

Double Smoothing for Time-Varying Distributed Multiuser Optimization

Andrea Simonetto and Geert Leus

Faculty of EEMCS, Delft University of Technology, 2826 CD Delft, The Netherlands

e-mails: {a.simonetto, g.j.t.leus}@tudelft.nl

Abstract—Constrained optimization problems that couple different cooperating users sharing the same communication network are often referred to as multiuser optimization programs. We are interested in convex discrete-time time-varying multiuser optimization, where the problem to be solved changes at each time step. We study a distributed algorithm to generate a sequence of approximate optimizers of these problems. The algorithm requires only one round of communication among neighboring users between subsequent time steps and, under mild assumptions, converges linearly to a bounded error floor whose size is dependent on the variability of the optimization problem in time. To develop the algorithm we employ a double regularization both in the primal and in the dual space. This increases the convergence rate and helps us in the convergence proof. Numerical results support the theoretical findings.

I. INTRODUCTION

We are interested in time-varying multiuser optimization problems arising often in network resource management. These problems are constrained optimization programs associated with a finite set of users $i \in \{1, \dots, N\}$. Each user i has, at discrete-time k , a *private* convex cost function $f_i^k(\mathbf{x}_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$ that depends on its decision vector \mathbf{x}_i only. The decision vectors are typically coupled by a convex constraint set and, sometimes, by an extra coupled term in the cost function. The time-varying multiuser optimization problem can be formulated as the following convex minimization for each discrete-time $k \geq 0$

$$\underset{\mathbf{x}}{\text{minimize}} \quad f^k(\mathbf{x}) := \sum_{i=1}^N f_i^k(\mathbf{x}_i) + c^k(\mathbf{x}) \quad (1a)$$

$$\text{subject to} \quad d_j^k(\mathbf{x}) \leq 0, \quad \text{for } j = 1, \dots, m \quad (1b)$$

$$\mathbf{x}_i \in X_i, \quad \text{for } i = 1, \dots, N, \quad (1c)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the stacked vector of all the local decision variables \mathbf{x}_i , the function $c^k(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ is a coupling convex function, each of the $d_j^k(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ is a convex function that describes the j -th coupled constraint, and finally X_i is a local convex set (often a box constraint).

In this paper, we seek distributed (iterative) algorithms that generate a sequence of approximate decision variables $\mathbf{v}(k)$ which eventually converges to an optimizer of (1). The algorithms we consider are distributed, since they will allow each user to communicate only with neighboring users, and they are time-varying, since they will require only one communication round between subsequent discrete time steps.

Different distributed algorithms can solve the multiuser problem (1) when it is *time-invariant*, e.g., the saddle-point

method of [1], distributed double smoothing [2], [3], and the alternating direction method of multipliers [4]–[6]. These techniques are iterative, and in the *time-varying* case, they would require (theoretically) an infinite amount of communication among neighboring nodes (between consecutive time steps k and $k+1$) to converge to an optimizer of (1). Specific methods that account for a finite number of messages and still guarantee convergence have been proposed in [7]–[16], but they are all limited to restricted versions of (1), typically, strongly convex and (often) unconstrained problems.

In this paper, we attack the constrained time-varying multiuser problem in its generality and we prove that the method of [2] (originally intended for time-invariant scenarios) works also in the time-varying case under some standard assumptions on the variability of the optimization problem in time. We approximate the original problem (1) by employing a double regularization (both in the primal and dual space), and show *linear* convergence of the proposed primal-dual algorithm to the optimizer of the regularized problem up to a bounded error dependent on the variability of the optimization problem in time. This fast convergence, together with the price of a limited and a priori quantifiable (and tunable) suboptimality, makes the chosen regularization particularly suitable in time-varying cases. Interestingly, the double regularization (also known as double smoothing) massages the original problem (1) in a way that it becomes easier to manipulate and offers us an unexpected advantage in the time-varying convergence proof.

II. ASSUMPTIONS AND FORMAL PROBLEM FORMULATION

We work under the following setting and assumptions.

Assumption 1: The cost function $f^k(\mathbf{x})$ and the constraint functions $d_j^k(\mathbf{x})$ are convex and continuously differentiable for each $k \geq 0$. The constraint sets X_i are nonempty, closed, convex, and bounded.

We define the gradient map $\mathbf{g}^k(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ as $\mathbf{g}^k(\mathbf{x}) :=$

$$\left(\nabla_{\mathbf{x}_1} (f_1^k(\mathbf{x}_1) + c^k(\mathbf{x}))^\top, \dots, \nabla_{\mathbf{x}_N} (f_N^k(\mathbf{x}_N) + c^k(\mathbf{x}))^\top \right)^\top,$$

and assume

Assumption 2: The gradient map $\mathbf{g}^k(\mathbf{x})$ is Lipschitz continuous with constant L over the set $X := \prod_{i=1}^N X_i$ for $k \geq 0$,

$$\|\mathbf{g}^k(\mathbf{x}) - \mathbf{g}^k(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad \text{for all } \mathbf{x}, \mathbf{y} \in X, k \geq 0,$$

and in addition, the gradient $\nabla_{\mathbf{x}_j} d_j^k(\mathbf{x})$ is also Lipschitz continuous over X with a constant $L_j > 0$.

Assumption 3: (Slater Condition) There exists a Slater vector $\bar{\mathbf{x}}^k \in X$ such that $d_j^k(\bar{\mathbf{x}}^k) < 0$ for all $j = 1, \dots, m$.

This research was supported in part by STW under the D2S2 project from the ASSYS program (project 10561).

We introduce a time-invariant undirected communication network through which the users communicate. We label as $G = (V, E)$ the network graph comprised of a vertex set $V = \{1, \dots, N\}$ and an edge set (the communication links). In particular, if user i and l can communicate, then $(i, l) \in E$. We assume that,

Assumption 4: User i can communicate with all the users it is coupled with via the constraint set and/or the cost. I.e., for $k \geq 0$, for each user i ,

- 1) if the gradient $\nabla_{\mathbf{x}_i} c^k(\mathbf{x})$ is a function of \mathbf{x}_l , then either $(i, l) \in E$ or $i = l$;
- 2) if $d_j^k(\mathbf{x})$ is a function of \mathbf{x}_i , then it cannot be also a function of \mathbf{x}_l if $(i, l) \notin E$.

Assumptions 1 till 3 are used often in multiuser problems and they are not very restrictive. In particular, we do not assume strong convexity of the cost function (often required in time-varying schemes). Even Assumption 4 is not strict: it means that the users are coupled only locally (with their neighbors), which is rather reasonable for networked systems.

By Assumptions 1 and 3, there exists an optimizer of the multiuser convex program (1) for each discrete time k , which we indicate with $\mathbf{x}^{k,*}$, and a unique minimum $f^{k,*}$.

Our aim is to find $\mathbf{x}^{k,*}$ by a distributed iterative scheme: distributed, i.e., each user i communicates only via the links in the edge set E to determine its own decision variable $\mathbf{x}_i^{k,*}$; iterative, i.e., we seek algorithms that generate a sequence of approximate optimizers $\{\mathbf{v}(k)\}$ for which

$$\lim_{k \rightarrow \infty} \|\mathbf{v}(k) - \mathbf{x}^{k,*}\| = 0. \quad (2)$$

Condition (2) could be realized if the users were allowed to exchange an infinite amount of messages between subsequent time steps. Here, we look at time-varying algorithms which allow the users only one round of communication at each time step k , and thus we expect convergence to $\mathbf{x}^{k,*}$ up to an error floor dependent on the variability of the optimization problem in time. An additional challenge that one could face with problems as (1) is that they are not strictly convex, and the available distributed methods for such problems (in the time-invariant scenario) exhibit either slow convergence with low computational complexity (e.g., the subgradient method of [1] that converges as $O(1/k)$) or faster convergence (up to linear, i.e., $O(r^k)$, $r < 1$) but higher computational complexity (e.g., ADMM [5]). We see next how one can have both linear convergence and the computational complexity of first-order gradient methods in the same algorithm. In particular, we see how to accelerate convergence by double smoothing and generate a fast converging sequence of approximate optimizers.

III. DOUBLE SMOOTHING AND DISTRIBUTED ALGORITHM

We introduce a distributed gradient-based optimization method that employs a fixed regularization in the primal and dual spaces. This regularization serves to approximate the primal problem (1) in a way that can be solved by gradient-based methods with improved convergence properties.

Let $\lambda_j \in \mathbb{R}_+$ be the dual variable associated to the inequality constraint $d_j^k(\mathbf{x}) \leq 0$, and let $\nu > 0$, $\epsilon > 0$ be strictly positive scalars. Let $\boldsymbol{\lambda}$ be the stacked vector of all the dual

variables (for all j). Motivated by [2], we define a regularized Lagrangian function associated to the primal problem (1) as

$$\mathcal{L}_{\nu, \epsilon}^k(\mathbf{x}, \boldsymbol{\lambda}) := f^k(\mathbf{x}) + \frac{1}{2}\nu\|\mathbf{x}\|^2 + \sum_{j=1}^m \lambda_j d_j^k(\mathbf{x}) - \frac{1}{2}\epsilon\|\boldsymbol{\lambda}\|^2. \quad (3)$$

This regularized Lagrangian function is by definition a strictly convex function of the primal variable \mathbf{x} and a strictly concave function of the dual variable $\boldsymbol{\lambda}$.

The idea of double smoothing (that is a regularization both in the primal and dual space) is to construct the (easier to solve) approximate saddle-point problem

$$\underset{\mathbf{x}_i \in X_i, \forall i}{\text{minimize}} \underset{\lambda_j \in \mathbb{R}_+, \forall j}{\text{maximize}} \mathcal{L}_{\nu, \epsilon}^k(\mathbf{x}, \boldsymbol{\lambda}) \quad (4)$$

whose unique primal-dual optimizer is the couple $(\mathbf{x}^{k, \text{opt}}, \boldsymbol{\lambda}^{k, \text{opt}})$ and whose primal objective value is $f^{k, \text{opt}}$ (which exists due to Assumptions 1 and 3). In general, it is expected that the solutions of the primal problem (1) and the regularized saddle-point problem (4) are different, meaning $\|\mathbf{x}^{k,*} - \mathbf{x}^{k, \text{opt}}\| \neq 0$ and $\|f^{k,*} - f^{k, \text{opt}}\| \neq 0$. Furthermore, the solution of the regularized problem (4) does not necessarily satisfy the inequality constraints of the primal problem (1). However, it is possible to bound the suboptimality and the distance from the primal optimizer, along with the constraint violation, by some function of the regularization parameters ν and ϵ . Thus, while we are solving an approximation of the primal problem (1) we have bounds on the distance from the primal optimal solution. In this context, the regularization procedure can be seen as a way to speed up the convergence of standard gradient-like methods, and it may lead to a closer iterate to the optimum $f^{k,*}$ of the original problem given a finite number of iterations even though an approximate regularized problem is being solved. This last property is actually very important when (as in our case) at each discrete time the regularized problem is changing due to its time-varying nature. For further details we refer to the original works on (time-invariant) regularization and double smoothing techniques [2], [17].

Based on the regularized time-varying program (4), we propose the following primal-dual gradient-based iterations. We equip each user i with an initial local approximate minimizer $\mathbf{v}_i(0)$ at time 0, and an initial approximate value for the dual optimizer $\mu_j(0) = 0$ for all j for which $\nabla_{\mathbf{x}_i} d_j^k(\mathbf{x}) \neq 0$. We also indicate $\boldsymbol{\mu}(k)$ as the stacked version of $\mu_j(k)$ at time k . Then we compute, for $i = 1, \dots, N$, and for $j = 1, \dots, m$

$$\mathbf{v}_i(k+1) = \text{P}_{X_i} \left[\mathbf{v}_i(k) - \alpha \nabla_{\mathbf{x}_i} \mathcal{L}_{\nu, \epsilon}^k(\mathbf{x}, \boldsymbol{\lambda}) \Big|_{\mathbf{v}(k), \boldsymbol{\mu}(k)} \right] \quad (5a)$$

$$\mu_j(k+1) = \text{P}_{\mathbb{R}_+} \left[\mu_j(k) + \alpha \nabla_{\lambda_j} \mathcal{L}_{\nu, \epsilon}^k(\mathbf{x}, \boldsymbol{\lambda}) \Big|_{\mathbf{v}(k), \boldsymbol{\mu}(k)} \right], \quad (5b)$$

where P is the projection operator, and $\alpha > 0$ a suitable fixed stepsize. By nature of the multiuser problem, the iterations (5) are *structurally* distributed. To see this, it is sufficient to expand the gradients of the regularized Lagrangian as

$$\nabla_{\mathbf{x}_i} \mathcal{L}_{\nu, \epsilon}^k(\mathbf{x}, \boldsymbol{\lambda}) \Big|_{\mathbf{v}(k), \boldsymbol{\mu}(k)} = \left(\nabla_{\mathbf{x}_i} f_i^k(\mathbf{x}_i) + \nabla_{\mathbf{x}_i} c^k(\mathbf{x}) + \nu \mathbf{x}_i + \sum_{j=1}^m \lambda_j \nabla_{\mathbf{x}_i} d_j^k(\mathbf{x}) \right) \Big|_{\mathbf{v}(k), \boldsymbol{\mu}(k)},$$

$$\nabla_{\lambda_j} \mathcal{L}_{\nu, \epsilon}^k(\mathbf{x}, \boldsymbol{\lambda})|_{\mathbf{v}(k), \mu_j(k)} = \left(d_j^k(\mathbf{x}) - \epsilon \lambda_j \right)|_{\mathbf{v}(k), \mu_j(k)},$$

and notice that under Assumption 4 each user i has all the elements to compute the gradient w.r.t. its decision variable \mathbf{x}_i and w.r.t. the coupling multipliers λ_j . The final algorithm is summarized in Algorithm 1. The main difference with its time-invariant version in [2] is that, at each time step, not only the local variables get updated, but also the optimization program changes. It is important to note that this *seemingly innocent* modification could completely jeopardize the convergence properties of the algorithm. In the next section, we formally prove that this is actually not the case under some quite reasonable assumptions on the variability of (4).

Algorithm 1 Double smoothing distributed algorithm

Initialize locally $\mathbf{v}_i(0), \mu_j(0) = 0$. Then for $k \geq 0$:

- 1) communicate within the neighborhood $\mathbf{v}_i(k)$;
 - 2) update local variable $\mathbf{v}_i(k)$ and the coupling multipliers $\mu_j(k)$ by using (5);
 - 3) go to step 1.
-

IV. PROPERTIES OF THE PROPOSED SOLUTION

We analyze now the convergence properties of Algorithm 1; in particular, we are interested in its ability to yield a sequence of primal-dual approximate optimizers $\{\mathbf{v}(k), \boldsymbol{\mu}(k)\}$ that inevitably converge to the primal-dual optimal solution of the regularized problem (4) up to a bounded error floor. We need an additional assumption that bounds the variability of (4) between subsequent time steps.

Assumption 5: The difference of the optimizers of (4) at two subsequent time steps is bounded as

$$\|\mathbf{x}^{k+1, \text{opt}} - \mathbf{x}^{k, \text{opt}}\| \leq \delta_{\mathbf{x}}, \quad k \geq 0,$$

and the difference in the constraint sets at optimality is bounded as

$$|d_j^{k+1}(\mathbf{x}^{k+1, \text{opt}}) - d_j^k(\mathbf{x}^{k, \text{opt}})| \leq \delta_d, \quad j = 1, \dots, m, k \geq 0.$$

Assumption 5 on the variability of the optimizers is normally asked in time-varying optimization. The difference with the current literature is that we need to require also a bounded variability condition on the constraint set. This is due to the fact that our problem (in contrast with the ones presented in the current literature) is *constrained*. This constraint variability bound is in fact a bound on the variability of the dual optimizers, as one might have already guessed. This relation is clearly unveiled in the next proposition.

Proposition 1: Define the map of all constraints $\mathbf{d}^k(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ as $\mathbf{d}^k(\mathbf{x}) := (d_1^k(\mathbf{x}), \dots, d_m^k(\mathbf{x}))^\top$. The dual optimizers of (4) satisfy for $k \geq 0$

$$\|\boldsymbol{\lambda}^{k+1, \text{opt}} - \boldsymbol{\lambda}^{k, \text{opt}}\| \leq \|\mathbf{d}^{k+1}(\mathbf{x}^{k+1, \text{opt}}) - \mathbf{d}^k(\mathbf{x}^{k, \text{opt}})\| / \epsilon \leq \sqrt{m} \delta_d / \epsilon.$$

Proof: The proof is based on the optimality conditions of (4) and is made easier by the regularization term in the dual space (ϵ). The complementary slackness condition for (4) is¹

$$\lambda_j^{k, \text{opt}} (d_j^k(\mathbf{x}^{k, \text{opt}}) - \epsilon \lambda_j^{k, \text{opt}}) = 0, \quad \text{for } j = 1, \dots, m.$$

¹This can be shown in different ways, one of which is by finding the fixed points of (5) at given k as in [2].

Since $\lambda_j^{k, \text{opt}} \geq 0$, we have $\lambda_j^{k, \text{opt}} = \max\{0, d_j^k(\mathbf{x}^{k, \text{opt}}) / \epsilon\}$, or

$$\lambda_j^{k+1, \text{opt}} - \lambda_j^{k, \text{opt}} = (\max\{0, d_j^{k+1}(\mathbf{x}^{k+1, \text{opt}})\} - \max\{0, d_j^k(\mathbf{x}^{k, \text{opt}})\}) / \epsilon,$$

and finally

$$|\lambda_j^{k+1, \text{opt}} - \lambda_j^{k, \text{opt}}| = |\max\{0, d_j^{k+1}(\mathbf{x}^{k+1, \text{opt}})\} - \max\{0, d_j^k(\mathbf{x}^{k, \text{opt}})\}| / \epsilon \leq |d_j^{k+1}(\mathbf{x}^{k+1, \text{opt}}) - d_j^k(\mathbf{x}^{k, \text{opt}})| / \epsilon,$$

where the last inequality comes by direct calculation. Stacking these last inequalities together for all j and by Assumption 5 the claim follows. ■

We note that the proof of Proposition 1 is made possible by the employed regularization and a similar result would be more difficult to derive for a generic constrained problem. We are now ready for the convergence result. First of all, we define the Lipschitz constant $L_{\nu, \epsilon}$ [2, Lemma 3.4]

$$L_{\nu, \epsilon} = \sqrt{(L + \nu + M_d + M_\nu L_d)^2 + (M_d + \epsilon)^2}, \quad (6)$$

$$L_d = \|(L_1, \dots, L_m)\|_2,$$

where L and L_j are defined in Assumption 2, $M_d = \max_{\mathbf{x} \in X} \|\nabla_{\mathbf{x}} d_j^k(\mathbf{x})\|$ (for all j, k), and $M_\nu = \max \|\boldsymbol{\mu}(k)\|$, for all k^2 .

Theorem 1: Let $\boldsymbol{\zeta}(k) := (\mathbf{v}(k)^\top, \boldsymbol{\mu}(k)^\top)^\top$ be the stacked vector of the primal-dual variables generated using Algorithm 1. Let $\mathbf{z}^{k, \text{opt}} := (\mathbf{x}^{k, \text{opt}^\top}, \boldsymbol{\lambda}^{k, \text{opt}^\top})^\top$ be the primal-dual solution of the regularized problem (4) for a given choice of ν and ϵ . Let $\rho = \min\{\nu, \epsilon\}$ and let $L_{\nu, \epsilon}$ be defined as (6). Under Assumptions 1 till 5, if the stepsize is picked as $\alpha < 2\rho / L_{\nu, \epsilon}^2$, then convergence goes as

$$\|\boldsymbol{\zeta}(k+1) - \mathbf{z}^{k+1, \text{opt}}\| \leq r \|\boldsymbol{\zeta}(k) - \mathbf{z}^{k, \text{opt}}\| + \sqrt{\delta_{\mathbf{x}}^2 + m \delta_d^2 / \epsilon^2},$$

with linear convergence rate $r = \sqrt{1 - 2\rho\alpha + \alpha^2 L_{\nu, \epsilon}^2} < 1$.

Proof: (Sketch) The proof uses similar arguments as the one in Theorem 3.5 of [2] to establish

$$\|\boldsymbol{\zeta}(k+1) - \mathbf{z}^{k, \text{opt}}\| \leq r \|\boldsymbol{\zeta}(k) - \mathbf{z}^{k, \text{opt}}\|,$$

successively, it employs the triangle inequality to bound $\|\boldsymbol{\zeta}(k+1) - \mathbf{z}^{k+1, \text{opt}}\|$ as

$$\|\boldsymbol{\zeta}(k+1) - \mathbf{z}^{k+1, \text{opt}}\| \leq \|\boldsymbol{\zeta}(k+1) - \mathbf{z}^{k+1, \text{opt}}\| + \sqrt{\delta_{\mathbf{x}}^2 + \|\boldsymbol{\lambda}^{k+1, \text{opt}} - \boldsymbol{\lambda}^{k, \text{opt}}\|^2},$$

and finally uses Proposition 1 to link $\|\boldsymbol{\lambda}^{k+1, \text{opt}} - \boldsymbol{\lambda}^{k, \text{opt}}\|$ to δ_d , from which the claim is derived. ■

Theorem 1 says that Algorithm 1 produces a sequence converging linearly to the primal-dual optimizer of the regularized problem, up to a bounded error. This error depends on the variability of the problem in time. It is quite straightforward, to determine such a bound given Theorem 1 (by using the properties of geometric series),

$$\liminf_{k \rightarrow \infty} \|\boldsymbol{\zeta}(k+1) - \mathbf{z}^{k+1, \text{opt}}\| \leq \frac{1}{1-r} \sqrt{\delta_{\mathbf{x}}^2 + m \delta_d^2 / \epsilon^2}. \quad (7)$$

²We can show that either $\boldsymbol{\mu}(k)$ stays bounded, or we can project it onto a bounded set, since the optimal multipliers are bounded [2].

Remark 1: As we are using a regularized problem to solve the original optimization (1) to increase convergence speed, we expect to introduce a certain degree of suboptimality (i.e., $|f^{k,\text{opt}} - f^{k,*}|$) and constraint violation. These effects are quantified in [2, Lemma 3.3] for a fixed k , and not reported here for sake of space.

Remark 2: Theorem 1 is often only an existence result: it says that there exists a small enough stepsize α that induces convergence. However, since $L_{\nu,\epsilon}$ is difficult to determine in practice, this α needs to be determined by trial-and-error. These “small enough” results are typical in distributed optimization [18].

V. NUMERICAL CASE STUDY

We use now a case study to show the numerical properties of the proposed algorithm (the time-invariant detailed description is given in [2]). The network has a set of N users, sharing a set of m links (see Figure 1 for an example). A user $i \in N$ has a cost function $f_i^k(x_i)$ of its traffic rate $x_i \in \mathbb{R}$ as

$$f_i^k(x_i) = -\sigma_i(k) \log(1 + x_i), \quad \text{for } i = 1, \dots, N,$$

where $\sigma_i(k)$ is a time-varying non-negative scalar. Each user selects an origin-destination pair of nodes on this network and faces congestion as

$$c(\mathbf{x}) = \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x},$$

where $\mathbf{A} \in \mathbb{N}_{\{0,1\}}^{m \times N}$ is the matrix that encodes the network structure, that is: $A_{ji} = 1$ if the traffic of user i goes through link j and 0 otherwise. The coupling in the constraints has the form $\mathbf{A}_j \mathbf{x} \leq C_j$, for all j , for which C_j is the maximum capacity of link j . The multiuser problem is

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{i=1}^N -\sigma_i(k) \log(1 + x_i) + \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} \quad (8a)$$

$$\text{subject to} \quad \mathbf{A}_j \mathbf{x} \leq C_j, \quad \text{for } j = 1, \dots, m \quad (8b)$$

$$x_i \in X_i, \quad \text{for } i = 1, \dots, N, \quad (8c)$$

and fits the generic model problem (1).

We use the same parameters of [2], we set $\alpha = 0.02$, $\nu = 0.1$, $\epsilon = 0.01$, and we let the $\sigma_i(k)$ to vary as

$$\sigma_i(k+1) = \sigma_i(0)((1-\eta) + \eta \cos(2\pi qk + \varphi)^2), \quad \eta \in [0, 1],$$

where the $\sigma_i(0)$'s are selected as the fixed ones in [2], $q = 0.005$, and $\varphi = 2\pi\mathcal{U}_{[0,1]}$. Assumptions 1 till 5 are verified in the chosen setting, and the values of δ_x and δ_d can be tuned by the parameter η .

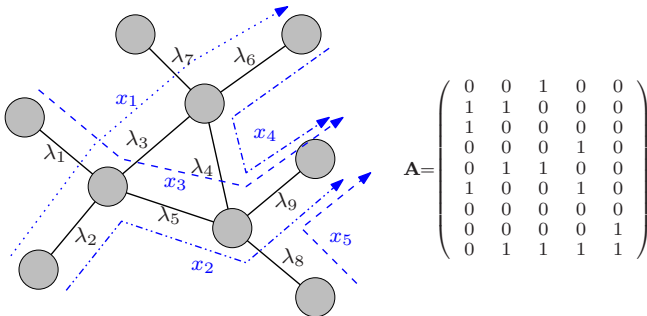


Fig. 1. The network with 5 users and 9 links used in the case study

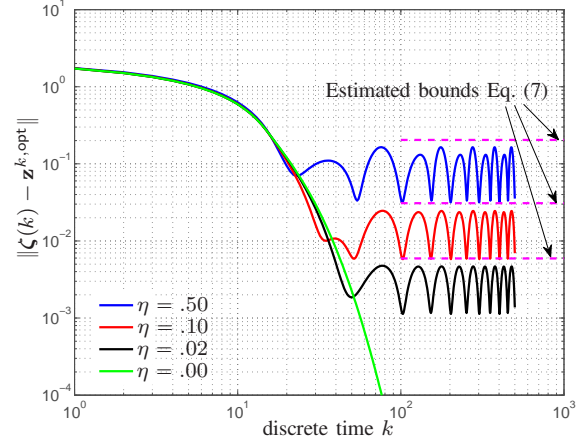


Fig. 2. Convergence results for different η and estimated theoretical bounds.

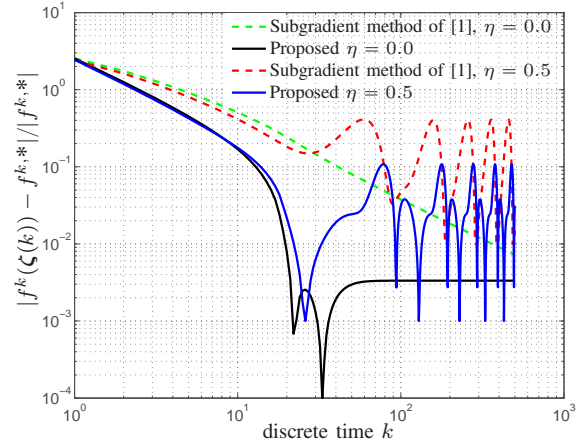


Fig. 3. Suboptimality results for different η and different distributed methods.

Convergence results to the primal-dual optimizer of the regularized problem are displayed in Figure 2 for different values of η (for $\eta = 0$ the problem becomes time-invariant and therefore convergence is exact). We also show the estimated bounds, computed by using (7) and by estimating the rate r by curve fitting on the linearly convergent part (see Remark 2). We notice that the bound is rather tight. The average relative error $\|\zeta(k) - \mathbf{z}^{k,\text{opt}}\| / \|\mathbf{z}^{k,\text{opt}}\|$ in the interval $k \in [100, 500]$ is .050, .008, and .002, for $\eta = .50$, .10, and .02, respectively.

Relative suboptimality w.r.t. the optimum of the original problem is shown in Figure 3, where we also compare the proposed method with the subgradient method of [1] (which converges to the optimum of the original problem in time-invariant cases but it is not proven to be convergent for time-varying scenarios). As we see, the proposed method converges faster to an approximate optimum, and it performs even better than [1] for $\eta > 0$.

Future research directions will consider Nesterov-type fast gradient methods and linearized versions of ADMM as [19] to obtain similar fast converging and low complexity solutions, as well as non-smooth cost functions, e.g., l_1 -regularizers, which are often employed in statistical signal processing.

REFERENCES

- [1] A. Nedić and A. Ozdaglar, “Subgradient Methods for Saddle-Point Problems,” *Journal of Optimization Theory and Applications*, vol. 142, no. 1, pp. 205 – 228, 2009.
- [2] J. Koshal, A. Nedić, and U. Y. Shanbhag, “Multiuser Optimization: Distributed Algorithms and Error Analysis,” *SIAM Journal on Optimization*, vol. 21, no. 3, pp. 1046 – 1081, 2011.
- [3] A. Simonetto, T. Keviczky, and M. Johansson, “A Regularized Saddle-Point Algorithm for Networked Optimization with Resource Allocation Constraints,” in *Proceedings of the 51st IEEE Conference on Decision and Control*, Maui, USA, December 2012, pp. 7476 – 7481.
- [4] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, “Consensus in Ad Hoc WSNs With Noisy Links— Part I: Distributed Estimation of Deterministic Signals,” *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 350 – 364, 2008.
- [5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers,” *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1 – 122, 2011.
- [6] A. Simonetto and G. Leus, “Distributed Maximum Likelihood Sensor Network Localization,” *IEEE Transactions on Signal Processing*, vol. 62, no. 6, pp. 1424 – 1437, 2014.
- [7] M. Kamgarpour and C. Tomlin, “Convergence Properties of a Decentralized Kalman Filter,” in *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico, December 2008, pp. 3205 – 3210.
- [8] P. Braca, S. Marano, V. Matta, and P. Willett, “Asymptotic Optimality of Running Consensus in Testing Binary Hypotheses,” *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 814 – 825, 2010.
- [9] F. S. Cattivelli and A. H. Sayed, “Diffusion Strategies for Distributed Kalman Filtering and Smoothing,” *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2069 – 2084, 2010.
- [10] S.-Y. Tu and A. H. Sayed, “Mobile Adaptive Networks,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 4, pp. 649 – 664, 2011.
- [11] D. Bajovic, D. Jakovetic, J. Xavier, B. Sinopoli, and J. M. F. Moura, “Distributed Detection via Gaussian Running Consensus: Large Deviations Asymptotic Analysis,” *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4381 – 4396, 2011.
- [12] R. L. G. Cavalcante and S. Stanczak, “A Distributed Subgradient Method for Dynamic Convex Optimization Problems Under Noisy Information Exchange,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 2, pp. 243 – 256, 2013.
- [13] F. Y. Jakubiec and A. Ribeiro, “D-MAP: Distributed Maximum a Posteriori Probability Estimation of Dynamic Systems,” *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 450 – 466, 2013.
- [14] Q. Ling and A. Ribeiro, “Decentralized Dynamic Optimization Through the Alternating Direction Method of Multipliers,” *IEEE Transactions on Signal Processing*, vol. 62, no. 5, pp. 1185 – 1197, 2014.
- [15] A. Simonetto and G. Leus, “Distributed Asynchronous Time-Varying Constrained Optimization,” in *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, USA, November 2014.
- [16] A. Simonetto, L. Kester, and G. Leus, “Distributed Time-Varying Stochastic Optimization and Utility-based Communication,” *Submitted to IEEE Transactions on Control of Network Systems*, 2014, available at <http://arxiv.org/abs/1408.5294>.
- [17] O. Devolder, F. Glineur, and Yu. Nesterov, “Double Smoothing Technique for Large-Scale Linearly Constrained Convex Optimization,” *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 702 – 727, 2012.
- [18] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Belmont, Massachusetts, 1997.
- [19] Q. Ling and A. Ribeiro, “Decentralized Linearized Alternating Direction Method of Multipliers,” in *Proceedings of the IEEE International Conference in Acoustics, Speech, and Signal Processing*, Firenze, Italy, May 2014, pp. 5484 – 5488.