2022 IEEE Conference on Control Technology and Applications (CCTA)
August 22-25, 2022. Trieste, Italy

Distributed CPU Scheduling Subject to Nonlinear Constraints

Mohammadreza Doostmohammadian, Alireza Aghasi, Apostolos I. Rikos, Andreas Grammenos,
Evangelia Kalyvianaki, Christoforos N. Hadjicostis, Karl H. Johansson, Themistoklis Charalambous

***Abstract*— This paper considers a network of collaborating agents for local resource allocation subject to nonlinear model constraints. In many applications, it is required (or desirable) that the solution be anytime feasible in terms of satisfying the sum-preserving global constraint. Motivated by this, sufficient conditions on the nonlinear mapping for anytime feasibility (or non-asymptotic feasibility) are addressed in this paper. For the two proposed distributed solutions, one converges over directed weight-balanced networks and the other one over undirected networks. In particular, we elaborate on uniform quantization and discuss the notion of ε-accurate solution, which gives an estimate of how close we can get to the exact optimizer subject to different quantization levels. This work, further, handles general (possibly non-quadratic) strictly convex objective functions with application to CPU allocation among a cloud of data centers via distributed solutions. The results can be used as a coordination mechanism to optimally balance the tasks and CPU resources among a group of networked servers while addressing quantization or limited server capacity.**

***Index Terms*— multi-agent systems, sum-preserving resource allocation, distributed optimization, anytime feasibility**

I. Introduction

Allocation of resources and utilities over a multi-agent network is considered in this paper. This problem finds application in different control scenarios ranging from coverage control and task allocation to electricity power scheduling [1]–[4]. The general idea is to optimally determine the allocated amount of resources from a fixed total among a group of users or agents. Recently, the emergence of Internet-of-Things (IoT) has motivated distributed solutions over networks, where agents locally solve the problem in their neighborhood with no direct knowledge of distant agents or global information. In many large-scale applications, localized processing, and cloud computing motivate such *distributed* resource allocation strategies instead of traditional centralized solutions. Example applications include managing the balance between energy resources and energy demand over the smart grid, allocating the fixed amount of tasks over a multi-agent network, or assigning the amount of computing load to the network of data servers [5]–[7]. In the context of resource management in Cloud infrastructures, we particularly focus on the latter application where some networked data centers (computing nodes) need to be assigned by CPU cycles (resources) in a distributed fashion. The total sum of resources is limited and fixed and the computing nodes follow a distributed algorithm to locally balance the CPU utilization by local information-exchange with other nodes. In general, in CPU scheduling the jobs are allocated in quantized (or discrete) values. Further, other than quantized CPU allocation and in general applications, the data-sharing setup is typically involved with bandwidth efficiency and limited capacity concerns, and thus, mandates quantized information exchange. This quantization issue needs to be addressed in general networked scenarios.

M. Doostmohammadian and T. Charalambous are with the School of Electrical Engineering at Aalto University, Finland, Email: firstname.lastname@aalto.fi. A. Aghasi is with Georgia State University, GA, USA, email: aaghasi@gsu.edu. A. Grammenos is with the Department of Computer Science and Technology, University of Cambridge, Cambridge, and the Alan Turing Institute, London, UK. E-mail: ag926@cl.cam.ac.uk. E. Kalyvianaki is with the Department of Computer Science and Technology, University of Cambridge, Cambridge, UK. Email: ek264@cl.cam.ac.uk. C. N. Hadjicostis and T. Charalambous are with the Department of Electrical and Computer Engineering, University of Cyprus, Cyprus, email: {chadjic,charalambous.themistoklis}@ucy.ac.cy. A. I. Rikos and K. H. Johansson are with the Division of Decision and Control Systems, KTH Royal Institute of Technology, Sweden, email: {rikos,kallej}@kth.se.

This work is supported in part by the European Commission through the H2020 Project Finest Twins under Agreement 856602.

A. The problem

The problem of sum-preserving resource allocation is in the following standard form,

$$\min_{\mathbf{x}} \quad F(\mathbf{x}) = \sum_{i=1}^{n} f_i(\mathbf{x}_i) \qquad \text{s.t.} \quad \sum_{i=1}^{n} \mathbf{x}_i = b, \qquad \mathbf{x}_i \in \mathcal{X}_i \tag{1}$$

with $\mathbf{x}_i, b \in \mathbb{R}$, $f_i : \mathbb{R} \to \mathbb{R}$, and $\mathcal{X}_i \subseteq \mathbb{R}$ representing a range of admissible values for states $\mathbf{x}_i$. The latter represents the so-called *box constraints* for $\mathbf{x}_i \in \mathbb{R}$ in the form $\mathbf{x}_i \in [m_i \ M_i]$. As discussed later, the problem can be extended to the case where $\mathbf{x}_i \in \mathbb{R}^{d_i}$ and $\mathcal{X}_i \subseteq \mathbb{R}^{d_i}$ where the *local constraints* are defined in the form [8],

$$\mathcal{X}_i = \{\mathbf{x} \in \mathbb{R}^{d_i} : g_i^j(\mathbf{x}) \leq 0, j = 1, \dots, p_i\} \tag{2}$$

with $g_i^j : \mathbb{R}^{d_i} \to \mathbb{R}$ as convex and twice-differentiable functions on $\mathcal{X}_i$. In general, the sum-preserving global constraint can be also of higher-order form with $\mathbf{x}_i, b \in \mathbb{R}^m$.

Among the existing solutions, other than the classic linear ones [1]–[3], [9], the work by [8] suggests a *local reallocation* optimization algorithm at every iteration to address all-time feasibility. On the other hand, there exist many primal-dual solutions that do not guarantee primal-feasibility (or anytime-feasibility), but instead asymptotically reach feasibility [10], [11]. Many existing works focus on linear solutions with ideal communication and actuation at the node dynamics. However, in reality, multi-agent systems (e.g., mobile robotic networks, connected generators over the smart grid, or collaborating distributed data centers) are subject to practical nonlinearities. For example, the shared

2022 IEEE Conference on Control Technology and Applications (CCTA) | 978-1-6654-7338-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/CCTA49430.2022.9966048

information for task/CPU scheduling among data centers (or servers) are quantized [6] or robot actuators performing coverage allocation are subject to saturation [12]. The work by [13] further addresses the notion of ε-accuracy over star multi-agent networks, i.e., the number of communication bits needed to reach the ε-neighborhood of the exact optimizer. In the same line of research, Ref. [14] considers unconstrained distributed optimization via single-bit information-exchange over limited-capacity communication networks. Other than the mentioned nonlinearities imposed by the nature of the actuation and communication, other kinds of nonlinearities are added for the purpose of improving the convergence rate or to reach the optimal value in (prescribed) fixed-time [15] or finite-time [14]. These further motivate the nonlinear model consideration in this paper.

B. Main Contributions

In this paper, the main contributions are: (i) we address possible nonlinearities in the dynamics of the agents due to imperfect actuation and limited communication capabilities. This is motivated, for example, by limited and/or quantized range of action in actuators and, similarly, possible clipping and quantization in communication channels. Other node-based and link-based nonlinearities are further applicable to address, for example, robustness to disturbances and pre-defined (or fixed) convergence time. Some examples regarding nonlinear *consensus* protocols are discussed in [16]–[18]. In this paper, we discuss convergence subject to both sector-based and non-sector-based nonlinearities, for example, logarithmic quantization and uniform quantization. (ii) We show exact convergence under sector-based nonlinearities, while for uniform quantization (as an example of non-sector-based nonlinearity) we prove convergence to the ε-neighborhood of the optimizer. In the latter case, the concept of ε-accuracy is considered. This notion implies the quantization level to ensure reaching ε-neighborhood of the optimal point. On the other hand, for a given quantization level (or the number of bits) one can address the best ε-accurate solution that can be achieved while satisfying the feasibility constraint at all times. In particular, (iii) we discuss the application in resource allocation and CPU scheduling over networked servers [6]. (iv) Unlike some works restricted to quadratic costs [6], this work can address general strictly (and strongly) convex cost functions (possibly non-quadratic) due to, e.g., the use of different barrier functions and penalty functions addressing the local constraints to advance the quadratic cost model in [6]. The results can further address different types of practical nonlinearities imposed on the coordination mechanism among the servers, for example, saturated capacity, quantization scheme of different sizes, and fast sign-based solutions. Further, (v) we advance the assumption in [6], [8] by considering uniform-connectivity over time instead of all-time connected networks.

C. Some Preliminary Concepts

Following the Karush-Kuhn-Tucker (KKT) condition, the following lemma finds the condition on the optimizer $\mathbf{x}^*$ as the solution of (1). Define the gradient vector $\nabla F = [\partial_{x_1} f_1(\mathbf{x}_1); \dots; \partial_{x_n} f_n(\mathbf{x}_n)]$.

Lemma 1: The optimizer $\mathbf{x}^*$ as the solution of (1) is in the form $\nabla F \in \text{span}(\mathbf{1}_n)$, i.e., $\partial_{x_j} f_j(\mathbf{x}_j^*) = \partial_{x_i} f_i(\mathbf{x}_i^*)$ for all i, j.

See the proof and more details in [16], [17]. Note that the above lemma holds for the equality-constraint problem (1) without local constraints (2). The box constraints ($d_i = 1$) are addressed by additive penalty terms discussed later in Section II-A. One can reformulate the problem and extend it to *weighted*-sum-preserving constraints as follows,

$$\begin{aligned} \min_{\mathbf{y}} \quad & \widetilde{F}(\mathbf{y}) = \textstyle\sum_{i=1}^n \widetilde{f}_i(\mathbf{y}_i) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^n a_i \mathbf{y}_i = b, \qquad \mathbf{y}_i \in \mathcal{Y}_i \end{aligned} \tag{3}$$

By change of variable in the form $a_i \mathbf{y}_i = \mathbf{x}_i$, the above problem takes the form (1) and follows similar solution. Notice that a_is need to satisfy composition conditions [19, Section 3.2.4] to ensure convexity of the local sets $\mathcal{X}_i$s after change of variables (as a composition of $\mathcal{Y}_i$s and linear transformation $a_i \mathbf{y}_i = \mathbf{x}_i$).

D. The Assumptions

The following assumptions on the cost functions hold throughout the paper:

1) The local cost functions f_i are strictly (or strongly) convex and smooth[1].
2) The feasible solution set of problem (1) is non-empty and compact.

The first assumption allows to address the unique optimizer via KKT conditions and is widely considered in the literature. The second assumption is particularly challenging if there are different local constraints $\mathbf{x}_i \in \mathcal{X}_i$ and the combination of these $\mathcal{X}_i$s and the sum-preserving constraint $\sum_{i=1}^n \mathbf{x}_i = b$ needs to be feasible. Algorithms are proposed in [1], [8] to render feasible initialization for such cases.

The following assumptions (for the proof of convergence) hold on possible nonlinearities on the agents' dynamics:

(i) The nonlinearities satisfy $0 < \underline{\alpha} \leq \frac{h(z)}{z} \leq \overline{\alpha}$ (sector-based), i.e., they are strongly sign-preserving and monotonically non-decreasing nonlinear mapping.
(ii) $h(z)$ is an odd mapping, i.e., $h(-z) = -h(z)$ and $h(0) = 0$.

The following are standard assumptions on the multi-agent network (or the graph topology) in the consensus literature:

(I) The network is undirected with symmetric weights.
(II) The network is uniformly-connected or B-connected, i.e., the union of the networks over every time-interval B is connected.

Note that for some special cases we relax the assumption (I) to general *weight-balanced directed networks*. In terms of network connectivity, Assumption (II) advances existing solutions [8], [13] to dynamic (possible disconnected) networks, i.e., the cases for which the network might be

[1]For the proof of convergence only strict convexity is used. In order to determine the *rate of convergence* v-strongly convex assumption is adopted.

disconnected during some time instances but their union is connected over a finite time interval B. This occurs in mobile multi-agent applications with limited communication resources where the links over the network come and go as the agents (e.g., robots) move in and out of the communication range of the other agents.

E. Paper Organization

The rest of the paper is as follows. Section II introduces the distributed solutions subject to possible nonlinearities. In Section III, the convergence of uniform quantization (as a non-sector-based nonlinearity) and the notion of ε-accurate solution are discussed. Section IV provides an example application in CPU scheduling and related simulations. Finally, Section V concludes the paper.

II. Nonlinear Distributed Solutions

Two nonlinear distributed gradient-Laplacian solutions are considered in this paper. The continuous-time (CT) solutions are in the form,

$$\dot{\mathbf{x}}_i = \eta \sum_{j\in\mathcal{N}_i} W_{ji}(t) h(\partial_{x_j} f_j(t) - \partial_{x_i} f_i(t)), \tag{4}$$

$$\dot{\mathbf{x}}_i = \eta \sum_{j\in\mathcal{N}_i} W_{ji}(t) (h(\partial_{x_j} f_j(t)) - h(\partial_{x_i} f_i(t))), \tag{5}$$

The CT solutions find application, e.g., in economic dispatch problem and power generation scheduling, see [16], [17]. In discrete-time (DT),

$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) + \overline{\eta} \sum_{j\in\mathcal{N}_i} W_{ji}(k) h(\partial_{x_j} f_j(k) - \partial_{x_i} f_i(k)), \tag{6}$$

$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) + \overline{\eta} \sum_{j\in\mathcal{N}_i} W_{ji}(k) (h(\partial_{x_j} f_j(k)) - h(\partial_{x_i} f_i(k))), \tag{7}$$

with $h(\cdot)$ representing possible node-based or actuation nonlinearity (protocols (4) and (6)) or link-based or communication nonlinearity (protocols (5) and (7)) at the agents' dynamics. This nonlinear function could be either (i) imposed by the nature of the agents' dynamics, e.g., due to control saturation and/or quantization, or (ii) added purposefully by the designer, e.g., to improve the convergence rate and/or robustness properties with respect to noise and disturbances by using sign-based solutions.

Lemma 2 (Convergence): Let the assumptions in Section I-D hold. The continuous-time solutions (4)-(5) and discrete-time solutions (6)-(7) converge to the exact optimizer $\mathbf{x}^*$ as the solution of problem (1).

The detailed proof for convergence and uniqueness of the solution under CT dynamics (4)-(5) are given in [16], [17] assuming general strictly convex cost functions. The proof can be extended to the DT case using the following lemma.

Lemma 3: Let Assumptions (1)-(2) hold. Consider two points $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$, and $\delta\mathbf{x} := \mathbf{x}_1 - \mathbf{x}_2$. There exist $0 < \alpha < 1$ and $\widehat{\mathbf{x}} = \alpha\mathbf{x}_1 + (1-\alpha)\mathbf{x}_2$ such that,

$$F(\mathbf{x}_1) = F(\mathbf{x}_2) + \nabla F(\mathbf{x}_2)^\top \delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^\top \nabla^2 F(\widehat{\mathbf{x}})\delta\mathbf{x}. \tag{8}$$

Then,

$$F(\mathbf{x}_1) \geq F(\mathbf{x}_2) + \nabla F(\mathbf{x}_2)^\top \delta\mathbf{x} + v\delta\mathbf{x}^\top\delta\mathbf{x}, \tag{9}$$

$$F(\mathbf{x}_1) \leq F(\mathbf{x}_2) + \nabla F(\mathbf{x}_2)^\top \delta\mathbf{x} + u\delta\mathbf{x}^\top\delta\mathbf{x}. \tag{10}$$

Define the Lyapunov function as the residual $\overline{F}(k) = F(\mathbf{x}(k)) - F(\mathbf{x}^*)$. For two consecutive (feasible) states $\mathbf{x}(k+1), \mathbf{x}(k)$ define $\delta\mathbf{x}(k) := \mathbf{x}(k+1) - \mathbf{x}(k)$. To satisfy $\overline{F}(k+1) \leq \overline{F}(k)$, from Lemma 3 one can prove that,

$$\nabla F^\top \delta\mathbf{x} + u\delta\mathbf{x}^\top\delta\mathbf{x} \leq 0. \tag{11}$$

Recall that for a weight-balanced connected graph $\mathcal{G}$ and its associated Laplacian matrix $L_g = D - W$ with $D = \text{diag}[\sum_{j\in\mathcal{N}_i} W_{ji}] = \text{diag}[\sum_{j\in\mathcal{N}_i} W_{ij}]$, define λ_n, λ_2 as the largest and smallest non-zero eigenvalue of L_g. For $\mathbf{x} \in \mathbb{R}^n$ and $\overline{\mathbf{x}} := \mathbf{x} - \frac{\mathbf{1}_n^\top \mathbf{x}}{n}\mathbf{1}_n$,

$$\mathbf{x}^\top L_g \mathbf{x} = \overline{\mathbf{x}}^\top L_g \overline{\mathbf{x}}, \tag{12}$$

$$\lambda_2 \|\overline{\mathbf{x}}\|_2^2 \leq \mathbf{x}^\top L_g \mathbf{x} \leq \lambda_n \|\overline{\mathbf{x}}\|_2^2 \tag{13}$$

Using (12)-(13) and substituting $\delta\mathbf{x}$ from Eq. (6)-(7), further assume strongly convex functions satisfying $2v \leq \partial_x^2 f_i(\mathbf{x}_i) \leq 2u$ and sector-based nonlinearities satisfying $\underline{\alpha} \leq \frac{h(\mathbf{z})}{\mathbf{z}} \leq \overline{\alpha}$. Then, similar Lyapunov analysis as in [16], [17], one can prove convergence for any step-rate $\overline{\eta} > 0$ satisfying,

$$\overline{\eta} \leq \frac{2\underline{\alpha}\lambda_2}{u\lambda_n^2\overline{\alpha}}. \tag{14}$$

Then, the linear convergence rate follows as,

$$\frac{\overline{F}(k+1)}{\overline{F}(k)} \leq 1 - \overline{\eta}v(\underline{\alpha}\lambda_2 - \frac{u}{2}\lambda_n^2\overline{\alpha}\overline{\eta}). \tag{15}$$

The proof can be easily extended to B-connected graphs with L_g as the Laplacian matrix of the union graph over the time-interval B, i.e., considering $\frac{\overline{F}(k+B)}{\overline{F}(k)}$ in the above formula. See [16], [17] for more information.

Remark 1: In problem (3), following the KKT conditions, the optimizer satisfies $\nabla\widehat{F}(\mathbf{y}^*) \in \text{span}(\mathbf{a})$.

A. The Local Constraints

The local constraints $\mathbf{x}_i \in \mathcal{X}_i$ can be addressed via adding penalty functions [20] or barrier functions [8] to the local costs f_i. Some commonly used penalty functions to address the box-constraints are discussed here. The cost function is updated as,

$$f_i^c(\mathbf{x}_i) = f_i(\mathbf{x}_i) + c[\mathbf{x}_i - M_i]^+ + c[m_i - \mathbf{x}_i]^+ \tag{16}$$

with $[u]^+ = \max\{u, 0\}$ and $c > 0$ penalizing the deviation from the admissible range of values. It is known that the solution of this penalized case can become arbitrary close to the exact optimizer by choosing c sufficiently small [21]. This non-smooth function can be substituted by the following smooth equivalents [21], [22],

$$L(u,\mu) = \frac{1}{\mu}\log(1+\exp(\mu u)) \tag{17}$$

$$[u]^{+\kappa} = ([u]^+)^\kappa,\ \kappa > 1, \kappa \in \mathbb{N} \tag{18}$$

It can be shown that the maximum gap between the two functions $[u]^+$ and (17) inversely scales with μ, i.e.,

$$L(u,\mu) - [u]^+ \leq \frac{1}{\mu}$$

and the two can become arbitrarily close by selecting μ sufficiently large [23]. In general, for local constraints in the form (2), the penalty functions can be written as $c\sum_{j=1}^{p_i}[g_i^j(\mathbf{x})]^+$. Similarly, some barrier functions $\mathcal{B}_i^j(\mathbf{x}_i)$ are proposed in the literature [8], [24] to be added to the local costs in the form $f_i^c(\mathbf{x}_i) = f_i(\mathbf{x}_i) + c\sum_{j=1}^{p_i}\mathcal{B}_i^j(\mathbf{x}_i)$. Following (2), $\mathcal{B}_i^j(\mathbf{x}_i)$ is defined real valued for $\mathbf{x}_i \in \mathcal{X}_i$, i.e., $g_i^j(\mathbf{x}_i) < 0$, and following Assumption (1),

1) The barrier function needs to be convex and smooth.
2) If $g_i^j(\mathbf{x}_i) \to 0^-$ (i.e., the function approaching zero from negative values), then $\mathcal{B}_i^j(\mathbf{x}_i) \to \infty$.

Some standard example barrier functions are given as [24],

$$\mathcal{B}_i^j(\mathbf{x}_i) = -\log(-g_i^j(\mathbf{x}_i)) \quad (19)$$

$$\mathcal{B}_i^j(\mathbf{x}_i) = \frac{-1}{g_i^j(\mathbf{x}_i)} \quad (20)$$

These are respectively known as logarithmic and inverse barrier functions.

B. The global Constraint: Anytime Feasibility

As mentioned in the introduction, many applications mandate solution feasibility at all times, i.e., the global constraint $\sum_{i=1}^n \mathbf{x}_i = b$ hold at all times along the solution dynamics. This implies that at any termination time, the resulting outcome $\mathbf{x}$ of the proposed anytime-feasible protocols (4)-(7) satisfy $\sum_{i=1}^n \mathbf{x}_i = b$. In application, e.g., the economic dispatch problem, this means that the produced power and the demand are balanced at all times to avoid system breakdown [1], [8]. Similarly, in balancing the CPU utilization among a group of data centers, the algorithm needs to be feasible at all times such that the allocated CPU resources meet the workloads required by the servers [6], [7].

Lemma 4 (Anytime Feasibility): Suppose that Assumption (2), Assumption (ii), and Assumption (I) hold. By any feasible initialization, the state of agents remain feasible under the CT dynamics (4)-(5) for all $t > 0$ and under the DT dynamics (6)-(7) for all $k \geq 1$.

The proof for CT case over uniformly-connected undirected graphs is discussed in [16], [17]. For the DT case, the proof similarly follows. First, note that from Assumption (2), the feasible solution exists. For protocol (6),

$$\sum_{i=1}^n \mathbf{x}_i(k+1) = \sum_{i=1}^n \mathbf{x}_i(k) + \overline{\eta}\sum_{i=1}^n\sum_{j\in\mathcal{N}_i} W_{ji}(k)h(\partial_{x_j}f_j(k) - \partial_{x_i}f_i(k)), \quad (21)$$

Following Assumption (ii) and Assumption (I), the last term is equal to zero. This is because for two neighboring agents i, j, we have $W_{ij} = W_{ji}$ and

$$h(\partial_{x_j}f_j(k) - \partial_{x_i}f_i(k)) = -h(\partial_{x_i}f_i(k) - \partial_{x_j}f_j(k)).$$

The feasibility proof of (7) for undirected graphs similarly follows. For link-based nonlinearities (5) and (7) one can extend the proof even to weight-balanced directed graphs.

Corollary 1: For protocols (5) and (7) over a weight-balanced graph,

$$\sum_{i=1}^n \mathbf{x}_i(k+1) = \sum_{i=1}^n \mathbf{x}_i(k) + \overline{\eta}\sum_{i=1}^n\sum_{j\in\mathcal{N}_i} W_{ji}(k)h(\partial_{x_j}f_j(k)) - h(\partial_{x_i}f_i(k)), \quad (22)$$

Recall that for a weight-balanced graph $\mathcal{G}$ and its associated Laplacian matrix L_g, we have $\mathbf{1}_n^\top L_g\mathbf{z} = 0$, where $\mathbf{z} \in \mathbb{R}^n$ and $\mathbf{1}_n$ as the vector of 1s. Now considering $\mathbf{z} = [h(\partial_{x_1}f_1(k)); \ldots; h(\partial_{x_n}f_n(k))]$, the last term in (22) is zero and Corollary 1 follows.

III. Quantization and ε-Accuracy

In this section, we compare the convergence for two cases: sector-based nonlinearities satisfying Assumption (i)-(ii), and sign-preserving (but not strongly) odd nonlinear mapping. Note that the main difference of the two cases is that for the second case $\frac{dh}{dx}(0) = 0$ while for the first case $\frac{dh}{dx}(0) > 0$. In particular, we consider logarithmic quantization versus uniform quantization respectively as examples of the first and second case. Following Lemma 2, for sector-based nonlinearities the exact convergence is achieved, i.e., substituting the strongly sign-preserving function $h(z) = \text{sgn}(z)\exp\left(q\left[\frac{\log(|z|)}{q}\right]\right)$ in (4)-(7) the solution reaches the exact optimizer of (1). In contrast, for uniform quantization, one can define ε-accuracy as a trade-off between the quantization level and convergence to the ε-neighborhood of the exact optimizer $\mathbf{x}^*$. We consider nonlinear CT protocol (5) and DT protocol (7) with $h(\partial_{x_i}f_i(k)) = q\left[\frac{\partial_{x_i}f_i(k)}{q}\right]$ with $[\cdot]$ as rounding to the nearest integer and q as the quantization level. Note that, from the definition, for x_i satisfying $-0.5q \prec \partial_{x_i}f_i - \partial_{x_i}f_i^* \prec 0.5q$ we have $h(\partial_{x_i}f_i) = h(\partial_{x_i}f_i^*)$ and for the optimizer we have $\partial_{x_j}f_j^* = \partial_{x_i}f_i^*$. Define a new variable $\xi(\mathbf{x}) := \nabla F(\mathbf{x}) - \frac{\sum_{i=1}^n \partial_{x_i}f_i}{n}\mathbf{1}_n$. Then, from the definition,

$$\nabla F - \nabla F^* = \xi + \frac{\sum_{i=1}^n \partial_{x_i}f_i}{n}\mathbf{1}_n - \nabla F^* \quad (23)$$

$$= \xi + \frac{\sum_{i=1}^n \partial_{x_i}f_i}{n}\mathbf{1}_n - \frac{\sum_{i=1}^n \partial_{x_i}f_i^*}{n}\mathbf{1}_n \quad (24)$$

where we simplified the notation as $\nabla F(\mathbf{x}^*) = \nabla F^*$ and $\partial_{x_i}f_i(x_i^*) = \partial_{x_i}f_i^*$. Recall the following lemma.

Lemma 5: For $\mathbf{z} \in \mathbb{R}^n$, $\overline{\mathbf{z}} := \mathbf{z} - \frac{\mathbf{1}_n^\top\mathbf{z}}{n}\mathbf{1}_n$, and laplacian matrix L of a weight-balanced graph: $\mathbf{z}^\top L\mathbf{z} = \overline{\mathbf{z}}^\top L\overline{\mathbf{z}}$.

Putting $L = I_n$ and $\mathbf{z} = \nabla F - \nabla F^*$ in the above lemma along with (24),

$$\xi^\top\xi = (\nabla F - \nabla F^*)^\top(\nabla F - \nabla F^*). \quad (25)$$

For $|\partial_{x_i} f_i - \partial_{x_i} f_i^*| < 0.5q$ (or $|\partial_{x_i} f_i - \partial_{x_j} f_j| < q$) we have,

$$\xi^\top \xi < 0.25 q^2 \mathbf{1}_n^\top \mathbf{1}_n = 0.25 n q^2. \tag{26}$$

From Lemma 3, substituting $\mathbf{x}_1 = \mathbf{x}$ and $\mathbf{x}_2 = \mathbf{x}^*$ we get,

$$\delta\mathbf{x}^\top \nabla F^* + v\delta\mathbf{x}^\top \delta\mathbf{x} \le \overline{F} \le \delta\mathbf{x}^\top \nabla F^* + u\delta\mathbf{x}^\top \delta\mathbf{x} \tag{27}$$

It is clear that for any two feasible states $\delta\mathbf{x}^\top \mathbf{1}_n = 0$ and,

$$\delta\mathbf{x}^\top \nabla F^* = \delta\mathbf{x}^\top \xi(\mathbf{x}^*) = 0, \tag{28}$$

since $\xi(\mathbf{x}^*) = \mathbf{0}_n$ from the definition. Further, following the results in [1] one can show that for any feasible state the residual $\overline{F}(\mathbf{x}) = F(\mathbf{x}) - F(\mathbf{x}^*)$ satisfies,

$$\frac{1}{4u}\xi^\top \xi \le \overline{F} \le \frac{1}{4v}\xi^\top \xi \tag{29}$$

where we dropped the dependence on $\mathbf{x}$ for notation simplicity. Eq. (28)-(29) along with Lemma 3 result in the following.

Lemma 6: Let Assumptions (1)-(2) hold and $2v \le \partial_x^2 f_i(\mathbf{x}_i) \le 2u$. Then,

$$v\|\mathbf{x} - \mathbf{x}^*\|_2^2 \le \overline{F} \le u\|\mathbf{x} - \mathbf{x}^*\|_2^2, \tag{30}$$

$$\frac{\|\xi\|_2}{2u} \le \|\mathbf{x} - \mathbf{x}^*\|_2 \le \frac{\|\xi\|_2}{2v}. \tag{31}$$

From (31) and (26) and given quantization level q,

$$\|\mathbf{x} - \mathbf{x}^*\|_2 \le \frac{\|\xi\|_2}{2v} < \frac{\sqrt{n}q}{4v} = \varepsilon. \tag{32}$$

This gives an estimate that how close we can get to the optimizer $\mathbf{x}^*$ for uniform quantization with level q, i.e., the so-called ε-accuracy. For a given demanded accuracy level ε, any quantization level $q > \frac{4v\varepsilon}{\sqrt{n}}$ may not guarantee such ε-accuracy and should be redesigned. One can find similar ε-bound for the node-based CT protocol (4) and DT protocol (6) following the same line of reasoning.

Remark 2: Note that the proposed nonlinear solutions are not limited to the quadratic cost model discussed in [6]. In general, any cost function satisfying Assumption (1) is valid in this work. Therefore, although the consensus-based solution in [6] reaches the exact optimizer for quadratic costs, it is not applicable for general *non-quadratic costs*. Further, the proposed solutions can address penalty and barrier functions discussed in Section II-A which are non-quadratic in general. On the other hand, the proposed protocols (4)-(7) can address other types of sector-based nonlinearities with exact optimality. Solutions based on fixed-time convergent algorithms can also be discussed as in [15], [17].

IV. Possible Applications and Simulations

A. CPU Scheduling in Data Centers

Consider the problem of balancing the CPU utilization over a cloud of $n = 12$ data servers in order to optimally assign the CPU resources to the workloads [5], [6]. The CPU costs at each node follow the quadratic form,

$$f_i(\mathbf{x}_i) = \frac{1}{2}\pi_i(\mathbf{x}_i - \frac{\rho_i + u_i}{\pi_i})^2 \tag{33}$$

with scalar $\pi_i > 0$ representing the capacity of node i, $\rho_i \in \mathbb{R}$ as the number of CPU cycles needed, and $u_i \in \mathbb{R}$ as

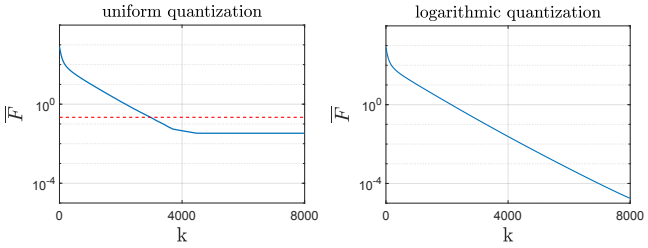

Fig. 1. The residual under two quantization approaches: (left) uniform, and (right) logarithmic quantization with level $q = 1$. Logarithmic quantizer as a sector-based nonlinearity is "strongly" sign-preserving as $\lim_{z\to 0} \frac{h(z)}{z} \ge (1 - \frac{q}{2}) > 0$ and the residual converges to zero. In contrast, the uniform quantizer with $\frac{h(z)}{z} = 0$ for $-\frac{h}{2} < z < \frac{h}{2}$ results in steady-state residual and converges to the ε-neighborhood of the exact optimizer defined by Eq. (32) and represented by the red dashed line on the left figure.

the number of occupied cycles due to predicted or known utilization from already running tasks on the server i (see more details in [5], [6]). For the simulation we choose $\pi_i = 2$, random $\rho_i, u_i \in [0\ 50]$ and assume scalar box constraints on the workloads/jobs at each node as,

$$m_i = 0 \le \mathbf{x}_i \le 100 = M_i \tag{34}$$

These constraints are addressed via quadratic penalty function (18) with $\kappa = 2$. Each node locally computes the optimal proportion of its workload out of $b = \sum_{i=1}^n (\rho_i + u_i) = 563$. The communication network is considered as a simple undirected cyclic network. Let assume admissible quantization level $q = 0.125$. Substituting $v = 1$ in Eq. (32), the solution under the nonlinear (uniformly-quantized) protocol (6) (for sufficiently small $\overline{\eta}$) is guaranteed to reach the ε-neighborhood of the optimizer $\mathbf{x}^*$ satisfying,

$$\|\mathbf{x} - \mathbf{x}^*\|_2 < \frac{0.125\sqrt{12}}{4} = \varepsilon. \tag{35}$$

Comparison between logarithmic quantization and uniform quantization is shown in Fig. 1.

B. Non-Quadratic Cost Model

As mentioned in the introduction, in contrast to consensus-based solutions that only consider quadratic cost functions [6], the proposed nonlinear solution in this paper can solve resource scheduling with non-quadratic cost models. As an example, the cost function can be in the form [25],

$$\sum_{i=1}^n f_i(\mathbf{x}_i) = \sum_{i=1}^n \omega_i(\mathbf{x}_i - \alpha_i)^4 \tag{36}$$

with random $\alpha_i \in [-2\ 4]$, $\omega_i \in [0\ 1]$. Further, the box constraints $-2 \le \mathbf{x}_i \le 5$ can be addressed by non-quadratic (logarithmic) penalty functions (17) with $\mu = 1$. For this simulation, actuation saturation (protocol (6)) is compared with the linear solution in Fig. 2(left). Such clipping may occur due to the maximum capacity at nodes, for example, because of some resource utilization due to previous tasks still being processed. In general, linear dynamics to solve the resource allocation converge slowly and asymptotically. To improve the convergence rate and to reach fixed-time convergence, sign-based solutions can be adopted. It is known that nonlinear dynamics in the form $\dot{\mathbf{z}} = \mathrm{sgn}^{\mu_1}(\mathbf{z}) + \mathrm{sgn}^{\mu_2}(\mathbf{z})$

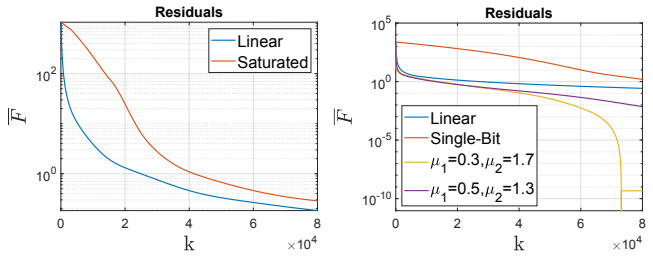

Fig. 2. (left) This figure compares the evolution of the residuals under the linear solution and the node-based protocol (6) subject to saturation level equal to 20. (right) The solution under linear and different nonlinear sign-based solutions are shown. Adding sign-based nonlinearities can improve the convergence rate as compared to the linear and single-bit solutions.

converge to the equilibrium in fixed (or prescribed) time [15]. Choosing nonlinear function $h(\mathbf{z}) = \text{sgn}^{\mu_1}(\mathbf{z}) + \text{sgn}^{\mu_2}(\mathbf{z})$ one can improve the convergence rate of the proposed protocols (4)-(7) to reach faster convergence as compared to the existing linear solutions [9]. The simulation results are shown in Fig. 2 for two cases with $\mu_1 = 0.5, \mu_2 = 1.3$ and $\mu_1 = 0.3, \mu_2 = 1.7$ for protocol (6) along with the single-bit protocol by [14] (with $\eta = 3 \times 10^{-5}$). Due to non-Lipschitz continuity of the sign-based solutions, in *discrete-time*, the steady-state residual is biased (known as the so-called *chattering* phenomena). This bias can be reduced by decreasing the step rate η.

V. Discussions and Concluding Remarks

This paper considers node-based and link-based nonlinearities on the agents' dynamics to optimally solve resource allocation subject to global sum-preserving constraints and local box constraints. In particular, the application to CPU scheduling subject to logarithmic quantization (sector-based nonlinearity) and uniform quantization (non-sector-based nonlinearity) are compared and for the latter ε-accuracy is addressed. As an extension and future research direction, the higher-order state dimension at agents can be considered as,

$$\begin{aligned} \min_{\mathbf{y}} \quad & \widetilde{F}(\mathbf{y}) = \textstyle\sum_{i=1}^n \widetilde{f}_i(\mathbf{y}_i) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^n A_i \mathbf{y}_i = \mathbf{b} \\ & \mathbf{y}_i \in \mathcal{Y}_i \end{aligned} \tag{37}$$

with $\mathbf{y}_i \in \mathbb{R}^{d_i}, \mathbf{b} \in \mathbb{R}^m$, $\widetilde{f}_i : \mathbb{R}^{d_i} \to \mathbb{R}$, $\mathcal{Y}_i \subseteq \mathbb{R}^{d_i}$, and $A_i \in \mathbb{R}^{m \times d_i}$ as a full row-rank matrix. Note that the feasibility constraint $\sum_{i=1}^n A_i \mathbf{y}_i = \mathbf{b}$ is the summation of some local constraints (of higher dimension). One point to notice is the convexity of the local constraints to admit certain composition conditions as discussed in [19, Section 3.2.4]. Such extensions based on the results of [8] can be addressed as a promising direction of future research.

References

[1] A. Cherukuri and J. Cortés, "Distributed generator coordination for initialization and anytime optimization in economic dispatch," *IEEE Trans. on Control of Net. Systems*, vol. 2, no. 3, pp. 226–237, 2015.

[2] A. Cherukuri and J. Cortes, "Initialization-free distributed coordination for economic dispatch under varying loads and generator commitment," *Automatica*, vol. 74, pp. 183–193, 2016.

[3] T. T. Doan and C. L. Beck, "Distributed lagrangian methods for network resource allocation," in *IEEE Conference on Control Technology and Applications*, 2017, pp. 650–655.

[4] M. Vrakopoulou, B. Li, and J. L. Mathieu, "Chance constrained reserve scheduling using uncertain controllable loads part i: Formulation and scenario-based analysis," *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 1608–1617, 2017.

[5] E. Makridis, K. Deliparaschos, E. Kalyvianaki, A. Zolotas, and T. Charalambous, "Robust dynamic cpu resource provisioning in virtualized servers," *IEEE Transactions on Services Computing*, 2020.

[6] A. I. Rikos, A. Grammenos, E. Kalyvianaki, C. N. Hadjicostis, T. Charalambous, and K. H. Johansson, "Optimal CPU scheduling in data centers via a finite-time distributed quantized coordination mechanism," in *60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 6276–6281.

[7] E. Kalyvianaki, T. Charalambous, and S. Hand, "Self-adaptive and self-configured CPU resource provisioning for virtualized servers using kalman filters," in *6th International Conference on Autonomic Computing*, 2009, pp. 117–126.

[8] X. Wu, S. Magnusson, and M. Johansson, "A new family of feasible methods for distributed resource allocation," in *60th IEEE Conference on Decision and Control*, 2021, pp. 3355–3360.

[9] L. Xiao and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," *Journal of Optimization Theory and Applications*, vol. 129, no. 3, pp. 469–488, 2006.

[10] N. Serhat Aybat and E. Yazdandoost Hamedani, "Distributed primal-dual method for multi-agent sharing problem with conic constraints," in *50th IEEE Asilomar Conference on Signals, Systems and Computers*, 2016, pp. 777–782.

[11] A. Nedić, A. Olshevsky, and W. Shi, "Improved convergence rates for distributed resource allocation," in *IEEE Conference on Decision and Control*, 2018, pp. 172–177.

[12] M. Doostmohammadian, H. Sayyaadi, and M. Moarref, "A novel consensus protocol using facility location algorithms," in *IEEE Conf. on Control Applications & Intelligent Control*, 2009, pp. 914–919.

[13] S. Magnússon, C. Enyioha, N. Li, C. Fischione, and V. Tarokh, "Communication complexity of dual decomposition methods for distributed resource allocation optimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 4, pp. 717–732, 2018.

[14] M. Doostmohammadian, "Single-bit consensus with finite-time convergence: Theory and applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 4, pp. 3332–3338, 2020.

[15] K. Garg, M. Baranwal, and D. Panagou, "A fixed-time convergent distributed algorithm for strongly convex functions in a time-varying network," in *59th IEEE Conference on Decision and Control*, 2020, pp. 4405–4410.

[16] M. Doostmohammadian, M. Aghasi, A. Vrakopoulou, and T. Charalambous, "1st-order dynamics on nonlinear agents for resource allocation over uniformly-connected networks," *arXiv preprint arXiv:2109.04822*, 2021.

[17] M. Doostmohammadian, A. Aghasi, M. Pirani, E. Nekouei, U. A. Khan, and T. Charalambous, "Fast-convergent dynamics for distributed allocation of resources over switching sparse networks with quantized communication links," *arXiv preprint arXiv:2012.08181*, 2021.

[18] J. Wei, X. Yi, H. Sandberg, and K. H. Johansson, "Nonlinear consensus protocols with applications to quantized communication and actuation," *IEEE Trans. on Control of Network Systems*, vol. 6, no. 2, pp. 598–608, 2019.

[19] S. P Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[20] D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar, "Convexity, duality, and lagrange multipliers," *Lecture Notes, MIT Press*, 2001.

[21] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*, vol. 87, Springer Science & Business Media, 2013.

[22] M. Doostmohammadian, A. Aghasi, T. Charalambous, and U. A. Khan, "Distributed support-vector-machine over dynamic balanced directed networks," *IEEE Control Systems Letters*, vol. 6, pp. 758 – 763, 2021.

[23] D. Jurafsky and J. Martin, *Speech and Language Processing*, 2020.

[24] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 1997.

[25] T. T. Doan and A. Olshevsky, "Distributed resource allocation on dynamic networks in quadratic time," *Systems & Control Letters*, vol. 99, pp. 57–63, 2017.