ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Distributed Newton Optimization with ADMM-Based Consensus

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:
Carnevale, G., Bastianello, N., Notarstefano, G., Carli, R. (2025). Distributed Newton Optimization with ADMM-Based Consensus. Berlino : Springer [10.1007/978-3-031-83191-1_20].

Availability:
This version is available at: https://hdl.handle.net/11585/1025533 since: 2025-10-16

Published:

DOI: http://doi.org/10.1007/978-3-031-83191-1_20

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/).
When citing, please refer to the published version.

(Article begins on next page)

12 November 2026

Distributed Newton Optimization with ADMM-Based Consensus

Guido Carnevale[1], Nicola Bastianello[2,*], Giuseppe Notarstefano[1], and Ruggero Carli[3]

[1] Alma Mater Studiorum - Università di Bologna, Italy,
[2] KTH Royal Institute of Technology, Sweden,
nicolba@kth.se
[3] University of Padova, Italy

Abstract. In this paper we propose two novel second-order distributed algorithms for consensus optimization both employing ADMM as a consensus protocol. The idea is to benefit from the robustness properties of ADMM on the one hand, and the enhanced performance of Newton-based algorithms on the other. The designed algorithms use GIANT- (a recent second-order optimization method) and Jacobi-like descent directions. We employ tools from system theory, especially singular perturbations, to prove the linear convergence of our distributed schemes with strongly convex costs. We conclude with numerical simulations that confirm our theoretical results and compare the proposed algorithm with state-of-the-art alternatives.

Keywords: distributed optimization, Newton method, ADMM, singular perturbations

1 Introduction

In recent years, multi-agent systems have been deployed in an increasing number of engineering domains, *e.g.*, robotics, power grids, traffic networks, IoT [17]. This is made possible by interconnecting devices equipped with computational and communication capabilities, with the goal of cooperatively performing different tasks, *e.g.*, learning, resource allocation, and exploration. These tasks can be formulated as consensus optimization problems, and the objective thus is to design resilient and efficient distributed algorithms to solve them. Different classes of algorithms have been proposed, which can be categorized in first-order, second-order, and dual methods (ADMM); we reference [15, 17, 23] for recent comprehensive surveys. Second-order distributed algorithms have been proposed for example in [5, 12–14, 21], with different convergence guarantees. In particular, [13] proposes a fully distributed version of [20] employing a tracking mechanism and Newton-like updates.

First- and second-order methods both employ (dynamic) average consensus to enable cooperation (*e.g.*, to perform gradient tracking). However, the resulting algorithms have been shown to be fragile to some of the challenges that arise

in practice, such as asynchrony and packet losses [5, 19], and noisy communications [4]. It is therefore necessary to design more robust algorithms, and one approach is to replace the average consensus protocol with a more resilient one, see *e.g.* [5, 8, 19]. In this paper, we follow the approach of [7, 8] in *choosing distributed ADMM to implement dynamic consensus.*

Dual methods, and ADMM in particular, are a powerful class of algorithms with provable convergence guarantees in different scenarios, including asynchrony and packet losses [2]. The drawback of these algorithms is that the agents' local updates require the solution of a minimization problem which, in general, does not have closed form solution. However, dynamic consensus can be cast as an online quadratic problem, for which the ADMM updates simplify to a linear updates [1], making it a suitably robust and efficient consensus protocol.

In this paper thus we combine ADMM as a consensus protocol with distributed methods mimicking the second-order GIANT algorithm proposed in [20] and the well-known Jacobi method, benefiting from the best of both: the enhanced performance of Newton-based methods, and the robustness of ADMM. In the Jacobi version of the algorithm, the agents' Hessians are approximated via their diagonals only, significantly reducing the computational complexity while still improving performance with respect to first-order methods. Following a recent successful trend [9, 10], the analysis of the proposed distributed algorithms is conducted using system theory tools based on singular perturbations. In detail, we prove the algorithms' linear convergence in the case of strongly convex problems.

The paper is organized as follows. Section 2 describes the problem setup. Section 3 describes the proposed distributed algorithms and provides their convergence properties. Section 4 theoretically proves the stated convergence properties. Finally, in Section 5, we test the proposed schemes via numerical simulations.

Notation The symbol I_m denote the identity matrix in $\mathbb{R}^{m\times m}$. The symbol $\otimes$ denotes the Kronecker product. The vector of N ones and N zeroes are denoted by 1_N and 0_N, respectively while $\mathbf{1}_{N,n} := 1_N \otimes I_n$ $\mathbf{0}_{N,n} := 0_N \otimes I_n$. Given a finite set S, we use the symbol $|S|$ to denote its cardinality. Given N column vectors $v_1, \dots, v_N$, we denote with $\textsc{col}(v_i)_{i\in\{1,\dots,N\}}$ their vertical concatenation. Given $A \in \mathbb{R}^{n\times n}$, we denote as $\mathrm{diag}\{A\} \in \mathbb{R}^{n\times n}$ the matrix having the main diagonal coinciding with A and 0 elsewhere.

2 Problem Setup

We consider $N \in \mathbb{N}$ agents in a network whose goal is to cooperatively solve the optimization problem

$$\min_{x\in\mathbb{R}^n} \sum_{i=1}^{N} f_i(x), \tag{1}$$

where each $f_i : \mathbb{R}^n \to \mathbb{R}$ is the local objective function of agent $i \in \{1, \dots, N\}$ and all depend on the common decision variable $x \in \mathbb{R}^n$. We use $f(x) := \sum_{i=1}^{N} f_i(x)$

to denote the global cost function of problem (1). We are interested in developing an optimization method to iteratively address (1) according to the distributed paradigm. Namely, we want to obtain an algorithm that can be executed by peer-to-peer networks of units each using only local data (e.g., the local function f_i) and neighboring communication. In particular, we assume that the agents communicate through an undirected graph $\mathcal{G} = (\{1, \dots, N\}, \mathcal{E})$, with the set of edges $\mathcal{E} \subset \{1, \dots, N\} \times \{1, \dots, N\}$ such that $(i, j) \in \mathcal{E}$ if i and j can exchange information. We use the symbol $\mathcal{N}_i := \{j \in \{1, \dots, N\} \mid (i, j) \in \mathcal{E}\}$ to indicate the neighbors of agent i, the symbol $d_i := |\mathcal{N}_i|$ to indicate the so-called degree of agent i, and define $\mathbf{d} := \sum_{i=1}^{N} d_i = 2|\mathcal{E}|$.

We enforce the following assumption about the class of graphs considered in this work.

Assumption 1. *[Graph] The graph $\mathcal{G}$ is undirected and connected.* □

Assumption 2. *[Cost functions] For all $i \in \{1, \dots, N\}$, the objective function f_i is $\mathcal{C}^3$, c-strongly convex, and its gradient ∇f_i is L-Lipschitz continuous.* □

We notice that Assumption 2 ensures that f is Nc-strongly convex and, thus, problem (1) admits a unique solution $x^\star \in \mathbb{R}^n$.

3 GIANT-ADMM and Jacobi-ADMM

To iteratively address problem (1), given the iteration index $t \in \mathbb{N}$, we maintain a solution estimate $x_i^t \in \mathbb{R}^n$ in each agent i and update it according to

$$x_i^{t+1} = x_i^t + u_i^t, \tag{2}$$

where $u_i^t \in \mathbb{R}^n$ is the corresponding update direction. In particular, we need to design each u_i^t to (i) force consensus among all the agents' estimates, and (ii) converge to the solution of problem (1). We pursue the first goal by including the consensus action $\frac{1}{N}\sum_{j=1}^{N} x_j^t - x_i^t$ into u_i^t. As for the second purpose, with an eye to second order methods, we leverage on the term $\frac{\delta}{N}\mathcal{H}_i(x_i^t)^{-1}\sum_{j=1}^{N}\nabla f_j(x_j^t)$, in which $\delta > 0$ is the stepsize and we allow for two possible choices of $\mathcal{H}_i : \mathbb{R}^n \to \mathbb{R}^{n\times n}$, namely

$$\begin{cases} \mathcal{H}_i(x_i) &= \nabla^2 f_i(x_i)^{-1} & \text{GIANT-like update} \\ \mathcal{H}_i(x_i) &= \operatorname{diag}\{\nabla^2 f_i(x_i)\}^{-1} & \text{Jacobi-like update.} \end{cases} \tag{3}$$

Therefore, the whole update direction u_i^t would read as

$$u_i^t = \left(\frac{1}{N}\sum_{j=1}^{N} x_j^t - x_i^t\right) - \frac{\delta}{N}\mathcal{H}_i(x_i^t)^{-1}\sum_{j=1}^{N}\nabla f_j(x_j^t). \tag{4}$$

However, in a distributed setting, the global terms $\frac{1}{N}\sum_{j=1}^{N} x_j^t$ and $\frac{1}{N}\sum_{j=1}^{N}\nabla f_j(x_j^t)$ are not available for agent i. Independently on the choice about $\mathcal{H}_i$, we thus

modify (4) by introducing in each agent two auxiliary variables $y_i^t, s_i^t \in \mathbb{R}^n$ aimed at reconstructing $\frac{1}{N}\sum_{j=1}^N x_j^t$ and $\frac{1}{N}\sum_{j=1}^N \nabla f_j(x_j^t)$, respectively, thus obtaining

$$u_i^t = (y_i^t - x_i^t) - \delta \mathcal{H}_i(x_i^t)^{-1} s_i^t. \tag{5}$$

It is worth noting that when $y_i^t = \frac{1}{N}\sum_{j=1}^N x_j^t$ and $s_i^t = \frac{1}{N}\sum_{j=1}^N \nabla f_j(x_j^t)$ the desired update (4) and the implementable one (5) coincide. For this reason, we follow the approach in [1,7,8] by pursuing the consensus goals of y_i^t and s_i^t by considering, for each iteration $t \geq 0$, the online optimization problem

$$\min_{\substack{(y_1,\dots,y_N)\in\mathbb{R}^{Nn}\\(s_1,\dots,s_N)\in\mathbb{R}^{Nn}}} \sum_{i=1}^N \ell_i^t(y_i, s_i) \qquad \text{s.t.: } \begin{bmatrix} y_i \\ s_i \end{bmatrix} = \begin{bmatrix} y_j \\ s_j \end{bmatrix} \ \forall (i,j) \in \mathcal{E}, \tag{6}$$

where, for all $i \in \{1,\dots,N\}$, the cost function $\ell_i^t : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is defined as

$$\ell_i^t(y_i, s_i) = \frac{1}{2}\left\|y_i - x_i^t\right\|^2 + \frac{1}{2}\left\|s_i - \nabla f_i(x_i^t)\right\|^2.$$

Indeed, if the graph $\mathcal{G}$ is connected, then [1] guarantees that problem (6) admits a unique solution $(y_\star^t, s_\star^t) \in \mathbb{R}^{2Nn}$ defined as

$$\begin{bmatrix} y_\star^t \\ s_\star^t \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{N,n}\frac{1}{N}\sum_{j=1}^N x_j^t \\ \mathbf{1}_{N,n}\frac{1}{N}\sum_{j=1}^N \nabla f_j(x_j^t) \end{bmatrix}.$$

Thus, by focusing on problem (6), we choose the distributed ADMM proposed in [2] to update y_i^t and s_i^t, namely, in each agent i, we introduce an additional variable $z_{ij}^t \in \mathbb{R}^{2n}$ for each neighbor $j \in \mathcal{N}_i$ and implement

$$\begin{bmatrix} y_i^t \\ s_i^t \end{bmatrix} = \underset{\substack{y_i\in\mathbb{R}^n\\ s_i\in\mathbb{R}^n}}{\arg\min} \left\{ \ell_i^t(y_i, s_i) - \begin{bmatrix} y_i^\top & s_i^\top \end{bmatrix} \sum_{j\in\mathcal{N}_i} z_{ij}^t + \frac{\rho d_i}{2}\left(\|y_i\|^2 + \|s_i\|^2\right) \right\} \tag{7a}$$

$$z_{ij}^{t+1} = (1-\alpha) z_{ij}^t + \alpha\left(-z_{ji}^t + 2\rho \begin{bmatrix} y_j^t \\ s_j^t \end{bmatrix}\right), \tag{7b}$$

where $\rho > 0$ and $\alpha \in (0,1)$. By observing that each function ℓ_i^t is quadratic, we note that problem (7a) admits a closed-form solution given by

$$\begin{bmatrix} y_i^t \\ s_i^t \end{bmatrix} = \frac{1}{1+\rho d_i}\left(\begin{bmatrix} x_i^t \\ \nabla f_i(x_i^t) \end{bmatrix} + \sum_{j\in\mathcal{N}_i} z_{ij}^t\right). \tag{8}$$

Further, in order to pursue a timescale separation approach, we add a tuning gain $\gamma > 0$ to (2) to arbitrarily control the rate of variation of x_i^t and, thus, allow for a proper interconnection with (8), namely

$$x_i^{t+1} = x_i^t + \gamma u_i^t. \tag{9}$$

By combining (5), (7), (8), and (9) we obtain two different distributed algorithms based on the choice made for $\mathcal{H}_i$ (see (3)). In particular, by choosing $\mathcal{H}_i = \nabla^2 f_i(x_i^t)$, we get Algorithm 1 and we name it GIANT-ADMM.

Algorithm 1 GIANT-ADMM (Agent i)

Initialization: $x_i^0 \in \mathbb{R}^n, z_{ij}^0 \in \mathbb{R}^{2n}$ for each $j \in \mathcal{N}_i$.
for $t = 0, 1, \ldots$ **do**
  $\begin{bmatrix} y_i^t \\ s_i^t \end{bmatrix} = \frac{1}{1+\rho d_i} \left(\begin{bmatrix} x_i^t \\ \nabla f_i(x_i^t) \end{bmatrix} + \sum_{j \in \mathcal{N}_i} z_{ij}^t \right)$
  $x_i^{t+1} = x_i^t + \gamma(y_i^t - x_i^t) - \gamma\delta \nabla^2 f_i(x_i^t)^{-1} s_i^t$
  for $j \in \mathcal{N}_i$ **do**
    $z_{ij}^{t+1} = (1-\alpha) z_{ij}^t + \alpha \left(-z_{ji}^t + 2\rho \begin{bmatrix} y_j^t \\ s_j^t \end{bmatrix} \right)$
  end for
end for

On the contrary, in Algorithm 2, we report the distributed algorithm obtained with $\mathcal{H}_i = \text{diag}\{\nabla^2 f_i(x_i^t)\}$ and we name it Jacobi-ADMM.

Algorithm 2 Jacobi-ADMM (Agent i)

Initialization: $x_i^0 \in \mathbb{R}^n, z_{ij}^0 \in \mathbb{R}^{2n}$ for each $j \in \mathcal{N}_i$.
for $t = 0, 1, \ldots$ **do**
  $\begin{bmatrix} y_i^t \\ s_i^t \end{bmatrix} = \frac{1}{1+\rho d_i} \left(\begin{bmatrix} x_i^t \\ \nabla f_i(x_i^t) \end{bmatrix} + \sum_{j \in \mathcal{N}_i} z_{ij}^t \right)$
  $x_i^{t+1} = x_i^t + \gamma(y_i^t - x_i^t) - \gamma\delta \text{diag}\{\nabla^2 f_i(x_i^t)\}^{-1} s_i^t$
  for $j \in \mathcal{N}_i$ **do**
    $z_{ij}^{t+1} = (1-\alpha) z_{ij}^t + \alpha \left(-z_{ji}^t + 2\rho \begin{bmatrix} y_j^t \\ s_j^t \end{bmatrix} \right)$
  end for
end for

Both Algorithm 1 and 2 can be implemented by satisfying the desired distributed paradigm since when running them each agent i only uses local quantities and performs neighboring communication. More in detail, in order to execute Algorithm 1 and 2, each agent i must maintain the local solution estimate $x_i^t \in \mathbb{R}^n$ and the auxiliary variable $z_i^t := \text{COL}(z_{ij}^t)_{j \in \mathcal{N}_i} \in \mathbb{R}^{2nd_i}$ stacking all the variables z_{ij}^t. Hence, each agent i needs to store vectors with overall size $(1 + 2d_i)n$. As for the communication burden, we note that each agent i needs to transmit the vector $-z_{ij}^t + 2\rho \left[(y_i^t)^\top \ (s_i^t)^\top \right]^\top \in \mathbb{R}^{2n}$ to each neighbor $j \in \mathcal{N}_i$. We also remark that it is possible to exactly recover the ADMM-Tracking Gradient distributed algorithm developed in [8] by choosing $\mathcal{H}_i(x_i) = I_n$ for all $i \in \{1, \ldots, N\}$. Next, we formally

claim the convergence features of Algorithm 1 and Algorithm 2. Indeed, as will become clearer in the analysis, the similarities between the considered schemes and our system-theoretic approach allow for a unified examination.

Theorem 1. *Consider Algorithm 1 and 2 and let Assumptions 1 and 2 hold. Then, for all* $(x_i^0, z_i^0) \in \mathbb{R}^n \times \mathbb{R}^{2nd_i}$ *for all* $i \in \{1, \dots, N\}$*, there exist* $\bar{\gamma}, \bar{\delta}, c_1, c_2 > 0$ *such that, for any* $\rho > 0$*,* $\alpha \in (0,1)$*,* $\gamma \in (0, \bar{\gamma})$*,* $\delta \in (0, \bar{\delta})$*, it holds*

$$\left\|x_i^t - x^\star\right\| \leq c_1 \exp(-c_2 t). \qquad \square$$

Theorem 1 is proved in Section 4.4. In detail, with tools from system theory, we will show semi-global exponential stability of $(\mathbf{1}_{N,n} x^\star, z_\perp^\star)$ for a suitable and restricted reformulation of the aggregate forms of GIANT-ADMM and Jacobi-ADMM, where the quantity $z_\perp^\star$ will be specified later.

4 Convergence Analysis of GIANT-ADMM and Jacobi-ADMM

In this section, we analyze Algorithm 1 and 2 by resembling the system-theoretic approach performed in [8] First, in Section 4.1, we provide the aggregate form of GIANT-ADMM and Jacobi-ADMM and show that it is possible to interpret it as a *singularly perturbed* system. By taking advantage of this form, in Section 4.2 and Section 4.3 we then separately analyze the stability properties of two auxiliary systems associated to the original one. These results are finally exploited in Section 4.4 to prove Theorem 1 by leveraging on singular perturbations arguments.

4.1 GIANT-ADMM and Jacobi-ADMM as Singularly Perturbed Systems

We start by building a dynamical system describing the aggregate form of GIANT-ADMM and Jacobi-ADMM (according to the choice made for $\mathcal{H}_i(x_i^t)$). To this end, we introduce the permutation matrix $P \in \mathbb{R}^{2n\mathbf{d} \times 2n\mathbf{d}}$ that swaps the ij-th element with the ji-th one, the matrices $A_x \in \mathbb{R}^{2n\mathbf{d} \times Nn}$, $A_\nabla \in \mathbb{R}^{n\mathbf{d} \times Nn}$, $A \in \mathbb{R}^{2n\mathbf{d} \times Nn}$, $M \in \mathbb{R}^{Nn \times Nn}$, $\mathcal{M} \in \mathbb{R}^{2Nn \times 2Nn}$, and $\mathcal{H} : \mathbb{R}^{Nn} \times \mathbb{R}^{Nn \times Nn}$ defined as

$$A_x := \begin{bmatrix} \mathbf{1}_{d_1,n} & & \\ \mathbf{0}_{d_1,n} & \ddots & \\ & \ddots & \mathbf{1}_{d_N,n} \\ & & \mathbf{0}_{d_N,n} \end{bmatrix}, \; A_\nabla := \begin{bmatrix} \mathbf{0}_{d_1,n} & & \\ \mathbf{1}_{d_1,n} & \ddots & \\ & \ddots & \mathbf{0}_{d_N,n} \\ & & \mathbf{1}_{d_N,n} \end{bmatrix},$$

$$A := \begin{bmatrix} \mathbf{1}_{d_1,2n} & & \\ & \ddots & \\ & & \mathbf{1}_{d_N,2n} \end{bmatrix}, \mathcal{H}(x) := \begin{bmatrix} \mathcal{H}_1(x_1) & & \\ & \ddots & \\ & & \mathcal{H}_N(x_N) \end{bmatrix}$$

$$M := \begin{bmatrix} \frac{1}{1+\rho d_1} I_n & & \\ & \ddots & \\ & & \frac{1}{1+\rho d_N} I_n \end{bmatrix}, \mathcal{M} := \begin{bmatrix} \frac{1}{1+\rho d_1} I_{2n} & & \\ & \ddots & \\ & & \frac{1}{1+\rho d_N} I_{2n} \end{bmatrix}$$

Then, by defining the stacking vectors $x^t := \text{COL}(x_1^t, \dots, x_N^t) \in \mathbb{R}^{Nn}$ and $z^t := \text{COL}(z_1^t, \dots, z_N^t) \in \mathbb{R}^{2n\mathbf{d}}$, the aggregate form of Algorithm 1 and 2 reads as

$$x^{t+1} = x^t + \gamma \left(M \left(x^t + A_x^\top z^t \right) - x^t \right) - \gamma \delta \mathcal{H}(x^t)^{-1} M \left(G(x^t) + A_\nabla^\top z^t \right) \quad (10a)$$

$$z^{t+1} = z^t - \alpha (I + P - 2\rho P A \mathcal{M} A^\top) z^t + 2\alpha\rho P A \mathcal{M} v(x^t), \quad (10b)$$

where, given $x := \text{COL}(x_1, \dots, x_N) \in \mathbb{R}^{Nn}$ with $x_i \in \mathbb{R}^n$ for all $i \in \{1, \dots, N\}$, the operators $G : \mathbb{R}^{Nn} \to \mathbb{R}^{Nn}$ and $v : \mathbb{R}^{Nn} \to \mathbb{R}^{2Nn}$ are defined as

$$G(x) := \left[\nabla f_1(x_1)^\top \dots \nabla f_N(x_N)^\top \right]^\top \quad (11)$$

$$v(x) := \left[x_1^\top \nabla f_1(x_1)^\top \dots x_N^\top \nabla f_N(x_N)^\top \right]^\top. \quad (12)$$

We know that, by construction, the matrix $(I + P - 2\rho P A \mathcal{M} A^\top)$ has some eigenvalues in 0 [2]. Moreover, these eigenvalues are semi-simple [2]. By relying on this property, we now follow the same approach as in [8] to obtain a convenient restricted system description. In particular, we employ a decomposition that expresses z^t according to a basis of the subspace corresponding to the null space of $(I + P - 2\rho P A \mathcal{M} A^\top)$. Specifically, let $b \in \mathbb{N}$ denote the dimension of the subspace $\mathcal{S}$ generated by the eigenvectors of $(I + P - 2\rho P A \mathcal{M} A^\top)$ associated with the eigenvalue 0, $B \in \mathbb{R}^{2n\mathbf{d} \times b}$ be the matrix having columns corresponding to an orthonormal basis of $\mathcal{S}$, and $B_\perp \in \mathbb{R}^{2n\mathbf{d} \times n_d}$ the matrix such that $B_\perp^\top B_\perp = I$ and $B^\top B_\perp = 0$, with $n_d := 2n\mathbf{d} - b$. The matrices B and $B_\perp$ have some useful properties formalized as follows.

Lemma 1. *[[8, Lemma III.1]] The matrices B and $B_\perp$ satisfy*

$$A_x^\top B = 0, \quad A_\nabla^\top B = 0 \quad (13a)$$

$$B^\top P A = 0 \quad (13b)$$

$$B^\top (I + P - 2\rho P A \mathcal{M} A^\top) = 0. \quad (13c)$$

Following the same approach in [8], we now introduce the change of variables

$$\bar{z} := B^\top z, \quad z_\perp := B_\perp^\top z. \quad (14)$$

Lemma 1 and the novel coordinates (14) allow for reformulating system (10) as

$$x^{t+1} = x^t + \gamma \left(M \left(x^t + A_x^\top B_\perp z_\perp^t \right) - x^t \right) - \gamma \delta \mathcal{H}(x^t)^{-1} M \left(G(x^t) + A_\nabla^\top B_\perp z_\perp^t \right) \quad (15a)$$

$$\bar{z}^{t+1} = \bar{z}^t \quad (15b)$$

$$z_\perp^{t+1} = z_\perp^t - \alpha B_\perp^\top (I + P - 2\rho P A \mathcal{M} A^\top) B_\perp z_\perp^t + 2\alpha\rho B_\perp^\top P A \mathcal{M} v(x^t). \quad (15c)$$

Notice that this is the same structure derived in [8], with the difference that now $\mathcal{H}(x^t)$ contains second order information (see (6)). In particular, the state $\bar{z}^t$ does not influence the other states of (15). Hence, we study system (15) without considering the variable $\bar{z}^t$ and using the novel coordinate $\tilde{x}^t := x^t - \mathbf{1}_{N,n} x^\star$. Therefore, we focus on

$$\tilde{x}^{t+1} = \tilde{x}^t + \gamma \left(M \left(x^t + A_x^\top B_\perp z_\perp^t \right) - x^t \right) - \gamma \delta \mathcal{H}(x^t)^{-1} M \left(G(x^t) + A_\nabla^\top B_\perp z_\perp^t \right) \quad (16a)$$

$$z_\perp^{t+1} = z_\perp^t - \alpha F z_\perp^t + 2\alpha\rho B_\perp^\top P A \mathcal{M} v(x^t), \quad (16b)$$

in which we also use

$$F := B_\perp^\top (I + P - 2\rho PA\mathcal{M}A^\top) B_\perp. \tag{17}$$

Notably, we remark that $(I - \alpha F)$ is Schur for all $\alpha \in (0, 1)$ [8, Lemma III.2].

System (16) matches the form of the so-called singularly perturbed systems since it exhibits the following two key properties. First, the variations of the subsystem (16a) can be arbitrarily reduced via the tuning parameter γ and, accordingly, plays the role of the so-called slow system. Second, the subsystem (16b) plays the role of the so-called fast system since its equilibria are parametrized in the slow state $\tilde{x}^t$ through the function $z_\perp^{\mathrm{eq}} : \mathbb{R}^{Nn} \to \mathbb{R}^{n_d}$ defined as

$$z_\perp^{\mathrm{eq}}(\tilde{x}^t) := 2\rho F^{-1} B_\perp^\top PA\mathcal{M}v(\tilde{x}^t + \mathbf{1}_{N,n} x^\star). \tag{18}$$

We conclude the section by reporting the following useful properties about z^{eq}. Given any $\tilde{x} \in \mathbb{R}^{Nn}$ and the corresponding $x = \tilde{x} + \mathbf{1}_{N,n} x^\star$, it holds

$$MA_x^\top B_\perp z_\perp^{\mathrm{eq}}(\tilde{x}) = \frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N} x - Mx \tag{19a}$$

$$MA_\nabla^\top B_\perp z_\perp^{\mathrm{eq}}(\tilde{x}) = \frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N} G(x) - MG(x). \tag{19b}$$

4.2 Boundary Layer System

In this section, we study the so-called boundary layer system associated to (16), i.e., the auxiliary system obtained by plugging an arbitrarily fixed state $\tilde{x}^t \equiv \tilde{x} \in \mathbb{R}^{Nn}$ into the fast dynamics (16b) and using the error coordinate $\tilde{z}_\perp^t := z_\perp^t - z_\perp^{\mathrm{eq}}(\tilde{x})$. By using the definition of $z_\perp^{\mathrm{eq}}$ (cf. (18)), such a boundary layer system reads as

$$\tilde{z}_\perp^{t+1} = \tilde{z}_\perp^t - \alpha F \tilde{z}_\perp^t. \tag{20}$$

Not surprisingly, since the consensus-oriented part of GIANT-ADMM and Jacobi-ADMM is the same as the one characterizing the distributed method provided in [8], the boundary layer system (20) associated to (16) is exactly the same considered in [8]. Then, we already know the stability properties of (20).

Lemma 2. *[[8, Lemma III.2]] Consider* (20)*. Then, for all* $\alpha \in (0, 1)$*, there exist* $U : \mathbb{R}^{n_d} \to \mathbb{R}$ *and* $b_1, b_2, b_3, b_4 > 0$ *such that*

$$b_1 \left\|\tilde{z}_\perp\right\|^2 \le U(\tilde{z}_\perp) \le b_2 \left\|\tilde{z}_\perp\right\|^2 \tag{21a}$$

$$U(\tilde{z}_\perp^t - \alpha F \tilde{z}_\perp^t) - U(\tilde{z}_\perp) \le -b_3 \left\|\tilde{z}_\perp\right\|^2 \tag{21b}$$

$$|U(\tilde{z}_\perp) - U(\tilde{z}'_\perp)| \le b_4 \left\|\tilde{z}_\perp - \tilde{z}'_\perp\right\| \left\|\tilde{z}_\perp\right\| + b_4 \left\|\tilde{z}_\perp - \tilde{z}'_\perp\right\| \left\|\tilde{z}'_\perp\right\|, \tag{21c}$$

for all $\tilde{z}_\perp, \tilde{z}'_\perp \in \mathbb{R}^{n_d}$*.* □

4.3 Reduced System

Here, we focus on the auxiliary system associated to (16) named reduced system. The latter is obtained from (16a) by plugging $z_\perp^t = z_\perp^{\mathrm{eq}}(\tilde{x}^t)$ for all $t \in \mathbb{N}$. Then, by also using (19), the reduced system of (16) reads as

$$\tilde{x}^{t+1} = \tilde{x}^t - \gamma\left(I - \frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}\right)\tilde{x}^t - \gamma\delta\mathcal{H}(x^t)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(\tilde{x}^t + \mathbf{1}_{N,n}x^\star). \quad (22)$$

The upcoming lemma establishes a Lyapunov function for system (22).

Lemma 3. *Consider* (22)*. Then, there exist* $W : \mathbb{R}^{Nn} \to \mathbb{R}$, $\bar{\gamma}_1, \bar{\delta}, c_1, c_2, c_3, c_4 > 0$ *such that, for all* $\gamma \in (0, \bar{\gamma}_1)$ *and* $\delta \in (0, \bar{\delta})$*, it holds*

$$c_1 \|\tilde{x}\|^2 \leq W(\tilde{x}) \leq c_2 \|\tilde{x}\|^2 \quad (23a)$$

$$W(\tilde{x}^{t+1}) - W(\tilde{x}^t) \leq -c_3\gamma \|\tilde{x}\|^2 \quad (23b)$$

$$|W(\tilde{x}) - W(\tilde{x}')| \leq c_4 \|\tilde{x} - \tilde{x}'\| \|\tilde{x}\| + c_4 \|\tilde{x} - \tilde{x}'\| \|\tilde{x}'\|, \quad (23c)$$

for all $\tilde{x}, \tilde{x}', \tilde{x}^t \in \mathbb{R}^{Nn}$*, and* $\tilde{x}^{t+1}$ *computed according to* (22) *starting from* x^t*.*

Proof. Following the same derivation as in [8], but applying it to the novel Algorithms 1 and 2, we take advantage of suitable novel coordinates. To this end, let $R \in \mathbb{R}^{Nn\times(N-1)n}$ be a matrix such that $R^\top \mathbf{1}_{N,n} = 0$ and $R^\top R = I_{(N-1)n}$. Then, let $\bar{x} \in \mathbb{R}^n$ and $x_\perp \in \mathbb{R}^{(N-1)n}$ be defined as

$$\bar{x} := \frac{\mathbf{1}_{N,n}^\top}{N}\tilde{x}, \quad x_\perp := R^\top \tilde{x}. \quad (24)$$

By employing the change of variables (24), $\mathbf{1}_{N,n} \in \ker(I_{Nn} - \frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N})$, and $R^\top \frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N} = 0$, system (22) turns out to be equivalent to

$$\bar{x}^{t+1} = \bar{x}^t - \gamma\delta\frac{\mathbf{1}_{N,n}^\top}{N}\mathcal{H}(x^t)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(x^t) \quad (25a)$$

$$x_\perp^{t+1} = (1-\gamma)x_\perp^t - \gamma\delta R^\top\mathcal{H}(x^t)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(x^t), \quad (25b)$$

in which we used $x^t = \mathbf{1}_{N,n}\bar{x}^t + Rx_\perp^t$ in the argument of $\mathcal{H}$ and G. Let $\bar{G} : \mathbb{R}^n \times \mathbb{R}^{(N-1)n} \to \mathbb{R}^n$ and $G_\perp : \mathbb{R}^n \times \mathbb{R}^{(N-1)n} \to \mathbb{R}^n$ be defined as

$$\begin{aligned}\bar{G}(\bar{x}, x_\perp) := &-\frac{\mathbf{1}_{N,n}^\top}{N}\mathcal{H}(x)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(\mathbf{1}_{N,n}\bar{x}+Rx_\perp+\mathbf{1}_{N,n}x^\star)\\ &+\frac{\mathbf{1}_{N,n}^\top}{N}\mathcal{H}(x)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(\mathbf{1}_{N,n}\bar{x}+\mathbf{1}_{N,n}x^\star)\\ G_\perp(\bar{x}, x) := &-R^\top\mathcal{H}(x)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(\mathbf{1}_{N,n}\bar{x}+Rx_\perp+\mathbf{1}_{N,n}x^\star),\end{aligned}$$

in which, we used $x = \mathbf{1}_{N,n}\bar{x} + Rx_\perp$ in the argument of $\mathcal{H}$. Let us inspect the term $\frac{\mathbf{1}_{N,n}^\top}{N}\mathcal{H}(x)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(\mathbf{1}_{N,n}\bar{x} + \mathbf{1}_{N,n}x^\star)$. In particular, it holds

$$\frac{\mathbf{1}_{N,n}^\top}{N}\mathcal{H}(x)^{-1}\frac{\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top}{N}G(\mathbf{1}_{N,n}\bar{x} + \mathbf{1}_{N,n}x^\star) = \frac{1}{N^2}\left(\sum_{i=1}^N \mathcal{H}_i(x_i)^{-1}\right)\nabla f(\bar{x} + x^\star),$$

in which we lighten the notation by denoting with $x_i \in \mathbb{R}^n$ the i-th block of the vector $x = \mathbf{1}_{N,n}\bar{x} + \mathbf{1}_{N,n}x^\star + Rx_\perp$. By adding $\pm\frac{\gamma\delta}{N^2}\left(\sum_{i=1}^N \mathcal{H}_i(x_i^t)^{-1}\right)\nabla f(\bar{x}^t + x^\star)$ into (25a), we compactly rewrite system (25) as

$$\bar{x}^{t+1} = \bar{x}^t - \frac{\gamma\delta}{N^2}\left(\sum_{i=1}^N \mathcal{H}_i(x_i^t)^{-1}\right)\nabla f(\bar{x}^t + x^\star) + \gamma\delta\bar{G}\left(\bar{x}^t, x_\perp^t\right) \tag{27a}$$

$$x_\perp^{t+1} = (1-\gamma)x_\perp^t + \delta\gamma G_\perp(\bar{x}^t, x_\perp^t). \tag{27b}$$

Now, we prove that the origin is a globally exponentially stable equilibrium point of (27). To this end, we consider the Lyapunov function $\tilde{W} : \mathbb{R}^n \times \mathbb{R}^{(N-1)n} \to \mathbb{R}$ defined as

$$\tilde{W}(\tilde{x}) := \underbrace{f(\bar{x} + x^\star) - f(x^\star)}_{:=\tilde{W}_{\bar{x}}(\bar{x})} + \underbrace{\frac{1}{2}\left\|x_\perp\right\|^2}_{:=\tilde{W}_{x_\perp}(x_\perp)}. \tag{28}$$

We recall that each ∇f_i is L-Lipschitz continuous (cf. Assumption 2) and, thus, by using the Descent Lemma [3, Prop. 6.1.2], for all $\bar{x} \in \mathbb{R}^n$, we get

$$f(\bar{x} + x^\star) - f(x^\star) \leq \nabla f(x^\star)^\top\bar{x} + \frac{NL}{2}\left\|\bar{x}\right\|^2 \overset{(a)}{\leq} \frac{NL}{2}\left\|\bar{x} - x^\star\right\|^2, \tag{29}$$

where in (a) we use the fact that $\nabla f(x^\star) = 0$ since $x^\star$ is the solution of problem (1). Moreover, since f is Nc-strongly convex (cf. Assumption 2), it holds

$$f(\bar{x} + x^\star) - f(x^\star) \geq \frac{Nc}{2}\left\|\bar{x}\right\|^2, \tag{30}$$

for all $\bar{x} \in \mathbb{R}^n$. Hence, by using the bounds (29) and (30) and the fact that $\tilde{W}_{x_\perp}$ is quadratic, it holds

$$\min\{Nc, 1\}/2\left(\left\|\bar{x}\right\|^2 + \left\|x_\perp\right\|^2\right) \leq \tilde{W}(\bar{x}, x_\perp) \leq \max\{NL, 1\}/2\left(\left\|\bar{x}\right\|^2 + \left\|x_\perp\right\|^2\right), \tag{31}$$

for all $\bar{x} \in \mathbb{R}^n$ and $x_\perp \in \mathbb{R}^{(N-1)n}$. Let us now evaluate $\Delta\tilde{W}_{\bar{x}}(\bar{x}^t) := \tilde{W}_{\bar{x}}(\bar{x}^{t+1}) - \tilde{W}_{\bar{x}}(\bar{x}^t)$ along (27a) using the Descent Lemma [3, Prop. 6.1.2] and thus obtaining

$$\begin{aligned}
\Delta\tilde{W}_{\bar{x}}(\bar{x}^t) = & -\frac{\gamma\delta}{N^2}\nabla f(\bar{x}^t + x^\star)^\top \left(\sum_{i=1}^N \mathcal{H}_i(\bar{x} + x^\star)^{-1}\right) \nabla f(\bar{x}^t + x^\star) \\
& + \gamma\delta \nabla f(\bar{x}^t + x^\star)^\top \bar{G}(\bar{x}^t, x_\perp^t) \\
& + \frac{\gamma^2\delta^2 NL}{2} \left\| \frac{1}{N^2}\nabla f(\bar{x}^t + x^\star) + \bar{G}(\bar{x}^t, x_\perp^t) \right\|^2 \\
\overset{(a)}{\leq} & -\frac{\gamma\delta}{N^2}\nabla f(\bar{x}^t + x^\star)^\top \left(\sum_{i=1}^N \mathcal{H}_i(\bar{x} + x^\star)^{-1}\right) \nabla f(\bar{x}^t + x^\star) \\
& + \gamma\delta \left\|\nabla f(\bar{x}^t + x^\star)\right\| \left\|\bar{G}(\bar{x}^t, x_\perp^t)\right\| \\
& + \gamma^2\delta^2 NL \left\| \frac{1}{N^2}\nabla f(\bar{x}^t + x^\star) \right\|^2 + \gamma^2\delta^2 NL \left\|\bar{G}(\bar{x}^t, x_\perp^t)\right\|^2, \qquad (32)
\end{aligned}$$

where in (a) we bound the second term using the Cauchy-Schwarz inequality and the Young's inequality for the last one. Being each cost f_i c-strongly convex with L-Lipschitz continuous gradients (cf. Assumption 2), both choices in (3) satisfy

$$\frac{1}{L_\mathcal{H}} I_n \leq \sum_{i=1}^N \mathcal{H}_i(\bar{x})^{-1} \leq \frac{1}{c_\mathcal{H}} I_n, \qquad (33)$$

for all $\bar{x} \in \mathbb{R}^n$, with $L_\mathcal{H} = L/N$ and $c_\mathcal{H} = c/N$ if $\mathcal{H}_i(x_i) = \nabla^2 f_i(x_i)$ and $L_\mathcal{H} = NL$ and $c_\mathcal{H} = Nc$ if $\mathcal{H}_i(x_i) = \mathrm{diag}\{\nabla^2 f_i(x_i)\}$. The bounds in (33) lead to

$$-\nabla f(\bar{x})^\top \left(\sum_{i=1}^N \mathcal{H}_i(\bar{x})^{-1}\right) \nabla f(\bar{x}) \leq -\frac{1}{L_\mathcal{H}} \left\|\nabla f(\bar{x})\right\|^2 \overset{(a)}{\leq} -\frac{N^2 c^2}{L_\mathcal{H}} \left\|\bar{x}\right\|^2, \qquad (34)$$

where (a) uses the fact that f is Nc-strongly convex (cf. Assumption 2). Moreover, we recall that $\nabla f(x^\star) = 0$ and that the gradients ∇f_i are L-Lipschitz continuous (cf. Assumption 2). Hence, we have

$$\left\|\nabla f(\bar{x} + x^\star)\right\| = \left\|\nabla f(\bar{x} + x^\star) - \nabla f(x^\star)\right\| \overset{(a)}{\leq} LN \left\|\bar{x}\right\|, \qquad (35)$$

for all $\bar{x} \in \mathbb{R}^n$. Further, the bound (33), the Lipschitz continuity of each ∇f_i (cf. Assumption 2), $\|\mathbf{1}_{N,n}^\top / N\| = \sqrt{N}$, $\left\|\frac{1}{N}\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top\right\| = 1$, and $\|R\| = 1$, lead to

$$\left\|\bar{G}(\bar{x}^t, x_\perp^t)\right\| \leq \frac{L\sqrt{N}}{c_\mathcal{H}} \left\|x_\perp^t\right\|. \qquad (36)$$

By combining (34), (35), and (36), we bound (32) as

$$\begin{aligned}
\Delta\tilde{W}_{\bar{x}}(\bar{x}^t) \leq & -\gamma\delta\frac{N^2 c^2}{L_\mathcal{H}} \left\|\bar{x}^t\right\|^2 + \gamma^2\delta^2 \frac{N\sqrt{N}L^2}{c_\mathcal{H}} \left\|\bar{x}^t\right\| \left\|x_\perp^t\right\| \\
& + \gamma^2\delta^2 \frac{L^3}{N} \left\|\bar{x}^t\right\|^2 + \gamma^2\delta^2 \frac{N\sqrt{N}L^2}{c_\mathcal{H}} \left\|x_\perp^t\right\|^2. \qquad (37)
\end{aligned}$$

Let us now consider $\tilde{W}_{x_\perp}$. Along the trajectories (27b), we bound $\Delta\tilde{W}_{x_\perp}(x_\perp^t) := \tilde{W}_{x_\perp}(x_\perp^{t+1}) - \tilde{W}_{x_\perp}(x_\perp^t)$ as

$$\begin{aligned}\Delta\tilde{W}_{x_\perp}(x_\perp^t) \leq -\gamma\left(1-\frac{\gamma}{2}\right)\left\|x_\perp^t\right\|^2 + \gamma\delta(1-\gamma)\left\|x_\perp^t\right\|\left\|G_\perp(\bar{x}^t, x_\perp^t)\right\| \\ + \gamma^2\delta^2\frac{1}{2}\left\|G_\perp(\bar{x}^t, x_\perp^t)\right\|^2. \end{aligned} \tag{38}$$

Since $\mathcal{H}_i(\bar{x})^{-1} \leq \frac{1}{c}$ in light of Assumption 2), each ∇f_i is L-Lipschitz continuous (cf. Assumption 2), $\left\|\mathbf{1}_{N,n}\mathbf{1}_{N,n}^\top/N\right\| = 1$, and $\|R\| = 1$, it holds

$$\left\|G_\perp(\bar{x}^t, x_\perp^t)\right\| \leq \frac{NL}{c_\mathcal{H}}\left\|x_\perp^t\right\|,$$

which allows us to further bound (38) as

$$\Delta\tilde{W}_{x_\perp}(x_\perp^t) \leq -\gamma\left(1-\frac{\gamma}{2}\right)\left\|x_\perp^t\right\|^2 + \gamma\delta(1-\gamma)\frac{NL}{c_\mathcal{H}}\left\|x_\perp^t\right\|^2 + \gamma^2\delta^2\frac{N^2L^2}{2c_\mathcal{H}^2}\left\|x_\perp^t\right\|^2. \tag{39}$$

By defining $\Delta\tilde{W}(\tilde{x}^t) := \Delta\tilde{W}_{\bar{x}}(\bar{x}^t) + \Delta\tilde{W}_{x_\perp}(x_\perp^t)$ and using (37) and (39), we get

$$\Delta\tilde{W}(\tilde{x}^t) \leq -\gamma\begin{bmatrix}\|\bar{x}^t\| \\ \|x_\perp^t\|\end{bmatrix}^\top Q\begin{bmatrix}\|\bar{x}^t\| \\ \|x_\perp^t\|\end{bmatrix}, \tag{40}$$

where we introduced the matrix $Q(\gamma,\delta) = Q(\gamma,\delta)^\top \in \mathbb{R}^{2\times 2}$ given by

$$Q(\gamma,\delta) := \begin{bmatrix}\delta\frac{N^2c^2}{L_\mathcal{H}} - \gamma\delta^2\frac{L^3}{N} & \gamma\delta^2\frac{N\sqrt{N}L^2}{2c_\mathcal{H}} \\ \gamma\delta^2\frac{N\sqrt{N}L^2}{2c_\mathcal{H}} & (1-\gamma/2) - p(\gamma,\delta)\end{bmatrix},$$

where

$$p(\gamma,\delta) = \gamma\delta^2\frac{N\sqrt{N}L^2}{c_\mathcal{H}} - \delta(1-\gamma)\frac{NL}{c_\mathcal{H}}\left\|x_\perp^t\right\|^2 - \gamma\delta^2\frac{N^2L^2}{2c_\mathcal{H}^2}\left\|x_\perp^t\right\|^2.$$

Let us arbitrarily choose $q \in \min\{\frac{Nc^2}{L^2}, 1\}$. By Sylvester Criterion, it holds

$$Q(\gamma,\delta) > \begin{bmatrix}\delta q & 0 \\ 0 & q\end{bmatrix} \iff \begin{cases}\delta\frac{N^2c^2}{L_\mathcal{H}} - \gamma\delta^2\frac{L^3}{N} > \delta q \\ \delta\left(\delta\frac{N^2c^2}{L_\mathcal{H}} - \gamma\delta^2\frac{L^3}{N} - q\right)\left((1-\gamma/2) - p(\gamma,\delta) - q\right) > \gamma^2\delta^4\frac{N^3L^4}{4c_\mathcal{H}^2}.\end{cases} \tag{41}$$

Hence, let $\bar{\gamma}_1 := \min\left\{2(1-q), \left(\frac{N^2c^2}{L_\mathcal{H}} - q\right)\frac{N}{L^3}\right\}$ and fix $\gamma \in (0, \bar{\gamma}_1)$. Thus, it holds

$$\begin{cases}\delta\frac{N^2c^2}{L_\mathcal{H}} - \gamma\delta^2\frac{L^3}{N} > \delta q \\ (1-\gamma/2) > q,\end{cases}$$

for all $\delta \in (0,1)$ and, thus, the second condition in (41) reduces to

$$q^2 > \gamma^2\delta^3\frac{N^3L^4}{4c_{\mathcal{H}}^2} + qp(\gamma,\delta)$$

Moreover, we note that $\lim_{\delta\to 0}\left(\gamma^2\delta^3N^3L^4/4c_{\mathcal{H}}^2 + qp(\gamma,\delta)\right) = 0$, for all $\gamma \in \mathbb{R}$ and, thus, since γ has been fixed, there exists $\bar{\delta}_1 > 0$ such that $\gamma^2\delta^3\frac{N^3L^4}{4c_{\mathcal{H}}^2} + qp(\gamma,\delta) < q^2$ for all $\delta \in (0,\bar{\delta}_1)$. Therefore, by defining $\bar{\delta} := \min\{1,\bar{\delta}_1\}$ and choosing $\gamma \in (0,\bar{\gamma}_1)$ and $\delta \in (0,\bar{\delta})$, we can bound (40) as

$$\Delta\tilde{W}(\bar{x}^t, x_\perp^t) \leq -\gamma q(\delta\left\|\bar{x}^t\right\|^2 + \left\|x_\perp^t\right\|^2).$$

By combining the last inequality and the bounds (31), we ensure that the origin is a globally exponentially stable equilibrium of (22) [11, Th. 13.2]. With this result at hand, the existence of the Lyapunov function W satisfying (23) follows by invoking the Converse Lyapunov Theorem (see, e.g., [6, Th. 5.8]). □

4.4 Proof of Theorem 1

Since the origin is a globally exponentially stable equilibrium of the boundary-layer system (20) (cf. Lemma 2) and the origin is a globally exponentially stable equilibrium of the reduced system (22) (cf. Lemma 3), we now prove that the origin is an exponentially stable equilibrium of the original interconnected system (16) via singular perturbations by slightly extending [8, Th. II.3].

More in detail, the mentioned result requires the two stability properties provided by Lemma 2 and Lemma 3, the Lipschitz continuity of the equilibrium function z^{eq} (cf. (18)), and the Lipschitz continuity of the dynamics characterizing (16). The Lipschitz continuity of the gradients of the objective functions (cf. Assumption 2) allows for claiming the Lipschitz continuity of z^{eq} but, as customary for second-order schemes, we cannot globally guarantee the desired property for the dynamics characterizing (16) due to the presence of $\mathcal{H}_i(x_i^t)$. However, for each algorithm initial condition, by defining $V(\tilde{x},\tilde{z}_\perp) := U(\tilde{z}_\perp) + W(\tilde{x})$ (with U and W provided by Lemma 2 and 3, respectively) and the level set $\Omega_0 := \{(\tilde{x},\tilde{z}_\perp) \mid V(\tilde{x},\tilde{z}_\perp) \leq V(\tilde{x}^0,\tilde{z}_\perp^0)\}$, we can use the continuity of each $\mathcal{H}_i$ (cf. Assumption 2) and the compactness of Ω_0 (see (21a) and (23a)), to ensure the desired Lipschitz continuity within Ω_0. Once this property is achieved, we can easily adapt the arguments of [8, Th. II.3] to guarantee the existence of $\bar{\gamma} \in (0,\bar{\gamma}_1)$ such that (i) Ω_0 is forward invariant for system (16) for all $\gamma \in (0,\bar{\gamma})$ and (ii) the origin is a semi-global exponentially stable equilibrium of system (16). The proof follows by using the exponential stability definition and reverting to the coordinates (x,z).

5 Numerical Simulations

This section reports numerical simulations comparing the proposed GIANT-ADMM, Jacobi-ADMM, with ADMM-Tracking Gradient [7, 8], and the well-known Gradient Tracking algorithm [16, 18, 22]. We focus on a classification task

over a network of $N = 20$ agents. Each agent i locally stores $m_i \in \mathbb{N}$ data points $p_{i,1}, \dots, p_{i,m_i} \in \mathbb{R}^{n-1}$ and each of them is labeled with $l_{i,k} \in \{-1, 1\}$ for all $k \in \{1, \dots, m_i\}$. The goal is to train a linear classification model by solving the problem

$$\min_{w \in \mathbb{R}^{n-1}, b \in \mathbb{R}} \sum_{i=1}^{N} \sum_{k=1}^{m_i} \log\left(1 + e^{-l_{i,k}(w^\top p_{i,k} + b)}\right) + \frac{C}{2} \left\| \begin{bmatrix} w \\ b \end{bmatrix} \right\|^2,$$

where $C > 0$ is the regularization weight, which ensures the strong convexity of the cost. We choose $n = 2$ and employ randomly generated data. The agents communicate according to a ring graph. We set $\alpha = 0.9$, $\rho = 0.9$, and $\gamma = 0.3$, and $\delta = 1$ for GIANT-ADMM, $\alpha = 0.9$, $\rho = 0.9$, and $\gamma = 0.3$, and $\delta = 1$ for Jacobi-ADMM, $\alpha = 0.9$, $\rho = 0.9$, $\gamma = 0.5$, and $\delta = 1$ for ADMM-Tracking Gradient, and $\gamma = 0.01$ for Gradient Tracking. Fig. 1 compares the algorithms' performance by providing the evolution of $\|x^t - \mathbf{1}_{N,n} x^\star\|$ over the iterations t. In particular, Fig. 1 confirms the linear convergence claimed by Theorem 1 and shows that both GIANT-ADMM and Jacobi-ADMM outperform the other considered distributed schemes thus proving the advantages of using second-order updates.

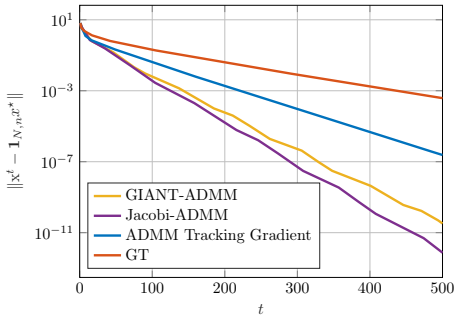

Fig. 1: Logistic regression scenario: comparison between GIANT-ADMM, Jacobi-ADMM, ADMM-Tracking Gradient, and Gradient Tracking (GT).

6 Conclusions

In this paper, we proposed GIANT-ADMM and Jacobi-ADMM, namely, two novel distributed methods for consensus optimization. The two algorithms, respectively, combine GIANT-like and Jacobi-like methods with ADMM-consensus to reconstruct the unavailable global quantities. The designed algorithms have been analyzed using tools of system theory based on singular perturbations. This approach guaranteed the linear convergence of the algorithms in the case of strongly convex costs. Numerical simulations confirmed the algortihms' effectiveness.

References

1. Bastianello, N., Carli, R.: Admm for dynamic average consensus over imperfect networks. IFAC-PapersOnLine **55**(13), 228–233 (2022)
2. Bastianello, N., Carli, R., Schenato, L., Todescato, M.: Asynchronous distributed optimization over lossy networks via relaxed admm: Stability and linear convergence. IEEE Transactions on Automatic Control **66**(6), 2620–2635 (2020)
3. Bertsekas, D.P., Scientific, A.: Convex optimization algorithms. Athena Scientific Belmont (2015)
4. Bin, M., Notarnicola, I., Parisini, T.: Stability, linear convergence, and robustness of the wang-elia algorithm for distributed consensus optimization. In: 2022 IEEE 61st Conference on Decision and Control (CDC), pp. 1610–1615. IEEE (2022)
5. Bof, N., Carli, R., Notarstefano, G., Schenato, L., Varagnolo, D.: Multiagent Newton–Raphson Optimization Over Lossy Networks. IEEE Transactions on Automatic Control **64**(7), 2983–2990 (2019)
6. Bof, N., Carli, R., Schenato, L.: Lyapunov theory for discrete time systems. arXiv preprint arXiv:1809.05289 (2018)
7. Carnevale, G., Bastianello, N., Carli, R., Notarstefano, G.: Distributed consensus optimization via admm-tracking gradient. In: 2023 62nd IEEE Conference on Decision and Control (CDC), pp. 290–295. IEEE (2023)
8. Carnevale, G., Bastianello, N., Notarstefano, G., Carli, R.: Admm-tracking gradient for distributed optimization over asynchronous and unreliable networks. arXiv preprint arXiv:2309.14142 (2023)
9. Carnevale, G., Mimmo, N., Notarstefano, G.: A unifying system theory framework for distributed optimization and games. arXiv preprint arXiv:2401.12623 (2024)
10. Carnevale, G., Notarstefano, G.: Nonconvex distributed optimization via lasalle and singular perturbations. IEEE Control Systems Letters **7**, 301–306 (2022)
11. Chellaboina, V., Haddad, W.M.: Nonlinear dynamical systems and control: A Lyapunov-based approach. Princeton University Press (2008)
12. Mansoori, F., Wei, E.: A Fast Distributed Asynchronous Newton-Based Optimization Algorithm. IEEE Transactions on Automatic Control **65**(7), 2769–2784 (2019). DOI 10.1109/TAC.2019.2933607
13. Maritan, A., Sharma, G., Schenato, L., Dey, S.: Network-GIANT: Fully distributed Newton-type optimization via harmonic Hessian consensus (2023). DOI 10.48550/arXiv.2305.07898. ArXiv:2305.07898 [cs, math]
14. Mokhtari, A., Shi, W., Ling, Q., Ribeiro, A.: A Decentralized Second-Order Method with Exact Linear Convergence Rate for Consensus Optimization. IEEE Transactions on Signal and Information Processing over Networks **2**(4), 507–522 (2016). DOI 10.1109/TSIPN.2016.2613678
15. Nedić, A., Liu, J.: Distributed optimization for control. Annual Review of Control, Robotics, and Autonomous Systems **1**, 77–103 (2018)
16. Nedić, A., Olshevsky, A., Shi, W.: Achieving geometric convergence for distributed optimization over time-varying graphs. SIAM Journal on Optimization **27**(4), 2597–2633 (2017)
17. Notarstefano, G., Notarnicola, I., Camisa, A.: Distributed optimization for smart cyber-physical networks. Foundations and Trends® in Systems and Control **7**(3), 253–383 (2019)
18. Shi, W., Ling, Q., Wu, G., Yin, W.: EXTRA: An exact first-order algorithm for decentralized consensus optimization. SIAM Journal on Optimization **25**(2), 944–966 (2015)

19. Tian, Y., Sun, Y., Scutari, G.: Achieving Linear Convergence in Distributed Asynchronous Multiagent Optimization. IEEE Transactions on Automatic Control **65**(12), 5264–5279 (2020)
20. Wang, S., Roosta, F., Xu, P., Mahoney, M.W.: Giant: Globally improved approximate newton method for distributed optimization. In: Advances in Neural Information Processing Systems, vol. 31, pp. 2332–2342 (2018)
21. Wei, H., Qu, Z., Wu, X., Wang, H., Lu, J.: Decentralized Approximate Newton Methods for Convex Optimization on Networked Systems. IEEE Transactions on Control of Network Systems **8**(3), 1489–1500 (2021). DOI 10.1109/TCNS.2021.3070663
22. Xin, R., Khan, U.A.: A linear algorithm for optimization over directed graphs with geometric convergence. IEEE Control Systems Letters **2**(3), 315–320 (2018)
23. Yang, T., Yi, X., Wu, J., Yuan, Y., Wu, D., Meng, Z., Hong, Y., Wang, H., Lin, Z., Johansson, K.H.: A survey of distributed optimization. Annual Reviews in Control **47**, 278–305 (2019)