Newton-Raphson Consensus
for Distributed Convex Optimization

Damiano Varagnolo, Filippo Zanella, Angelo Cenedese,
Gianluigi Pillonetto, Luca Schenato

Abstract

We address the problem of distributed unconstrained convex optimization under separability assumptions, i.e., the framework where a network of agents, each endowed with a local private multidimensional convex cost and subject to communication constraints, wants to collaborate to compute the minimizer of the sum of the local costs. We propose a design methodology that combines average consensus algorithms and separation of time-scales ideas. This strategy is proved, under suitable hypotheses, to be globally convergent to the true minimizer. Intuitively, the procedure lets the agents distributedly compute and sequentially update an approximated Newton-Raphson direction by means of suitable average consensus ratios. We show with numerical simulations that the speed of convergence of this strategy is comparable with alternative optimization strategies such as the Alternating Direction Method of Multipliers. Finally, we propose some alternative strategies which trade-off communication and computational requirements with convergence speed.

Index Terms

Distributed optimization, unconstrained convex optimization, consensus, multi-agent systems, Newton-Raphson methods, smooth functions.

I. INTRODUCTION

Optimization is a pervasive concept underlying many aspects of modern life [3], [4], [5], and it also includes the management of distributed systems, i.e., artifacts composed by a multitude of interacting entities often referred to as “agents”. Examples are transportation systems, where the agents are both the vehicles and the traffic management devices (traffic lights), and smart electrical grids, where the agents are the energy producers-consumers and the power transformers-transporters.

Here we consider the problem of distributed optimization, i.e., the class of algorithms suitable for networked systems and characterized by the absence of a centralized coordination unit [6], [7], [8]. Distributed optimization tools have received an increasing attention over the last years, concurrently with the research on networked control systems. Motivations comprise the fact that the former methods let the networks self-organize and adapt to surrounding and changing environments, and that they are necessary to manage extremely complex systems in an autonomous way with only limited human intervention. In particular

D. Varagnolo is with the Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, Luleå Sweden. Email: damiano.varagnolo@ltu.se. F. Zanella, A. Cenedese, G. Pillonetto and L. Schenato are with the Department of Information Engineering, Università di Padova, Padova, Italy. Emails: {fzanella | angelo.cenedese | giapi | schenato}@dei.unipd.it.

The research leading to these results has received funding from the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n°257462 HYCON2 Network of excellence, from the Knuth and Alice Wallenberg Foundation and the Swedish Research Council. This paper is an extended and revised version of [1], [2].

DRAFT
we focus on unconstrained convex optimization, although there is a rich literature also on distributed constrained optimization such as Linear Programming [9].

**Literature review**

The literature on distributed unconstrained convex optimization is extremely vast and a first taxonomy can be based whether the strategy uses or not the Lagrangian framework, see, e.g., [5, Chap. 5].

Among the distributed methods exploiting Lagrangian formalism, the most widely known algorithm is Alternating Direction Method of Multipliers (ADMM) [10], whose roots can be traced back to [11]. Its efficacy in several practical scenarios is undoubted, see, e.g., [12] and references therein. A notable size of the dedicated literature focuses on the analysis of its convergence performance and on the tuning of its parameters for optimal convergence speed, see, e.g., [13] for Least Squares (LS) estimation scenarios or [14] for linearly constrained convex programs. Even if proved to be an effective algorithm, ADMM suffers from requiring synchronous communication protocols, although some recent attempts for asynchronous and distributed implementations have appeared [15], [16], [17].

On the other hand, among the distributed methods not exploiting Lagrangian formalisms, the most popular ones are the Distributed Subgradient Methods (DSMs) [18]. Here the optimization of non-smooth cost functions is performed by means of subgradient based descent/ascent directions. These methods arise in both primal and dual formulations, since sometimes it is better to perform dual optimization. Subgradient methods have been exploited for several practical purposes, e.g., to optimally allocate resources in Wireless Sensor Networks (WSNs) [19], to maximize the convergence speeds of gossip algorithms [20], to manage optimality criteria defined in terms of ergodic limits [21]. Several works focus on the analysis of the convergence properties of the DSM basic algorithm [22], [23], [24] (see [25] for a unified view of many convergence results). We can also find analyses for several extensions of the original idea, e.g., directions that are computed combining information from other agents [26], [27] and stochastic errors in the evaluation of the subgradients [28]. Explicit characterizations can also show trade-offs between desired accuracy and number of iterations [29].

These methods have the advantage of being easily distributed, to have limited computational requirements and to be inherently asynchronous as shown in [30], [31], [32]. However they suffer from low convergence rate since they require the update steps to decrease to zero as $1/t$ (being $t$ the time) therefore as a consequence the rate of convergence is sub-exponential. In fact, one of the current trends is to design strategies that improve the convergence rate of DSMs. For example, a way is to accelerate the convergence of subgradient methods by means of multi-step approaches, exploiting the history of the past iterations to compute the future ones [33]. Another is to use Newton-like methods, when additional smoothness assumptions can be used. These techniques are based on estimating the Newton direction starting from the Laplacian of the communication graph. More specifically, distributed Newton techniques have been proposed in dual ascent scenarios [34], [35], [36]. Since the Laplacian cannot be computed exactly, the convergence rates of these schemes rely on the analysis of inexact Newton methods [37]. These Newton methods are shown to have super-linear convergence under specific assumptions, but can be applied only to specific optimization problems such as network flow problems.

Recently, several alternative approaches to ADMM and DSM have appeared. For example, in [38], [39] the authors construct contraction mappings by means of cyclic projections of the estimate of the optimum onto the constraints. A similar idea based on contraction maps is used in F-Lipschitz methods [40] but it requires additional assumptions on the cost functions. Other methods are the control-based approach [41] which exploits distributed consensus, and the distributed randomized Kaczmarz method [42] for quadratic cost functions.
Here we propose a distributed Newton-Raphson optimization procedure, named Newton-Raphson Consensus (NRC), for the exact minimization of smooth multidimensional convex separable problems, where the global function is a sum of private local costs. With respect to the categorization proposed before, the strategy exploits neither Lagrangian formalisms nor Laplacian estimation steps. More specifically, it is based on average consensus techniques [43] and on the principle of separation of time-scales [44, Chap. 11]. The main idea is that agents compute and keep updated, by means of average consensus protocols, an approximated Newton-Raphson direction that is built from suitable Taylor expansions of the local costs. Simultaneously, agents move their local guesses towards the Newton-Raphson direction. It is proved that, if the costs satisfy some smoothness assumptions and the rate of change of the local update steps is sufficiently slow to allow the consensus algorithm to converge, then the NRC algorithm exponentially converges to the global minimizer.

The proposed algorithm has several advantages w.r.t. the aforementioned literature. The first advantage is that it is as easy to implement as the DSM but, if opportunely tuned, it generally shows faster convergence rates. Second, numerical simulations based on relevant applications in machine learning and real data show that our Newton-Raphson consensus approach can have comparable or better convergence rates of standard ADMM algorithms. Lastly, we see advantages in its potential extensions: since it is essentially based on average consensus algorithms, we foresee natural extensions that may manage asynchronous communications, time-varying network topologies [45], directed graphs [46], and packet-losses.

Structure of the paper

The paper is organized as follows: Section II collects the notation used through the whole paper, while Section III formulates the considered problem and provides some ancillary results that are then used to study the convergence properties of the main algorithm. Section IV proposes the main optimization algorithm, provides convergence results and describes some strategies to trade-off communication and computational complexities with convergence speed. Section V compares, via numerical simulations, the performance of the proposed algorithm with several distributed optimization strategies available in the literature. Finally, Section VI collects some final observations and suggests future research directions. We collect all the proofs in the Appendix.

II. Notation

We model the communication network as a graph $G = (\mathcal{N}, \mathcal{E})$ whose vertices $\mathcal{N} := \{1, 2, \ldots, N\}$ represent the agents and whose edges $(i, j) \in \mathcal{E}$ represent the available communication links. We assume that the graph is undirected and connected, and that the matrix $P \in \mathbb{R}^{N \times N}$ is stochastic, i.e., its elements are non-negative, it is s.t. $P 1 = 1$ (where $1 := [1 1 \cdots 1]^T \in \mathbb{R}^N$), symmetric, i.e., $P = P^T$ and consistent with the graph $G$, in the sense that each entry $p_{ij}$ of $P$ is $p_{ij} > 0$ only if $(i, j) \in \mathcal{E}$. We recall that if $P$ is stochastic and symmetric and includes all edges, i.e., $p_{ij} > 0$ if and only if $(i, j) \in \mathcal{E}$, then $\lim_{k \to \infty} P^k = \frac{1}{N} 1 1^T$. Such $P$’s are also often referred to as average consensus matrices. We will indicate with $\rho(P) := \max_{i,j \neq 1} |\lambda_i(P)|$ the spectral radius of $P$, with $\sigma(P) := 1 - \rho(P)$ its spectral gap.

We use fraction bars to indicate also Hadamard divisions, e.g., if $\mathbf{a} = [a_1, a_2, \ldots, a_N]^T$ and $\mathbf{b} = [b_1, b_2, \ldots, b_N]^T$ then $\mathbf{a} / \mathbf{b} = \begin{bmatrix} a_1/b_1 & \cdots & a_N/b_N \end{bmatrix}^T$. Also, fraction bars may indicate pre-multiplication with inverse matrices, i.e., if $A$ is a matrix then $x/A$ indicates $A^{-1}x$. We indicate with $n$ the dimensionality of the domains of the cost functions, $k$ a discrete time index, $t$ a continuous time index. For notational simplicity we denote differentiation with $\nabla$ operators, so that $\nabla f = \partial f/\partial x$ (for simplicity considered as a column vector) and $\nabla^2 f = \partial^2 f/\partial x^2$. With a little abuse of notation, we will define $\chi = (x, Z)$, where $x \in \mathbb{R}^n$ and $Z \in \mathbb{R}^{T \times q}$.
as the vector obtained by stacking in a column both the vector $x$ and the vectorized matrix $Z$ so that $\|\chi\|^2 = \|x\|^2 + \|Z\|_F^2$, where the subscript $F$ indicates the Frobenius norm.

We will use plain italic font, possibly with a subscript (usually $i$, e.g., $x_i \in \mathbb{R}^n$), then it refers to the local variable of the specific agent $i$, while we indicate with a bold italic font, e.g., $x$, then it refers to the collection of the decision variables of the various agents, e.g., $x := [x_1^T, \ldots, x_N^T]^T \in \mathbb{R}^{nN}$. We will also consider the following notation to indicate special variables:

\[
\bar{x} := \frac{1}{N} \sum_{i=1}^{N} x_i \quad \mathbb{R}^n
\]
\[
x_i^i := \mathbb{1}_N \otimes x_i \quad \mathbb{R}^{nN}
\]
\[
x_i^\perp := x_i - x_i^i \quad \mathbb{R}^{nN}
\]

As in [44, p. 116], we say that a function $V$ is a Lyapunov function for a specific dynamics if $V$ is continuously differentiable and satisfies $V(0) = 0$, $V(x) > 0$ for $x \neq 0$, and $\dot{V}(x) \leq 0$.

### III. Problem formulation and preliminary results

#### A. Structure of the section

Our main contribution is to characterize the convergence properties of the distributed Newton-Raphson (NR) scheme proposed in Section IV. In doing so we both exploit standard singular perturbation analysis tools [44, Chap. 11] [48] and a set of ancillary results, collected for readability in this section.

The logical flow of these ancillary results is the following: Section III-C claims that, under suitable assumptions, forward-Euler discretizations of stable continuous dynamics lead to stable discrete dynamics. This basic result enables reasoning on continuous-time systems. Then, Sections III-D and III-E respectively claim that single- and multi-agent continuous-time NR dynamics satisfy these discretization assumptions. Sections III-F and III-G then generalize these dynamics by introducing perturbation terms that mimic the behavior of the proposed main optimization algorithm, and characterize their stability properties. Summarizing, the ancillary results characterize the stability properties of systems that are progressive approximations of the dynamics under investigation.

#### B. Problem formulation

We assume that the $N$ agents of the network are endowed with cost functions $f_i : \mathbb{R}^n \mapsto \mathbb{R}$ so that

\[
\overline{f} : \mathbb{R}^n \mapsto \mathbb{R}, \quad \overline{f}(x) := \frac{1}{N} \sum_{i=1}^{N} f_i(x)
\]

is a well-defined global cost. We assume that the aim of the agents is to cooperate and distributedly compute the minimizer of $\overline{f}$, namely

\[
x^* := \arg \min_{x \in \mathbb{R}^n} \overline{f}(x) \quad (2)
\]

We now enforce the following simplifying assumptions, valid throughout the rest of the paper:

**Assumption 1 (Convexity)** The local costs $f_i$ in (1) are of class $C^3$. Moreover the global cost $\overline{f}$ is coercive, strictly convex, and with bounded positive Hessian, i.e., $0 < cI < \nabla^2 \overline{f}(x) < mI$ for some $c, m \in \mathbb{R}^+$ and $\forall x \in \mathbb{R}^n$. Moreover, w.l.o.g., $x^* = 0$ and $\overline{f}(0) = 0$. 

DRAFT
The assumptions \( x^* = 0 \) and \( f(0) = 0 \) are included just for notational convenience, while the scalar \( c \) is assumed to be known a-priori. Assumption 1 ensures that \( x^* \) in (2) exists and is unique. The strictly positive definite Hessian is moreover a mild sufficient condition to guarantee that the minimum \( x^* \) defined in (2) will be globally exponentially stable under the continuous and discrete Newton-Raphson dynamics described in the following Theorem 3. We also notice that, for the consequent results, in principle just the average function \( \bar{f} \) needs to have specific properties, and thus no conditions for the single \( f_i \)'s are required (that for example might be even non convex).

C. Stability of discretized dynamics

This subsection aims to show that, under suitable assumptions, forward-Euler discretization of suitable exponentially stable continuous-time dynamics maintains the same global exponential stability properties.

**Theorem 2** Let the continuous-time system

\[
\dot{x} = \phi(x)
\]  

admit \( x = 0 \in \mathbb{R}^n \) as an equilibrium, and let \( V(x) : \mathbb{R}^n \mapsto \mathbb{R} \) be a Lyapunov function for (3) for which there exist positive scalars \( a_1, a_2, a_3, a_4 \) s.t., \( \forall x \in \mathbb{R}^n \),

\[
\begin{cases}
  a_1 I \leq \nabla^2 V(x) \leq a_2 I \\
  \frac{\partial V(x)}{\partial x} \phi(x) \leq -a_3 \|x\|^2 \\
  \|\phi(x)\| \leq a_4 \|x\|.
\end{cases}
\] (4a) (4b) (4c)

Then:

a) the origin of system (3) is globally exponentially stable;

b) there exists a positive scalar \( \bar{\varepsilon} \) such that for every \( \varepsilon \in (0, \bar{\varepsilon}) \) the origin is globally exponentially stable for

\[
x(k + 1) = x(k) + \varepsilon \phi(x(k)).
\] (5)

D. Stability of single-agent NR dynamics

This subsection shows that the results of Section III-C apply to continuous NR dynamics\(^1\).

**Theorem 3** Let

\(^1\)We notice that other asymptotic properties of continuous time NR methods are available in the literature, e.g., [49], [50].
\begin{equation}
\phi_{\text{NR}}(x) := -\overline{h}(x)^{-1}\nabla \overline{f}(x)
\end{equation}

be defined by a generic function \( \overline{h}(x) \in \mathbb{R}^{n \times n} \) that satisfies the positive definiteness conditions \( cI \leq \overline{h}(x) = \overline{h}(x)^T \leq mI \) for all \( x \in \mathbb{R}^n \), and with \( c \) and \( m \) defined in Assumption 1. Let (6) define the dynamics

\begin{equation}
\dot{x} = \phi_{\text{NR}}(x),
\end{equation}

\begin{equation}
x(k + 1) = x(k) + \varepsilon \phi_{\text{NR}}(x(k)).
\end{equation}

Then under Assumption 1:

a) is a Lyapunov function for (7);

b) there exist positive scalars \( b_1, b_2, b_3, b_4 \) s.t., \( \forall x \in \mathbb{R}^n \),

\begin{equation}
\begin{aligned}
b_1 I \leq \nabla^2 V_{\text{NR}}(x) & \leq b_2 I \\
\partial V_{\text{NR}} / \partial x \phi_{\text{NR}}(x) & \leq -b_3 \|x\|^2 \\
\|\phi_{\text{NR}}(x)\| & \leq b_4 \|x\|.
\end{aligned}
\end{equation}

For suitable choices of \( \overline{h}(x) \) the dynamics (7) corresponds to continuous versions of well known descent dynamics. Indeed, the correspondences are

\begin{equation}
\overline{h}(x) = \begin{cases}
\nabla^2 \overline{f}(x) & \rightarrow \text{Newton-Raphson descent} \\
diag[\nabla^2 \overline{f}(x)] & \rightarrow \text{Jacobi descent} \\
I & \rightarrow \text{Gradient descent}
\end{cases}
\end{equation}

where \( \text{diag}[A] \) is a diagonal matrix containing the main diagonal of \( A \). Combining now Theorem 3 with Theorem 2 we are guaranteed that both continuous and discrete NR dynamics induced by (6) are globally exponentially stable:

**Lemma 4** Under Assumption 1, the origin is a globally exponentially stable point for dynamics (7). Moreover there exists \( \bar{\varepsilon} > 0 \) such that the origin is a globally exponentially stable point also for dynamics (8) for all \( \varepsilon < \bar{\varepsilon} \).

The previous lemma and theorems do not require \( \overline{h}(x) \) to be differentiable. However, differentiability may be used to linearize the system dynamics and obtain explicit rates of convergence. In fact, the linearized dynamics around the origin is given by

\begin{equation}
F := \frac{\partial \phi_{\text{NR}}(0)}{\partial x} = -\overline{h}(0)^{-1}\nabla^2 \overline{f}(0) - \frac{\partial \overline{h}(0)}{\partial x} \nabla \overline{f}(0).
\end{equation}

In particular, for the NR descent it holds that \( \overline{h}(x) = \nabla^2 \overline{f}(x) \). Thus in this case \( F(0) = -I \), since \( \nabla \overline{f}(0) = 0 \), and this says that the linearized continuous time NR dynamics is \( \dot{x} = -x \), independent of the cost \( \overline{f}(x) \) and whose rate of convergence is unitary and uniform along any direction.
E. Stability of multi-agent NR dynamics

We now generalize (7) by considering $N$ coupled dynamical systems that, when starting at the very same initial condition, behave like $N$ decoupled systems (7). This novel dynamics is the core of the slow-dynamics embedded in the main algorithm presented in Section IV. In this section we also include additional assumptions to show that the here presented generalization of (7) preserves global exponential stability and some other additional properties.

To this aim we introduce some additional notation: let then $h_i(x): \mathbb{R}^n \to \mathbb{R}^{n \times n}$, $i = 1, \ldots, N$ be generic continuously differentiable functions, and

\[
\bar{h}(x) := \frac{1}{N} \sum_{i=1}^{N} h_i(x), \quad \bar{h}(\bar{x}) := \frac{1}{N} \sum_{i=1}^{N} h_i(\bar{x}),
\]

additional composite functions defined starting from the $h_i$'s (recall that $x := [x_1^T, \ldots, x_N^T]^T \in \mathbb{R}^{nN}$ and that $\bar{x} := \frac{1}{N} \sum_{i=1}^{N} x_i \in \mathbb{R}^n$). Let moreover

\[
g_i(x) := h_i(x)x - \nabla f_i(x), \quad g(\bar{x}) := \bar{h}(\bar{x})x - \nabla f(\bar{x}),
\]

and $g(x), \bar{g}(x), \bar{g}(\bar{x})$ be defined accordingly as for $h_i$.

The definitions of $h_i$ and $g_i$ are instrumental to generalize the NR dynamics (7) to the distributed case. Indeed, let

\[
\psi(x) := \bar{h}(x)^{-1} \bar{g}(x), \quad \bar{h}(x)^{-1} \bar{g}(x) := \frac{1}{N} \sum_{i=1}^{N} h_i(x_i) x_i - \nabla f_i(x_i),
\]

that can be also equivalently written as

\[
\dot{x} = \varphi_{\text{NR}}(x) := -x + 1_N \otimes \psi(x),
\]

that is globally exponentially stable.

The question is then whether dynamics (14) is exponentially stable also in the general case where $x_i(0) \neq x_j(0)$. To characterize this case we exploit some additional global properties:
Assumption 5 (Global properties) The local costs \( f_1, \ldots, f_N \) defined in (1) are s.t. there exist positive scalars \( c, a_g, a_h, a_\psi \) s.t., \( \forall x, x' \in \mathbb{R}^{nN} \),

\[
\begin{align*}
& cI < \overline{h}(x) = \overline{h}^T(x) < mI \\
& \|\overline{g}_i(x)\| \leq m_g \\
& \|g_i(x) - g_i(x')\| \leq a_g \|x - x'\| \\
& \|h_i(x) - h_i(x')\| \leq a_h \|x - x'\| \\
& \|\psi(x) - \psi(x')\| \leq a_\psi \|x - x'\|
\end{align*}
\]

\[(16a)\]

Note that the definitions and Assumption 5 imply

\[
\begin{align*}
& \|\overline{g}(x) - \overline{g}(x')\| \leq a_g \|x - x'\| \\
& \|\overline{h}(x) - \overline{h}(x')\| \leq a_h \|x - x'\| \\
& \|g_i(x) - g_i(x')\| \leq a_g \|x - x'\| \\
& \|h_i(x) - h_i(x')\| \leq a_h \|x - x'\|
\end{align*}
\]

\[(17a)\]

Using the previous assumptions we can now prove global stability of dynamics (14):

**Theorem 6** Under Assumptions 1 and 5, and for a suitable positive scalar \( \eta \),

a) \[
V_{\text{PNR}}(x) := V_{\text{NR}}(x) + \frac{1}{2} \eta \|x^+\|^2 = \overline{f}(x) + \frac{1}{2} \eta \|x^+\|^2
\]

is a Lyapunov function for (14);

b) there exist positive scalars \( b_5, b_6, b_7, b_8 \) s.t., \( \forall x \in \mathbb{R}^{nN} \),

\[
\begin{align*}
& b_5 I \leq \nabla^2 V_{\text{PNR}}(x) \leq b_6 I \\
& \frac{\partial V_{\text{PNR}}}{\partial x} \phi_{\text{PNR}}(x) \leq -b_7 \|x\|^2 \\
& \|\phi_{\text{PNR}}(x)\| \leq b_8 \|x\|.
\end{align*}
\]

\[(19a)\]

\[(19b)\]

\[(19c)\]

As in Lemma 4, combining Theorem 6 with Theorem 2 it is possible to claim that (14) and its discrete-time counterpart are globally exponentially stable.

**F. Multi-agent NR dynamics under vanishing perturbations**

Let \( \chi^y := (\chi^y_1, \ldots, \chi^y_N) \) where \( \chi^y_i \in \mathbb{R}^n \), \( \chi^z := (\chi^z_1, \ldots, \chi^z_N) \) where \( \chi^z_i \in \mathbb{R}^{n \times n} \), and \( \chi := (\chi^y, \chi^z) \). Consider also the operator \( [\cdot]_c : \mathbb{R}^{nN \times n} \mapsto \mathbb{R}^{nN \times n} \), which defines the component-wise max-operation

\[
[z]_c = \begin{bmatrix} z_1 \\ \vdots \\ z_N \end{bmatrix}_c := \begin{bmatrix} z_1' \\ \vdots \\ z_N' \end{bmatrix} \quad z'_i = \begin{cases} z_i & \text{if } z_i \geq cI \\ cI & \text{otherwise}. \end{cases}
\]

\[(20)\]
With a little abuse of notation, here we also consider $x^* \in \mathbb{R}^n$ as a generic vector, and not just the solution of our optimization problem. Consider then the perturbed version of the multi-agent NR dynamics (14),

$$
\dot{x} = \phi_x(x, \chi) := -x - I_N \otimes x^* + \frac{\chi y + I_N \otimes \left( \overline{f}(x) + \overline{h}(x) \right) x^*}{\left[ \chi^2 + I_N \otimes \overline{h}(x) \right] \chi},
$$

(21)

where the division is an Hadamard division, as recalled in Section II. Direct inspection shows that

$$
\phi_x(x, 0) = \phi_{PNR}(x).
$$

(22)

The next lemma provides perturbations interconnection bounds that will be used in the main theorem in the next section.

**Lemma 7** Under Assumptions 1 and 5 there exist positive scalars $a_x, a_\Delta$ s.t.

$$
\left\{ \begin{array}{l}
\|\phi_x(x, \chi)\| \leq a_x (\|x\| + \|\chi\|) \\
\|\phi_x(x, \chi) - \phi_{PNR}(x)\| \leq a_\Delta \|\chi\|.
\end{array} \right.
$$

(23a)

(23b)

**G. Multi-agent NR dynamics under non-vanishing perturbations**

Let us now consider some additional properties of the flow (21) for some specific non-vanishing perturbation. Let us consider the following perturbation

$$
\xi = (\chi^y, \chi^z),
$$

where $\chi^y = 1_N \otimes \xi^y, \chi^z = 1_N \otimes \xi^z$, and $\xi = (\chi^y, \chi^z)$. We are interest in studying the equilibrium points of the equations $\phi_x(x, \xi) = 0$ which are characterized by the following theorem:

**Theorem 8** Let us consider the equation:

$$
\phi_x(x, \xi) = 0
$$

where $\xi = (\xi^y, \xi^z), \xi^y = 1_N \otimes \xi^y, \xi^z = 1_N \otimes \xi^z$, and $\xi^y \in \mathbb{R}^n$ and $\xi^z \in \mathbb{R}^{n \times n}$. Let also define $\xi = (\xi^y, \xi^z)$.

Then under Assumptions 1 and 5 there exists a positive scalar $r > 0$ such that $\forall \|\xi\| \leq r$ there exists a unique continuously differentiable function $x^{eq}(\xi)$ such that

a)

$$
\phi_x(x^{eq}(\xi), \xi) = 0, \quad x^{eq}(0) = 0, \quad x^{eq}(\xi) = 1_N \otimes x^*(\xi)
$$

where

b)

$$
x^{eq}(\xi) = \left( \overline{h}(x^{eq}(\xi)) + \xi^z \right)^{-1} \left( \overline{f}(x^{eq}(\xi)) + \xi^y \right);
$$

(24)

Theorem 8 allows to define

$$
\phi'_x(x, \xi) := \phi_x(x + 1_N \otimes x^*(\xi), \xi)
$$

(25)

and the corresponding dynamics

$$
\dot{x} = \phi'_x(x, \xi)
$$

(26)

which corresponds to the translated version of the original perturbed system $\phi_{\xi}(x, \xi)$, which has now the property that the origin is an equilibrium point, i.e., $\phi_{\xi}(0, \xi) = 0, \forall \|\xi\| \leq r.$
To prove the global exponential stability of (26) we need the flow $\phi'_x$ a global Lipschitz condition:

**Assumption 9 (Global Lipschitz perturbation)** There exists a positive scalar $a_\xi$ s.t.

$$\|\phi'_x(x, \xi) - \phi'_x(x, 0)\| \leq a_\xi \|\xi\| \|x\| \quad \forall \|\xi\| \leq r.$$ 

With these assumptions we can prove that the origin is a globally exponentially stable equilibrium for dynamics (26):

**Theorem 10** Under Assumptions 1, 5 and 9,

a) $V_{PNR}(x)$ defined in (18) is a Lyapunov function for (26);

b) there exist positive scalars $r, b'_7, b'_8$ s.t., $\forall x \in \mathbb{R}^N$,

$$\begin{align}
\frac{\partial V_{PNR}}{\partial x} \phi'_x(x, \xi) &\leq -b'_7 \|x\|^2 \\
\|\phi'_x(x, \xi)\| &\leq b'_8 \|x\|.
\end{align} \tag{27a} \tag{27b}$$

for all $\|\xi\| \leq r$.

Again, as in Lemma 4, combining Theorem 10 with Theorem 2 it is possible to claim that (26) and its discrete-time counterpart are globally exponentially stable.

**H. Quadratic Functions**

Before presenting the main algorithm, we show that quadratic costs satisfy all the previous assumptions. Consider then

$$f_i(x) = \frac{1}{2}(x - d_i)^T A_i (x - d_i) + e_i, \quad A_i = A_i^T$$

and assume that, w.l.o.g.,

$$\sum_i A_i d_i = 0, \quad \sum_i (d_i^T A_i d_i + e_i) = 0$$

condition that can be met by a simple change of coordinate and a constant translation of the cost functions.

**Theorem 11** Quadratic costs that satisfy

$$A := \sum_i A_i > 0$$

satisfy Assumptions 1, 5 and 9 for $h_i(x) = \nabla^2 f_i(x)$.

It is also important to remark that the used assumptions are tight, in the sense that if they are not satisfied it is possible to construct a dynamical system which is not globally stable. Indeed, the assumptions are very common in the Lyapunov literature and they are necessary to guarantee the existence of quadratic-type Lyapunov functions.
IV. NEWTON-RAPHSON CONSENSUS

Consider the following algorithm, where \( g(x(-1)) = 0 \) and \( h(x(-1)) = 0 \) in the initialization step should be intended as initialization of suitable registers and not as operations involving the quantity \( x(-1) \).

**Algorithm 1** Newton-Raphson Consensus (NRC)

<table>
<thead>
<tr>
<th>(storage allocation and constraints on the parameters)</th>
</tr>
</thead>
</table>
| 1: \( x_i(k), y_i(k) \in \mathbb{R}^n \) and \( z_i(k) \in \mathbb{R}^{n \times n} \) for all \( k \) and \( i = 1, \ldots, N; \) \( \varepsilon \in (0, 1), c > 0 \)
| (initialization) |
| 2: \( x_i(0) = 0; \) \( y_i(0) = g_i(x_i(-1)) = 0; \) \( z_i(0) = h_i(x_i(-1)) = 0 \)
| (main algorithm) |
| 3: for \( k = 1, 2, \ldots \) do |
| 4: \hspace{0.5cm} for \( i = 1, \ldots, N \) do |
| 5: \hspace{1.0cm} \( x_i(k) = (1 - \varepsilon)x_i(k - 1) + \varepsilon \left[ z_i(k - 1)^{-1} y_i(k - 1) \right] \)
| 6: \hspace{1.0cm} \( y_i(k) = \sum_{j=1}^{N} p_{ij} \left( y_j(k - 1) + g_j(x_j(k - 1)) - g_j(x_j(k - 2)) \right) \)
| 7: \hspace{1.0cm} \( z_i(k) = \sum_{j=1}^{N} p_{ij} \left( z_j(k - 1) + h_j(x_j(k - 1)) - h_j(x_j(k - 2)) \right) \)
| 8: \hspace{0.5cm} end for |
| 9: end for |

Intuitively, the algorithm functions as follows: if the dynamics of the \( x_i(k) \)s is sufficiently slow w.r.t. the dynamics of the \( y_i(k) \)s and \( z_i(k) \)s, then the two latter quantities tend to reach consensus. Then, the more these quantities reach consensus, the more the products \( [z_i(k)]^{-1} y_i(k) \) exhibit these two specific characteristics: \( i) \) being the same among the various agent; \( ii) \) representing Newton descent directions. Thus, the more the \( y_i(k) \)s and \( z_i(k) \)s are sufficiently close, the more the various \( x_i(k) \)s are driven by the same forcing term, that makes them converge to the same value, equal to the optimum \( x^* \).

We now characterize the convergence properties of Algorithm 1. Defining

\[
\xi_y := \frac{1}{N} \sum_{i=1}^{N} (y_i(0) - g_i(x_i(-1))) \\
\xi_z := \frac{1}{N} \sum_{i=1}^{N} (z_i(0) - h_i(x_i(-1)))
\]

the following applies:

**Theorem 12** Consider the dynamics defined by Algorithm 1 with possibly nonzero initial conditions. If \( \xi_y = 0 \) and \( \xi_z = 0 \), then under Assumptions 1 and 5 there exists a positive scalar \( \tau > 0 \) such that the local estimates \( x_i(k) \) converge to the global minimizer, i.e., such that

\[
\lim_{k \to +\infty} x_i(k) = x^* \quad \forall i = 1, \ldots, N.
\]

for all \( \varepsilon \in (0, \tau) \) and \( x_i(0) \in \mathbb{R}^n \).

Non-null initial conditions on \( \xi_y \) and \( \xi_z \) lead the algorithm to do not compute the global optimum \( x^* \). Nonetheless in this case the computed solution, as a function of the initial conditions, is a smooth function and thus small errors in the initial conditions
do not produce dramatic errors in the computation of the optimum:

**Theorem 13** Consider the dynamics defined by Algorithm 1 with possibly nonzero initial \( \xi_y \) and \( \xi_z \) but generic \( x_i(0) \)'s. Under Assumptions 1, 5 and 9 there exist a positive scalar \( r, \varepsilon \) and a continuously differentiable function \( \psi(\xi_y, \xi_z) : \mathbb{R}^n \times \mathbb{R}^n, n \rightarrow \mathbb{R}^n \) s.t.

\[
\lim_{k \rightarrow \infty} x_i(k) = \psi(\xi_y, \xi_z) \quad \forall i = 1, \ldots, N
\]

and satisfying

\[
\Vert \psi(\xi_y, \xi_z) - x^* \Vert \leq \alpha(\Vert \xi_y \Vert + \Vert \xi_z \Vert)
\]

for all \( \varepsilon \in (0, \pi) \), initial conditions \( x_i(0) \in \mathbb{R}^n \) and \( (\Vert \xi_y \Vert + \Vert \xi_z \Vert) \leq r \).

Assumptions 1, 5 and 9 are needed to ensure global convergence properties w.r.t. the initial conditions \( x_i(0) \)'s. Relaxing these hypotheses the convergence reduces to be local:

**Theorem 14** Consider the dynamics defined by Algorithm 1 with possibly nonzero initial conditions. Under the assumptions that the \( f_i \)'s are \( C^3 \) and that \( \nabla^2 J(x^*) > cI \), there exist positive scalars \( r, r', \pi \) and a continuously differentiable function \( \psi(\xi_y, \xi_z) : \mathbb{R}^n \times \mathbb{R}^n, n \rightarrow \mathbb{R}^n \) s.t.

\[
\lim_{k \rightarrow \infty} x_i(k) = \psi(\xi_y, \xi_z) \quad \forall i = 1, \ldots, N
\]

and satisfying

\[
\Vert \psi(\xi_y, \xi_z) - x^* \Vert \leq \alpha(\Vert \xi_y \Vert + \Vert \xi_z \Vert)
\]

for all \( \varepsilon \in (0, \pi) \) and initial conditions \( \|x_i(0) - x^*\| \leq r' \), \( \|y_i - \overline{f}(x^*)\| \leq r \), \( \|z_i - \overline{h}(x^*)\| \leq r \), \( \|g_i(x_i(-1)) - \overline{f}(x^*)\| \leq r \), \( \|h_i(x_i(-1)) - \overline{h}(x^*)\| \leq r \).

Numerical simulations suggest that the algorithm is robust w.r.t. numerical errors and quantization noise. We also notice that Theorem 12 guarantees the existence of a critical value \( \pi_\varepsilon \) but does not provide indications on its value. This is a known issue in all the systems dealing with separation of time scales. A standard rule of thumb is then to let the rate of convergence of the fast dynamics be sufficiently faster than the one of the slow dynamics, typically 2-10 times faster. In our algorithm the fast dynamics inherits the rate of convergence of the consensus matrix \( P \), given by its spectral gap \( \sigma(P) \), i.e., its spectral radius \( \rho(P) = 1 - \sigma(P) \). The rate of convergence of the slow dynamics is instead governed by (15), which is nonlinear and therefore possibly depending on the initial conditions. However, close to the equilibrium point the dynamic behavior is approximately given by \( \pi(t) \approx - (\pi(t) - x^*) \), thus, since \( x_i(k) \approx \pi(\varepsilon k) \), then the convergence rate of the algorithm approximately given by

\[
1 - \varepsilon.
\]

Thus we aim to let \( 1 - \rho(P) \gg 1 - (1 - \varepsilon) \), which provides the rule of thumb

\[
\varepsilon \ll \sigma(P). \tag{28}
\]
which is suitable for generic cost functions. We then notice that, although the spectral gap \( \sigma(P) \) might not be known in advance, it is possible to distributedly estimate it, see, e.g., [51]. However, such rule of thumb might be very conservative. In fact, if all the \( f_i \)'s are quadratic then the procedure reduces to

\[
\begin{align*}
y(k+1) &= Py(k) \\
z(k+1) &= Pz(k) \\
x(k+1) &= y(k+1) / z(k+1),
\end{align*}
\]

Thus:

**Theorem 15** Consider Algorithm 1 with arbitrary initial conditions \( x(0) \), quadratic cost functions \( f_i = \frac{1}{2} (x - d_i)^T A_i (x - d_i) \) with \( A_i > 0 \) and \( \varepsilon = 1 \). Then \( \| x(k) - x^* \| \leq \alpha (\rho(P))^k \) for all \( k \) and for a suitable positive \( \alpha \).

Thus, if the cost functions are close to be quadratic then the overall rate of convergence is limited by the rate of convergence of the embedded consensus algorithm. Moreover, the values of \( \varepsilon \) that still guarantee convergence can be much larger than those dictated by the rule of thumb (28).

### A. On the selection of the structure of \( h(x) \)

As introduced in Section III-D, by selecting different structures for \( h_i(x) \) one can obtain different procedures with different convergence properties and different computational/communication requirements. Some plausible choices for \( h_i \) are

\[
h_i(x) = \begin{cases} 
\nabla^2 f_i(x) \\
\text{diag}[\nabla^2 f_i(x)] \\
I
\end{cases}
\]

and the correspondences are the following:

- \( h_i(x) = \nabla^2 f_i(x) \rightarrow \text{Newton-Raphson Consensus (NRC)} \): in this case it is possible to rewrite the main algorithm and show that, for sufficiently small \( \varepsilon \), \( x_i(k) \approx \pi(\varepsilon k) \), where \( \pi(t) \) evolves according to the continuous-time Newton-Raphson dynamics

\[
\dot{\pi}(t) = - \left( \nabla^2 f_i(\pi(t)) \right)^{-1} \nabla f_i(\pi(t)).
\]

- \( h_i(x) = \text{diag}[\nabla^2 f_i(x)] \rightarrow \text{Jacobi Consensus (JC)} \): choice \( h_i(x) = \nabla^2 f_i(x) \) requires agents to exchange information on \( O(n^2) \) scalars, and this could pose problems under heavy communication bandwidth constraints and large \( n \)’s. Choice \( h_i(x) = \text{diag}[\nabla^2 f_i(x)] \) instead reduces the amount of information to be exchanged via the underlying diagonalization process, also called Jacobi approximation\(^2\). In this case, for sufficiently small \( \varepsilon \), \( x_i(k) \approx \pi(\varepsilon k) \), where \( \pi(t) \) evolves according to the continuous-time dynamics

\[
\dot{\pi}(t) = - \left( \text{diag}[\nabla^2 f_i(\pi(t))] \right)^{-1} \nabla f_i(\pi(t)),
\]

which can be shown to converge to the global optimum \( x^* \) with a convergence rate that in general is slower than the Newton-Raphson when the global cost function is skewed.

\(^2\)In centralized approaches, nulling the Hessian’s off-diagonal terms is a well-known procedure, see, e.g., [52]. See also [53], [35] for other Jacobi algorithms with different communication structures.
• \( h_i(x) = I \rightarrow \text{Gradient Descent Consensus (GDC): this choice is motivated in frameworks where the computation of the local second derivatives } \frac{\partial^2 f_i}{\partial x_m} \text{ is expensive, or where the second derivatives simply might not be continuous. With this choice the main algorithm reduces to a distributed gradient-descent procedure. In fact, for sufficiently small } \varepsilon, x_i(k) \approx \overline{x}(\varepsilon k) \text{ with } \overline{x}(t) \text{ evolving according to the continuous-time dynamics}

\[
\dot{\overline{x}}(t) = -\nabla \overline{f}(\overline{x}(t)),
\]

which one again is guaranteed to converge to the global optimum \( x^* \).

The following Table I summarizes the various costs of the previously proposed strategies.

<table>
<thead>
<tr>
<th>Choice</th>
<th>NRC, ( h_i(x) = \nabla f_i(x) )</th>
<th>JC, ( h_i(x) = \text{diag}[\nabla^2 f_i(x)] )</th>
<th>GDC, ( h_i(x) = I )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computational Cost</td>
<td>( O(n^3) )</td>
<td>( O(n) )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Communication Cost</td>
<td>( O(n^2) )</td>
<td>( O(n) )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Memory Cost</td>
<td>( O(n^2) )</td>
<td>( O(n) )</td>
<td>( O(n) )</td>
</tr>
</tbody>
</table>

Table I

**Computational, Communication and Memory Costs of NRC, JC, GDC per Single Unit and Single Step.**

We remark that \( \varepsilon \) in Theorem 12 depends also on the particular choice for \( h_i \). The list of choices for \( h_i \) given above is not exhaustive. For example, future directions are to implement distributed quasi-Newton procedures. To this regard, we recall that approximations of the Hessians that do not maintain symmetry and positive definiteness or are bad conditioned require additional modification steps, e.g., through Cholesky factorizations [54].

Finally, we notice that in scalar scenarios JC and NRC are equivalent, while GDC corresponds to algorithms requiring just the knowledge of first derivatives.

**V. Numerical Examples**

In Section V-A we analyze the effects of different choices of \( \varepsilon \) on the NRC on regular graphs and exponential cost functions. We then propose two machine learning problems in Section V-B, used in Sections V-C and V-D, and numerically compare the convergence performance of the NRC, JC, GDC algorithms and other distributed convex optimization algorithms on random geometric graphs.

**A. Effects of the choice of \( \varepsilon \)**

Consider a ring network of \( S = 30 \) agents that communicate only to their left and right neighbors through the consensus matrix

\[
P = \begin{bmatrix}
0.5 & 0.25 & 0.25 \\
0.25 & 0.5 & 0.25 \\
& \ddots & \ddots & \ddots \\
0.25 & 0.5 & 0.25 \\
0.25 & 0.25 & 0.5
\end{bmatrix},
\]

(30)
so that the spectral radius \( \rho(P) \approx 0.99 \), implying a spectral gap \( \sigma(P) \approx 0.01 \). Consider also scalar costs of the form
\[
f_i(x) = c_i e^{a_i x} + d_i e^{-b_i x}, \quad i = 1, \ldots, N,
\]
with \( a_i, b_i \sim \mathcal{U}[0, 0.2] \), \( c_i, d_i \sim \mathcal{U}[0, 1] \) and where \( \mathcal{U} \) indicates the uniform distribution.

Figure 1 compares the evolution of the local states \( x_i \) of the continuous system (39) for different values of \( \varepsilon \). When \( \varepsilon \) is not sufficiently small, then the trajectories of \( x_i(t) \) are different even if they all start from the same initial condition \( x_i(0) = 0 \). As \( \varepsilon \) decreases, the difference between the two time scales becomes more evident and all the trajectories \( x_i(k) \) become closer to the trajectory given by the slow NR dynamics \( \pi(\varepsilon k) \) given in (15) and guaranteed to converge to the global optimum \( x^* \).

![Figure 1](image_url)

**Figure 1.** Temporal evolution of system (39) for different values of \( \varepsilon \), with \( N = 30 \). The black dotted line indicates \( x^* \). The black solid line indicates the slow dynamics \( \pi(\varepsilon k) \) of Equation (15). As \( \varepsilon \) decreases, the difference between the time scale of the slow and fast dynamics increases, and the local states \( x_i(k) \) converge to the manifold of \( \pi(\varepsilon k) \).

In Figure 2 we address the robustness of the proposed algorithm w.r.t. the choice of the initial conditions. In particular, Figure 2(a) shows that if \( \alpha = \beta = 0 \) then the local states \( x_i(t) \) converge to the optimum \( x^* \) for arbitrary initial conditions \( x_i(0) \). Figure 2(b) considers, besides different initial conditions \( x_i(0) \), also perturbed initial conditions \( v(0), w(0), y(0), z(0) \) leading to non null \( \alpha \)'s and \( \beta \)'s. More precisely we apply Algorithm 1 to different random initial conditions s.t. \( \alpha, \beta \sim \mathcal{U}[-\sigma, \sigma] \).

Figure 2(b) shows the boxplots of the errors \( x_i(+\infty) - x^* \) for different \( \sigma \)'s based on 300 Monte Carlo runs with \( \varepsilon = 0.01 \) and \( N = 30 \).

**B. Optimization problems**

The first problem considered is the distributed training of a Binomial-Deviance based classifier, to be used, e.g., for spam-nonspam classification tasks [55, Chap. 10.5]. More precisely, we consider a database of emails \( E \), where \( j \) is the email index, \( y_j = -1, 1 \) denotes if the email \( j \) is considered spam or not, \( \chi_j \in \mathbb{R}^n \) numerically summarizes the \( n \) features of the \( j \)-th email (how many times the words “money”, “dollars”, etc., appear). If the \( E \) emails come from different users that do not want to disclose their private information, then it is meaningful to solve exploiting the distributed optimization algorithms described in the previous sections. More specifically, letting \( x = (x, x_0) \in \mathbb{R}^n \times \mathbb{R} \) represents a generic classification hyperplane, training a Binomial-Deviance based classifier corresponds to solve a distributed optimization problem where the local cost functions are...
Figure 2. Characterization of the dependency of the performance of Algorithm 1 on the initial conditions. In all the experiments $\varepsilon = 0.01$ and $N = 30$.

given by:

$$f_i(x) := \sum_{j \in E_i} \log \left( 1 + \exp \left( -y_j \left( \chi_j^T x + x_0 \right) \right) \right) + \gamma \|x\|^2_2. \quad (31)$$

where $E_i$ is the set of emails available to agent $i$, $E = \cup_{i=1}^N \cup E_i$, and $\gamma$ is a global regularization parameter. In the following numerical experiments we consider $|E| = 5000$ emails from the spam-nonspam UCI repository, available at [link](http://archive.ics.uci.edu/ml/datasets/Spambase), randomly assigned to 30 different users communicating as in graph of Figure 3. For each email we consider $n = 3$ features

(the frequency of words “make”, “address”, “all”) so that the corresponding optimization problem is 4-dimensional.

The second problem considered is a regression problem inspired by the UCI Housing dataset available at [link](http://archive.ics.uci.edu/ml/datasets/Housing).

In this task, an example $\chi_j \in \mathbb{R}^{n-1}$ is a vector representing some features of a house (e.g., per capita crime rate by town, index of accessibility to radial highways, etc.), and $y_j \in \mathbb{R}$ denotes the corresponding median monetary value of of the house.

The objective is to obtain a predictor of house value based on these data. Similarly as the previous example, if the datasets come from different users that do not want to disclose their private information, then it is meaningful to solve exploiting the distributed optimization algorithms. This problem can be formulated as a convex regression problem on the local costs

$$f_i(x) := \sum_{j \in E_i} \left( \frac{(y_j - \chi_j^T x - x_0)^2}{|y_j - \chi_j^T x - x_0| + \beta} + \gamma \|x\|^2_2 \right). \quad (32)$$

where $x = (x, x_0) \in \mathbb{R}^{n-1} \times \mathbb{R}$ is the vector of coefficient for the linear predictor $\hat{y} = \chi^T x + x_0$ and $\gamma$ is a common regularization parameter. The loss function $\frac{(\cdot)^2}{|\cdot|+\beta}$ corresponds to a smooth version of the Huber robust loss (being $C^2$) which is employed to minimize the effects of outliers.

The loss function $\frac{(\cdot)^2}{|\cdot|+\beta}$ corresponds to a smooth version of the Huber robust loss (being $C^2$), with $\beta$ a parameter dictating for
which arguments the loss is pseudo-linear or pseudo-quadratic and to be manually chosen to minimize the effects of outliers.

In our experiments we used $n = 4$ features, $\beta = 50$, $\gamma = 1$, and $|E| = 506$ total number of examples in the dataset randomly assigned to the $N = 30$ users communicating as in the graph of Figure 3.

In both the previous problems the optimum, in the following indicated for simplicity with $x^\ast$, has been computed with a centralized NR with the termination rule “stop when in the last 5 steps the norm of the guessed $x^\ast$ changed less than $10^{-9}$%”.

Figure 3. Random geometric graph exploited in the simulations relative to the optimization problem (31). For this graph $\rho(P) \approx 0.9338$, with $P$ the matrix of Metropolis weights.

C. Comparison of the NRC, JC and GDC algorithms

In Figure 4 we analyze the performance of the three proposed NRC, JC and GDC algorithms defined by the various choices for $h_i(x)$ in Algorithm 1 in terms of the relative Mean Square Error (MSE)

$$\text{MSE}(k) := \frac{1}{N} \sum_{i=1}^{N} \|x_i(k) - x^\ast\|^2 / \|x^\ast\|^2$$

for the classification and regression optimization problem described above. The consensus matrix $P$ has been by selecting the Metropolis-Hastings weights which are consistent with the communication graph [56]. Panels 4(a) and 4(c) report the MSE obtained at a specific iteration ($k = 40$) by the various algorithms, as a function of $\varepsilon$. These plots thus inspect the sensitivity w.r.t. the choice of the tuning parameters. Consistently with the theorems in the previous section, the GDC and JC algorithms are stable only for $\varepsilon$ sufficiently small, while NRC exhibit much larger robustness and best performance for $\varepsilon = 1$. Panels 4(b) and 4(d) instead report the evolutions of the relative MSE as a function of the number of iterations $k$ for the optimally tuned algorithms.

We notice that the differences between NRC and JC are evident but not resounding, due to the fact that the Jacobi approximations are in this case a good approximation of the analytical Hessians. Conversely, GDC presents a slower convergence rate which is a known drawback of gradient descent algorithms.

D. Comparisons with other distributed convex optimization algorithms

We now compare Algorithm 1 and its accelerated version, referred as Fast Newton-Raphson Consensus (FNRC) and described in detail below in Algorithm 2), with three popular distributed convex optimization methods, namely the DSM, the Distributed
Control Method (DCM) and the ADMM, described respectively in Algorithm 3, 4 and 5. The following discussion provides some details about these strategies.

- FNRC is an accelerated version of Algorithm 1 that inherits the structure of the so called second order diffusive schedules, see, e.g., [57], and exploits an additional level of memory to speed up the convergence properties of the consensus strategy. Here the weights multiplying the $g_i$’s and $h_i$’s are necessary to guarantee exact tracking of the current average, i.e., $\sum g_i(k) = \sum g_i(x(k-1))$ for all $k$. As suggested in [57], we set the $\varphi$ that weights the gradient and the memory to $\varphi = \frac{2}{1 + \sqrt{1 - \rho(P)^2}}$. This guarantees second order diffusive schedules to be faster than first order ones (even if this does not automatically imply the FNRC to be faster than the NRC). This setting can be considered a valid heuristic to be used when $\rho(P)$ is known. For the graph in Figure 3, $\varphi \approx 1.4730$.

- DSM, as proposed in [29], alternates consensus steps on the current estimated global minimum $x_i(k)$ with subgradient updates of each $x_i(k)$ towards the local minimum. To guarantee the convergence, the amplitude of the local subgradient steps should appropriately decrease. Algorithm 3 presents a synchronous DSM implementation, where $\varphi$ is a tuning parameter and $P$ is the matrix of Metropolis-Hastings weights.
Algorithm 2 Fast Newton-Raphson Consensus
1: storage allocation, constraints on the parameters and initialization as in Algorithm 1
2: for $k = 1, 2, \ldots$ do
3: for $i = 1, \ldots, N$ do
4: \[ x_i(k) = (1 - \epsilon)x_i(k - 1) + \epsilon |z_i(k - 1)|^{-1} y_i(k) \]
5: \[ \tilde{y}_i(k) = y_i(k - 1) + \frac{1}{\varphi} g_i(x_i(k - 1)) - g_i(x_i(k - 2)) - \frac{1 - \varphi}{\varphi} g_i(x_i(k - 3)) \]
6: \[ \tilde{z}_i(k) = z_i(k - 1) + \frac{1}{\varphi} h_i(x_i(k - 1)) - h_i(x_i(k - 2)) - \frac{1 - \varphi}{\varphi} h_i(x_i(k - 3)) \]
7: \[ y_i(k) = \varphi \sum_{j=1}^{N} (p_{ij} \tilde{y}_j(k)) + (1 - \varphi) y_i(k - 2) \]
8: \[ z_i(k) = \varphi \sum_{j=1}^{N} (p_{ij} \tilde{z}_j(k)) + (1 - \varphi) z_i(k - 2) \]
9: end for
10: end for

Algorithm 3 DSM [29]
(storage allocation and constraints on parameters)
1: $x_i(k) \in \mathbb{R}^N$ for all $i$, $g \in \mathbb{R}_+$
(initialization)
2: $x_i(0) = 0$
(main algorithm)
3: for $k = 0, 1, \ldots$ do
4: for $i = 1, \ldots, N$ do
5: \[ x_i(k + 1) = \sum_{j=1}^{N} p_{ij} \left( x_i(k) - \frac{\vartheta}{k} \nabla f_j(x_j(k)) \right) \]
6: end for
7: end for

- DCM, as proposed in [41], differentiates from the gradient searching because it forces the states to the global optimum by controlling the subgradient of the global cost. This approach views the subgradient as an input/output map and uses small gain theorems to guarantee the convergence property of the system. Again, each agent $i$ locally computes and exchanges information with its neighbors, collected in the set $\mathcal{N}_i := \{ j \mid (i, j) \in E \}$. DCM is summarized in Algorithm 4, where $\mu, \nu > 0$ are parameters to be designed to ensure the stability property of the system. Specifically, $\mu$ is chosen in the interval $0 < \mu < 2 \max_{i=1,\ldots,N} |\mathcal{N}_i| + 1$ to bound the induced gain of the subgradients. Also here the parameters have been manually tuned for best convergence rates.

- ADMM, instead, requires the augmentation of the system through additional constraints that do not change the optimal solution but allow the Lagrangian formalism. There exist different implementations of ADMM in distributed contexts, see, e.g., [7], [58], [12, pp. 253-261]. For simplicity we consider the following formulation,

$$\min_{x_1, \ldots, x_N} \sum_{i=1}^{N} f_i(x_i)$$
$$\text{s.t.} \ z_{(i,j)} = x_i, \ \forall i \in \mathcal{N}, \ \forall (i,j) \in E,$$
Algorithm 4 DCM [41]

(storage allocation and constraints on parameters)

1: \(x_i(k), z_i(k) \in \mathbb{R}^n\), for all \(i\), \(\mu, \nu \in \mathbb{R}_+\)

(initialization)

2: \(x_i(0) = z_i(0) = 0\) for all \(i\)

(main algorithm)

3: for \(k = 0, 1, \ldots\) do

4: for \(i = 1, \ldots, N\) do

5: \(z_i(k + 1) = z_i(k) + \mu \sum_{j \in \mathcal{N}_i} (x_i(k) - x_j(k))\)

6: \(x_i(k + 1) = x_i(k) + \mu \sum_{j \in \mathcal{N}_i} (x_j(k) - x_i(k)) + \mu \sum_{j \in \mathcal{N}_i} (z_j(k) - z_i(k)) - \mu \nu \nabla f_i(x_i(k))\)

7: end for

8: end for

where the auxiliary variables \(z_{(i,j)}\) correspond to the different links in the network, and where the local Augmented Lagrangian is given by

\[
L_i(x_i, k) := f_i(x_i) + \sum_{j \in \mathcal{N}_i} y_{(i,j)}(x_i - z_{(i,j)}) + \sum_{j \in \mathcal{N}_i} \frac{\delta}{2} ||x_i - z_{(i,j)}||^2,
\]

with \(\delta\) a tuning parameter (see [59] for a discussion on how to tune it) and the \(y_{(i,j)}\)'s Lagrange multipliers.

Algorithm 5 ADMM [7, pp. 253-261]

(storage allocation and constraints on parameters)

1: \(x_i(k), z_{(i,j)}(k), y_{(i,j)}(k) \in \mathbb{R}^n, \delta \in (0, 1)\)

(initialization)

2: \(x_i(0) = z_{(i,j)}(0) = y_{(i,j)}(0) = 0\)

(main algorithm)

3: for \(k = 0, 1, \ldots\) do

4: for \(i = 1, \ldots, N\) do

5: \(x_i(k + 1) = \arg\min_{x_i} L_i(x_i, k)\)

6: for \(j \in \mathcal{N}_i\) do

7: \(z_{(i,j)}(k + 1) = \frac{1}{2\delta} \left( y_{(i,j)}(k) + y_{(j,i)}(k) \right) + \frac{1}{2} \left( x_i(k + 1) + x_j(k + 1) \right)\)

8: \(y_{(i,j)}(k + 1) = y_{(i,j)}(k) + \delta \left( x_i(k + 1) - z_{(i,j)}(k + 1) \right)\)

9: end for

10: end for

11: end for

Figure 5 then compares the previously cited algorithms as did in Figure 4. The first panel thus reports the relative MSE of the various algorithms at a given number of iterations \((k = 40)\) as a function of the parameters. The second panel instead reports the temporal evolution of the relative MSE for the case of optimal tuning.
(a) Relative MSE at a given time $k$ as a function of the algorithms parameters for problem (31). For the DCM, $\nu = 1.7$.

(b) Relative MSE as a function of the time $k$ for the three fastest algorithms for problem (31). Their parameters are chosen as the best ones from Figure 5(a).

(c) Relative MSE at a given time $k$ as a function of the algorithms parameters for problem (32). For the DCM, $\nu = 1.7$. 

(d) Relative MSE as a function of the time $k$ for the three fastest algorithms for problem (32). Their parameters are chosen as the best ones from Figure 5(c).
We notice that the DCM and the DSM are both much slower than the NRC, FNRC and ADMM. Moreover, both the NRC and its accelerated version converge faster than the ADMM, even if not tuned at their best. These numerical examples seem to indicate that the proposed NRC might be a viable alternative to the ADMM, although further comparisons are needed to strengthen this claim. Moreover, a substantial potential advantage of NRC as compared to ADMM is that the former can be readily adapted to asynchronous and time-varying graphs, as preliminary made in [60]. Moreover, as in the case of the FNRC, the strategy can implement any improved linear consensus algorithm.

VI. CONCLUSION

We proposed a novel distributed optimization strategy suitable for convex, unconstrained, multidimensional, smooth and separable cost functions. The algorithm does not rely on Lagrangian formalisms and acts as a distributed Newton-Raphson optimization strategy by repeating the following steps: agents first locally compute and update second order Taylor expansions around the current local guesses and then they suitably combine them by means of average consensus algorithms to obtain a sort of approximated Taylor expansion of the global cost. This allows each agent to infer a local Newton direction, used to locally update the guess of the global minimum.

Importantly, the average consensus protocols and the local updates steps have different time-scales, and the whole algorithm is proved to be convergent only if the step-size is sufficiently slow. Numerical simulations show that, if suitably tuned, the algorithm is generally faster than DSMs and competitive with respect to ADMMs.

The set of open research paths is extremely vast. We envisage three main avenues. The first one is to study how the agents can dynamically and locally tune the speed of the local updates w.r.t. the consensus process, namely how to tune their local step-size $\varepsilon$. In fact large values of $\varepsilon$ gives faster convergence but might lead to instability. A second one is to let the communication protocol be asynchronous: in this regard we notice that some preliminary attempts can be found in [60]. A final branch is about the analytical characterization of the rate of convergence of the proposed strategies and the extensions to non-smooth convex functions.

REFERENCES


Proof (of Theorem 2) proof of a): integrating (4a) twice implies
\[
\frac{1}{2}a(x^2 + \alpha(x) = \frac{1}{2}a_1 ||x||^2
\]
that, jointly with (4b), immediately guarantee global exponential stability for (3) [44, Thm. 4.10].

Proof of b): consider
\[
\Delta V(x(k)) : = V(x(k + 1)) - V(x(k))
\]
To prove the claim we show that \(\Delta V(x(k)) \leq -d||x||^2\) for some positive scalar \(d\). To this aim, expand \(V(x(k + 1))\) with a second order Taylor expansion around \(x(k)\) with remainder in Lagrange form, to obtain
\[
V(x + \epsilon \phi(x)) = V(x) + \epsilon \frac{\partial V}{\partial x} \phi(x) + \frac{1}{2} \epsilon^2 \phi(x) \nabla^2 V(x_{\epsilon}) \phi(x)
\]
with \(x_{\epsilon} = x + \epsilon \phi(x)\) for \(\epsilon \in [0, \varepsilon]\). Using inequalities (4) we then obtain
\[
\Delta V(x(k)) = V(x(k + 1)) - V(x(k)) \leq -a_3||x(k)||^2 + \frac{1}{2} \epsilon^2 m a_4^2 ||x(k)||^2
\]
Thus, for all \(\varepsilon < \varepsilon = \frac{2a_3}{m a_4^2}\), the origin is globally exponentially stable.

Proof (of Theorem 3) proof of a): Assumption 1 guarantees that \(V_{\text{NR}}(0) = 0\) and \(V_{\text{NR}}(x) \geq 0\) for \(x \neq 0\). Moreover, for \(x \neq 0\),
\[
\frac{\partial V_{\text{NR}}}{\partial x} \phi_{\text{NR}}(x) = - (\nabla \tilde{f}(x))^T \tilde{h}(x)^{-1} \nabla \tilde{f}(x)
\]
To prove (10c) consider instead that (10a) guarantees \(c||x|| \leq \frac{1}{c} \nabla \tilde{f}(x)|| \leq m \frac{c}{||x||} \), that in its turn implies
\[
\frac{1}{c} \nabla \tilde{f}(x)|| \leq m \frac{c}{||x||} \leq m \frac{c}{||x||} \leq m \frac{c}{||x||} = b_4 \frac{c}{||x||}.
\]
To prove (10b) eventually consider then that (10c) implies
\[
\frac{\partial V_{\text{NR}}}{\partial x} \phi_{\text{NR}}(x) = - (\nabla \tilde{f}(x))^T \tilde{h}(x)^{-1} \nabla \tilde{f}(x)
\]
Proof (of Theorem 6) In the interest of clarity we analyze the case where the local costs \(f_i\) are scalar, i.e., \(n = 1\). The multivariable case is indeed a straightforward extension with just a more involved notation. We also recall the following equivalences:
\[
x = x^\| + x^\perp, \quad (x^\perp)^T x^\perp = 0,
\]
\[
||x||^2 = ||x^\|||^2 + ||x^\perp||^2 = N ||x^\perp||^2 + ||x^\perp||^2.
\]
Proof of a): \(V_{\text{NR}}(0) = 0\) and \(V_{\text{NR}}(x) \geq 0\) for \(x \neq 0\) follow immediately from the fact that \(V_{\text{NR}}(0) = 0\) and \(V_{\text{NR}}(x) \geq 0\) for \(x \neq 0\). \(\dot{V}_{\text{NR}} < 0\) is instead proved by proving (19b).

Proof of inequality (19a): given (18),
\[
\frac{\partial^2 V_{\text{NR}}(x)}{\partial x^2} = \frac{\partial^2 (V_{\text{NR}}(x) + \frac{1}{2} \eta ||x^\perp||^2)}{\partial x^2}.
\]
Since $0 \leq \|x^+\|^2 \leq \|x\|^2$ and
\[
\frac{\partial^2 V_{\text{NR}}(x)}{\partial x^2} = \frac{1}{N^2} \mathbf{1} \mathbf{1}^T \nabla^2 V_{\text{NR}}(x),
\]
thanks to (10a) it follows immediately that (19a) holds with
\[
b_5 = \min\{\frac{b_1}{N}, \eta\}, \quad b_6 = \max\{\frac{b_2}{N}, \eta\}.
\]

**Proof of inequality (19c):** being the origin of $\mathbf{f}$ a minimum, then $\nabla \mathbf{f}(0) = 0$, and thus $\nabla \mathbf{f}(0) = 0$ (cf. (11)). Thus also $\psi(0) = 0$, that in turn implies $\|\psi(x)\| \leq a_\psi \|x\|$ by Assumption 5. Therefore
\[
\|\psi_{\text{NR}}(x)\| \leq \|x\| + N\|\psi(x)\| \leq (1 + Na_\psi)\|x\| = b_8\|x\|.
\]

**Proof of inequality (19b):** since
\[
\frac{\partial \mathbf{x}}{\partial x} = \frac{1}{N} \mathbf{1}^N, \quad \frac{\partial x^+}{\partial x} = I - \frac{1}{N} \mathbf{1} \mathbf{1}^T =: \Pi,
\]
it follows that
\[
\frac{\partial V_{\text{NR}}}{\partial x} \psi_{\text{NR}}(x) = \left( \frac{\partial V_{\text{NR}}}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial x} + \frac{\partial V_{\text{NR}}}{\partial x^+} \frac{\partial x^+}{\partial x} \right) \phi_{\text{NR}}(x) = \left( \frac{\partial V_{\text{NR}}(x)}{\partial \mathbf{x}} \frac{1}{N} \mathbf{1}^N + \eta(x^+)^T \Pi \right) \phi_{\text{NR}}(x).
\]

Considering then (14), the definition of $\mathbf{x}$ and $x^+$, and the fact that $\Pi \mathbf{1}^N = 0$, it follows that
\[
\frac{\partial V_{\text{NR}}}{\partial x} \psi_{\text{NR}}(x) = \frac{\partial V_{\text{NR}}(x)}{\partial \mathbf{x}} (-\mathbf{x} + \psi(x)) + \eta(x^+)^T (-x^+)
\]
Adding and subtracting $\frac{\partial V_{\text{NR}}(x)}{\partial \mathbf{x}} (\psi(x))$ it then follows that
\[
\frac{\partial V_{\text{NR}}}{\partial x} \psi_{\text{NR}}(x) = \frac{\partial V_{\text{NR}}(x)}{\partial \mathbf{x}} (\psi(x) - \psi(x^+)) + \frac{\partial V_{\text{NR}}(x)}{\partial \mathbf{x}} (\psi(x) - \psi(x^+)) \leq -b_3x^2 - \eta \|x^+\|^2 + b_4|x| a_\psi \|x - x^+\| \leq -b_3x^2 - \eta \|x^+\|^2 + b_4a_\psi \|x\| \|x^+\| \leq -b_3x^2 - \eta \left( \|x\| + \|x^+\| \right)^2 \leq -b_3x^2 + \frac{2}{2N^2} \|x\|^2 + \|x^+\|^2 = -b_7\|x\|^2
\]
where the last inequality is valid for $\eta > \frac{b_3^2a_\psi^2}{4b_3N^2}$.

**Proof of Lemma 7** proof of (23a): notice that $\phi_x(x, \chi)$ is globally defined since $[\cdot]_c$ ensures there is no division by zero.

Also note that, since $\mathbf{h}(x) > cI$ by Assumption 5, then there exists $r > 0$ such that, for $\|x\| + \|\chi\| \leq r$.
\[
\phi_x(x, \chi) = -x - \mathbf{1}_N \otimes x^+ + \chi_\chi + \mathbf{1}_N \otimes (\mathbf{1}(x) + \mathbf{h}(x)x^+) \chi_\chi + \mathbf{1}_N \otimes \mathbf{h}(x)
\]
that implies that $\phi_x$ is locally differentiable in $\|x\| + \|\chi\| \leq r$. In addition to this local differentiability, also observe that
\[
\phi_x(0, 0) = 0, \text{ therefore there must exist } a_1 > 0 \text{ s.t.}
\]
\[
\|\phi_x(x, \chi)\| \leq a_1(\|x\| + \|\chi\|), \quad \forall(\|x\| + \|\chi\|) \leq r.
\]
(34)
Moreover \( \phi_s(x, \chi) \) cannot grow more than linearly globally. In fact,

\[
\| \phi_s(x, \chi) \| \leq \|x\| + N\|x^*\| + \frac{1}{r} \left\| \chi + \mathbb{I} \otimes (\varphi(x) + \overline{h}(x)x^*) \right\|
\]

\[
\leq \|x\| + N\|x^*\| + \frac{1}{r} \left\| \chi + \frac{N}{2} \left( \|\varphi(x)\| + \|x^*\| \|\overline{h}(x)\| \right) \right\|
\]

\[
\leq \|x\| + N\|x^*\| + \frac{1}{r} \left\| \chi + \frac{N}{2} a_5 \|x\| + \frac{N}{2} a_6 \|x^*\| \right\|
\]

\[
\leq a_2 + a_3 (\|x\| + \|\chi\|), \quad \forall x, \chi
\]

where we used Assumption 5 and where \( a_2, a_3 \) are suitable positive scalars. In particular inequality (35) is valid for \( (\|x\| + \|\chi\|) > r \). Inequalities (34) and (35) define two cones, one affine (shifted by \( a_2 \)) and one proper. Therefore, it is possible to find a cone that guarantees (34) for \( (\|x\| + \|\chi\|) \leq r \) and (35) for \( (\|x\| + \|\chi\|) > r \) simultaneously. Simple geometrical arguments imply then that

\[
\| \phi_s(x, \chi) \| \leq a_4 (\|x\| + \|\chi\|) \quad \forall x, \chi
\]

where

\[
a_4 := \max \left\{ a_1, \frac{a_2 + a_3 r}{r} \right\}.
\]

Proof of (23b): Let \( \Delta(x, \chi) := \phi_s(x, \chi) - \phi_{\text{PSR}}(x) \), with \( \phi_{\text{PSR}} \) as in (14). Then there exists a positive scalar \( r > 0 \) such that, for all \( \|\chi\| + \|x\| \leq r \),

\[
\Delta(x, \chi) = -(\chi_x + 1_N \otimes \overline{h}(x)) \frac{\chi_x + 1_N \otimes \overline{h}(x)}{\chi_x + 1_N \otimes \overline{h}(x)} - 1_N \otimes \psi(x)
\]

\[
= \chi_x + 1_N \otimes \overline{h}(x) - 1_N \otimes \varphi(x) - 1_N \otimes \overline{h}(x)
\]

which implies that the function is differentiable for \( \|\chi\| + \|x\| \leq r \). Moreover, since \( \Delta(x, 0) = 0 \), then there exists a positive scalar \( a_4 > 0 \) s.t.

\[
\| \Delta_s(x, \chi) \| \leq a_4 \| \chi \| \quad \| \chi \| + \| x \| \leq r.
\]

By using (16a) and (16b) we can then show that \( \Delta_s(x, \chi) \) cannot grow more than linearly in the variable \( \chi \), since

\[
\| \Delta_s(x, \chi) \| = \frac{\chi_x + 1_N \otimes \overline{h}(x)}{\chi_x + 1_N \otimes \overline{h}(x)} - 1_N \otimes \left( x^* + \frac{\varphi(x)}{\overline{h}(x)} \right)
\]

\[
\leq \frac{1}{r} (\|\chi\| + 2N\|\varphi(x)\| + N\|x^*\| \|\overline{h}(x)\| + N\|x^*\|)
\]

\[
\leq a_5 + a_6 \| \chi \|, \quad \forall x, \chi
\]

for suitable positive scalars \( a_5 \) and \( a_6 \). Repeating the same geometrical arguments used above we then obtain

\[
\| \Delta_s(x, \chi) \| \leq a_\Delta \| \chi \|, \quad \forall x, \chi
\]

with

\[
a_\Delta := \max \left\{ a_3, \frac{a_5 + a_6 r}{r} \right\}.
\]

Proof of (Theorem 8) For notational brevity we omit the dependence on \( \xi \), i.e., let \( x^{eq} = x^{eq}(\xi) \) and \( x^{eq} = x^{eq}(\xi) \).

Proof of a): assume that such \( x^{eq}(\xi) \) exists for \( \|\xi\| \leq r \). If \( r \) is sufficiently small, then, since \( \overline{h}(x) > cI \) by Assumption 1,

\[
[\xi_x + 1_N \otimes \overline{h}(x)] = \xi_x + 1_N \otimes \overline{h}(x) = 1_N \otimes (\overline{h}(x) + \xi_x).
\]
This implies that for \( ||\xi|| \leq r \) we have

\[
\phi_x(x^{eq}, \xi) = -x - I_N \otimes (x^* - (\xi_* + \overline{t}(x))^{-1}(\xi_* + \overline{g}(x) + \overline{h}(x)x^*))
\]

Therefore

\[
\phi_x(x^{eq}, \xi) = 0 \iff x_i = -x^* + (\xi_* + \overline{h}(x))^{-1}(\xi_* + \overline{g}(x) + \overline{h}(x)x^*), \forall i
\]

which implies that solution must satisfy \( x^{eq} = 1 \otimes x^{eq} \).

**Proof of b)**: if we substitute the necessary condition \( x^{eq} = 1 \otimes x^{eq} \) into the definition of \( \phi_x(x^{eq}, \xi) \), we obtain that

\[
x^{eq} + x^* = (\overline{h}(x^{eq}) + \xi_*^{-1}(\overline{g}(x^{eq}) + \overline{h}(x^{eq})x^*)
\]

where we used properties (13a) and (13b) that lead to \( \overline{h}(I_N \otimes x) = \overline{h}(x) \) and \( \overline{g}(I_N \otimes x) = \overline{g}(x) \).

Moreover, Assumption 5 ensures that \( \overline{h}(x^*) > cI \). Thus, for the continuity assumptions in Assumption 1, there exists a sufficiently small \( r > 0 \) s.t. \( ||\xi|| \leq ||\xi|| \leq r \) then \( \overline{h}(x^*) + \xi_* \) is still invertible. Therefore

\[
\overline{f}(x^{eq}) + \xi_* + \overline{h}(x^{eq})x^* = \overline{h}(x^{eq})x^{eq} + x^* + \xi_* (x^{eq} + x^*)
\]

Exploiting now the equivalence \( \overline{f}(x^{eq}) = \overline{h}(x^{eq})x^{eq} - \nabla \overline{f}(x^{eq}) \), it follows that \( x^{eq} \) must satisfy the following condition:

\[
\nabla \overline{f}(x^{eq}) - \xi_* + \xi_* (x^{eq} + x^*) = 0
\]

Given Assumption 1, the left-hand side of the previous inequality is a continuously differentiable function, since

\[
\frac{\partial (\nabla \overline{f}(x^{eq}) - \xi_* + \xi_* (x^{eq} + x^*))}{\partial x^{eq}} = \nabla^2 \overline{f}(x^{eq}) + \xi_*
\]

Notice moreover that if \( r \) is sufficiently small (i.e., \( ||\xi|| \) is sufficiently small) then the differentiation is an invertible matrix, since once again \( \nabla^2 \overline{f}(x^*) > cI \) by assumption. Therefore, by the Implicit Function Theorem, \( x^{eq}(\xi) \) exists, is unique and continuously differentiable.

**Proof of Theorem 10**

**Proof of a)**: \( V_{PNR}(0) = 0 \) and \( V_{PNR}(x) \geq 0 \) for \( x \neq 0 \) have been proved before. \( V_{PNR} < 0 \) is instead proved by proving (27a).

**Proof of b)**: as for (27a), consider that, \( \forall x \in \mathbb{R}^n \),

\[
\frac{\partial V_{PNR}}{\partial x} \phi'_x(x, \xi) = \frac{\partial V_{PNR}}{\partial x} \phi'_x(x, 0) + \frac{\partial V_{PNR}}{\partial x} \phi'_x(x, \xi) - \phi'_x(x, 0)
\]

\[
\leq \frac{\partial V_{PNR}}{\partial x} \phi'_x(x, \xi) + ||\phi'_x(x, \xi) - \phi'_x(x, 0)||
\]

\[
\leq -b_7 ||x||^2 + b_6 ||x||^2 ||\xi|| ||x||
\]

\[
\leq -(b_7 - b_6 \alpha \xi) ||x||^2 \leq -b_7 ||x||^2.
\]

Notice that this inequality is meaningful for \( r < \frac{b_7}{b_6 \alpha \xi} \).

As for (27b), consider that, \( \forall x \in \mathbb{R}^n \),

\[
||\phi'_x(x, \xi)|| = ||\phi'_x(x, 0)|| + ||\phi'_x(x, \xi) - \phi'_x(x, 0)||
\]

\[
\leq (b_8 + a \xi) ||x|| \leq b_8 ||x||.
\]
Proof Since w.l.o.g. $\sum_i A_id_i = 0$ and $\sum_i (d_i^2 A_i d_i + e_i) = 0$, it follows that $\overline{f}(x) = x^T A x$. Clearly $\overline{f}(x)$ satisfies Assumption 1 since $\nabla^2 \overline{f}(x) = A > 0$ is independent of $x$. Considering then $h_i(x) = \nabla^2 f_i(x) = A_i$ it follows

$$\overline{f}(x) = A$$

$$\overline{g}_j(x) = A_i x - A_i (x - d_i) = A_id_i$$

$$g(x) - g(x') = 0$$

$$\overline{g}(x) = \frac{1}{N} \sum_i A_i d_i = 0$$

$$h(x) - h(x') = 0$$

$$\overline{h}(x) = \frac{1}{N} \sum_i A_i = A$$

$$\phi(x) = \overline{h}^{-1}(x) \overline{g}(x) = 0$$

$$x^{eq}(\xi) = (A + \xi z)^{-1} \xi y$$

$$\phi'_\nu(x, \xi) = \phi'_\nu(x, 0) = -x$$

and thus also the other assumptions are satisfied.

Proof (of Theorem 12) The proof considers the system as an autonomous singularly perturbed system, and proceeds as follows:

a) show that $x^*$ is an equilibrium; b) perform a change of variables; c) construct a Lyapunov function for the boundary layer system; d) construct a Lyapunov function for the reduced system; e) join the two Lyapunov functions into one, and show (by cascading the previously introduced Lemmas and Theorems) that the complete system (39) converges to $x^*$ while satisfying the hypotheses of Theorem 2. By doing so it follows that (38), i.e., Algorithm 1, is exponentially stable.

For notational simplicity we let $x^* := I_N \otimes x^*$. We also use all the notation collected in Section II.

- **Discrete to continuos dynamics** The dynamics of Algorithm 1 can be written in state space as

$$\begin{cases}
    u(k) = g(x(k-1)) \\
    w(k) = h(x(k-1)) \\
    y(k) = P y(k-1) + g(x(k-1)) - v(k-1) \\
    z(k) = P z(k-1) + h(x(k-1)) - w(k-1) \\
    x(k) = (1 - \varepsilon) x(k-1) + \frac{y(k-1)}{z(k-1)_{eq}}
\end{cases}$$

with suitable initial conditions. (38) can then be interpreted as the forward-Euler discretization of

$$\begin{cases}
    \varepsilon \dot{u}(t) = -u(t) + g(x(t)) \\
    \varepsilon \dot{w}(t) = -w(t) + h(x(t)) \\
    \varepsilon \dot{y}(t) = -Ky(t) + (I - K) [g(x(t)) - u(t)] \\
    \varepsilon \dot{z}(t) = -Kz(t) + (I - K) [h(x(t)) - w(t)] \\
    \dot{x}(t) = -x(t) + \frac{y(t)}{z(t)_{eq}}
\end{cases}$$

with null initial conditions, where $\varepsilon$ is the discretization time interval and $K := I - P$. Notice that, as for $P$, if $n$ is the dimension of the local costs then $P = P' \otimes I_n$ with $P'$ a doubly-stochastic average consensus matrix. Nonetheless for brevity we will omit the superscripts $'$. 

DRAFT
\( b) \) let

\[
\begin{align*}
    v' &:= v - g(x) \\
    w' &:= w - h(x) \\
    y' &:= y - v' - 1_N \otimes \overline{y}(x) \\
    z' &:= z - w' - 1_N \otimes \overline{h}(x) \\
    x' &:= x - x^* \\
\end{align*}
\]

and

\[
\phi_x(x') := \frac{\partial g}{\partial x'} - 1_N \otimes \frac{\partial g}{\partial x'} \\
\phi_y(x') := \frac{\partial h}{\partial x'} - 1_N \otimes \frac{\partial h}{\partial x'} \\
\phi_s(x', \chi) := -x'(t) - x^* + \\
\frac{y'(t) + v'(t) + 1_N \otimes \overline{y}(x'(t)) + x^*}{x'(t) + w'(t) + 1_N \otimes \overline{h}(x(t)) + x^*}
\]

with \( \chi := (v', w', y', z') \), so that (39) becomes

\[
\begin{align*}
    \varepsilon \dot{v}'(t) &= -v'(t) - \varepsilon \frac{\partial g}{\partial x'} x'(t) \\
    \varepsilon \dot{w}'(t) &= -w'(t) - \varepsilon \frac{\partial h}{\partial x'} x'(t) \\
    \varepsilon \dot{y}'(t) &= -K y'(t) + \varepsilon \phi_y(x') x'(t) \\
    \varepsilon \dot{z}'(t) &= -K z'(t) + \varepsilon \phi_s(x') x'(t) \\
    \dot{x}'(t) &= \phi_x(x', \chi)
\end{align*}
\]

with initial conditions

\[
\begin{align*}
    v'(0) &= v(0) - g(x(0)) \\
    w'(0) &= w(0) - h(x(0)) \\
    y'(0) &= y(0) - v(0) + g^+(x(0)) \\
    z'(0) &= z(0) - w(0) + h^+(x(0)) \\
    x'(0) &= x(0) - x^*
\end{align*}
\]

where \( g^+(x) := g(x) - 1_N \otimes \overline{y}(x) \) (equivalent definition for \( h^+ \)). Notice that (40) has the origin as an equilibrium point.

Moreover this dynamics exploits the function \( \phi_s \) defined in (21), with \( \chi_v = y' + v' \), and \( \chi_z = z' + w' \).

The next step is to exploit the structure of \( K \) (more precisely, the fact that it contains an average consensus matrix) to reduce the dynamics, i.e., to eliminate the dynamics of the average since the latter does not change in time. To this aim, we analyze the behavior of the average of the \( y \)'s, i.e., the behavior of \( (1^T_N \otimes I_n) y' \). To this point, consider the third equation in (40). Recalling that \( (A \otimes B)(C \otimes D) = AB \otimes CD \), and exploiting the fact that \( 1^T_N P' = 0 \), we notice that \( (1^T_N \otimes I_n) K = 0 \).

Moreover, from the definitions of \( g \) and \( \overline{y} \),

\[
\left(1^T_N \otimes I_n\right) \frac{\partial g(x')}{\partial x'} = N \frac{\partial \overline{y}(x')}{\partial x'}.
\]

Since \( N = 1^T_N 1_N \), it follows also that

\[
\left(1^T_N \otimes I_n\right) \phi_x(x') = 0
\]

for all \( t \geq 0 \), i.e., \( 1^T y'(t) = 1^T y'(0) = 0 \). Similarly it is possible to show that \( z'(t) = 0 \). This eventually implies that

\[
y'^{(0)}(t) = 0 \quad z'^{(0)}(t) = 0 \quad \forall t
\]
that means, recalling that $y' = y^\parallel + y^\perp$ and $z' = z^\parallel + z^\perp$, that (40) can be equivalently rewritten as

$$
\begin{align*}
\varepsilon \dot{y}'(t) &= -\dot{y}'(t) - \varepsilon \frac{\partial y}{\partial \chi} \phi_\varepsilon(x', \chi') \\
\varepsilon \dot{w}'(t) &= -\dot{w}'(t) - \varepsilon \frac{\partial w}{\partial \chi} \phi_\varepsilon(x', \chi') \\
\varepsilon \dot{y}'^\perp(t) &= -K y^\perp(t) + \varepsilon \phi_\varepsilon(x') \phi_\varepsilon(x', \chi') \\
\varepsilon \dot{z}'^\perp(t) &= -K z^\perp(t) + \varepsilon \phi_\varepsilon(x') \phi_\varepsilon(x', \chi')
\end{align*}
$$

(41)

where now $\chi' := (v, w, y'^\perp, z'^\perp)$ and where the novel initial conditions for the changed variables are

$$
\begin{align*}
y'^\perp(0) &= y^\perp(0) - v^\perp(0) + g^\perp(x(0)) \\
z'^\perp(0) &= z^\perp(0) - w^\perp(0) + h^\perp(x(0))
\end{align*}
$$

• $c)$ the boundary layer of (41) is computed by setting $x'(t) = \tilde{x}$. Since a constant $x'$ implies $\dot{x}' = \phi_\varepsilon = 0$, this boundary layer reduces to a linear system globally exponentially converging to the origin. Notice that this implies that, in the original coordinates system,

$$
v = g(x), \quad w = h(x), \quad y = \mathbbm{1}_N \otimes g(x), \quad z = \mathbbm{1}_N \otimes h(x).
$$

In the novel coordinates system we thus consider, as a Lyapunov function, $\frac{1}{2} ||\chi'||^2$.

• $d)$ the reduced system of (41) is computed by plugging $v'(t) = 0$, $w'(t) = 0$, $y'^\perp(t) = 0$, $z'^\perp(t) = 0$, i.e., $\chi' = 0$ into the equations. Defining

$$
f_i^*(x') := f_i(x' + x^*), \quad h_i^*(x') := h_i(x' + x^*),
$$

we obtain

$$
\dot{x}'(t) = -x'(t) - x^* + \mathbbm{1}_N \otimes \nabla^* f_i(x'(t)) \nabla h_i(x'(t))
\quad - \nabla^* f_i(x^* - x')
$$

where $\psi^*$ and $\phi_\varepsilon^*$ are the starred versions of $\psi$ and $\phi_\varepsilon$ in (12) and (14), respectively. Thus the reduced system, thanks to Theorem 6, admits $x^*$ as a global exponentially stable equilibrium, and admits $V_{\varepsilon}^*$ in (18) as a Lyapunov function.

• $e)$ we now notice that the interconnection of the boundary layer and reduced systems maintains the global stability, since their Lyapunov functions are quadratic type. Thus (see [44, pp. 453]) the global system is asymptotically globally stable. To check that forward-Euler discretizations of the system preserve these stability properties we then consider as a global Lyapunov function the function

$$
V(x', \chi') = (1 - d) V_{\varepsilon}^*(x') + d \frac{1}{2} ||\chi'||^2,
$$

that is clearly positive definite for every $d \in (0, 1)$, and prove that inequalities (4) of Theorem 2 are satisfied.

**proof that (4a) holds:** from (19a) and the structure of $V$ it follows immediately that

$$
((1 - d) b_1 + d) I \leq \nabla^2 V(x', \chi') \leq ((1 - d) b_6 + d) I.
$$
proof that (4c) holds: applying (17) and (23a) to (41) it follows that (4c) holds with

\[ a_4 = a V := \max \{1 + 2\varepsilon a_y a_z, 1 + 2\varepsilon a_h a_z, a_z\}. \]

proof that (4b) holds: the part relative to the slow dynamics is already characterized by (27a). For the part relative to the fast dynamics, since \( \frac{\partial}{\partial x} \| \chi \|_2 = \chi^T \) to check that (4b) corresponds to check the negativity of the terms

\[
-v'^T v' - \varepsilon v'^T \frac{\partial g}{\partial x} (x', \chi') \\
-w'^T w' - \varepsilon w'^T \frac{\partial h}{\partial x} (x', \chi') \\
-(y'^T)^T K y' + \varepsilon (y'^T)^T \phi_y (x') \phi_x (x', \chi') \\
-(z'^T)^T K z' + \varepsilon (z'^T)^T \phi_h (x') \phi_x (x', \chi')
\]

These terms can then be majorized using (17) and (23a). E.g., the third term can be majorized with

\[
-\sigma(P) \| y' \|^2 + 2\varepsilon a_y a_z \| y' \| (\| x \| + \| \chi \|)
\]

where \( \sigma(P) \) is the spectral gap of \( P \). Applying similar concepts also to the other terms it follows that (4b) holds with

\[ a_3 = \min \{ \sigma(P) - 2\varepsilon a_y a_z, \sigma(P) - 2\varepsilon a_h a_z \}. \]

---

1. Proof (of Theorem 13) The proof is identical to the one of Theorem 12 with the exception that the substitution is now \( x'' = x - x^* - \mathbb{1}_N \otimes \psi(\xi_y, \xi_z) \). Indeed one can prove the stability of the novel system using the same Lyapunov function of Theorem 12. Notice that we are ensured that there exists a sufficiently small neighborhood of the origin for which the function \( \psi \) exists due to the smoothness conditions in Assumption 1.

2. Proof (of Theorem 12) The proof is the local version of the one in Theorem 13. Indeed the local versions of Assumptions 1, 5 and 9 always hold, i.e., they hold when considering \( x \) s.t. \( \| x \| \leq r' \), and one can thus repeat that reasonings using local perspectives.

3. Proof (of Theorem 15) Consider for simplicity the scalar case. Let \( y^* := \sum_i A_i d_i \) and \( z^* := \sum_i A_i \), so that \( x^* = y^* / z^* \). Since \( y(k+1) = P y(k) \) and \( z(k+1) = P z(k) \), given the assumptions on \( P \), there exist positive \( \alpha_y, \alpha_z \) independent of \( x(0) \) s.t. \( |y(k) - y^*| \leq \alpha_y (\rho(P))^k \) and \( |z(k) - z^*| \leq \alpha_z (\rho(P))^k \). The claim thus follows considering that \( x_i(k) = \frac{y_i(k)}{z_i(k)} \), and that, since the elements of \( P \) are non negative, all the \( z_i(k) \) are non smaller than \( e \) for all \( k \geq 0 \) (i.e., the operator \( [\cdot]_e \) is always performing as the identity operator).