Systematic Analysis of Distributed Optimization Algorithms over Jointly-Connected Networks

Bryan Van Scoy Laurent Lessard

To Appear, IEEE Conference on Decision and Control, 2020

Abstract

We consider the distributed optimization problem, where a group of agents work together to optimize a common objective by communicating with neighboring agents and performing local computations. For a given algorithm, we use tools from robust control to systematically analyze the performance in the case where the communication network is *time-varying*. In particular, we assume only that the network is jointly connected over a finite time horizon (commonly referred to as B-connectivity), which does *not* require connectivity at each time instant. When applied to the distributed algorithm DIGing, our bounds are orders of magnitude tighter than those available in the literature.

1 Introduction

Many recent and emerging applications in multi-agent systems require groups of agents to cooperatively solve problems. Examples of agents include computing nodes, robots, or mobile sensors connected in a network. In this paper, we consider the *distributed optimization problem*, where each agent $i \in \{1, \dots, n\}$ has a local function $f_i : \mathbb{R}^d \to \mathbb{R}$, and agents cooperate to minimize the sum of the functions over all agents,

$$\underset{y \in \mathbb{R}^d}{\text{minimize}} \ \sum_{i=1}^{n} f_i(y). \tag{1}$$

Each agent is capable of evaluating its local gradient ∇f_i, communicating information with neighboring agents, and performing local computations. This problem is relevant in applications such as large-scale machine learning [2], distributed spectrum sensing [1], and sensor networks [7].

We are interested in cases where the communication network among agents is *time varying*, which occurs in mobile agents with range-limited communication and systems with noisy and unreliable communication.

B. Van Scoy and L. Lessard were both with the University of Wisconsin–Madison, Madison, WI 53706, USA at the time of initial submission. B. Van Scoy is now with the Department of Electrical and Computer Engineering at Miami University, Oxford, OH 45056, USA, and L. Lessard is now with the Department of Mechanical and Industrial Engineering at Northeastern University, Boston, MA 02115, USA.

bvanscoy@miamioh.edu, l.lessard@northeastern.edu

This material is based upon work supported by the National Science Foundation under Grants No. 1750162 and 1936648.

In the past several years, numerous algorithms have been proposed for distributed optimization (see [8] and the reference therein). While some algorithms have been studied in the time-varying scenario [5,9], the analysis is typically performed on a case-by-case basis, resulting in lengthy convergence proofs and conservative bounds. On the other hand, the recent work [8] provides a systematic framework for deriving convergence bounds for a large class of distributed algorithms; this approach yields a straightforward comparison between various algorithms, but the analysis requires the network to be connected at *every* iteration, which is often unrealistic.

In this work, we provide a systematic analysis in the time-varying scenario for distributed algorithms whose state update is synchronous, homogeneous across agents, time invariant, and linear in the state, local gradient, and a weighted average among neighbors. Unlike [8], we only assume that the union of every B consecutive networks is connected; this assumption is common in the consensus literature and is often referred to as *B-connectivity* [3,4]. Our main contributions are the following.

- **Generality.** Our analysis applies to a large class of distributed algorithms, allowing for straighforward comparisons without deriving lengthy convergence proofs for each algorithm individually.
- **Tightness.** For the gradient tracking algorithm DIGing [5,6], we improve on existing bounds. While the available bounds scale with the number of agents n (and become vacuous as $n \to \infty$), our bounds are orders of magnitude tighter and independent of n.

In Section 2, we state our assumptions on the local functions and communication network, as well as describe the class of algorithms considered. We then describe our analysis and present our main result in Section 3. We conclude with a case study for DIGing in Section 4, where we compare our results with those in the literature.

Notation. We use $\mathbf{1}$ and $\mathbf{0}$ to denote the $n \times 1$ vectors of all ones and zeros, and $\mathbb{R}^{m \times m}$ ($\mathbb{S}^m$) to denote the set of $m \times m$ real (symmetric) matrices. A matrix $A \in \mathbb{S}^m$ is denoted positive (semi)definite by $A \succ 0$ ($A \succeq 0$). We denote the vertical concatenation of a list of matrices or vectors by $\mathsf{vcat}(A_1, \dots, A_n)^\mathsf{T} = \begin{bmatrix} A_1^\mathsf{T} & \dots & A_n^\mathsf{T} \end{bmatrix}$. Subscripts i and j refer to agents, and index k denotes the discrete time index. For a signal $x_i(k)$ on agent i at time k, we denote the aggregation over all agents as

$x(k) = \mathsf{vcat}(x_1(k), \ldots, x_n(k))$.

2 Problem setup

We now discuss the objective functions, communication networks, and algorithms that we consider in this paper.

2.1 Objective function

We assume that the objective function has the form (1), where each agent i can evaluate its local gradient ∇f_i. Furthermore, we assume that the gradient of each local function satisfies the following *sector bound*.

Assumption 1. There exist parameters $0 < m \leq L$ such that each local function f_i is continuously differentiable and satisfies the inequality

$$\big(\nabla f_i(y) - \nabla f_i(y_{\text{opt}}) - m\,(y - y_{\text{opt}})\big)^{\mathsf{T}} \times \big(\nabla f_i(y) - \nabla f_i(y_{\text{opt}}) - L\,(y - y_{\text{opt}})\big) \leq 0$$

for all $y \in \mathbb{R}^d$, where $y_{\text{opt}} \in \mathbb{R}^d$ is the optimizer of (1).

Remark. One way to satisfy Assumption 1 is if each ∇f_i is L-Lipschitz continuous and each f_i is m-strongly convex, though in general, Assumption 1 is much weaker.

The condition ratio $\kappa := L/m$ captures how much the curvature of the objective function varies. If f_i is twice differentiable, then κ is an upper bound on the condition number of the Hessian $\nabla^2 f_i$. In general, as $\kappa \to \infty$, the functions become poorly conditioned and are therefore more difficult to optimize using first-order methods.

2.2 Communication network

We represent the communication network among agents as a time-varying directed graph. Each agent corresponds to a node in the graph, and a directed edge from node i to node j indicates that agent i sends information to agent j. Each agent processes the communicated data by computing a weighted sum of the information from its neighbors (the set of agents from which it receives information). We characterize this diffusion process by a *gossip matrix* $W(k) \in \mathbb{R}^{n \times n}$, where the discrete time index k denotes the iteration of the algorithm. We make the following assumptions on the gossip matrices.

Assumption 2. The set of gossip matrices $\{W(k)\}_{k=0}^{\infty}$ satisfies the following properties at each iteration k.

1. **Graph sparcity:** $W_{ij}(k) = 0$ if agent i does not receive information from agent j at time k.
2. **Weight-balanced:** $W(k)\mathbf{1} = W(k)^{\mathsf{T}}\mathbf{1} = \mathbf{1}$.
3. **Spectrum property:** $\|\frac{1}{n}\mathbf{1}\mathbf{1}^{\mathsf{T}} - W(k)\|_2 \leq 1$.
4. **Joint-spectrum property:** There exists a positive integer B and a scalar $\sigma \in [0, 1)$, called the *spectral gap*, such that

$$\left\| \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathsf{T}} - \prod_{\ell=k}^{k+B-1} W(\ell) \right\|_2^{1/B} \leq \sigma.$$

Remark. Our assumption on the gossip matrices does not require the graph to be connected at each iteration if $B > 1$ and is common in the consensus and distributed optimization literature; see [4, 5].

2.3 Algorithm

We now describe a broad class of algorithms that may be used to have the group of agents solve the distributed optimization problem (1), where each agent may perform local computations and communicate with neighboring agents. At each iteration k, each agent i has a local state variable $x_i(k) \in \mathbb{R}^{s \times d}$ that it updates as follows:

$$\begin{bmatrix} x_i(k+1) \\ y_i(k) \\ z_i(k) \end{bmatrix} = \begin{bmatrix} A & B_u & B_v \\ C_y & D_{yu} & D_{yv} \\ C_z & D_{zu} & D_{zv} \end{bmatrix} \begin{bmatrix} x_i(k) \\ u_i(k) \\ v_i(k) \end{bmatrix}, \tag{2a}$$

$$u_i(k) = \nabla f_i\big(y_i(k)\big), \tag{2b}$$

$$v_i(k) = \sum_{j=1}^{n} W_{ij}(k)\, z_j(k). \tag{2c}$$

The local gradient ∇f_i is evaluated[1] at $y_i(k) \in \mathbb{R}^{1 \times d}$ in (2b), and the quantity $z_i(k) \in \mathbb{R}^{c \times d}$ is transmitted to neighboring agents in (2c). We also allow for linear state-input invariants to be enforced with

$$\sum_{j=1}^{n} \big(F_x\, x_j(k) + F_u\, u_j(k)\big) = 0. \tag{2d}$$

Such invariants typically arise from requiring a particular initialization for the algorithm.

The matrices $A \in \mathbb{R}^{s \times s}$, $D_{yu} \in \mathbb{R}^{1 \times 1}$, and $D_{zv} \in \mathbb{R}^{c \times c}$ are square with the other matrices having compatible dimensions. Here, s is the number of local states on each agent and c is the number of variables that each agent communicates with its neighbors at each iteration.

We want each agent's trajectory of algorithm (2) to converge to the optimizer of the distributed optimization problem (1), that is, $y_i(k) \to y_{\text{opt}}$ for all $i \in \{1, \ldots, n\}$. To obtain this, we need (i) the algorithm to have a fixed point corresponding to the optimal solution, and (ii) the trajectory to converge to this fixed point. Existence of such a fixed point places requirements on the structure of the algorithm, which we characterize in the following proposition; we prove the result in the Appendix.

Proposition 1. An algorithm of the form (2) has a fixed point such that $y_1^\star = \ldots = y_n^\star$ and $\sum_{i=1}^n \nabla f_i(y_i^\star) = 0$ for any local functions and any set of weight-balanced gossip matrices if and only if the algorithm matrices satisfy

$$\mathsf{null}\left(\begin{bmatrix} A - I & B_v \\ C_z & D_{zv} - I \\ F_x & 0 \end{bmatrix}\right) \cap \mathsf{row}\big(\begin{bmatrix} C_y & D_{yv} \end{bmatrix}\big) \neq \{0\} \tag{3a}$$

$$\text{and} \quad \begin{bmatrix} B_u \\ D_{yu} \\ D_{zu} \end{bmatrix} \in \mathsf{col}\left(\begin{bmatrix} A - I \\ C_y \\ C_z \end{bmatrix}\right). \tag{3b}$$

[1] We interpret the gradient ∇f_i as a mapping from $\mathbb{R}^{1 \times d}$ to $\mathbb{R}^{1 \times d}$.

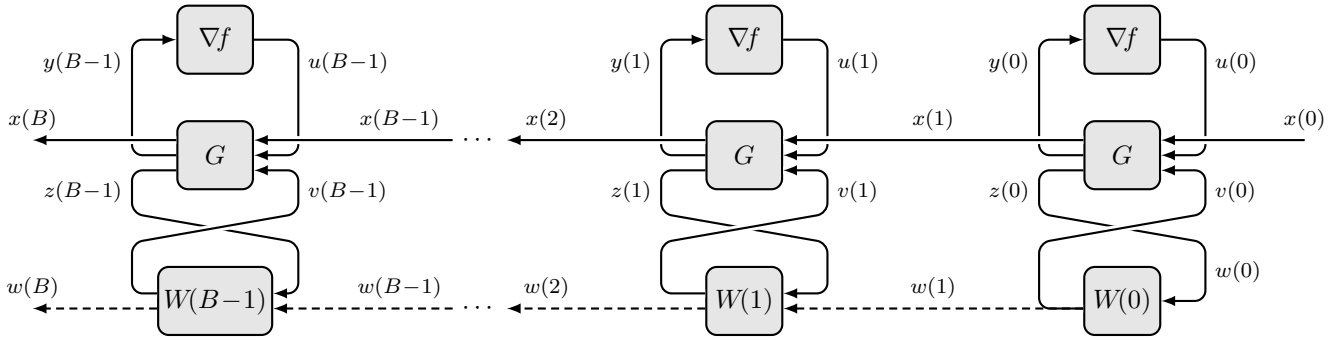

Figure 1: Block diagram used to analyze a distributed algorithm over a B-connected communication network. The algorithm is unravelled over B time steps, where we have introduced the virtual signal $w(k)$ (dashed lines) to multiply the gossip matrix $W(k)$ at each iteration (this signal is used only in the analysis and is *not* part of the algorithm).

3 A systematic analysis

We now describe our systematic method for analyzing the convergence properties of an algorithm of the form (2) when the communication network is B-connected (that is, satisfies Assumption 2). We first motivate our analysis with the main ideas and then state our main result.

3.1 Motivation

The main idea behind our analysis is to first unravel the algorithm for B time steps, and then to systematically search for a Lyapunov function for this unravelled system using linear matrix inequalities.

Consider the first B iterations of the algorithm as shown in Figure 1, where the matrix

$$G = \begin{bmatrix} A & B_u & B_v \\ C_y & D_{yu} & D_{yv} \\ C_z & D_{zu} & D_{zv} \end{bmatrix} \tag{4}$$

describes the algorithm update in (2a). In the diagram, we have introduced the virtual signal $w_i(k)$ on agent i at iteration k, defined recursively by

$$w_i(k+1) = \sum_{j=1}^{n} W_{ij}(k)\, w_j(k)$$

with $w_i(0) = z_i(0)$. While this virtual signal is not part of the algorithm, we make use of it in the analysis.

To prove convergence of this unravelled system, we search for a quadratic *Lyapunov function* of the form

$$V(\xi) = \big\langle \xi, \big(\underbrace{\tfrac{1}{n}\mathbf{1}\mathbf{1}^{\mathsf{T}} \otimes P}_{\text{consensus}} + \underbrace{(I - \tfrac{1}{n}\mathbf{1}\mathbf{1}^{\mathsf{T}}) \otimes Q}_{\text{disagreement}} \big)\, \xi \big\rangle, \tag{5}$$

where ξ is the state of the unravelled system. The first term in V describes the state in the *consensus direction*, that is, the component of the agents' states which are in agreement, and is characterized by the matrix P. The second term describes the *disagreement directions*, which are all directions orthogonal to the consensus direction, and is characterized by the matrix Q. If there exist P and Q such that the function is both positive definite and sufficiently decreasing along trajectories of the algorithm, then V is a valid Lyapunov function that we can use to certify convergence of the state to a fixed point. For algorithms satisfying the conditions in Proposition 1, this fixed point corresponds to the optimal solution of the distributed optimization problem as desired.

To find a valid Lyapunov function, we systematically search over the matrices P and Q using linear matrix inequalities. Such LMIs arise by replacing any nonlinear and/or unknown blocks in Figure 1 with constraints on their input and output signals. In particular, we use the following:

- The input $y(k)$ and output $u(k)$ of the gradient of the objective function satisfy a quadratic inequality from Assumption 1.
- The inputs $z(k)$ and $w(k)$ and corresponding outputs $v(k)$ and $w(k+1)$ of the gossip matrix $W(k)$ satisfy a quadratic inequality due to the spectrum property in Assumption 2.
- The input $w(0)$ and output $w(B)$ of the product $\prod_{\ell=0}^{B-1} W(\ell)$ satisfy a quadratic inequality due to the joint spectrum property in Assumption 2.

Note that we include the virtual signal $w(k)$ in order to make use of the joint spectrum property. Our analysis then consists of solving two LMIs, the *consensus LMI* which searches for P and the *disagreement LMI* which searches for Q.

Remark. We could use a modified version of the joint spectrum constraint from Assumption 2 in our analysis. For example, if the Laplacian $\mathcal{L}(k) = I - W(k)$ satisfies

$$\Big\| I - \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathsf{T}} - \prod_{\ell=k}^{k+B-1} \mathcal{L}(\ell) \Big\|_2^{1/B} \;\leq\; \sigma \quad \text{for all } k,$$

then we would instead define the virtual signal $w(k)$ by the recursion $w(k+1) = \mathcal{L}(k)\, w(k)$ with $w(0) = z(0)$, which would require modifying the quadratic inequalities involving $w(k)$ accordingly.

3.2 Main result

Given an algorithm of the form (2), parameters (m, L) from Assumption 1 and (σ, B) from Assumption 2, and a prospective convergence rate $\rho \in (0,1)$, we now construct the consensus and disagreement LMIs used to find the matrices P and Q in (5) and then state our main result.

Map from basis to iterates. To construct the LMIs, we first define a set of matrices that map the basis

$$\eta_i(k) = \mathsf{vcat}\big(x_i(k), u_i(k), \dots, u_i(k+B-1), v_i(k), \dots, v_i(k+B-1), w_i(k+2), \dots, w_i(k+B)\big) \quad (6)$$

to the corresponding iterates of the algorithm. The basis has size $b \times d$, where $b = s - c + B\,(2c+1)$ (recall that s is the number of states on each agent and c is the number of variables communicated per iteration). In particular, we define the sets of matrices

$$\begin{aligned}
\boldsymbol{u}(\ell) &\in \mathbb{R}^{1\times b} && \ell \in \{0,\dots,B-1\}\\
\boldsymbol{v}(\ell) &\in \mathbb{R}^{c\times b} && \ell \in \{0,\dots,B-1\}\\
\boldsymbol{w}(\ell) &\in \mathbb{R}^{c\times b} && \ell \in \{0,\dots,B\}\\
\boldsymbol{x}(\ell) &\in \mathbb{R}^{s\times b} && \ell \in \{0,\dots,B\}\\
\boldsymbol{y}(\ell) &\in \mathbb{R}^{1\times b} && \ell \in \{0,\dots,B-1\}\\
\boldsymbol{z}(\ell) &\in \mathbb{R}^{c\times b} && \ell \in \{0,\dots,B-1\}
\end{aligned}$$

such that the concatenated matrix

$$\mathsf{vcat}\big(\boldsymbol{x}(0), \boldsymbol{u}(0), \dots, \boldsymbol{u}(B-1), \boldsymbol{v}(0), \dots, \boldsymbol{v}(B-1), \boldsymbol{w}(2), \dots, \boldsymbol{w}(B)\big)$$

is the $b \times b$ identity matrix, $\boldsymbol{w}(0) = \boldsymbol{z}(0)$ and $\boldsymbol{w}(1) = \boldsymbol{v}(0)$, and the matrices satisfy the algorithm update

$$\begin{bmatrix}\boldsymbol{x}(\ell+1)\\ \boldsymbol{y}(\ell)\\ \boldsymbol{z}(\ell)\end{bmatrix} = G \begin{bmatrix}\boldsymbol{x}(\ell)\\ \boldsymbol{u}(\ell)\\ \boldsymbol{v}(\ell)\end{bmatrix}, \qquad \ell \in \{0,\dots,B-1\},$$

where G is defined in (4). These matrices are constructed such that multiplying each matrix on the right by the basis vector (6) yields the corresponding iterate.

Lyapunov function. Using these matrices, we define the matrices mapping the basis vector to the current and next state of the Lyapunov function as

$$\begin{aligned}
\boldsymbol{\xi} &= \mathsf{vcat}\big(\boldsymbol{x}(0), \boldsymbol{u}(0), \dots, \boldsymbol{u}(B-2), \boldsymbol{v}(0), \dots, \boldsymbol{v}(B-2)\big)\\
\boldsymbol{\xi}_+ &= \mathsf{vcat}\big(\boldsymbol{x}(1), \boldsymbol{u}(1), \dots, \boldsymbol{u}(B-1), \boldsymbol{v}(1), \dots, \boldsymbol{v}(B-1)\big)
\end{aligned}$$

with dimensions $a \times b$, where $a = s + (c+1)(B-1)$.

Consensus LMI. Let the matrix Ψ be a basis for

$$\begin{aligned}
&\mathsf{null}\left(\begin{bmatrix} I_B \otimes F_x & I_B \otimes F_u\end{bmatrix}\begin{bmatrix}\mathsf{vcat}\big(\boldsymbol{x}(0),\dots,\boldsymbol{x}(B-1)\big)\\ \mathsf{vcat}\big(\boldsymbol{u}(0),\dots,\boldsymbol{u}(B-1)\big)\end{bmatrix}\right)\\
&\quad \cap \mathsf{null}\Big(\mathsf{vcat}\big(\boldsymbol{v}(0)-\boldsymbol{z}(0),\dots,\boldsymbol{v}(B)-\boldsymbol{z}(B)\big)\Big)\\
&\quad \cap \mathsf{null}\Big(\mathsf{vcat}\big(\boldsymbol{w}(0)-\boldsymbol{w}(1),\dots,\boldsymbol{w}(B)-\boldsymbol{w}(B+1)\big)\Big).
\end{aligned}$$

The consensus LMI is then

$$\begin{aligned}
0 &\succeq X(P,\lambda) && && (7a)\\
0 &\prec P && && (7b)\\
0 &\le \lambda(\ell) && \ell \in \{0,\dots,B-1\} && (7c)
\end{aligned}$$

with variables $P \in \mathbb{S}^a$ and $\lambda(\ell) \in \mathbb{R}$, where the symmetric matrix X is given by

$$\begin{aligned}
X(P,\lambda) = \Psi^{\mathsf{T}}\Bigg(&\boldsymbol{\xi}_+^{\mathsf{T}} P\, \boldsymbol{\xi}_+ - \rho^2\,(\boldsymbol{\xi}^{\mathsf{T}} P\, \boldsymbol{\xi})\\
&+ \sum_{\ell=0}^{B-1} \lambda(\ell) \begin{bmatrix}\boldsymbol{y}(\ell)\\ \boldsymbol{u}(\ell)\end{bmatrix}^{\mathsf{T}} M_0 \begin{bmatrix}\boldsymbol{y}(\ell)\\ \boldsymbol{u}(\ell)\end{bmatrix}\Bigg)\Psi
\end{aligned}$$

with $M_0 = \left[\begin{smallmatrix} -2mL & L+m \\ L+m & -2\end{smallmatrix}\right]$.

Disagreement LMI. The disagreement LMI is

$$\begin{aligned}
0 &\succeq Y(Q,R,S,\lambda) && && (8a)\\
0 &\prec Q && && (8b)\\
0 &\preceq R && && (8c)\\
0 &\preceq S(\ell) && \ell \in \{0,\dots,B-1\} && (8d)\\
0 &\le \lambda(\ell) && \ell \in \{0,\dots,B-1\} && (8e)
\end{aligned}$$

with variables $Q \in \mathbb{S}^a$, $R \in \mathbb{S}^c$, $S(\ell) \in \mathbb{S}^{2c}$, and $\lambda(\ell) \in \mathbb{R}$, where the $b \times b$ symmetric matrix Y is given by

$$\begin{aligned}
Y(Q,R,S,\lambda) = &\boldsymbol{\xi}_+^{\mathsf{T}} Q\, \boldsymbol{\xi}_+ - \rho^2\,(\boldsymbol{\xi}^{\mathsf{T}} Q\, \boldsymbol{\xi})\\
&+ \sum_{\ell=0}^{B-1} \lambda(\ell) \begin{bmatrix}\boldsymbol{y}(\ell)\\ \boldsymbol{u}(\ell)\end{bmatrix}^{\mathsf{T}} M_0 \begin{bmatrix}\boldsymbol{y}(\ell)\\ \boldsymbol{u}(\ell)\end{bmatrix}\\
&+ \begin{bmatrix}\boldsymbol{w}(0)\\ \boldsymbol{w}(B)\end{bmatrix}^{\mathsf{T}} (M_1 \otimes R) \begin{bmatrix}\boldsymbol{w}(0)\\ \boldsymbol{w}(B)\end{bmatrix}\\
&+ \sum_{\ell=0}^{B-1} \begin{bmatrix}\boldsymbol{z}(\ell)\\ \boldsymbol{w}(\ell)\\ \boldsymbol{v}(\ell)\\ \boldsymbol{w}(\ell+1)\end{bmatrix}^{\mathsf{T}} \big(M_2 \otimes S(\ell)\big) \begin{bmatrix}\boldsymbol{z}(\ell)\\ \boldsymbol{w}(\ell)\\ \boldsymbol{v}(\ell)\\ \boldsymbol{w}(\ell+1)\end{bmatrix}
\end{aligned}$$

with $M_1 = \left[\begin{smallmatrix} \sigma^{2B} & 0 \\ 0 & -1\end{smallmatrix}\right]$ and $M_2 = \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1\end{smallmatrix}\right]$.

Remark. The consensus and disagreement LMIs are coupled through the variable $\lambda(\ell)$.

We now use the consensus and disagreement LMIs to state our main result, which characterizes the worst-case convergence rate of algorithm (2); we prove the result in the Appendix.

Theorem 1 (Main result)**.** Consider the optimization problem (1) solved using a distributed algorithm of the form (2) that satisfies the fixed-point conditions (3), and suppose that Assumptions 1–2 hold. If the consensus and disagreement LMIs in (7) and (8) are feasible for some scalar $\rho > 0$, then for any initial condition, there exists a constant $\gamma > 0$ such that

$$\|x_i(k) - x_i^\star\| \le \gamma\, \rho^k \quad (9)$$

for all agents $i \in \{1,\dots,n\}$ and all iterations $k \ge 0$, where $x_i^\star$ is a fixed point corresponding to the optimal solution of (1).

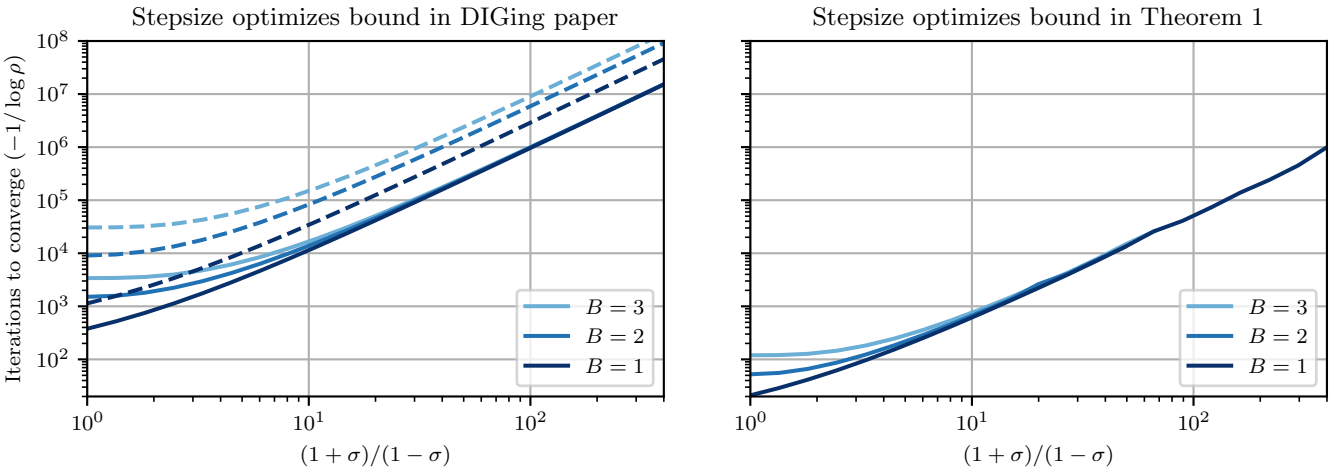

Figure 2: Upper bound on the number of iterations to converge for DIGing as a function of the spectral gap σ with condition ratio $\kappa = 10$ and connectivity parameter $B \in \{1, 2, 3\}$ using Theorem 1 (solid lines) and the bound from the original DIGing paper [5, Theorem 3.14] (dashed lines) with the stepsize α chosen to optimize the bound in the DIGing paper (left) and Theorem 1 (right). Note that the bound in the DIGing paper depends on the number of agents; we use $n = 2$ for the plot on the left, and the bound is vacuous for the stepsizes used in the plot on the right.

4 Case study: DIGing

To illustrate our results, we applied our analysis to the gradient tracking algorithm DIGing [5,6], which has been analyzed under the same assumptions[2].

The DIGing algorithm is given by the recursion

$$x(k+1) = W(k)\,x(k) - \alpha\,y(k)$$
$$y(k+1) = W(k)\,y(k) + \nabla f\big(x(k+1)\big) - \nabla f\big(x(k)\big)$$

with initial condition $y_i(0) = \nabla f_i(x_i(0))$ and stepsize α.

If we define the state as $\mathsf{vcat}(x_i(k), y_i(k), \nabla f_i(x_i(k)))$, then DIGing is equivalent to our algorithm form (2) with

$$\left[\begin{array}{c|c|c} A & B_u & B_v \\ \hline C_y & D_{yu} & D_{yv} \\ \hline C_z & D_{zu} & D_{zv} \\ \hline F_x & F_u & \end{array}\right] = \left[\begin{array}{c|cc|cc|cc} 0 & -\alpha & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & -\alpha & 0 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & -1 & 0 & & \end{array}\right].$$

From the dimensions of the matrices, we see that each agent has $s = 3$ state variables and communicates $c = 2$ variables to neighbors at each iteration.

We compare our convergence bound from Theorem 1 with that from the original DIGing paper [5, Thm. 3.14] in Figure 2. In the plot on the left, we use the stepsize

$$\alpha = \frac{1.5\left(\sqrt{J^2 - (1-\delta^2)\,J} - \delta\,J\right)^2}{m\,J\,(J+1)^2}$$

with $J = 3\kappa B^2(1 + 4\sqrt{n\kappa})$, which optimizes the worst-case linear rate $\rho = (1 - \alpha\,m/1.5)^{1/2B}$ from the DIGing paper. While our bound depends on the spectral gap σ and condition ratio κ, this bound also depends on the number of agents n and is vacuous in the limit as $n \to \infty$. In the plot on the right, we use the (much larger) stepsize which optimizes our bound from Theorem 1, for which the bound from the DIGing paper is vacuous.

To summarize, our analysis is tighter than previous bounds for DIGing (for both small and large stepsizes), is independent of the number of agents, and is applicable to any algorithm of the form (2).

[2] While the authors of [5] do not explicitly assume the spectrum property from Assumption 2, they make use of this property in [5, Equation (14)] to prove their convergence bound.

References

[1] J. A. Bazerque and G. B. Giannakis. Distributed spectrum sensing for cognitive radio networks by exploiting sparsity. *IEEE Transactions on Signal Processing*, 58(3):1847–1862, 2009.

[2] P. A. Forero, A. Cano, and G. B. Giannakis. Consensus-based distributed support vector machines. *Journal of Machine Learning Research*, 11:1663–1707, 2010.

[3] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[4] A. Nedić, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis. On distributed averaging algorithms and quantization effects. *IEEE Transactions on Automatic Control*, 54(11):2506–2517, 2009.

[5] A. Nedić, A. Olshevsky, and W. Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

[6] G. Qu and N. Li. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2018.

[7] M. Rabbat and R. Nowak. Distributed optimization in sensor networks. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pages 20–27. ACM, 2004.

[8] A. Sundararajan, B. Van Scoy, and L. Lessard. Analysis and design of first-order distributed optimization algorithms over time-varying graphs. *IEEE Transactions on Control of Network Systems*, 2020. (to appear).

[9] J. Xu, S. Zhu, Y. C. Soh, and L. Xie. Convergence of asynchronous distributed gradient methods over stochastic networks. *IEEE Transactions on Automatic Control*, 63(2):434–448, 2018.

Appendix

Proof of Proposition 1. Suppose that the algorithm matrices satisfy the conditions in (3), and let $y_i^\star = y_{\text{opt}} \in \mathbb{R}^{1\times d}$ for all i with $\sum_{i=1}^n \nabla f_i(y_{\text{opt}}) = 0$. Then there exist matrices $\bar{x} \in \mathbb{R}^{s\times d}$, $\hat{x} \in \mathbb{R}^s$, and $\bar{v} \in \mathbb{R}^{1\times d}$ such that

$$\begin{bmatrix} 0 \\ y_{\text{opt}} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} A-I & B_v \\ C_y & D_{yv} \\ C_z & D_{zv}-I \\ F_x & 0 \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{v} \end{bmatrix}, \quad \begin{bmatrix} B_u \\ D_{yu} \\ D_{zu} \end{bmatrix} = \begin{bmatrix} A-I \\ C_y \\ C_z \end{bmatrix} \hat{x}.$$

For all agents $i \in \{1,\dots,n\}$, use these to define

$$\begin{aligned} x_i^\star &= \bar{x} - \hat{x}\,\nabla f_i(y_{\text{opt}}), & y_i^\star &= y_{\text{opt}}, & z_i^\star &= \bar{v}, \\ u_i^\star &= \nabla f_i(y_{\text{opt}}), & v_i^\star &= \bar{v}, \end{aligned}$$

which is a fixed point of (2) corresponding to y_{opt}.

Now suppose $(x_i^\star, y_i^\star, z_i^\star, u_i^\star, v_i^\star)$ is a fixed point of (2) such that $y_i^\star = y_{\text{opt}}$ and $\sum_{i=1}^n \nabla f_i(y_{\text{opt}}) = 0$. Define the average state as $\bar{x} = (1/n)\sum_{i=1}^n x_i^\star$, and similarly for the other points. Then $\bar{u} = 0$, so the concatenated matrix $\mathsf{vcat}(\bar{x}, \bar{v})$ must be nonzero and in the space (3a). Now let q be any nonzero vector such that $q^\mathsf{T}\mathbf{1} = 0$. For the fixed point to not depend on the sequence of gossip matrices, $v_i = \bar{v}$ must be in consensus. Then multiplying (2a) by q_i and summing over $i \in \{1,\dots,n\}$,

$$0 = \begin{bmatrix} A-I \\ C_y \\ C_z \end{bmatrix} \sum_{i=1}^n (q_i\, x_i^\star) + \begin{bmatrix} B_u \\ D_{yu} \\ D_{zu} \end{bmatrix} \sum_{i=1}^n (q_i\, u_i^\star).$$

This must hold for all objective functions, which implies the condition in (3b). □

Proof of Theorem 1. Let $(x_i, y_i, z_i, u_i, v_i)$ denote a trajectory of algorithm (2). From Proposition 1, there exists a fixed point $(x_i^\star, y_i^\star, z_i^\star, u_i^\star, v_i^\star)$ with $y_i^\star = y_{\text{opt}}$ for all i, where $y_{\text{opt}} \in \mathbb{R}^{1\times d}$ is the optimizer of (1). We denote the error coordinates as $\tilde{x}_i(k) := x_i(k) - x^\star$, and similarly for the other signals.

From the invariant (2c) and the gossip matrix being weight-balanced, there exists $\tilde{s}(k)$ such that

$$\Psi\,\tilde{s}(k) = \frac{1}{\sqrt{n}} \sum_{i=1}^n \tilde{\eta}_i(k),$$

where $\eta_i(k)$ is the basis in (6) and $\tilde{\eta}_i(k)$ the corresponding error signal. Multiplying X in the consensus LMI on the right and left by $\tilde{s}(k)$ and its transpose, respectively, and defining $\Pi = \frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T}$, we obtain the consensus inequality

$$\begin{aligned} 0 \geq\ & \langle \tilde{\xi}(k+1), (\Pi\otimes P)\,\tilde{\xi}(k+1)\rangle \\ & - \rho^2\,\langle \tilde{\xi}(k), (\Pi\otimes P)\,\tilde{\xi}(k)\rangle \\ & + \sum_{\ell=0}^{B-1} \lambda(\ell) \left\langle \begin{bmatrix} \tilde{y}(k) \\ \tilde{u}(k) \end{bmatrix}, (M_0\otimes\Pi) \begin{bmatrix} \tilde{y}(k) \\ \tilde{u}(k) \end{bmatrix} \right\rangle, \end{aligned} \tag{10a}$$

where $\langle A, B\rangle = \mathrm{tr}(A^\mathsf{T}B)$ is the Frobenius inner product. Now choose vectors $q_2,\dots,q_n \in \mathbb{R}^n$ so that the matrix $\begin{bmatrix}\mathbf{1}/\sqrt{n} & q_2 & \dots & q_n\end{bmatrix}$ is orthonormal, and multiply the matrix Y in the disagreement LMI on the right and left by the weighted sum $\sum_{i=1}^n (q_m)_i\,\tilde{\xi}_i(k)$ and its transpose, respectively, and sum over $m \in \{2,\dots,n\}$. This results in the disagreement inequality

$$\begin{aligned} 0 \geq\ & \langle \tilde{\xi}(k+1), \big((I-\Pi)\otimes Q\big)\,\tilde{\xi}(k+1)\rangle \\ & - \rho^2\,\langle \tilde{\xi}(k), \big((I-\Pi)\otimes Q\big)\,\tilde{\xi}(k)\rangle \\ & + \sum_{\ell=0}^{B-1} \lambda(\ell) \left\langle \begin{bmatrix} \tilde{y}(k) \\ \tilde{u}(k) \end{bmatrix}, \big(M_0\otimes(I-\Pi)\big) \begin{bmatrix} \tilde{y}(k) \\ \tilde{u}(k) \end{bmatrix} \right\rangle \\ & + \left\langle \begin{bmatrix} \tilde{w}(k) \\ \tilde{w}(k+B) \end{bmatrix}, \big(M_1\otimes(I-\Pi)\otimes R\big) \begin{bmatrix} \tilde{w}(k) \\ \tilde{w}(k+B) \end{bmatrix} \right\rangle \\ & + \sum_{\ell=0}^{B-1} \left\langle \begin{bmatrix} \tilde{z}(\ell) \\ \tilde{w}(\ell) \\ \tilde{v}(\ell) \\ \tilde{w}(\ell+1) \end{bmatrix}, \big(M_2\otimes(I-\Pi)\otimes S(\ell)\big) \begin{bmatrix} \tilde{z}(\ell) \\ \tilde{w}(\ell) \\ \tilde{v}(\ell) \\ \tilde{w}(\ell+1) \end{bmatrix} \right\rangle \end{aligned} \tag{10b}$$

where we used that $\{q_m\}_{m=1}^n$ form an orthonormal basis for $\mathbb{R}^n$. Summing the inequalities in (10), we obtain

$$\begin{aligned} 0 \geq\ & V\big(\tilde{\xi}(k+1)\big) - \rho^2\,V\big(\tilde{\xi}(k)\big) \\ & + \sum_{\ell=0}^{B-1} \lambda(\ell) \left\langle \begin{bmatrix} \tilde{y}^k \\ \tilde{u}^k \end{bmatrix}, (M_0\otimes I) \begin{bmatrix} \tilde{y}^k \\ \tilde{u}^k \end{bmatrix} \right\rangle \\ & + \left\langle \begin{bmatrix} \tilde{w}(k) \\ \tilde{w}(k+B) \end{bmatrix}, \big(M_1\otimes(I-\Pi)\otimes R\big) \begin{bmatrix} \tilde{w}(k) \\ \tilde{w}(k+B) \end{bmatrix} \right\rangle \\ & + \sum_{\ell=0}^{B-1} \left\langle \begin{bmatrix} \tilde{z}(\ell) \\ \tilde{w}(\ell) \\ \tilde{v}(\ell) \\ \tilde{w}(\ell+1) \end{bmatrix}, \big(M_2\otimes(I-\Pi)\otimes S(\ell)\big) \begin{bmatrix} \tilde{z}(\ell) \\ \tilde{w}(\ell) \\ \tilde{v}(\ell) \\ \tilde{w}(\ell+1) \end{bmatrix} \right\rangle \end{aligned}$$

where the Lyapunov function V is defined in (5). Each quadratic form in the first summation is nonnegative from Assumption 1, the following term is nonnegative from the joint spectrum property in Assumption 2, and each term in the last summation is nonnegative from the spectrum property in Assumption 2; see [8, Prop. 15–16]. Then using the slight abuse of notation $V(k) = V(\tilde{\xi}(k))$, we have the decrease condition $V(k+1) \leq \rho^2\,V(k)$, which implies $V(k) \leq \rho^{2k}\,V(0)$. Now define the matrix $T := \Pi\otimes P + (I-\Pi)\otimes Q$, and note that $T \succ 0$ since P and Q are positive definite. Letting $\mathsf{cond}(T)$ denote the condition number of T, we have the bound

$$\|x_i(k) - x_i^\star\|^2 \leq \mathsf{cond}(T)\,V(k) \leq \rho^{2k}\,\mathsf{cond}(T)\,V(0),$$

so (9) holds with $\gamma = \sqrt{\mathsf{cond}(T)\,V(0)}$. □