An $O(1/k)$ Gradient Method for Network Resource Allocation Problems

Amir Beck, Angelia Nedić, Asuman Ozdaglar, and Marc Teboulle

***Abstract*—We present a fast distributed gradient method for a convex optimization problem with linear inequalities, with a particular focus on the network utility maximization (NUM) problem. Most existing works in the literature use (sub)gradient methods for solving the dual of this problem which can be implemented in a distributed manner. However, these (sub)gradient methods suffer from an $O(1/\sqrt{k})$ rate of convergence (where k is the number of iterations). In this paper, we assume that the utility functions are strongly concave, an assumption satisfied by most standard utility functions considered in the literature. We develop a completely distributed fast gradient method for solving the dual of the NUM problem. We show that the generated primal sequences converge to the unique optimal solution of the NUM problem at rate $O(1/k)$.**

***Index Terms*—Gradient methods, convex functions, network utility maximization.**

I. Introduction

THE unprecedented growth in the scale of communication networks has increased the importance and urgency of efficient scalable and decentralized algorithms for the allocation of resources in such networks. Optimization formulations of the corresponding resource allocation problem provide a powerful approach as exemplified by the canonical *network utility maximization* (NUM) problem proposed in [7] (see also [4], [11], and [24]). NUM problems are characterized by a fixed network and a set of sources, which send information over the network along a predetermined set of links. Each source has a local utility function of the rate at which it sends information. The goal is to determine the source rates that maximize the sum of utilities subject to link capacity constraints.

Existing work has exploited the convexity of the NUM formulation, resulting from the concavity of the utility functions and the linearity of the capacity constraints, to derive a decentralized algorithm using a dual-based (sub)gradient method with convergence rate of $O(1/\sqrt{k})$, where k is the number of iterations. Although this approach has proved to be rich and useful both in theory and practice, our starting point in this paper is that in most practically relevant cases, a significant improvement is possible. The reason for this is that in most such applications, utility functions are not just concave but also *strongly concave*. An important implication of this property is that the dual function is not only differentiable but also has a Lipschitz continuous gradient enabling the use of fast gradient methods (FGMs) with much improved convergence rate. For some very recent works in that direction, see, e.g., [13] and [22].

In this paper, we derive a decentralized fast dual gradient algorithm for the NUM problem and investigate its implications for the resulting generated primal solutions. Our analysis considers a more general convex optimization problem with linear constraints given by

$$(P) \qquad g_{\rm opt} = \max \quad g(\boldsymbol{x}) \quad \text{s.t.} \quad \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{c}, \boldsymbol{x} \in X$$

where $\boldsymbol{A}$ is an $m \times n$ matrix, $X \subseteq \mathbb{R}^n$ is a closed convex set, and g is a strongly concave function over X with a parameter $\sigma > 0$. Under the assumption that each utility function u_i is strongly concave over a compact interval $I_i = [0, M_i]$ (where M_i is the maximum allowed rate for source i), the NUM problem is a special case of this problem with $g(\boldsymbol{x}) = \sum_{i \in \mathcal{S}} u_i(x_i)$, where $\mathcal{S}$ is the set of sources and $X = \Pi_{i \in \mathcal{S}} I_i$. Standard utility functions considered in the literature such as the α-fair utility functions (see [12]) satisfy the strong concavity assumption over the compact interval $I_i = [0, M_i]$.

Under a mild condition, i.e., Slater's condition, strong duality holds and we can solve problem (P) through the use of its dual. We first show that the dual problem of problem (P) can be expressed in terms of conjugate function of the primal objective function $g(\boldsymbol{x})$. We then use an important equivalence relation between the differentiability of a convex function and the strong convexity of its conjugate. The equivalence relation enables us to establish that the gradient mapping of the dual function is Lipschitz continuous, thus allowing us to apply an FGM [17] with rate $O(1/k^2)$ to the dual problem. We show that the primal sequence generated by the method converge to the unique optimal solution of problem (P) at rate of $O(1/k)$. We also show that the primal infeasibility converges to 0, and that the objective function value converges to the optimal value at a rate of $O(1/k)$. Our algorithm and results are different from those obtained in a recent paper [13], where more general nonlinear (convex) constraints have been considered. In particular, in [13], a different fast gradient method has been proposed with the convergence rate of $O(1/k^2)$ for a given target level of accuracy in computing the optimal function value, and the primal

Manuscript received July 17, 2013; accepted December 21, 2013. Date of publication March 5, 2014; date of current version April 9, 2014. The work was supported by the BSF Grant #2008100. Recommended by Associate Editor A. Jadbabaie.

A. Beck is with the Faculty of Industrial Engineering and Management, Technion—Israel Institute of Technology, Haifa 32000, Israel.

A. Nedić is with the Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, Champaign, IL 61801, USA (e-mail: angelia@illinois.edu).

A. Ozdaglar is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA.

M. Teboulle is with the School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv 69978, Israel.

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

Digital Object Identifier 10.1109/TCNS.2014.2309751

infeasibility decrease rate of $O(1/\sqrt{k})$. In contrast, our convergence rate result for the primal infeasibility is $O(1/k)$.

We demonstrate that a direct application of the fast method to the NUM problem will require a centralized implementation since the stepsize needed to ensure convergence (which is a function of the Lipschitz constant of the gradient mapping of the dual function) relies on global information. We therefore develop a scaled version of the FGM in which each variable uses a different stepsize that depends on local information only, enabling the method to be implemented in a distributed manner while retaining the $O(1/k^2)$ rate of convergence of the dual sequence.

Our paper is also related to the recent literature on distributed second-order methods for solving network flow and NUM problems (see [6] and [25]–[29]). Similar to the stepsize rule used by the fast weighted gradient projection method used in this paper, the recent paper [29] has presented a distributed backtracking stepsize rule that involves each variable using a different stepsize that can be computed using local information. Although these methods provide superlinear convergence in outer iterations, they involve inexact computations in the inner loop at each iteration and, therefore, can only guarantee convergence to a neighborhood of the optimal solution. This is in contrast with the exact convergence results presented in this paper.

The paper is organized as follows. Section II contains the formulations of the NUM problem and its dual, and presents a dual-based gradient method for this problem. An FGM is discussed in Section III, together with its fully distributed implementation. Section IV presents our simulation setting and reports our numerical results, and Section V provides some concluding remarks.

A. Notation, Terminology, and Basics

We view a vector as a column vector, and we denote by $\boldsymbol{x}^T\boldsymbol{y}$ the inner product of two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. We use $\|\boldsymbol{y}\|_2$ to denote the standard Euclidean norm (or l_2 norm), $\|\boldsymbol{y}\|_2 = \sqrt{\boldsymbol{y}^T\boldsymbol{y}}$ (we drop the subscript and refer to it as $\|\boldsymbol{y}\|$ whenever it is clear from the context). Occasionally, we also use the standard l_1 norm and l_∞ norm denoted, respectively, by $\|\boldsymbol{y}\|_1$ and $\|\boldsymbol{y}\|_\infty$, i.e., $\|\boldsymbol{y}\|_1 = \sum_i |y_i|$ and $\|\boldsymbol{y}\|_\infty = \max_i |y_i|$. For an $m \times n$ matrix $\boldsymbol{M}$, we use the following induced matrix norm: given any vector norm $\|\cdot\|$, the corresponding induced matrix norm, also denoted by $\|\cdot\|$, is defined by

$$\|\boldsymbol{M}\| = \max\{\|\boldsymbol{M}\boldsymbol{x}\| : \|x\| = 1\|\}.$$

We next list some standard properties of the induced norm which will be used in our analysis (see [5, Sec. V-F] for more details).

Lemma I.1: Given any vector norm $\|\cdot\|$ and the induced matrix norm, we have: 1) $\|\boldsymbol{M}\boldsymbol{x}\| \le \|\boldsymbol{M}\|\|\boldsymbol{x}\|$ for all $m \times n$ matrices $\boldsymbol{M}$ and all vectors $\boldsymbol{x} \in \mathbb{R}^n$, and $\|\boldsymbol{N}\boldsymbol{M}\| \le \|\boldsymbol{N}\|\|\boldsymbol{M}\|$ for all matrices $\boldsymbol{N}$ and $\boldsymbol{M}$ (with proper dimensions); and 2) $\rho(\boldsymbol{M}^T\boldsymbol{M}) \le \|\boldsymbol{M}^T\boldsymbol{M}\|$, where $\rho(\boldsymbol{M}^T\boldsymbol{M})$ is the spectral radius of matrix $\boldsymbol{M}^T\boldsymbol{M}$ (i.e., the maximum of the magnitudes of the eigenvalues of $\boldsymbol{M}^T\boldsymbol{M}$).

Moreover, $\|\boldsymbol{M}\|_2 = \|\boldsymbol{M}^T\|_2$, $\|\boldsymbol{M}\|_1 = \|\boldsymbol{M}^T\|_\infty$, and $\|\boldsymbol{M}\|_2^2 = \rho(\boldsymbol{M}^T\boldsymbol{M})$.

For a concave function $g : \mathbb{R}^n \to [-\infty, \infty)$, we denote the domain of g by $\mathrm{dom}(g)$, where

$$\mathrm{dom}(g) = \{\boldsymbol{x} \in \mathbb{R}^n | g(\boldsymbol{x}) > -\infty\}.$$

We say that $\boldsymbol{d} \in \mathbb{R}^n$ *is a subgradient of a concave function* $g(\boldsymbol{x})$ *at a given vector* $\bar{\boldsymbol{x}} \in \mathrm{dom}(g)$ if the following relation holds:

$$g(\bar{\boldsymbol{x}}) + \boldsymbol{d}^T(\boldsymbol{x} - \bar{\boldsymbol{x}}) \ge g(\boldsymbol{x}), \quad \text{for all } \boldsymbol{x} \in \mathrm{dom}(g). \tag{1}$$

The set of all subgradients of g at $\bar{\boldsymbol{x}}$ is denoted by $\partial g(\bar{\boldsymbol{x}})$.

Given a nonempty convex set $C \subseteq \mathbb{R}^n$, a function $g : C \to \mathbb{R}$ is said to be strongly concave over C with a parameter $\sigma > 0$ (in a norm $\|\cdot\|$) if for all $\boldsymbol{x}, \boldsymbol{y} \in C$ and all $\gamma \in [0, 1]$

$$g(\gamma\boldsymbol{x} + (1-\gamma)\boldsymbol{y}) \ge \gamma g(\boldsymbol{x}) + (1-\gamma)g(\boldsymbol{y}) + \gamma(1-\gamma)\frac{\sigma}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2.$$

We will use the following equivalent characterization of a strongly concave function in our analysis: a function $g : C \to \mathbb{R}$ is strongly concave over C with parameter $\sigma > 0$ if and only if for all $\boldsymbol{x}, \boldsymbol{y} \in C$ and all $\boldsymbol{d} \in \partial g(\boldsymbol{y})$

$$g(\boldsymbol{x}) \le g(\boldsymbol{y}) + \boldsymbol{d}^T(\boldsymbol{x} - \boldsymbol{y}) - \frac{\sigma}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2. \tag{2}$$

For a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with Lipschitz gradients L_h, we have the so-called descent Lemma (see, e.g., [3]): for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$

$$h(\boldsymbol{x}) \le h(\boldsymbol{y}) + \langle \boldsymbol{x} - \boldsymbol{y}, \nabla h(\boldsymbol{y})\rangle + \frac{L_h}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2. \tag{3}$$

II. NUM Problem

Consider a network consisting of a finite set $\mathcal{S}$ of sources and a finite set $\mathcal{L}$ of undirected capacitated links, where a link l has capacity $c_l \ge 0$. Let $\mathcal{L}(i) \subseteq \mathcal{L}$ denote the set of links used by source i, and let $\mathcal{S}(l) = \{i \in \mathcal{S} \mid l \in \mathcal{L}(i)\}$ denote the set of sources that use link l.

Each source i is associated with a utility function $u_i : [0, \infty) \to [0, \infty)$, i.e., each source i gains a utility $u_i(x_i)$ when it sends data at rate x_i. We further assume that the rate x_i is constrained to lie in the interval $I_i = [0, M_i]$ for all $i \in \mathcal{S}$, where the scalar M_i denotes the maximum allowed rate for source i. We adopt the following assumption on the source utility functions.

Assumption 1: For each i, the function $u_i : [0, \infty) \to [0, \infty)$ is continuous, increasing, and strongly concave over the interval $I_i = [0, M_i]$.

The goal of the network utility maximization problem (abbreviated NUM), first proposed in [7] (see also [11] and [24]), is to allocate the source rates as the optimal solution of the following problem:

$$\begin{aligned} \max \quad & g_{\mathrm{N}}(\boldsymbol{x}) \equiv \sum\nolimits_{i\in\mathcal{S}} u_i(x_i) \\ \text{s.t.} \quad & \sum\nolimits_{i\in\mathcal{S}(l)} x_i \le c_l, \quad \text{for all } l \in \mathcal{L} \\ & x_i \in I_i, \quad \text{for all } i \in \mathcal{S}. \end{aligned}$$

Let us consider the $|\mathcal{L}| \times |\mathcal{S}|$ network matrix $\boldsymbol{A}$ with entries given by

$$A_{li} = \begin{cases} 1, & l \in \mathcal{L}(i) \\ 0, & \text{else.} \end{cases} \tag{4}$$

By letting $\boldsymbol{x} = (x_1, \ldots, x_{|\mathcal{S}|})^T$ and $\boldsymbol{c} = (c_1, \ldots, c_{|\mathcal{L}|})^T$, the problem can be compactly represented as

$$\text{(N-P)} \quad \begin{array}{ll} \max & g_{\mathrm{N}}(\boldsymbol{x}) = \sum_{i \in \mathcal{S}} u_i(x_i) \\ \text{s.t.} & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{c}, x_i \in I_i, \text{ for all } i \in \mathcal{S}. \end{array}$$

In our analysis, we will also consider a more general model of a linearly constrained maximization problem

$$\text{(P)} \quad \begin{array}{lll} \mathrm{g}_{\mathrm{opt}} = & \max & \mathrm{g}(\boldsymbol{x}) \\ & \text{s.t.} & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{c}, \boldsymbol{x} \in \mathrm{X} \end{array}$$

where $\boldsymbol{A}$ is an $m \times n$ matrix, the set $X \subseteq \mathbb{R}^n$ is closed and convex, and the function g is a strongly concave over X with a parameter $\sigma > 0$ in the Euclidean norm.

Problem (P), as a problem of maximizing a concave function over a convex set, is a convex problem. Moreover, by the strong concavity assumption on the function g, problem (P), whenever feasible, has a unique solution, denoted by $\boldsymbol{x}^*$. Problem (N-P) obviously fits into the general model (P) with $g(\boldsymbol{x}) = g_{\mathrm{N}}(\boldsymbol{x}) = \sum_{i \in \mathcal{S}} u_i(x_i)$ and $X = \Pi_{i \in \mathcal{S}} I_i$ and $g_{\mathrm{N}}(\boldsymbol{x})$ a strongly concave function over X with the constant $\sigma = \min_{i \in \mathcal{S}} \sigma_i$.

A. The Dual of (P) and Its Properties

We will assume that Slater's condition is satisfied.

Assumption 2: There exists a vector $\tilde{\boldsymbol{x}}$ in the relative interior of set X such that $\boldsymbol{A}\tilde{\boldsymbol{x}} \leq \boldsymbol{c}$.

It is well-known (see [19]) that, under Assumption 2, strong duality holds for problem (P). We let $\tilde{g}$ denote the extended-valued function associated with the objective function g and the set X, which is given by

$$\tilde{g}(\boldsymbol{x}) = \begin{cases} g(\boldsymbol{x}), & \boldsymbol{x} \in X \\ \infty, & \text{else.} \end{cases}$$

In what follows, we also use the notion of the conjugate of an extended-valued function h given by

$$h^*(\boldsymbol{y}) = \sup_{\mathbf{x}} \{\boldsymbol{x}^T \boldsymbol{y} - h(\boldsymbol{x})\}.$$

Equipped with the above notations, we can write the dual objective function of (P) as

$$\begin{aligned} q(\boldsymbol{\lambda}) &= \max_{x \in X} \{g(\boldsymbol{x}) - \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{c})\} \\ &= (-\tilde{g})^*(-\boldsymbol{A}^T\boldsymbol{\lambda}) + \boldsymbol{\lambda}^T\boldsymbol{c} \end{aligned} \tag{5}$$

for every $\boldsymbol{\lambda} \in \mathbb{R}_+^{|\mathcal{L}|}$. The dual problem is given by

$$\text{(D)} \quad \begin{array}{lll} q_{\mathrm{opt}} = & \min & (-\tilde{g})^*(-\boldsymbol{A}^T\boldsymbol{\lambda}) + \boldsymbol{\lambda}^T\boldsymbol{c} \\ & \text{s.t.} & \boldsymbol{\lambda} \geq \boldsymbol{0}. \end{array}$$

By the strong duality property that holds for the pair (P) and (D), we have $g_{\mathrm{opt}} = q_{\mathrm{opt}}$.

Recall that by duality theory (see, e.g., [3]), the dual objective function q is in fact differentiable (by the strong concavity of the primal) and its gradient is given by

$$\nabla q(\boldsymbol{\lambda}) = -(\boldsymbol{A}\boldsymbol{x}(\boldsymbol{\lambda}) - \boldsymbol{c}) \tag{6}$$

where the unique maximizer $\boldsymbol{x}(\boldsymbol{\lambda})$ is given by

$$x(\boldsymbol{\lambda}) = \arg\max_{x \in X} \{g(\boldsymbol{x}) - \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{c})\}.$$

We use the important equivalence between the differentiability of a convex function and the strong convexity of its conjugate, see [20, Proposition 12.60, p. 565].

Lemma II.1: Let $h : \mathbb{E} \to (-\infty, \infty]$ be a proper, lower semicontinuous, and convex function, and let $\sigma > 0$. The following statements are equivalent.

1) The function h is differentiable and its gradient mapping ∇h is Lipschitz continuous in some norm $\|\cdot\|_{\mathbb{E}}$ with constant $\frac{1}{\sigma}$.
2) The conjugate function $h^* : \mathbb{E}^* \to (-\infty, \infty]$ is σ-strongly convex with respect to the dual norm $\|\cdot\|_{\mathbb{E}}^*$.

We work with the Euclidean norm, which coincides with its dual norm, and the function $-\tilde{g}$ is σ-strongly convex in this norm, since $-g$ is σ-strongly convex. Coming back to the NUM problem (N-P), we can exploit the special structure of the objective function to obtain

$$\begin{aligned} (-\tilde{g}_{\mathrm{N}})^*(-\boldsymbol{A}^T\boldsymbol{\lambda}) &= \sum_{i \in \mathcal{S}} (-\tilde{u}_i)^*(-(\boldsymbol{A}^T\boldsymbol{\lambda})_i) \\ &= \sum_{i \in \mathcal{S}} (-\tilde{u}_i)^*(-\pi_i(\boldsymbol{\lambda})) \end{aligned}$$

where $\pi_i(\boldsymbol{\lambda}) = \sum_{l \in \mathcal{L}(i)} \lambda_l$ and $\tilde{u}_i$ is the extended-valued function associated with the function u_i and the set I_i

$$\tilde{u}_i(x) = \begin{cases} u_i(x), & x \in I_i \\ \infty, & \text{else.} \end{cases}$$

Consequently, the dual problem of the NUM problem (N–P) is given by

$$\text{(N-D)} \quad \begin{array}{ll} \min & q_{\mathrm{N}}(\boldsymbol{\lambda}) \equiv \sum_{i \in \mathcal{S}} (-\tilde{u}_i)^*(-(\boldsymbol{A}^T\boldsymbol{\lambda})_i) + \boldsymbol{c}^T\boldsymbol{\lambda} \\ \text{s.t.} & \boldsymbol{\lambda} \geq 0. \end{array}$$

Moreover, as recorded before, with this special choice of g, the resulting dual objective function q_{N} is differentiable and its gradient is given by

$$\nabla q_{\mathrm{N}}(\boldsymbol{\lambda}) = -(\boldsymbol{A}\boldsymbol{x}(\boldsymbol{\lambda}) - \boldsymbol{c})$$

where

$$x_i(\boldsymbol{\lambda}) = \arg\max_{x_i \in I_i} \{u_i(x_i) - \pi_i(\boldsymbol{\lambda})x_i\}, \quad \text{for every } i \in \mathcal{S}.$$

Since we assume that the functions $u_i, i \in \mathcal{S}$ are not only strictly concave, but also, in fact, *strongly* concave over I_i the gradient of the objective function ∇q_{N} is Lipschitz continuous by Lemma II.1.

B. A Dual-Based Gradient Method

One approach for constructing a solution method for (D) [and thus also for (P)] is to disregard the Lipschitz continuity of the gradient ∇q and employ a gradient projection method for solving the dual problem with a constant stepsize α_G. The method generates dual variables $\boldsymbol{\lambda}^k$ according to the following rule:

Gradient Method with Constant Stepsize

Step 0: Choose $\boldsymbol{\lambda}_0 \geq bf0$.

Step k: (for $k \geq 1$)

$$\boldsymbol{x}^{k-1} = \arg\max_{\boldsymbol{x}\in X}\{g(\boldsymbol{x}) - (\boldsymbol{\lambda}^{k-1})^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{c})\} \tag{7}$$

$$\boldsymbol{\lambda}^k = \left[\boldsymbol{\lambda}^{k-1} + \alpha_G(\boldsymbol{A}\boldsymbol{x}^{k-1} - \boldsymbol{c})\right]_+ \tag{8}$$

where $[\cdot]_+$ is the projection on the non-negative orthant in $\mathbb{R}^m$.

For the NUM problem (i.e., $g = g_{\mathrm{N}}$ and $q = q_{\mathrm{N}}$), the constraint set X and the objective function are separable in components of the variables vector $\boldsymbol{x}$ since $X = \Pi_{i\in\mathcal{S}} I_i$ and $g_{\mathrm{N}}(\boldsymbol{x}) = \sum_{i\in S} u_i(x_i)$. This allows decoupling step (7) as

$$x_i^{k-1} = \arg\max_{x_i\in I_i}\left\{u_i(x_i) - \left(\sum_{l\in\mathcal{L}(i)}\lambda_l^{k-1}\right)x_i\right\} \tag{9}$$

for all $i \in \mathcal{S}$. Moreover, step (8) can be written as

$$\lambda_l^k = \left[\lambda_l^{k-1} + \alpha\left(\sum_{i\in\mathcal{S}(l)} x_i^{k-1} - c_l\right)\right]_+ \tag{10}$$

for all $l \in \mathcal{L}$. Thus, a link l can update its dual variable λ_l in step (10) by using the aggregated rates $\sum_{i\in\mathcal{S}(l)} x_i^{k-1}$ of users that utilize the link and its own link capacity value c_l. Moreover, each source i can update its rate in step (9) by using its own utility function u_i and the aggregated dual variables $\sum_{l\in\mathcal{L}(i)}\lambda_l^{k-1}$ for the links that serve the source. Hence, as long as there is a feedback mechanism that sends the aggregated information (along the links used by a source) back to the source (which is the case in practical flow control protocols), the preceding updates can be implemented using local information available to each source and destination.

The decomposition properties of these two steps have been observed in [7], which motivated interest in using dual decomposition and subgradient projection methods to solve network resource allocation problems (see, e.g., [4], [7], [11], [21], and [24]). To address the rate of convergence of such dual methods, a subgradient method with averaging has been considered in [14] and [15], which is motivated by a primal-recovery approach proposed in [16], see also [8]–[10] and [23]. The primal recovery approach constructs the primal sequence, denoted by $\{\hat{\boldsymbol{x}}^k\}$, as a running average of the iterate sequence $\{\boldsymbol{x}^k\}$

$$\hat{\boldsymbol{x}}^k = \frac{1}{k+1}\sum_{t=0}^{k}\boldsymbol{x}^t.$$

As seen in [14], the averages of the iterates generated by the method with a constant stepsize do not necessarily converge. However, the function values approach the optimal value within an error depending on the stepsize, while the feasibility violation diminishes at rate $O(1/k)$.

None of the aforementioned works makes use of the strong concavity of the utility functions and, thus, the results there remain within the domain of non-smooth convex optimization. The major disadvantages of such an approach are: 1) it suffers from the slow $O(1/\sqrt{k})$ rate of convergence of subgradient methods[1]; and 2) the distributed implementation dictates a constant stepsize choice which essentially does not guarantee convergence to the optimal value but rather to a value in an interval surrounding the optimal value. In Section III, we will show how to overcome the mentioned disadvantages by exploiting the Lipschitz continuity of the gradient of the dual objective function. Indeed, it is well-known that when the objective function of a concave program is strongly concave, then the resulting dual objective is differentiable with Lipschitz gradient, see, e.g., [18, Ch. 9]. Lemma II.2 records this fact and provides an explicit computation of the Lipschitz gradient constant for our specific dual objective given in (5).

Lemma II.2: The dual objective function $q(\boldsymbol{\lambda}) = (-\tilde{g})^*(-\boldsymbol{A}^T\boldsymbol{\lambda}) + \boldsymbol{\lambda}^T\boldsymbol{c}$ defined in (5) has a Lipschitz continuous gradient with constant $\frac{\rho(\boldsymbol{A}^T\boldsymbol{A})}{\sigma}$, where $\rho(\boldsymbol{A}^T\boldsymbol{A})$ is the spectral radius of the matrix $\boldsymbol{A}^T\boldsymbol{A}$.

Proof: By the definition of q, we have

$$\nabla q(\boldsymbol{\lambda}) = -\boldsymbol{A}\nabla(-\tilde{g})^*(-\boldsymbol{A}^T\boldsymbol{\lambda}) + \boldsymbol{c}.$$

The function $-\tilde{g}$ is proper, convex lower-semicontinuous, and hence it conicides with its bi-conjugate (see, e.g., [19, Th. 12.2, p. 104]). Since $\tilde{g}$ is also strongly convex with parameter σ, applying Lemma II.1, with $h := (-\tilde{g})^*$, we have that $(-\tilde{g})^*$ has a Lipschitz continuous gradient with the constant $\frac{1}{\sigma}$. Combining this with the properties of induced norms given in Lemma I.1, we obtain for all $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2 \in \mathbb{R}^{|\mathcal{L}|}$

$$\begin{aligned}\|\nabla q(\boldsymbol{\lambda}_1) - \nabla q(\boldsymbol{\lambda}_2)\| &\leq \frac{1}{\sigma}\|\boldsymbol{A}\|\|\boldsymbol{A}^T(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)\| \\ &\leq \frac{1}{\sigma}\|\boldsymbol{A}\|\|\boldsymbol{A}^T\|\|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}\| = \frac{\rho(\boldsymbol{A}^T\boldsymbol{A})}{\sigma}\|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2\|\end{aligned}$$

proving the stated Lipschitz gradient property for q. □

III. An FGM for the Dual

Our approach is to utilize the Lipschitz gradient property of the dual objective function by applying an $O(1/k^2)$ FGM to the dual problem. We will show in Section III-A, an $O(1/k)$ rate of convergence of the primal sequence can be established without the need of any primal averaging.

[1]Despite the fact that the dual objective function is differentiable, if we do not assume that it has a Lipschitz gradient, the convergence results are no better than those known for the nonsmooth case.

A. The Method

Since the objective function of the dual problem (D) has a Lipschitz gradient, in order to solve the problem (D), we can invoke an FGM, such as the one devised by Nesterov in 1983 [17] (see also [2]). At this point, we will not concern ourselves with the exact FGM, i.e., or can be used, and instead we will assume that there exists an algorithm that generates a sequence $\{\boldsymbol{\lambda}^k\}_{k=1}^{\infty}$ satisfying

$$q(\boldsymbol{\lambda}^k) - q^* \leq \frac{C}{k^2}, \quad \text{for } k = 1, 2, \ldots \tag{11}$$

where $C > 0$ is some constant. The above inequality is quite often interpreted as follows: in order to obtain an ε-optimal solution of the dual problem (D), one requires at most $O(1/\sqrt{\varepsilon})$ iterations. Of course, we can also define a corresponding primal sequence for $k = 1, 2, \ldots,$

$$\boldsymbol{x}^k = \arg\max_{\boldsymbol{x} \in X} \{g(\boldsymbol{x}) - (\boldsymbol{\lambda}^k)^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{c})\}. \tag{12}$$

The primal iterates $\boldsymbol{x}^k$ are not necessarily feasible [in fact, if $\boldsymbol{x}^k$ is feasible for some k, then it coincides with the *optimal* solution of (P)] and the natural question is whether the sequence $\boldsymbol{x}^k$ converges to the unique optimal solution $\boldsymbol{x}^*$ and, if so, at what rate? These questions are answered in Theorem 1.

Theorem 1: Suppose that $\{\boldsymbol{\lambda}^k\} \subseteq \mathbb{R}_+^m$ is a sequence satisfying (11) and let $\{\boldsymbol{x}^k\}$ be the sequence defined by (12). Then, for all $k \geq 1$ we have:

1) $\|\boldsymbol{x}^k - \boldsymbol{x}^*\| \leq \sqrt{\frac{2C}{\sigma}}\frac{1}{k}$;
2) $[\boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{c}]_+ \leq (\|\boldsymbol{A}\|_{2,\infty}\sqrt{\frac{2C}{\sigma}}\frac{1}{k})\boldsymbol{e}_m$, where $\boldsymbol{e}_m$ is the m-dimensional vector with all entries equal to 1 and $\|\boldsymbol{A}\|_{2,\infty} \equiv \max\{\|\boldsymbol{A}\boldsymbol{x}\|_\infty : \|\boldsymbol{x}\|_2 = 1\}$;
3) If g is Lipschitz continuous over X with a constant L_g, then $g_{\text{opt}} - g(\boldsymbol{x}^k) \leq L_g\sqrt{\frac{2C}{\sigma}}\frac{1}{k}$.

Proof: For an arbitrary $\boldsymbol{\lambda} \geq 0$ and any $\boldsymbol{x} \in \boldsymbol{X}$, let

$$h(\boldsymbol{x}, \boldsymbol{\lambda}) := g(\boldsymbol{x}) - \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{c}).$$

$h(\boldsymbol{x}(\boldsymbol{\lambda}), \boldsymbol{\lambda})$, with $\boldsymbol{x}(\boldsymbol{\lambda}) = \arg\max\{h(\boldsymbol{x}, \boldsymbol{\lambda}) : \boldsymbol{x} \in X\}$. Since $g(\cdot)$ is a strongly concave function with parameter σ, so is the function $\boldsymbol{x} \to h(\boldsymbol{x}, \boldsymbol{\lambda})$, and thus, it follows that for every $\boldsymbol{x} \in \boldsymbol{X}$

$$h(\boldsymbol{x}(\boldsymbol{\lambda}), \boldsymbol{\lambda}) - h(\boldsymbol{x}, \boldsymbol{\lambda}) \geq \frac{\sigma}{2}\|\boldsymbol{x}(\boldsymbol{\lambda}) - \boldsymbol{x}\|^2. \tag{13}$$

On the other hand, let $\boldsymbol{x}^*$ be the optimal solution of (P) and $\boldsymbol{\lambda}^* \geq 0$ an optimal solution of the dual problem (D). Then, by the definition of h we have for any $\boldsymbol{\lambda} \geq 0$

$$\begin{aligned} h(\boldsymbol{x}(\boldsymbol{\lambda}), \boldsymbol{\lambda}) - h(\boldsymbol{x}^*, \boldsymbol{\lambda}) &= g(\boldsymbol{x}(\boldsymbol{\lambda})) - \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x}(\boldsymbol{\lambda}) - \boldsymbol{c}) \\ &\quad - g(\boldsymbol{x}^*) + \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{c}) \\ &= q(\boldsymbol{\lambda}) - q(\boldsymbol{\lambda}^*) + \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{c}) \\ &\leq q(\boldsymbol{\lambda}) - q(\boldsymbol{\lambda}^*) \end{aligned}$$

where the second equality uses $g(\boldsymbol{x}^*) = q(\boldsymbol{\lambda}^*)$ [which holds by strong duality for the primal problem (P)], and where the last inequality follows from the facts $\boldsymbol{A}\boldsymbol{x}^* \leq \boldsymbol{c}$ and $\boldsymbol{\lambda} \geq 0$. Therefore, with (13) we thus obtain

$$\frac{\sigma}{2}\|\boldsymbol{x}(\boldsymbol{\lambda}) - \boldsymbol{x}^*\|^2 \leq q(\boldsymbol{\lambda}) - q(\boldsymbol{\lambda}^*) \quad \forall \boldsymbol{\lambda} \geq 0.$$

1) Using the later with $\boldsymbol{\lambda} = \boldsymbol{\lambda}^k$ and $\boldsymbol{x}(\boldsymbol{\lambda}) = \boldsymbol{x}^k$, together with (11), we thus obtain for all $k = 1, \ldots,$

$$\frac{\sigma}{2}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 \leq q(\boldsymbol{\lambda}^k) - q(\boldsymbol{\lambda}^*) \leq \frac{C}{k^2}$$

establishing the first part of the theorem.

2) We have

$$\begin{aligned} &\|\boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{c} - (\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{c})\|_\infty = \|\boldsymbol{A}(\boldsymbol{x}^k - \boldsymbol{x}^*)\|_\infty \\ &\leq \|\boldsymbol{A}\|_{2,\infty}\|\boldsymbol{x}^k - \boldsymbol{x}^*\| \leq \|\boldsymbol{A}\|_{2,\infty}\sqrt{\frac{2C}{\sigma}}\frac{1}{k} \end{aligned}$$

and in particular

$$\boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{c} - (\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{c}) \leq \left(\|\boldsymbol{A}\|_{2,\infty}\sqrt{\frac{2C}{\sigma}}\frac{1}{k}\right)\boldsymbol{e}_m.$$

Since $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{c} \leq 0$, it follows that $-(\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{c}) \geq 0$, thus implying that

$$\boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{c} \leq \left(\|\boldsymbol{A}\|_{2,\infty}\sqrt{\frac{2C}{\sigma}}\frac{1}{k}\right)\boldsymbol{e}_m$$

and the stated result for the feasibility violation follows.

3) A direct consequence of first part of the theorem. □

We have thus shown that $\boldsymbol{x}^k \to \boldsymbol{x}^*$ with a rate of $O(1/k)$, and that the constraint violation measured by $[\boldsymbol{A}\boldsymbol{x}^k - \boldsymbol{c}]_+$ is also of the order $O(1/k)$. Therefore, we obtain the interesting fact that although the convergence rate of the sequence of dual objective functions is of the rate $O(1/k^2)$, the convergence rate of the primal sequence and its corresponding objective function values, is of the order $O(1/k)$. Next, we will show how such a dual-based method can be implemented for the NUM problem.

As an example, in order to solve the NUM problem, we can use the FGM of Nesterov [17] (see also [2]) for solving problem (N–D) and obtain the following method:

Fast Gradient Method

Input: L_{N}—a Lipschitz constant of ∇q_N.

Step 0. Take $\boldsymbol{\xi}^1 = \boldsymbol{\lambda}^0 \in \mathbb{R}^{|\mathcal{L}|}$, $t_1 = 1$.

Step k. ($k \geq 1$) Compute

$$\begin{aligned} \boldsymbol{\lambda}^k &= \left[\boldsymbol{\xi}^k - \frac{1}{L_{\text{N}}}\nabla q_{\text{N}}(\boldsymbol{\xi}^k)\right]_+ \\ t_{k+1} &= \frac{1 + \sqrt{1 + 4t_k^2}}{2} \\ \boldsymbol{\xi}^{k+1} &= \boldsymbol{\lambda}^k + \left(\frac{t_k - 1}{t_{k+1}}\right)(\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}). \end{aligned}$$

Then, the following convergence result holds [2], [17].

Theorem 2: Let $\{\boldsymbol{\lambda}^k\}$ be the sequence generated by the FGM. Then, for all $k \geq 0$

$$q_{\mathrm{N}}(\boldsymbol{\lambda}^k) - q_{\mathrm{opt}} \leq \frac{2L_{\mathrm{N}}\|\boldsymbol{\lambda}^{bf0} - \boldsymbol{\lambda}^*\|^2}{(k+1)^2}.$$

The main problem in applying such a scheme in a distributed way is that a backtracking procedure for determining the stepsize is not possible. It turns out that utilization of a constant stepsize that ensures convergence requires the knowledge of the Lipschitz constant of ∇q_{N}, which regretfully depends on the information from all the sources. An illustration of this fact is shown in Lemma III.1 that derives a Lipschitz constant.

Lemma III.1: The following is a Lipschitz constant for the mapping ∇q_{N}:

$$L_{\mathrm{N}} = \left(\max_{i\in\mathcal{S}} \frac{1}{\sigma_i}\right) \cdot \max_{i\in\mathcal{S}} |\mathcal{L}(i)| \cdot \max_{l\in\mathcal{L}} |\mathcal{S}(l)|. \tag{14}$$

Proof: First, let

$$h(\boldsymbol{\mu}) \equiv \sum_{i\in\mathcal{S}} (-\tilde{u}_i)^*(\mu_i), \quad \boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{S}|}$$

so that $q_{\mathrm{N}}(\boldsymbol{\lambda}) = h(-\boldsymbol{A}^T\boldsymbol{\lambda}) + \boldsymbol{c}^T\boldsymbol{\lambda}$ and $\nabla q_{\mathrm{N}}(\boldsymbol{\lambda}) = -\boldsymbol{A}\nabla h(-\boldsymbol{A}^T\boldsymbol{\lambda}) + \boldsymbol{c}$. Since for every $i \in \mathcal{S}$ the function u_i, and hence also $\tilde{u}_i$, is strongly concave with parameter $\sigma_i > 0$, it follows that $(-\tilde{u}_i)^*$ has a Lipschitz derivative with constant $\frac{1}{\sigma_i}$. Therefore, for every $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathbb{R}^{|\mathcal{S}|}$

$$\|\nabla h(\boldsymbol{\mu}_1) - \nabla h(\boldsymbol{\mu}_2)\| \leq \left(\max_{i\in\mathcal{S}} \frac{1}{\sigma_i}\right)\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|.$$

Thus, using the properties of the induced norm given in Lemma I.1, for every $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2 \in \mathbb{R}^{|\mathcal{L}|}$, we have

$$\begin{aligned}
&\|\nabla q_{\mathrm{N}}(\boldsymbol{\lambda}_1) - \nabla q_{\mathrm{N}}(\boldsymbol{\lambda}_2)\| \\
&\leq \|-\boldsymbol{A}(\nabla h(-\boldsymbol{A}^T\boldsymbol{\lambda}_1) - \nabla h(-\boldsymbol{A}^T\boldsymbol{\lambda}_2))\| \\
&\leq \left(\max_{i\in\mathcal{S}} \frac{1}{\sigma_i}\right)\|\boldsymbol{A}\| \cdot \|\boldsymbol{A}^T(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)\| \\
&\leq \left(\max_{i\in\mathcal{S}} \frac{1}{\sigma_i}\right)\|\boldsymbol{A}\|\|\boldsymbol{A}^T\|\|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2\| \\
&= \left(\max_{i\in\mathcal{S}} \frac{1}{\sigma_i}\right)\|\boldsymbol{A}\|^2\|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2\|.
\end{aligned}$$

In addition,

$$\begin{aligned}
\|\boldsymbol{A}\|^2 &= \rho(\boldsymbol{A}^T\boldsymbol{A}) \leq \|\boldsymbol{A}^T\boldsymbol{A}\|_\infty \\
&\leq \|\boldsymbol{A}^T\|_\infty\|\boldsymbol{A}\|_\infty = \|\boldsymbol{A}\|_1\|\boldsymbol{A}\|_\infty.
\end{aligned}$$

With $\boldsymbol{A}$ defined in (4) we have

$$\|\boldsymbol{A}\|_1 = \max_{i\in\mathcal{S}} |\mathcal{L}(i)|, \quad \|\boldsymbol{A}\|_\infty = \max_{l\in\mathcal{L}} |\mathcal{S}(l)|$$

establishing the desired result. □

Computation of a Lipschitz constant of ∇q_{N}, such as L_{N} given in (14), will require communication between all the sources in the network, and this is not possible when only local communication is permitted. We are therefore led to discuss scaled versions of FGMs in which each variable has its own stepsize which depends only on local information.

B. A Distributed Implementation of FGM for NUM

In this section, we will show how to exploit the special structure of the dual problem (N–D) in order to establish a fully distributed FGM for solving it. For ease of notation, we will rewrite problem (N–D) as

$$\begin{aligned}
\min \quad & \textstyle\sum_{i\in\mathcal{S}} h_i\Big(-\sum_{l\in\mathcal{L}(i)} \boldsymbol{\lambda}_l\Big) + \boldsymbol{c}^T\boldsymbol{\lambda} \\
\text{s.t.} \quad & \boldsymbol{\lambda} \geq bf0
\end{aligned} \tag{15}$$

where $h_i(\lambda) \equiv (-\tilde{u}_i)^*(\lambda)$. For an index set I, the vector $\boldsymbol{\lambda}_I$ denotes the subvector of $\boldsymbol{\lambda}$ consisting of the variables $\lambda_j, j \in I$ [e.g., $\boldsymbol{\lambda}_{\{1,3,4\}} = (\lambda_1, \lambda_3, \lambda_4)^T$]. We can thus also rewrite (15) as

$$\begin{aligned}
\min \quad & \textstyle\sum_{i\in\mathcal{S}} H_i\big(-\boldsymbol{\lambda}_{\mathcal{L}(i)}\big) + \boldsymbol{c}^T\boldsymbol{\lambda} \\
\text{s.t.} \quad & \boldsymbol{\lambda} \geq \boldsymbol{0}
\end{aligned} \tag{16}$$

where for an index set I, $H_i(\boldsymbol{\lambda}_I) = h_i(\sum_{i\in I} \lambda_i)$. Recall that h_i has a Lipschitz derivative with constant $\frac{1}{\sigma_i}$. Therefore, from its definition, it follows that H_i has a Lipschitz gradient with constant $\frac{|\mathcal{L}(i)|}{\sigma_i}$.

Now, for every $i \in \mathcal{S}$, we can write the descent lemma for the function H_i

$$\begin{aligned}
H_i(\boldsymbol{\lambda}_{\mathcal{L}(i)}) \leq\ & H_i(\boldsymbol{\mu}_{\mathcal{L}(i)}) + \langle \nabla H_i(\boldsymbol{\mu}_{\mathcal{L}(i)}), \boldsymbol{\lambda}_{\mathcal{L}(i)} - \boldsymbol{\mu}_{\mathcal{L}(i)}\rangle \\
&+ \frac{|\mathcal{L}(i)|}{2\sigma_i}\|\boldsymbol{\lambda}_{\mathcal{L}(i)} - \boldsymbol{\mu}_{\mathcal{L}(i)}\|^2.
\end{aligned}$$

Summing the above inequality over $i \in \mathcal{S}$ and using the fact that $q_{\mathrm{N}}(\boldsymbol{\lambda}) \equiv \sum_{i\in\mathcal{S}} H_i(-\boldsymbol{\lambda}_{\mathcal{L}(i)}) + \boldsymbol{c}^T\boldsymbol{\lambda}$, we obtain

$$\begin{aligned}
q_{\mathrm{N}}(\boldsymbol{\lambda}) \leq\ & q_{\mathrm{N}}(\boldsymbol{\mu}) + \langle \nabla q_{\mathrm{N}}(\boldsymbol{\mu}), \boldsymbol{\lambda} - \boldsymbol{\mu}\rangle \\
&+ \frac{1}{2}(\boldsymbol{\lambda} - \boldsymbol{\mu})^T\boldsymbol{W}(\boldsymbol{\lambda} - \boldsymbol{\mu})
\end{aligned} \tag{17}$$

where $\boldsymbol{W}$ is a positive definite diagonal matrix whose lth diagonal element is given by

$$\boldsymbol{W}_{ll} = \sum_{i\in\mathcal{S}(l)} \frac{|\mathcal{L}(i)|}{\sigma_i}.$$

It is well-known that the key ingredient in proving convergence of gradient-type methods is the existence of a corresponding descent lemma. The weighted descent lemma given by (17) can also be used in order to prove the convergence of a corresponding scaled gradient projection method. Indeed, it is very easy to see that the analysis of [2] can be easily extended to the weighted case and the resulting fast gradient projection method will have the following form:

Fast Weighted Gradient Projection Method

Step 0. Initialize $\eta^1 = \boldsymbol{\lambda}^0 \in \mathbb{R}_+^{|\mathcal{L}|},\ t_1 = 1.$

Step k. ($k \geq 1$) Compute

$$\boldsymbol{\lambda}^k = \left[\boldsymbol{\eta}^k - \boldsymbol{W}^{-1}\nabla q_{\mathrm{N}}(\eta^k)\right]_+ \tag{18}$$

$$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2} \tag{19}$$

$$\boldsymbol{\eta}^{k+1} = \boldsymbol{\lambda}^k + \left(\frac{t_k - 1}{t_{k+1}}\right)(\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}). \tag{20}$$

As already mentioned, the convergence analysis of [2] can be easily extended to the weighted case when all the l_2-norms $\|\boldsymbol{x}\|$ are replaced by the weighted norm $\|\boldsymbol{x}\|_{\boldsymbol{W}}^2 = \sum_i W_{ii}x_i^2$, and the convergence result will be the following.

Theorem 3: Let $\{\boldsymbol{\lambda}^k\}, \{\boldsymbol{\eta}^k\}$ be generated by the fast weighted gradient projection method. Then, for any $k \geq 1$

$$q_{\mathrm{N}}(\boldsymbol{\lambda}^k) - q(\boldsymbol{\lambda}^*) \leq \frac{2\|\boldsymbol{\lambda}^0 - \boldsymbol{\lambda}^*\|_{\boldsymbol{W}}^2}{(k+1)^2}. \tag{21}$$

Note that even disregarding the issues of distributive optimization, the convergence result (21) is better than the one obtained when the Lipschitz constant L_{N} given in (14) since $\boldsymbol{W} \preceq L_{\mathrm{N}}\boldsymbol{I}$ implying that $\|\boldsymbol{x}\|_{\boldsymbol{W}}^2 \leq L_{\mathrm{N}}\|\boldsymbol{x}\|^2$. The additional attribute of this method is of course that it lends itself to a decentralized implementation. The method is described in details below.

Fast Dual-Based Method for Solving NUM

Initialization. For each link $l \in \mathcal{L}$, select λ_l^0 and set $\eta_l^1 = \lambda_l^0$. Let $t_1 = 1$ and

$$\alpha_l = \left(\sum_{i\in\mathcal{S}(l)} \frac{|\mathcal{L}(i)|}{\sigma_i}\right)^{-1}.$$

Step k. For $k \geq 1$, execute the following steps:

(A) **Source-Rate Update:** for all $i \in \mathcal{S}$

$$x_i^{k-1} = \arg\max_{x_i \in I_i}\left\{u_i(x_i) - \left(\sum_{l\in\mathcal{L}(i)} \eta_l^{k-1}\right)x_i\right\}.$$

(B) **Link-Price Update**: for all $l \in \mathcal{L}$

$$\lambda_l^k = \left[\eta_l^{k-1} + \alpha_l\left(\sum_{i\in\mathcal{S}(l)} x_i^{k-1} - c_l\right)\right]_+.$$

(C) **Two-Step Network-Price Update:**

(C.1) $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$

(C.2) $\boldsymbol{\eta}^{k+1} = \boldsymbol{\lambda}^k + (\frac{t_k-1}{t_{k+1}})(\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1})$.

For each $l \in \mathcal{L}$, the step sizes α_l only depend on the sources i that use link l [i.e., $\mathcal{S}(l)$] and it is assumed that at the beginning of the process each source i sends its strong convexity constant σ_i and the number of links it uses $|\mathcal{L}(i)|$ to all the links on its path (i.e., all the links it uses). This is the only additional communication that is required for the fast method. By Theorems 1 and 3, the sequence $\{\boldsymbol{x}^k\}$ converges to the optimal allocation vector at the rate of $O(1/k)$.

IV. Numerical Experiments

We compare the performance of the fast weighted gradient method developed in Section III-B with two other distributed algorithms commonly used in the literature for solving the NUM problem: 1) (dual) gradient method explained in Section II-B; and 2) Newton-type diagonally scaled (dual) gradient method introduced in [1]. We have implemented all three algorithms both on small deterministic networks and also on a randomly generated collection of networks. Our simulation results demonstrate that the proposed FGM significantly outperforms the standard gradient methods in terms of the number of iterations.

We have assumed that all sources have identical utility functions given by $u_i(x_i) = 20\log(x_i + 0.1)$, where 0.1 is added to the argument of the logarithmic function to prevent numerical instability when x_i is close to 0. We have also assumed that all links have identical capacity given by 1. Thus, the NUM problem has the form

$$\begin{aligned}&\max_x \sum_{i\in\mathcal{S}} 20\log(x_i + 0.1)\\ &\text{s.t.} \sum_{i\in\mathcal{S}(l)} x_i \leq 1, \quad \text{for all } l \in \mathcal{L}\\ &x_i \geq 0, \quad \text{for all } i \in \mathcal{S}.\end{aligned}$$

For all three algorithms, we used constant stepsize rules that can guarantee convergence. More specifically, in the price update (8) for the gradient method, we used a stepsize α_G given by

$$\alpha_G = \frac{2\sigma}{N_p N_s}$$

where σ is a strong convexity constant for the utility functions [taken to be $\sigma = \frac{20}{(1+0.1)^2}$ for these experiments]. The scalars N_p and N_s are defined, respectively, as the longest path length among all sources and the maximum number of sources sharing a particular link

$$N_p = \max_{s\in\mathcal{S}} |\mathcal{L}(s)|, \qquad N_s = \max_{l\in\mathcal{L}} |\mathcal{S}(l)|.$$

Since in a distributed setting, we do not have information on N_p and N_s, we use the total number of links and sources, i.e., $|\mathcal{L}|$ and $|\mathcal{S}|$, as upper bounds on N_p and N_s, respectively. For the diagonally scaled gradient method, we used a stepsize α_{DS} that satisfies

$$\alpha_{DS} < \frac{2\epsilon\sigma}{N_p N_s}$$

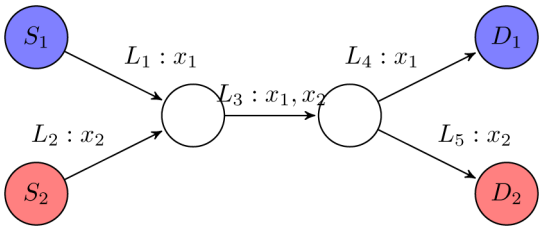


Fig. 1. Sample network. Each source–destination pair is displayed with the same color. We use x_i to denote the flow corresponding to the ith source and L_i to denote the ith link.

where scalars N_p, N_s, and σ are defined as above and ε is a positive scalar used to guarantee the Hessian approximation H which is positive definite, i.e., if $H_{ll} < \epsilon$ then we use ϵ for that element to avoid singularity.[2] We set $\epsilon = 0.1$ in our experiments, which is the same value used in [1]. For the fast weighted gradient method, we used the stepsize rule of Section III-B with $\sigma = \frac{20}{(1+0.1)^2}$.

In our first experiment, we considered the network shown in Fig. 1 with two sources (and destinations determined by the set of links used by the sources). The links used by the sources are identified using the flows corresponding to each source. Fig. 2 illustrates a sample evolution of the objective function value for each of the three algorithms. The iteration count on the horizontal axis is log-scaled. The dotted horizontal lines indicate $\pm 5\%$ interval around the optimal objective function value. The fast weighted gradient method outperforms the standard gradient method. In this particular example, it also converges faster than the diagonally scaled gradient method.

To test the performance of the algorithms over general networks, we generated 50 random networks, with a random number of links taking (integer) values in range $[1, 40]$ and a random number of sources taking values in the interval $[1, 25]$ (generated independently). Each routing matrix consists of $|\mathcal{L}| \times |\mathcal{S}|$ Bernoulli random variables.[3] All three methods are implemented over the 50 networks. The methods were terminated when all of the following conditions are satisfied at an iteration k:

1) primal objective function value satisfies $|\frac{f(x^{k+1})-f(x^k)}{f(x^k)}| \leq 0.01$;
2) dual variable satisfies $\|\lambda^{k+1} - \lambda^k\|_\infty \leq 0.01$;
3) primal feasibility satisfies $[c - Ax^k]_l \geq -0.01$ for all links l.

To display the results properly, we capped the number of iterations at 250000 (this cap was not exceeded in the trials, except a few times with the gradient method). We record the number of iterations upon termination for all three methods and results are shown in Fig. 3 on a log scale. The mean number

[2]The Hessian approximation is given as

$$H_{ll} = -\frac{x_l^k - x_l^{k-1}}{\lambda_l^k - \lambda_l^{t-1}}$$

where x_l^k is the flow on link l and λ_l^k is the dual variable associated with link l at iteration k. Hence depending on the initial conditions, the approximated value of H_{ll} may be smaller than ϵ, even though the elements in the exact Hessian are lower bounded by the scalar α.

[3]When a source does not use any link or a link is not used by any source, we discard the routing matrix and generate another one.

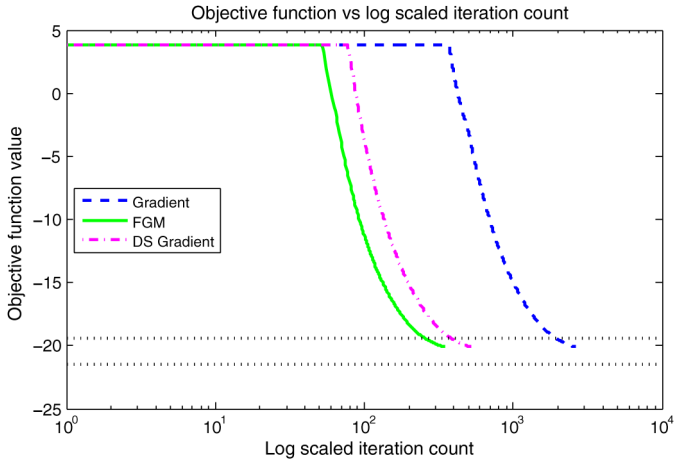


Fig. 2. Sample objective function value of all three methods against log-scaled iteration count for network in Fig. 1. The dotted horizontal lines denote $\pm 5\%$ interval of the optimal objective function value.

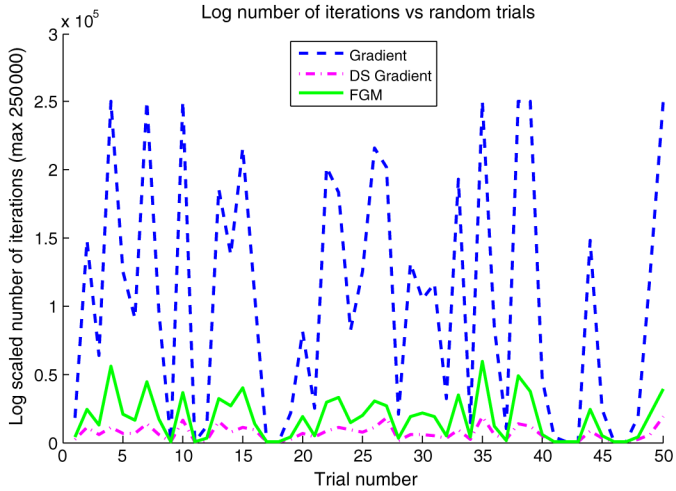


Fig. 3. Log-scaled iteration count for the three methods implemented over 50 randomly generated networks with random sizes.

of iterations to convergence from the 50 trials is 6584.2 for the diagonally scaling gradient method, 17871.6 for the fast weighted gradient method and 103265.9 for the gradient method.

To further study the scaling properties of the algorithm with respect to the network size, we generated another set of 50 random networks, each with 20 sources and 50 links ($|\mathcal{S}| = 20$ and $|\mathcal{L}| = 50$). We repeated the same experiment as before and recorded the results in Fig. 4. The average number of iterations is 91221 for the diagonally scaled gradient method, 61430 for the fast weighted gradient method, and 247628.6 for the gradient method. These results are qualitatively different from those in Fig. 3, as the fast weighted gradient method is faster than the diagonally scaled gradient method. This can be explained by the difference in stepsize rules used in the two methods. In the diagonally scaled gradient method, the stepsize is proportional to the global quantity $\frac{1}{|\mathcal{L}||\mathcal{S}|}$, whereas in the fast weighted gradient method, the stepsize is proportional to the local path lengths $\frac{1}{|\mathcal{L}(i)|}$. The latter quantity, in general, results in larger stepsize values.

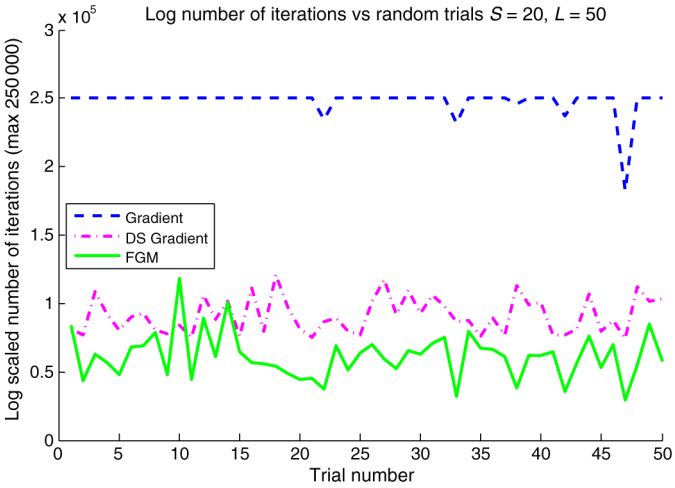


Fig. 4. Log-scaled iteration count for the three methods implemented over 50 random networks, each with 20 sources and 50 links.

Thus, for large networks, the FGM tends to converge faster than the diagonally scaled gradient method.

V. Conclusion

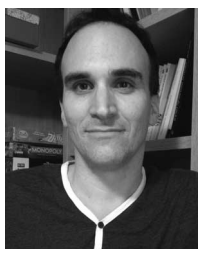
We have considered the NUM problem and proposed a fast distributed dual-based gradient method for solving the problem. Our focus was on the NUM problem with strongly concave utility functions. We established the convergence rate of the order $1/k$ for the primal iterate sequences, which demonstrates the superiority of these methods over the standard dual-gradient methods with convergence rate of the order $1/\sqrt{k}$. Furthermore, we have proposed a fully distributed implementation of the FGM. Our numerical results indicate that the proposed method is a viable alternative to the standard gradient method, but also to the Newton-type diagonally scaled dual-gradient method of [1].

Acknowledgments

The authors thank E. Wei for useful comments and her help with the numerical experiments. They also thank the three reviewers for their useful comments.

References

[1] S. Athuraliya and S. Low, "Optimization flow control with Newton-like algorithm," *J. Telecommun. Syst.*, vol. 15, pp. 345–358, 2000.
[2] A. Beck and M. Teboulle, "Gradient-based algorithms with applications to signal recovery problems," in *Convex Optimization in Signal Processing and Communications*, D. Palomar and Y. Eldar, Eds., Cambridge, U.K.: Cambridge Univ. Press, 2009, pp. 139–162.
[3] D. P. Bertsekas, in *Nonlinear Programming*, 2nd ed. Belmont, MA, USA: Athena Scientific, 1999.
[4] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.
[5] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge, U.K.: Cambridge Univ. Press, 1985.
[6] A. Jadbabaie, A. Ozdaglar, and M. Zargham, "A distributed newton method for network optimization," in *Proc. 48th IEEE Conf. Decision and Control* (CDC), 2009, pp. 2736–2741.
[7] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, 1998.
[8] T. Larsson, M. Patriksson, and A. Strömberg, "Ergodic results and bounds on the optimal value in subgradient optimization," in *Operations Research Proceedings*, P. Kelinschmidt *et al.*, Eds. Berlin, Germany: Springer-Verlag, 1995, pp. 30–35.
[9] T. Larsson, M. Patriksson, and A. Strömberg, "Ergodic convergence in subgradient optimization," *Optim. Methods Softw.*, vol. 9, pp. 93–120, 1998.
[10] T. Larsson, M. Patriksson, and A. Strömberg, "Ergodic primal convergence in dual subgradient schemes for convex programming," *Math. Program.*, vol. 86, pp. 283–312, 1999.
[11] S. H. Low and D. E. Lapsley, "Optimization flow control. I. Basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861–874, Dec. 1999.
[12] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.
[13] I. Necoara and V. Nedelcu, *Rate analysis of inexact dual first order methods: Application to distributed MPC for network systems* [Online]. Available: http://arxiv.org/pdf/1302.3129.pdf, 2013.
[14] A. Nedić and A. Ozdaglar, "On the rate of convergence of distributed subgradient methods for multi-agent optimization," in *Proc. 46th IEEE Conf. Decis. Control,* 2007, pp. 4711–4716.
[15] A. Nedić and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM J. Optim.*, vol. 19, no. 4, pp. 1757–1780, 2009.
[16] A. S. Nemirovskii and D. B. Yudin, "Cezare convergence of gradient method approximation of saddle points for convex-concave functions," *Dokl. Akad. Nauk SSSR*, vol. 239, pp. 1056–1059, 1978.
[17] Y. Nesterov, "A method for solving the convex programming problem with convergence rate $O(1/k^2)$," *Dokl. Akad. Nauk SSSR*, vol. 269, no. 3, pp. 543–547, 1983.
[18] B. T. Polyak, *Introduction to Optimization* (Transl.: in Mathematics and Engineering. New York: Optimization Software, 1987, translated from Russian, with a foreword by D. P. Bertsekas).
[19] R. T. Rockafellar, *Convex Analysis.* Princeton, NJ, USA: Princeton Univ. Press, 1970.
[20] R. T. Rockafellar and R. J. B. Wets, *Variational Analysis, vol. 317, Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Berlin, Germany: Springer-Verlag, 1998.
[21] S. Shakkottai and R. Srikant, "Network optimization and control," *Found. Trends Netw.*, vol. 2, no. 3, pp. 271–379, 2007.
[22] S. Shalev-Shwartz and T. Zhang, *Accelerated Proximal Stochastic Dual Coordinate Ascent for Regularized Loss Minimization*, [Online]. Available: http://arxiv.org/pdf/1309.2375v2.pdf, 2013.
[23] N. Z. Shor, *Minimization Methods for Nondifferentiable Functions,* vol. 3 *Springer Series in Computational Mathematics.* Berlin, Germany: Springer, 1985.
[24] R. Srikant, *The Mathematics of Internet Congestion Control. Systems & Control: Foundations & Applications.* Cambridge, MA, USA: Boston Inc., 2004.
[25] E. Wei, A. Ozdaglar, and A. Jadbabaie, "A distributed newton method for network utility maximization, part I: Algorithm," *IEEE Trans. Automat. Control*, vol. 58, no. 9, pp. 2162–2175, Sep. 2013.
[26] E. Wei, A. Ozdaglar, and A. Jadbabaie, "A distributed newton method for network utility maximization, part II: Convergence," *IEEE Trans. Automat. Control*, vol. 58, no. 9, pp. 2176–2188, Sep. 2013.
[27] E. Wei, M. Zargham, A. Ozdaglar, and A. Jadbabaie, "On dual convergence of the distributed newton method for network utility maximization," in *Proc. 50th IEEE Conf. Decision Control* (CDC), 2011, pp. 6612–6617.
[28] M. Zargham, A. Ribeiro, and A. Jadbabaie, "Accelerated dual descent for capacity constrained network flow optimization," in *Proc. 52nd IEEE Conf. Decision Control* (CDC), 2013, pp. 1037–1042.
[29] M. Zargham, A. Ribeiro, A. Ozdaglar, and A. Jadbabaie, "Accelerated dual descent for network optimization," *IEEE Trans. Automat. Control,* 2013, to be published.



Amir Beck received the B.Sc. degree in pure mathematics (cum laude) in 1991, the M.Sc. degree in operations research (summa cum laude), and the Ph.D. degree in operations research from Tel Aviv University, Tel Aviv, Israel, in 1997 and 2003, repectively.

From 2003 to 2005, he was a Postdoctoral Fellow at the Minerva Optimization Center, Technion, Haifa, Israel. He is currently an Associate Professor in the Department of Industrial Engineering at the Technion-Israel Institute of Technology, Haifa, Israel. His research interests include continuous optimization, including theory, algorithmic analysis, and its applications. He has published numerous papers and has given invited lectures at international conferences.

Dr. Beck was awarded the Salomon Simon Mani Award for Excellence in Teaching and the Henry Taub Research Prize award. He is in the editorial board of *Mathematics of Operations Research*, *Operations Research* and *Journal of Optimization Theory and Applications*. His research has been supported by various funding agencies, including the Israel Science Foundation, the German-Israeli Foundation, the Binational US-Israel foundation, the Israeli Science and Energy Ministries, and the European community.

Angelia Nedić received the B.S. degree in mathematics from the University of Montenegro, Podgorica, Montenegro, in 1987, the M.S. degree in mathematics from the University of Belgrade, Belgrade, Serbia, in 1990, and the Ph.D. degree in mathematics and mathematical physics from Moscow State University, Moscow, Russia, in 1994, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2002.

From 2002 to 2006, she has been at the BAE Systems Advanced Information Technology. Since 2006, she has been with the Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA, where she is currently an Associate Professor. She is also affiliated with the Electrical Engineering Department and the Coordinated Science Laboratory at UIUC. Her current research interest is focused on large-scale convex optimization, distributed multi-agent optimization, equilibrium problems, and duality theory.

Dr. Nedić received an NSF Faculty Early Career Development Award in 2007 in Operations Research and the Donald Biggar Willett Scholar title in 2013 from the College of Engineering at UIUC.

Asuman Ozdaglar received the B.S. degree in electrical engineering from the Middle East Technical University, Ankara, Turkey, in 1996, and the M.S. and the Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1998 and 2003, respectively.

She is currently a Professor in the Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology, Cambridge, MA, USA. She is also a Member of the Laboratory for Information and Decision Systems and the Operations Research Center. Her research expertise includes optimization theory, with emphasis on nonlinear programming and convex analysis, game theory, with applications in communication, social, and economic networks, distributed optimization and control, and network analysis with special emphasis on contagious processes, systemic risk, and dynamic control. She is the co-author of the book entitled *Convex Analysis and Optimization* (Athena Scientific, 2003). She is currently the co-editor for a new area of *Journal Operations Research*, entitled "Games, Information, and Networks."

Prof. Ozdaglar is the recipient of a Microsoft fellowship, the MIT Graduate Student Council Teaching award, the NSF Career award, the 2008 Donald P. Eckman award of the American Automatic Control Council, the Class of 1943 Career Development Chair, a 2011 Kavli Fellowship of the National Academy of Sciences, and the inaugural Steven and Renee Finn Innovation Fellowship. She served as a Member of the Board of Governors of the Control System Society and as an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL. She is currently the Chair of the Control System Society Technical Committee Networks and Communication Systems.

Marc Teboulle received the D.Sc. degree from the Technion-Israel Institute of Technology, Haifa, Israel, in 1985.

He is a Professor at the School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv, Israel. He has held a position of Applied Mathematician at the Israel Aircraft Industries, and academic appointments at Dalhousie University and the University of Maryland. He has also held visiting appointments at several institutions, including IMPA (Rio de Janeiro), the University of Montpellier, the University of Paris, the University of Lyon, the University of Michigan, the University of Texas at Austin, The National Sun Yat-sen University, Kaohsiung, Taiwan, China, and IPAM at The University of California, Los Angeles. His research interests include the area of continuous optimization, including theory, algorithmic analysis, and its applications in engineering. He has published numerous papers and co-authored two books, and has given invited lectures at many international conferences. His research has been supported by various funding agencies including, the US National Science Foundation, the French-Israeli Ministry of Sciences, the Bi-National Israel-United States Science Foundation, and the Israel Science Foundation. From 1999 to 2002, he served as Chairman of the Department of Statistics and Operations Research at the School of Mathematical Sciences, Tel-Aviv University. He is on the editorial boards of the European Series in *Applied and Industrial Mathematics*, *Control, Optimization, and Calculus of Variations*, *Journal of Optimization Theory and Applications*, and *Science China Mathematics*. Since July 2013, he has been the Area Editor of *Continuous Optimization for Mathematics of Operations Research*.