum_{t=1}^T \left(f_t(\phi(t)) - f_t(x^*) + L\|x_i(t)-\phi(t)\|\right). \tag{32}$$

Note that we can reformulate the first term on the right hand side of (32) as

$$\begin{aligned} f_t(\phi(t)) - f_t(x^*) = &\left(\frac{1}{n}\sum_{i=1}^n f_{t,i}(x_i(t)) - f_t(x^*)\right) \\ &+ \left(\frac{1}{n}\sum_{i=1}^n [f_{t,i}(\phi(t)) - f_{t,i}(x_i(t))]\right). \end{aligned} \tag{33}$$

Based on the convexity of $f_{t,i}$'s, we have

$$\begin{aligned} \sum_{t=1}^T \left(\frac{1}{n}\sum_{i=1}^n f_{t,i}(x_i(t)) - f_t(x^*)\right) \quad &\le \\ \sum_{t=1}^T \left(\frac{1}{n}\sum_{i=1}^n \langle g_i(t), x_i(t)-x^*\rangle\right), \end{aligned} \tag{34}$$

where $g_i(t)\in\partial f_{t,i}(x_i(t))$ is the sub-gradient of $f_{t,i}$ at $x_i(t)$. Thereby, we can express the regret bound based on (33), (34), and the L-Lipschitz continuity of $f_{t,i}$'s as,

$$\begin{aligned} R_T(x^*,x_i) \le \sum_{t=1}^T (&\frac{1}{n}\sum_{i=1}^n \langle g_i(t), x_i(t)-x^*\rangle \\ &+ \frac{L}{n}\sum_{i=1}^n \|x_i(t)-\phi(t)\| + L\|x_i(t)-\phi(t)\|). \end{aligned} \tag{35}$$

The first term on the right had side of (35) can be expanded as

$$\sum_{t=1}^{T}\left(\frac{1}{n}\sum_{i=1}^{n}\langle g_i(t), x_i(t)-x^*\rangle\right) = \sum_{t=1}^{T}(\frac{1}{n}\sum_{i=1}^{n}\langle g_i(t), x_i(t)-\phi(t)\rangle + \frac{1}{n}\sum_{i=1}^{n}\langle g_i(t), \phi(t)-x^*\rangle). \tag{36}$$

Now, we need to bound the terms on the right hand side of (36). The first term is bounded based on the convexity and L-Lipschitz continuity of $f_{t,i}$.[2] In other words,

$$\langle g_i(t), x_i(t)-\phi(t)\rangle \le L\|x_i(t)-\phi(t)\|. \tag{37}$$

Since $x_i(t)$ and $\phi(t)$ are the projections of $y_i(t)$ and $\bar{y}(t)$ respectively, the Lipschitz continuity of $\Pi_{\mathcal{X}}^{\psi}(.,\alpha)$ presented in Lemma 8 of the Appendix imposes a bound on $\|x_i(t)-\phi(t)\|$ as

$$\|x_i(t)-\phi(t)\| \le \alpha(t)\|\bar{y}(t)-y_i(t)\|_*, \tag{38}$$

where $||.||_*$ is the dual norm. Therefore, using the bound in Lemma 9 of the Appendix and noting that $\|g_i(t)\|_* \le L$, we can write (36) as

$$\sum_{t=1}^{T}\left(\frac{1}{n}\sum_{i=1}^{n}\langle g_i(t), x_i(t)-x^*\rangle\right) \le \frac{L}{n}\sum_{t=1}^{T}\sum_{i=1}^{n}\alpha(t)\|\bar{y}(t)-y_i(t)\|_* + \frac{L^2}{2}\sum_{t=2}^{T}\alpha(t-1) + \frac{1}{\alpha(T)}\psi(x^*). \tag{39}$$

Thus, (35), (38), and (39) imply that

$$R_T(x^*, x_i) \le \frac{L^2}{2}\sum_{t=2}^{T}\alpha(t-1) + \frac{1}{\alpha(T)}\psi(x^\star) + L\sum_{t=1}^{T}\alpha(t)\left(\|\bar{y}(t)-y_i(t)\|_* + \frac{2}{n}\sum_{i=1}^{n}\|\bar{y}(t)-y_i(t)\|_*\right). \tag{40}$$

On the other hand, Lemma 10 of the Appendix imposes an upper bound on the last term on the right hand side of (40). Thus, using (27) the regret is further bounded as

$$R_T(x^*, x_i) \le \frac{L^2}{2}\sum_{t=1}^{T-1}\alpha(t) + \frac{1}{\alpha(T)}\psi(x^*) + 3L^2\left(\frac{n}{1-\gamma} + n\delta\nu + 2(1-n)\right)\sum_{t=1}^{T}\alpha(t). \tag{41}$$

[2]Note that convexity of $f_{t,i}$ implies $\langle g_i(t), x-y\rangle \le f_{t,i}(x) - f_{t,i}(y)$. Therefore, based on L-Lipschitz continuity of $f_{t,i}$'s, we have $||g_i||_* \le L$ and we can deduce (37).

The statement of the theorem now follows from the integral test on $\alpha(t) = k/\sqrt{t}$ and $\psi(x^*) \le R^2$. ∎

Theorem 6 indicates the "good" performance of online-DWDA through sub-linear regret and highlights the importance of the underlying network topology through the parameters γ and ν examined in §VI.

Next we present the regret analysis for the (temporal) running average estimates at each agent exhibiting a similar dependence on the network connectivity.

Corollary 7. *Given the sequence $\widetilde{x}_i(t)$ generated by line 7 in* Algorithm 1 *for all $i \in [n]$ with $\psi(x^*) \le R^2$ and $\alpha(t) = k/\sqrt{t}$, we have*

$$R_T(x^*, \widetilde{x}_i) \le 2\left(\frac{R^2}{k} + kL^2\left(\frac{6n}{1-\gamma} + 6n\delta\nu + 1\right)\right)\sqrt{T}.$$

Proof: Since the cost function $f_t(x(t))$ is convex, $f_t(\widetilde{x}_i(t)) \le \frac{1}{t}\sum_{s=1}^{t} f_t(x_i(s))$. Therefore, we have

$$f_t(\widetilde{x}_i(t)) - f_t(x^*) \le \frac{1}{t}R_t(x^*, x_i). \tag{42}$$

Thus, the running average regret is bounded as

$$R_T(x^*, \widetilde{x}_i) \le \sum_{t=1}^{T}\left(\frac{1}{t}R_t(x^*, x_i)\right). \tag{43}$$

On the other hand, the regret bound (31) implies that

$$R_T(x^*, \widetilde{x}_i) \le \left(\frac{R^2}{k} + kL^2\left(\frac{6n}{1-\gamma} + 6n\delta\nu + 1\right)\right) \times \left(\sum_{t=1}^{T}\frac{1}{\sqrt{t}}\right). \tag{44}$$

The statement of the corollary now follows from the integral test on the right hand side of (44). ∎

VIII. Online Distributed Estimation

Adopting the least squares point of view, a model for online estimation over a distributed sensor network is presented in this section. The distributed sensor network aims to estimate a random vector $\theta \in \Theta = \left\{\theta \in \mathbb{R}^d | \, \|\theta\|_2 \le \theta_{\max}\right\}$. Note that Θ is a closed convex set containing the origin. The observation vector $z_{t,i} : \mathbb{R}^d \to \mathbb{R}^{p_i}$ represents the ith sensor measurement at time t which is uncertain and time-varying due to the sensor's susceptibility to unknown environmental factors such as jamming. The sensor is assumed (not necessarily accurately) to have a linear model of the form $h_i(\theta) = H_i\theta$, where $H_i \in \mathbb{R}^{p_i \times d}$ is the observation matrix of sensor i and $\|H_i\|_1 \le h_{\max}$ for all i. Consider now the interconnection topology between the sensors defined via the directed graph $\mathcal{G} = (V, E, W)$, where the set of n sensors are represented by V. The presence of an edge $(j,i) \in E$ indicates an information flow from sensor j to sensor i. The set of agents that are communicating with agent i is defined as the neighborhood set $N(i) = \{j \in V | (j,i) \in E\}$. Figure 1 graphically summarizes the problem setup. The objective is to find the argument $\hat{\theta}$ that minimizes the cost function

$$f_t(\hat{\theta}) = \frac{1}{n}\sum_{i=1}^{n} f_{t,i}(\hat{\theta}) \quad \text{subject to } \hat{\theta} \in \Theta, \tag{45}$$

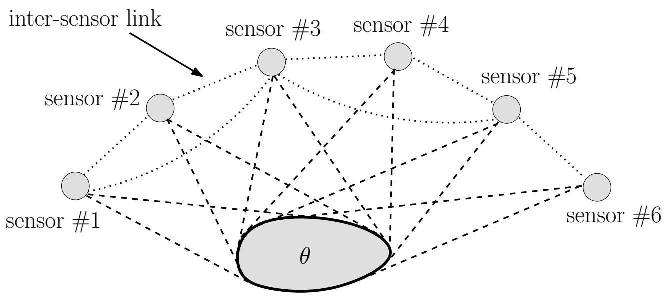

Figure 1. A graphical representation of a distributed sensor network.

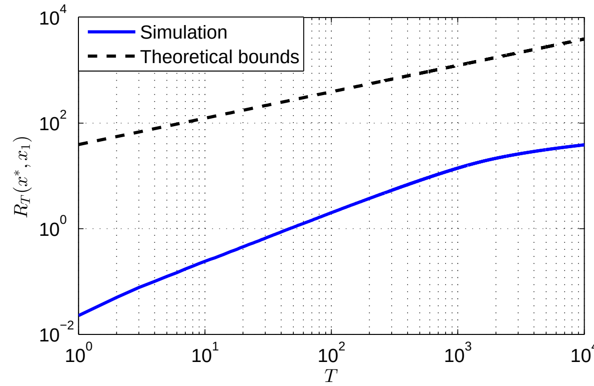

Figure 2. Accuracy of the bounds in (31) where $\mathcal{G}$ is a 100 node random directed graph with edge probability $p = 0.08$, $\nu = 5$, and $\gamma = 0.2034$.

where

$$f_{t,i}(\hat{\theta}) = \frac{1}{2}\left\|z_{i,t} - H_i\hat{\theta}\right\|_2^2 \tag{46}$$

is a convex cost function associated with sensor $i \in [n]$. It is assumed that the value of this local cost at time t is only revealed to the sensor after $\hat{\theta}(t)$ has been computed, that is, the local error functions are allowed to change over time in an unpredictable manner due to modeling errors and uncertainties in the environment. The (sub)gradient of the local estimation error (46),

$$\partial f_{t,i}(\hat{\theta}) = H_i^T\left(z_{t,i}(\theta) - H_i\hat{\theta}\right), \tag{47}$$

is also assumed to be known to the sensor and its neighbors. We note that the cumulative cost at time T is defined as $f(\hat{\theta}) = \sum_{t=1}^T f_t(\hat{\theta})$.

In an offline setting, for all $t \in [T]$, each sensor i has a noisy observation $z_{t,i} = H_i\theta + v_{t,i}$, where $v_{t,i}$ is generally assumed to be (independent) white noise. In this case, the centralized time-averaged optimal estimate for (45) is

$$\theta^* = \frac{1}{T}\sum_{t=1}^T\left(\sum_{i=1}^n H_i^T\Sigma_{t,i}^{-1}H_i\right)^{-1}\left(\sum_{i=1}^n H_i^T\Sigma_{t,i}^{-1}z_{i,t}\right), \tag{48}$$

where $\Sigma_{t,i}$ is the covariance of the error observed by sensor i at time t [40]. For the case where $\theta \in \mathbb{R}$, $\Sigma_{t,i} = I$, and $H_i = 1$, the optimal estimate is $\theta^* = \frac{1}{nT}\sum_{i=1}^n\sum_{t=1}^T z_{t,i}$. However, this approach to estimation problems is not suitable in scenarios where the noise characteristics are unknown. For example when a wireless sensor network is employed in an unknown and dynamic environment, the measurement signal can be blocked or degraded due to obstructions such as walls, furniture, trees, or buildings. This is known as the shadowing effect and usually modeled as a function of the environment in which the network is deployed. Another example is jamming of one or more sensors in the network. When the sensor resolution and noise characteristics are not known ahead of time, the dynamic weight selection procedure discussed in §V can be employed to eliminate the information from the jammed sensors.

An online framework is particularly suitable for such estimation problems without relying on prior assumption or knowledge of the statistical properties of the data. In the proposed distributed estimation algorithm, at time step t, each sensor i estimates $\hat{\theta}_i \in \Theta$ based on the local information available to it and then an "oracle" reveals the cost $f_t(\hat{\theta}_i)$.

The bounds presented in Theorem 6 apply after selecting $\psi(\hat{\theta}) = \frac{1}{2}\|\hat{\theta}\|_2^2$ and the parameter $\alpha(t)$ accordingly. In order to find the constants R and L featured in the result, we note that for $\hat{\theta} \in \Theta$, $\psi(\hat{\theta}) \leq \frac{1}{2}\theta_{\max}^2$, and thus $R \leq \frac{1}{\sqrt{2}}\theta_{\max}$. In this example, we assume that the observation for agent i at time t is of the form $z_{t,i} = a_t\theta + b_t$ for some $a \in (0, a_{\max})$ and $b \in (-b_{\max}, b_{\max})$. Therefore,

$$\sup_{\theta\in\chi}\|z_{t,i}(\theta)\|_2 \leq a_{\max}\theta_{\max} + b_{\max}.$$

Further, the function $f_{t,i}$ is Lipschitz as it is convex on a compact domain and the Lipschitz constant can be found by observing that

$$\begin{aligned}
&\left|f_{k,i}(\hat{\theta}) - f_{k,i}(\phi)\right| \\
&\leq \frac{1}{2}\left|\left(\hat{\theta}-\phi\right)^T H_i^T H_i\left(\hat{\theta}-\phi\right)\right| + \left|z^T H_i\left(\hat{\theta}-\phi\right)\right| \\
&\leq \left(\frac{1}{2}\|H_i\|_F^2\left\|\hat{\theta}-\phi\right\|_2 + \|z_{t,i}\|_2\|H_i\|_F\right)\left\|\hat{\theta}-\phi\right\|_2
\end{aligned}$$

and thus $L = \left(\frac{1}{2}\theta_{\max}h_{\max} + a_{\max}\theta_{\max} + b_{\max}\right)h_{\max}$. Hence $R_T(\theta^*, \hat{\theta}_i)/T \to 0$ and the algorithm performs as well as best fixed estimate θ^* in hindsight (48) "on average". For the case where $\theta_t = \theta_{t+1}$ for $t = 1, 2, \ldots, T$, θ^* is the optimal estimate.

The online-DWDA and DOA algorithms have been implemented on the described distributed sensor setup for $n = 100$ sensors. The objective is to estimate a scalar $\theta \in \left(-\frac{1}{2}, \frac{1}{2}\right)$ with a fixed $H_i \in \left(0, \frac{1}{4}\right)$ for each agent; hence $\sup_i|H_i| = \frac{1}{4}$. In this example, we have assumed $a \in (0,1)$, $b \in \left(-\frac{1}{4}, \frac{1}{4}\right)$, $\beta = 0.9$, and $k = \frac{1}{4}$. Thus, $d = 1$, $\Theta = \left(-\frac{1}{2}, \frac{1}{2}\right)$, $h_{\max} = \frac{1}{4}$, $\theta_{\max} = \frac{1}{2}$, $R = \frac{1}{2\sqrt{2}}$, and $L = \frac{13}{64}$.

The online-DWDA and DOA algorithms were also applied to random sensor network with edge probability $p = 0.08$. Figure 2 shows a qualitative agreement of the theoretical regret bound (31) and simulation results, indicating that $R_T(x^*, x_1) = O(\sqrt{T})$. The improved performance of the adaptive network topology has been emphasized in Figure 3 in the context of a jamming scenario, where a number of

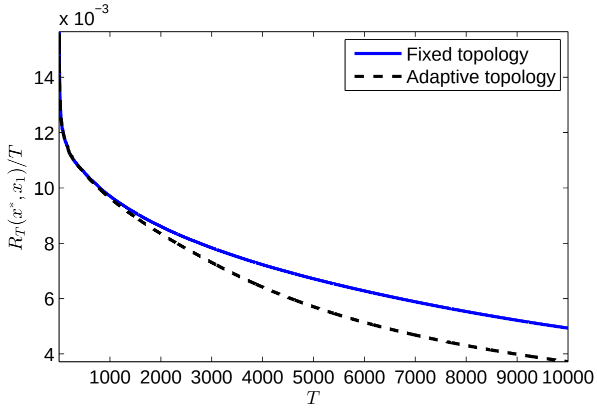

Figure 3. Regret performance over fixed and adaptive network topologies where $\mathcal{G}$ is a 100 node random 4-regular graph and 25 sensors are assumed to have been jammed. For the jammed sensors, $b_t = b_{\max}$ and $a_t = H_i$.

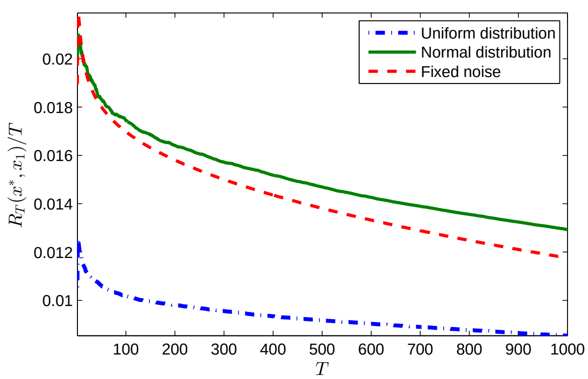

Figure 4. Regret performance for three different observation noise characteristics, where $\mathcal{G}$ is a 100 node random 4-regular graph. The noise signals have been generated from distributions with mean $-b_{\max}$ and standard deviation $b_{\max}$.

sensors in the random regular network are assumed to have been jammed. This figure also demonstrates that the adaptive sensor network has a better regret performance as compared with the fixed topology sensor network.

In addition, the performance of the proposed adaptive online distributed estimation in the presence of various noise types is presented in Figure 4. These simulation results indicate that $R_T(\theta^*, \hat{\theta}_1) = O(\sqrt{T})$ for all noise types considered without a prior assumption on the noise characteristics.

Furthermore, the role of network connectivity in the performance of the algorithm has been emphasized in Figure 5 for various classes of network topologies, directly correlated to the network connectivity measure γ. This result can be applied to designing sensor network topologies that operate in highly uncertain environments. Suitable metrics for such a topology design procedure include $\lambda_2(P(\mathcal{G}^0))$ that predictably scales with n, such as random regular graphs and expander graphs [41].

IX. Conclusion

This paper studies the problem of decentralized optimization on dynamic networks operating in an uncertain

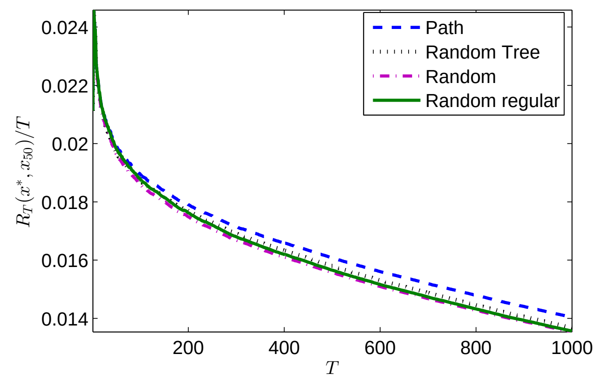

Figure 5. The performance of the online distributed estimation algorithm on four different 100 node graphs with $\gamma = \{0.8999,\ 0.8998,\ 0.7939,\ 0.4110\}$, for the path, random tree, random k-regular with $k = 4$, and random graph with edge probability $p = 0.08$, respectively, presented in increasing order of performance.

environment. An algorithm has been presented that evolves distributively using only local information available to the agents in the network. Our analysis provided a convergence rate of $O(1/\sqrt{T})$ and a sub-linear regret of $O(\sqrt{T})$ in the online setting. In addition, the convergence analysis of the distributed optimization algorithm highlighted the role of two measures of network connectivity.

A distributed dynamic weight selection procedure has also been proposed that on average, performs as well as the best strategy for information diffusion in hindsight. It was demonstrated that this approach improves the convergence rate by mitigating the network effects.

This work can be applied in the context of a range of applications such as mobile sensor networks where the network is susceptible to unknown errors, jamming, link failure, and a varying network topology. Moreover, this work can be extended in several directions. One such extension, which is the subject of our future work, involves examining online distributed filtering. More generally, the online approach can be adopted for a host of network dynamic systems that operate in unstructured environments, requiring that a learning algorithm is embedded in the network-level decision-making process.

X. Appendix

We note that Lemmas 8 and 9 have been shown by Duchi *et al.*, [10] and are presented here for reference.

Lemma 8. *[10] For any $u, v \in \mathbb{R}^m$, and under the conditions stated for proximal function ψ and step size $\alpha(t)$, we have $\|\Pi^{\psi}_{\chi}(u,\alpha) - \Pi^{\psi}_{\chi}(v,\alpha)\| \leq \alpha\|u - v\|_*$.*

Lemma 9. *[10] For any positive and non-increasing sequence $\alpha(t)$ and $x^* \in \chi$,*

$$\sum_{t=1}^{T}\langle \overline{g}(t), \phi(t) - x^*(t)\rangle \leq \frac{1}{2}\sum_{t=1}^{T}\alpha(t-1)\|\overline{g}(t)\|_\star^2 + \frac{1}{\alpha(T)}\psi(x^*),$$

where the sequence $\phi(t)$ is generated by (19).

The following result presents a bound on $\|\bar{y}(t) - y_i(t)\|_*$ proportional to the error incurred by the decentralized update in Algorithm 1.

Lemma 10. *For sequences $y_i(t)$ and $\bar{y}(t)$ generated by line 5 of Algorithm 1 and (18), respectively, we have,*

$$\|\bar{y}(t) - y_i(t)\|_* \leq L \sum_{k=0}^{t-2} \sum_{j=1}^{n} \left| P_{ij}^{(t-1,k+1)} - \pi_j \right| + 2L,$$

for all $i \in [n]$.

Proof: Reformulating the update in line 5 of Algorithm 1 for all $i \in [n]$, by induction through s steps we have,

$$y_i(t) = \sum_{j=1}^{n} P_{ij}^{(t-1,t-s)} y_j(t-s) + \sum_{k=t-s}^{t-2} \sum_{j=1}^{n} P_{ij}^{(t-1,k+1)} g_j(k) + g_i(t-1). \tag{49}$$

Since $\bar{y}(t)$ evolves as in (18), by setting $s = t$ in (49) and assuming $y_i(0) = 0$, we get,

$$\bar{y}(t) - y_i(t) = \sum_{k=0}^{t-2} \left(\sum_{j=1}^{n} \left(\pi_j - P_{ij}^{(t-1,k+1)} \right) g_j(k) \right) + \bar{g}(t-1) - g_i(t-1). \tag{50}$$

Thus, the dual norm of (50) is bounded as

$$\|\bar{y}(t) - y_i(t)\|_* \leq \| \sum_{k=0}^{t-2} \left(\sum_{j=1}^{n} \left(\pi_j - P_{ij}^{(t-1,k+1)} \right) g_j(k) \right) \|_* + \|\bar{g}(t-1) - g_i(t-1)\|_*, \tag{51}$$

and the right hand side of (51) can be bounded by

$$\|\bar{y}(t) - y_i(t)\|_* \leq \sum_{k=0}^{t-2} \sum_{j=1}^{n} \left| P_{ij}^{(t-1,k+1)} - \pi_j \right| \|g_j(k)\|_* + \|\bar{g}(t-1) - g_i(t-1)\|_*. \tag{52}$$

Since $\|g_i(t)\|_\star \leq L$,

$$\|\bar{y}(t) - y_i(t)\|_* \leq L \sum_{k=0}^{t-2} \sum_{j=1}^{n} \left| P_{ij}^{(t-1,k+1)} - \pi_j \right| + 2L. \tag{53}$$

■

The following proposition provides an upper bound on ν in the convergence rate (25) over fixed topology networks.

Proposition 11. *Consider a set $\mathcal{P}$ of stochastic matrices with positive diagonal elements, representing an arbitrarily strongly connected topologies over n nodes, i.e., $P^k \in \mathcal{P}$ for all positive integers k. Suppose that any two matrices P^{k_1} and P^{k_2} are of the same type.[3] Then, there exists an integer ν,*

$$1 \leq \nu \leq \min_{i\in[n]} \max_{j\in[n]} dist(j, i), \tag{54}$$

for which if the sequence $Q = P^{(m+\nu-2,m)}$ of matrices in $\mathcal{P}$ is not scrambling, $P^{m+\nu-1}Q$ is scrambling.

[3]The matrices A and B are of the same type if they have zero elements and positives elements in the same place.

Proof: Let $Q_1 = P^m$ and $Q_2 = P^{m+1}P^m$. Thus every entry of Q_2 is represented as

$$[Q_2]_{ij} = \sum_{k=1}^{n} \left[P^{m+1}\right]_{ik} \left[P^m\right]_{kj}.$$

Since $[P^m]_{ii} > 0$ for all $i \in [n]$ and integer $m \geq 1$, the entry $[Q_2]_{ij}$ is positive if $(j, i) \in E$, $(i, j) \in E$, or if there exists a node $k \in [n]$ in the directed path from node j to node i with $\text{dist}(j, i) = 2$. Thus, the corresponding zero entry of Q_1 that has one of the aforementioned properties will be positive in Q_2. By induction, it follows that the entry of $[Q_\nu]_{ij}$ will be positive if $(j, i) \in E$, $(i, j) \in E$, or if there exists a node $k \in [n]$ in the directed path from node j to node i with $\text{dist}(j, i) = \nu_i$. Therefore, for each row i of Q_ν, all entries will be positive when

$$\nu_i = \max_{j\in[n]} \text{dist}(j, i).$$

Note that every element of any row of the sequence $Q = P^{(m+\nu-2,m)}$ of matrices in $\mathcal{P}$ is positive, the matrix Q is scrambling and ν satisfies the bound (54). ■

A similar observation for the adjacency matrix of $\mathcal{G}$ can be found in the algebraic graph theory literature such as [42].

References

[1] S. Boyd, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[2] D. Jakovetic, J. Xavier, and J. M. F. Moura, "Cooperative Convex Optimization in Networked Systems: Augmented Lagrangian Algorithms With Directed Gossip Communication," *IEEE Transactions on Signal Processing*, vol. 59, no. 8, pp. 3889–3902, Aug. 2011.

[3] D. Mosk-Aoyama, T. Roughgarden, and D. Shah, "Fully distributed algorithms for convex optimization problems," *Distributed Computing*, vol. 4731, pp. 492–493, 2007.

[4] Y. Kim and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 116–120, 2006.

[5] K. Chi and X. Jiang, "Topology design of network-coding-based multicast networks," *IEEE Transaction on Parallel and Distributed Systems*, vol. 19, no. 5, pp. 627–640, 2008.

[6] R. Dai and M. Mesbahi, "Optimal topology design for dynamic networks," *IEEE Conference on Decision and Control and European Control Conference*, pp. 1280–1285, 2011.

[7] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control,*, vol. 54, pp. 48–61, 2009.

[8] I. Lobel and A. Ozdaglar, "Distributed subgradient methods for convex optimization over random networks," *IEEE Transactions on Automatic Control,*, pp. 1291–1306, 2011.

[9] I. Lobel, A. Ozdaglar, and D. Feijer, "Distributed multi-agent optimization with state-dependent communication," *Mathematical programming*, vol. 129, pp. 255–284, 2011.

[10] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: convergence analysis and network scaling," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 592–606, 2012.

[11] S. Lee and A. Nedic, "Distributed random projection algorithm for convex optimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 221–229, 2013.

[12] S. Ram, A. Nedic, and V. Veeravalli, "Incremental stochastic subgradient algorithms for convex optimization," *SIAM Journal on Optimization*, pp. 1–30, 2009.

[13] S. Sundhar Ram, A. Nedić, and V. V. Veeravalli, "Distributed Stochastic Subgradient Projection Algorithms for Convex Optimization," *Journal of Optimization Theory and Applications*, vol. 147, no. 3, pp. 516–545, 2010.
[14] A. Agarwal and J. Duchi, "Distributed delayed stochastic optimization," *IEEE Conference on Decision and Control*, pp. 5451–5452, 2012.
[15] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," *International Conference on Machine Learning*, pp. 421–422, 2003.
[16] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Machine Learning*, vol. 69, pp. 169–192, 2007.
[17] L. Xiao, "Dual Averaging Methods for Regularized Stochastic Learning and Online Optimization," *Journal of Machine Learning Research*, vol. 11, pp. 2543–2596, 2010.
[18] F. Yan, S. Sundaram, S. V. N. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, pp. 2483 – 2493, 2013.
[19] M. Raginsky, N. Kiarashi, and R. Willett, "Decentralized Online Convex Programming with Local Information," *American Control Conference*, pp. 5363–5369, 2011.
[20] S. Hosseini, A. Chapman, and M. Mesbahi, "Online Distributed Optimization via Dual Averaging," *IEEE Conference on Decision and Control*, pp. 1484 – 1489, 2013.
[21] A. Ghosh and S. Boyd, "Growing well-connected graphs," *IEEE Conference on Decision and Control*, pp. 6605–6611, 2006.
[22] 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.
[23] Y. Wan, S. Roy, and A. Saberi, "Network design problems for controlling virus spread," *IEEE Conference on Decision and Control*, pp. 3925–3932, 2007.
[24] A. Chapman and M. Mesbahi, "Semi-autonomous consensus: network measures and adaptive trees," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 19–31, 2013.
[25] A. Chapman, E. Schoof, and M. Mesbahi, "Distributed Online Topology Design for Disturbance Rejection," *IEEE Conference on Decision and Control*, pp. 817 – 822, 2013.
[26] S. Aldosari and J. Moura, "Topology of sensor networks in distributed detection," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1061–1064, 2006.
[27] S. Kar and J. Moura, "Sensor networks with random links: Topology design for distributed consensus," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3315–3326, 2008.
[28] A. Laszka, L. Buttyán, and D. Szeszlér, "Designing robust network topologies for wireless sensor networks in adversarial environments," *Pervasive and Mobile Computing*, vol. 9, no. 4, pp. 546–563, 2013.
[29] N. Littlestone and M. Warmuth, "The weighted majority algorithm," *Information and Computation*, vol. 108, pp. 212–261, 1994.
[30] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.
[31] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. NJ: Princeton University Press, 2010.
[32] S. Shalev-Shwartz, "Online learning and online convex optimization," *Foundations and Trends in Machine Learning*, vol. 4, pp. 107–194, 2012.
[33] S. Bubeck, "Introduction to Online Optimization," *Lecture Notes*, 2011.
[34] E. Hazan, "The Convex Optimization Approach to Regret Minimization," *Optimization for machine learning*, pp. 287–294, 2011.
[35] Y. Nesterov, "Primal-dual subgradient methods for convex problems," *Mathematical Programming*, vol. 120, pp. 221–259, 2007.
[36] C. W. Wu, "On bounds of extremal eigenvalues of irreducible and m-reducible matrices," *Linear Algebra and its Applications*, vol. 402, pp. 29–45, 2005.
[37] J. Anthonisse and H. Tijms, "Exponential convergence of products of stochastic matrices," *Journal of Mathematical Analysis and Applications*, vol. 59, no. 2, pp. 360–364, 1977.
[38] M. E. Mehran Mesbahi, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
[39] A. Jadbabaie and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.
[40] L. Xiao, S. Boyd, and S. Lall, "A space-time diffusion scheme for peer-to-peer least-squares estimation," *International Conference on Information Processing in Sensor Networks*, pp. 168 – 176, 2006.
[41] B. Bollobás, *Modern Graph Theory*. New York: Springer, 1998.
[42] C. Godsil and G. Royle, *Algebraic Graph Theory*. Springer, 2001.