DISTRIBUTED DUAL GRADIENT TRACKING FOR NONCONVEX RESOURCE ALLOCATION WITH LIMITED COMMUNICATION DATA RATE

Keli Fu
School of Mathematical Sciences
East China Normal University
China
zzufukeli@163.com

Xiaozheng Fu*
School of Mathematical Sciences
East China Normal University
China
fxz4926@163.com

Tao Li
School of Mathematical Sciences
East China Normal University
China
tli@math.ecnu.edu.cn

Article history:
Received 21.02.2025, Accepted 21.06.2025

Abstract

We study the distributed nonconvex resource allocation with Limited Communication Data Rate (LCDR) over a communication network. Each node in the network has its own private cost function and determines the optimal resource allocation through interactions solely with its neighboring nodes. The nodes need to cooperatively minimize the total cost function to achieve the optimal resource allocation under the constraint of constant total resources. First, we consider exact communication. By Lagrange dual method, we propose a successive convex approximation-based distributed dual gradient tracking algorithm to solve the distributed nonconvex resource allocation problems. Then, we consider the case of digital communication among nodes based on LCDR. The information transmission among nodes is based on the Dynamic Encoding and Decoding (DED) with finite-level uniform quantization. We propose a successive convex approximation-based distributed dual gradient tracking algorithm with LCDR and conduct numerical simulations. The numerical results show that the algorithm converges based on merely one-bit quantizers when appropriate step sizes and scaling functions are chosen.

Key words

Limited communication data rate, distributed resource allocation, nonconvex, gradient tracking, successive convex approximation.

1 Introduction

Distributed Resource Allocation Problems (DRAPs) study how to optimally allocate available resources to multiple users called nodes connected through a communication network. The global goal of DRAPs is to allocate the total resources to multiple nodes while minimizing the global cost. A primary characteristic of DRAPs is that every node computes its optimal resource allocation by interacting only with neighboring nodes over the communication network. DRAPs are applicable across a broad spectrum of practical scenarios such as economic dispatch ([Richard et al., 2013]), smart grid systems ([Fan et al., 2013]) as well as wireless sensor networks ([Georgiadis et al., 2006]). A typical application is distributed Economic Dispatch (ED), where the local cost functions are usually quadratic functions ([Yang et al., 2013; Chen and Li, 2020]). See [Ho et al., 1980; Xiao and Boyd, 2006; Hariharan and Daniela, 2008; Zhao et al., 2018; Zhang et al., 2020] for more applications of DRAPs.

DRAPs are optimization problems with globally coupled equality or inequality constraints caused by the limitation of the amount of resources to be allocated. The local resource allocation amounts are coupled together through the global constraint which is the sum of the local constraints, and each node has a local private constraint. This brings difficulties to design fully distributed algorithms. An effective way to decouple global constraints is using Langrange function. This kind of methods transform the original optimization problem with coupling constraints into a dual problem without constraints, and the dual problem can be solved by a fully distributed method (see [Xu et al., 2015; Xu et al., 2017a; Li and Hu, 2018; Alessandro et al., 2020; Yang et al., 2017; Yi et al., 2015; Yi et al., 2016; Liang et al., 2018; Chang et al., 2014]). [Xu et al., 2017b] developed a nonnegative surplus-based distributed optimization algorithm to solve DRAPs where the local cost functions are quadratic and the global equality constraint function is

*Corresponding author

linear. [Chang et al., 2015] investigated an Alternating Direction Method of Multipliers (ADMM) based distributed algorithm and proved the convergence of the algorithm under some convexity assumptions of local cost functions. And the distributed algorithms based on ADMM were further developed in [Chang, 2016; Aybat and Hamedani, 2019; Pham et al., 2023]. [Zhang et al., 2020] considered a Distributed Dual Gradient Tracking algorithm (DDGT) to solve DRAPs over an unbalanced network. They proved that the DDGT converges linearly with strongly convex local cost functions and linear constraints.

The above research mainly focused on those problems with convex cost functions, convex inequality constraints ([Yi et al., 2015; Yi et al., 2016; Liang et al., 2018; Chang et al., 2014]) or linear equality constraints ([Xu et al., 2017b; Chang et al., 2015; Chang, 2016; Aybat and Hamedani, 2019; Zhang et al., 2020]). While, the actual problems are often more complex. For example, in the ED for energy internet, the equality constraints may be nonlinear functions ([Li et al., 2019b; Soliman and Mantawy, 2012; Yalcinoz and Short, 1996; Wollenberg and Bruce, 1996; Saini and Ohri, 2023]), or the cost functions are often nonconvex ([Chiang, 2005]) due to the prevalence of valve-point loading effects. [Giulio et al., 2014] proposed a distributed algorithm based on the auction technique to solve distributed nonconvex ED problems by introducing an auction-based technique combined with a leaderless consensus protocol. The distributed EDs for the grid-connected microgrids were studied in [Li et al., 2019b; Chen and Li, 2020], where the cost functions as well as the equality constraints are quadratic. [Chen and Li, 2020] proposed a distributed ED algorithm for the grid-connected microgrid and proved the convergence of the algorithm. [Li et al., 2019b] combined frequency control methods with consensus protocols to balance real power between load and generation during ED, and proposed a distributed algorithm to solve distributed ED problems. Note that the above works have proposed targeted distributed algorithms for specific nonconvex ED problems, while distributed algorithms for more general Distributed Nonconvex Resource Allocation Problems (DNRAPs) need to be developed.

All the studies mentioned above assume that every node can acquire exact information about its neighbors via local communication, which implies that the communication channels among nodes have unlimited capacity (bandwidth) when the nodes' states are real-valued. However, the communication channels of real distributed networks often have only limited capacity, e.g., underwater vehicles or low-cost unmanned aerial vehicles only allow the exchange of digital information with limited bits among individuals ([Xiong et al., 2022]). To solve the problems of the communication limitations, a common technique is to design a distributed algorithm based on digital communication [Li et al., 2011]. In this context, a distributed gradient descent algorithm with LCDR was proposed in [Zhou et al., 2019] based on the DED in [Li et al., 2011] to solve DRAPs with linear equality constraints. To minimize the communication costs among nodes, [Li et al., 2021] proposed a continuous-time distributed algorithm with LCDR based on event-triggered communication mechanism to solve DRAPs with linear equality constraints. [Mohammadreza et al., 2022] proposed a continuous-time distributed algorithm with LCDR based on uniform quantization to solve DRAPs with time-varying communication networks, and proved the convergence of the algorithm. Based on the DED in [Li et al., 2011], [Ma et al., 2021; Xiong et al., 2022] proposed Distributed Gradient Tracking (DGT) algorithms with LCDR for distributed optimization and proved that the designed algorithms can converge to the unique solution of the optimization problems at a linear rate. However, the distributed algorithms with LCDR based on DGT for DNRAPs remain to be developed.

In this paper, we study DNRAPs with LCDR over a communication network. Each node in the network has an individual private cost function that evaluates the cost of its allocated resources, with the global objective being to cooperatively minimize the total resource allocation cost under a specified global constraint. First, by Lagrange dual method, we transform the original DNRAPs into its dual problem which can be transformed into a distributed convex optimization problem without constraints. Due to the possible nonconvex nature of the original problems, the Lagrange function of the original DNRAPs may be a nonconvex function with respect to the original variables. To this end, we make one step of the Successive Convex Approximation (SCA) ([Scutari et al., 2017]) method to minimize the Lagrange function in original variables and use the computed results to calculate the dual function's gradient. Then, for the case of exact communication, we propose a Successive Convex Approximation-based Distributed Dual Gradient Tracking (SCA-based DDGT) algorithm by using a DGT algorithm. Next, we consider the case of digital communication among nodes based on LCDR. We propose a SCA-based DDGT algorithm with LCDR based on the DED in [Li et al., 2011]. Finally, we test the SCA-based DDGT algorithm with LCDR by four examples. The numerical simulations show that the SCA-based DDGT algorithm with LCDR can solve DNRAPs, and when the appropriate step sizes and scaling functions are chosen, one-bit quantizers can ensure the convergence of our algorithm.

To summarize, the contributions of this paper are as follows.

1) We consider DNRAPs. Specifically, the local cost functions can be nonconvex and nonsmooth, and the equality constraints can be nonlinear functions. The local cost functions and the equality constraints of [Chen and Li, 2020; Li et al., 2019b] are quadratic functions.

In order to account for the valve-point loading effects, the quadratic cost function in [Giulio et al., 2014] is extended by including an additional rectified sinusoidal term, and the equality constraints are quadratic functions. Different from [Chen and Li, 2020; Li et al., 2019b; Giulio et al., 2014], the local cost functions and equality constraints can be more general functions without special structures in this paper, and the models considered in [Chen and Li, 2020; Li et al., 2019b; Giulio et al., 2014] are our special cases. The non-quadratic equality constraints are meaningful. For example, for the ED problems with polynomial transmission loss in the power network ([Jiang and Ertem, 1995]), the equality constraints are polynomial functions.

2) We consider the case of digital communication among nodes based on LCDR. Convex DRAPs with LCDR were studied in [Zhou et al., 2019; Li et al., 2021; Mohammadreza et al., 2022], where the local cost functions are convex, and the equality constraint functions are linear. Different from [Zhou et al., 2019; Li et al., 2021; Mohammadreza et al., 2022], the DRAPs in this paper can be nonconvex. And the models considered in [Zhou et al., 2019; Li et al., 2021; Mohammadreza et al., 2022] are our special cases.

3) We propose a SCA-based DDGT algorithm with LCDR. Different from the algorithms using logarithmic quantizers in [Li et al., 2021; Mohammadreza et al., 2022], we use a uniform quantizer with finite levels. Different from the algorithms based on gradient descent methods in [Zhou et al., 2019], our algorithm is based on DGT methods. The numerical results indicate that the proposed algorithm is able to solve DNRAPs, and when the appropriate step sizes and scaling functions are chosen, one-bit quantizers can ensure the convergence of the proposed algorithm.

The subsequent sections of this paper are arranged as follows. In Section 2, the mathematical model of DRAPs is introduced. Section 3 gives our algorithms. Section 4 gives several numerical simulations to demonstrate the effectiveness of our algorithms. Section 5 concludes this paper and outlines several potential future research topics.

Notation: $\mathbb{R}$: the set of real numbers; $\mathbb{R}^n$: the n dimensional Euclidean space; $\mathbf{0}_m$: the m dimensional vector with all zeros; $|\mathcal{S}|$: the cardinality of set $\mathcal{S}$; $D_1\widetilde{f}(x,y)$: the partial gradient of $\widetilde{f}(\cdot,\cdot)$ with respect to the first argument evaluated at (x,y); $\partial f(\bar{x})$: the sub-differential set of the convex function $f(\cdot)$ at $\bar{x}$, i.e., $\partial f(\bar{x}) = \{d|f(\bar{x}) + d^T(x-\bar{x}) \le f(x)\}$; conv$\{x_1,\dots,x_n\}$: the convex hull of $\{x_1,\dots,x_n\}$.

2 Problem Formulation

Consider a network with N nodes, and the objective of the nodes is to cooperatively solve the DRAPs that can be modeled as the following distributed optimization problem

$$\begin{aligned}\min_{x_1,\cdots,x_N} \quad & \sum_{i=1}^{N} f_i(x_i)\\ \text{subject to } & x_i \in \mathcal{X}_i,\ \sum_{i=1}^{N} h_i(x_i) = \mathbf{0}_m,\end{aligned} \tag{2.1}$$

where for $i = 1,\dots,N$, $f_i(\cdot) : \mathbb{R}^{n_i} \to \mathbb{R}$ is the local private cost function of node i ; $x_i \in \mathbb{R}^{n_i}$ represents the resource allocated to node i; $\mathcal{X}_i \subseteq \mathbb{R}^{n_i}$ is a local convex and closed constraint set. $\sum_{i=1}^{N} h_i(x_i) = \mathbf{0}_m$ represents the constraint on total available resources, which shows the coupling among nodes, where each $h_i : \mathbb{R}^{n_i} \to \mathbb{R}^m$ is known to node i only.

Remark 1. Many forms of DRAPs considered in the existing literature are the special cases of Problem (2.1). For example, the local constraint sets are bounded and closed intervals ([Chen and Li, 2020; Yang et al., 2013; Xu et al., 2017b; Xu et al., 2015; Yang et al., 2017; Li et al., 2019a]); the local equality constraints are affine functions ([Zhang et al., 2020; Yang et al., 2013; Li et al., 2019a; Xu et al., 2017b; Yang et al., 2017]) or quadratic functions ([Chen and Li, 2020; Li et al., 2019b]); the local cost functions are quadratic functions ([Chen and Li, 2020; Xu et al., 2017b; Xu et al., 2015; Li et al., 2019b]) or strongly convex functions ([Chen and Li, 2020; Yang et al., 2013; Zhang et al., 2020]). In the case of the distributed ED with valve-points loading in power systems, the local cost functions are nonconvex and nonsmooth ([Mohammadreza et al., 2022; Walters and Sheble, 1993]). See the numerical simulations in Section 4 for more examples.

The information structure of the network is modeled by a directed graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}_\mathcal{G}, \mathcal{A}_\mathcal{G}\}$, where $\mathcal{V} = \{1,\dots,N\}$ is the set of nodes, $\mathcal{E}_\mathcal{G} \subseteq \mathcal{V}\times\mathcal{V}$ is the set of edges, and $(j,i) \in \mathcal{E}_\mathcal{G}$ if node j can send information to node i directly, and node j is called the in-neighbor of node i, and the set of all the in-neighbors of node i is denoted by $\mathcal{N}_i^{in} = \{j \in \mathcal{V}|(j,i) \in \mathcal{E}_\mathcal{G}\}$. Similarly, if node i can send information to node j directly, then node j is called the out-neighbor of node i, and the set of all the out-neighbors of node i is denoted by $\mathcal{N}_i^{out} = \{j \in \mathcal{V}|(i,j) \in \mathcal{E}_\mathcal{G}\}$. The weight associated to edge (j,i) is defined as a_{ij} and $a_{ii} = 0$, $a_{ij} > 0 \Leftrightarrow j \in \mathcal{N}_i^{in}$. If any two nodes of $\mathcal{G}$ can reach each other, then $\mathcal{G}$ is strongly connected. If $\sum_{j\in\mathcal{N}_i^{in}} a_{ij} = \sum_{j\in\mathcal{N}_i^{out}} a_{ji}$ for all $i \in \mathcal{V}$, then $\mathcal{G}$ is balanced, otherwise, $\mathcal{G}$ is unbalanced.

3 Algorithm Design

We are now ready to formally introduce our SCA-based DDGT algorithm with LCDR to solve (2.1) over a directed network. First, we transform the dual problem of (2.1) into a form of distributed optimization. Then, we make one step of the SCA-based ([Scutari et al., 2017]) methods for minimizing the Lagrange function in original variables and use the computed results to compute the gradient of the dual function. Next, by using the DGT algorithms ([Nedić et al., 2017]), we propose a SCA-based DDGT algorithm with exact communication to solve DNRAPs. Finally, based on the DED in [Li et al., 2011], we propose a SCA-based DDGT algorithm with LCDR.

3.1 The Dual Problem

In this subsection, we will focus on the dual problem of (2.1). Consider the Lagrange function of (2.1)

$$L(\mathbf{x},\lambda)=\sum_{i=1}^{N}L_i(x_i,\lambda), \tag{3.1}$$

where $\mathbf{x}=[x_1^T,\cdots,x_N^T]^T\in\mathbb{R}^n$, $n=\sum_{i=1}^N n_i$, and $L_i(x_i,\lambda)=f_i(x_i)+\lambda^T h_i(x_i)$, $\lambda\in\mathbb{R}^m$ is the Lagrange multiplier. Then, we give the dual problem of (2.1) as following

$$\max_{\lambda\in\mathbb{R}^m}\inf_{\mathbf{x}\in\mathcal{X}}L(\mathbf{x},\lambda), \tag{3.2}$$

where $\mathcal{X}=\mathcal{X}_1\times\ldots\times\mathcal{X}_N$. Let $g_i(\lambda)=\sup_{x_i\in\mathcal{X}_i}[-L_i(x_i,\lambda)]$, from (3.1), we have

$$\begin{aligned}&\inf_{\mathbf{x}\in\mathcal{X}}L(\mathbf{x},\lambda)=\inf_{\mathbf{x}\in\mathcal{X}}\sum_{i=1}^{N}L_i(x_i,\lambda)\\ &=\sum_{i=1}^{N}\inf_{x_i\in\mathcal{X}_i}L_i(x_i,\lambda)=\sum_{i=1}^{N}-\sup_{x_i\in\mathcal{X}_i}[-L_i(x_i,\lambda)]\\ &=\sum_{i=1}^{N}-g_i(\lambda),\end{aligned}$$

which means that (3.2) is equivalent to the following unconstrained distributed optimization problem

$$\min_{\lambda\in\mathbb{R}^m}\sum_{i=1}^{N}g_i(\lambda). \tag{3.3}$$

From $L_i(x_i,\lambda)=f_i(x_i)+\lambda^T h_i(x_i)$ and $g_i(\lambda)=\sup_{x_i\in\mathcal{X}_i}[-L_i(x_i,\lambda)]$, we have that $g_i(\cdot)$ is continuous and for any $\lambda_1,\lambda_2\in\mathbb{R}^m$ and $\theta\in(0,1)$

$$\begin{aligned}&g_i(\theta\lambda_1+(1-\theta)\lambda_2)\\ &=\sup_{x_i\in\mathcal{X}_i}\{-f_i(x_i)-(\theta\lambda_1+(1-\theta)\lambda_2)^T h_i(x_i)\}\\ &=\sup_{x_i\in\mathcal{X}_i}\{-\theta(f_i(x_i)+\lambda_1^T h_i(x_i))\\ &\quad-(1-\theta)(f_i(x_i)+\lambda_2^T h_i(x_i))\}\\ &\le\sup_{x_i\in\mathcal{X}_i}\{-\theta(f_i(x_i)+\lambda_1^T h_i(x_i))\}\\ &\quad+\sup_{x_i\in\mathcal{X}_i}\{-(1-\theta)(f_i(x_i)+\lambda_2^T h_i(x_i))\}\\ &=\theta g_i(\lambda_1)+(1-\theta)g_i(\lambda_2).\end{aligned}$$

Thus, $g_i(\cdot)$ is convex, and problem (3.3) is an unconstrained convex optimization problem. By Danskins theorem [Bertsekas, 2016], if $\mathcal{X}_i$, $i=1,2,\ldots,N$ are compact, then

$$\partial g_i(\lambda)=\operatorname{conv}\{-h_i(x_i)\mid x_i\in S_i(\lambda)\}, \tag{3.4}$$

where $S_i(\lambda)=\{\bar{x}_i|L_i(\bar{x}_i,\lambda)=\min_{x_i\in\mathcal{X}_i}L_i(x_i,\lambda)\}$. In the next subsection, we will use the DGT algorithms to solve (3.3) and propose the SCA-based DDGT algorithm.

3.2 SCA-based DDGT Algorithm with Exact Communication

We will introduce the SCA-based DDGT algorithm with exact communication in this subsection. The DGT algorithms ([Nedić et al., 2017]) will be used to solve problem (3.3), which is the equivalent problem of (3.2), and the SCA-based methods ([Scutari et al., 2017]) will be used to solve the problem $\min\limits_{x_i\in\mathcal{X}_i}L_i(x_i,\lambda)$. We give the following iterative formula for the ith node

$$\begin{aligned}\lambda_i(k+1)=&\lambda_i(k)+\eta\sum_{j\in\mathcal{N}_i^{in}}a_{ij}\left(\lambda_j(k)-\lambda_i(k)\right)\\ &+\alpha y_i(k),\end{aligned} \tag{3.5a}$$

$$\begin{aligned}\widehat{x}_i(k+1)=&\operatorname*{argmin}_{x\in\mathcal{X}_i}\Big\{\widetilde{f}_i(x,x_i(k))\\ &+\lambda_i^T(k+1)\nabla h_i(x_i(k))(x-x_i(k))\Big\},\end{aligned} \tag{3.5b}$$

$$x_i(k+1)=x_i(k)+\beta\left(\widehat{x}_i(k+1)-x_i(k)\right), \tag{3.5c}$$

$$\begin{aligned}y_i(k+1)=&y_i(k)+\eta\sum_{j\in\mathcal{N}_i^{in}}a_{ij}\left(y_j(k)-y_i(k)\right)\\ &+h_i(x_i(k+1))-h_i(x_i(k)),\end{aligned} \tag{3.5d}$$

where $\lambda_i(k)\in\mathbb{R}^m$ and $x_i(k)\in\mathbb{R}^{n_i}$ represent the estimations of the optimal solution for the problems (3.3) and (2.1) by the ith node, respectively; $y_i(k)\in\mathbb{R}^m$ and $\widehat{x}_i(k)\in\mathbb{R}^{n_i}$ are the intermediate variables. The iterates are initiated with $x_i(0)\in\mathcal{X}_i$ and $y_i(0)=h_i(x_i(0))$, $\forall\, i\in\mathcal{V}$; $\alpha>0$, $\beta>0$ and $\eta>0$ are the step sizes. $\widetilde{f}_i(x,x_i(k))$ represents a convex approximation of $f_i(\cdot)$ at $x_i(k)$ that needs to satisfy the following conditions

1) $\widetilde{f}_i(x,y):\mathcal{X}_i\times\mathcal{X}_i\to\mathbb{R}$ is a strongly convex function and continuously differentiable with respect to x;

2) $D_1\widetilde{f}_i(y,y)=\nabla f_i(y)$, $\forall\, y\in\mathcal{X}_i$;

3) $D_1\widetilde{f}_i(\cdot,\cdot)$ is continuous on $\mathcal{X}_i\times\mathcal{X}_i$.

In practical applications, the function $\widetilde{f}_i(\cdot,\cdot)$ should be designed as a strongly convex function as simple as possible that preserves the first order properties of $f_i(\cdot)$. The most obvious choice for $\widetilde{f}_i(x,x_i(k))$ is the linearization of $f_i(\cdot)$ at $x_i(k)$,

$$\begin{aligned}\widetilde{f}_i(x_i,x_i(k))=&f_i(x_i(k))+\nabla f_i(x_i(k))^T(x_i-x_i(k))\\ &+\frac{\tau_i}{2}\|x_i-x_i(k)\|^2,\end{aligned} \tag{3.6}$$

where $\tau_i>0$ is the regularization coefficient, and the proximal regularization guarantees that $\widetilde{f}_i(x_i,x_i(k))$ is strongly convex with respect to x_i. For cost functions with different specific structures, many convex approximation methods satisfying conditions 1)-3) are introduced in [Scutari et al., 2017; Scutari and Sun, 2019], and will not be repeated here.

We call Algorithm 1 as the SCA-based DDGT algorithm and now explain it in detail. After initialization, four vectors $\lambda_i(k)$, $\widehat{x}_i(k)$, $x_i(k)$ and $y_i(k)$ are iteratively updated by each node i. To be specific, at each iteration step k, firstly, each node i sends its inner states $\lambda_i(k)$ and $y_i(k)$ to all its out-neighbors and receives $\lambda_j(k)$ and $y_j(k)$ from all its in-neighbors $j \in \mathcal{N}_i^{in}$ at the same time. Then the vectors $\lambda_i(k+1)$, $\widehat{x}_i(k+1)$, $x_i(k+1)$ and $y_i(k+1)$ are updated according to (3.5). This process repeats until terminated.

Algorithm 1 SCA-based DDGT algorithm for (2.1).

Initialization: $\lambda_i(0) = \mathbf{0}_m$, $x_i(0) \in \mathcal{X}_i$, $y_i(0) = h_i(x_i(0))$ for each $i \in \mathcal{V}$, $\alpha \in (0,1)$, $\beta \in (0,1]$, $\eta \in (0,1)$.
for $k = 0, 1, 2, \ldots$ **do**
  for $i = 1, \ldots, N$ **do**
    Communication: receive $\lambda_j(k)$ and $y_j(k)$ from in-neighbor $j \in \mathcal{N}_i^{in}$, and broadcast $\lambda_i(k)$ and $y_i(k)$ to each of i's out-neighbors;
    Updation: update $\lambda_i(k+1)$, $\widehat{x}_i(k+1)$, $x_i(k+1)$ and $y_i(k+1)$ via (3.5) .
  end for
end for
Return $\{x_i(k)\}_{i=1}^N$.

Remark 2. Algorithm 1 is actually a double loop algorithm, where (3.5a) and (3.5d) is its out-loop, the DGT algorithm is applied to solve the problem (3.3). Noting that $g_i(\cdot)$ may be nondifferentiable due to the possible nonconvex properties of problem (2.1), thus we use the subgradient of $g_i(\cdot)$ instead of the gradient in Algorithm 1. It is known from (3.4) that we must first solve the nonconvex minimization problem $\min_{x_i \in \mathcal{X}_i} L_i(x_i, \lambda)$ to find the subgradient of $g_i(\cdot)$. We make one step of the SCA-based methods in the inner loop of Algorithm 1 (i.e. (3.5b) and (3.5c)) to solve $\min_{x_i \in \mathcal{X}_i} L_i(x_i, \lambda_i(k+1))$ and use the calculated $x_i(k+1)$ to replace the optimal solution of $\min_{x_i \in \mathcal{X}_i} L_i(x_i, \lambda_i(k+1))$, then from (3.4), we get an approximate subgradient $-h_i(x_i(k+1))$ of $g_i(\cdot)$. In (3.5a) and (3.5d), $y_i(k)$ represents the value of $\frac{1}{N}\sum_{i=1}^N h_i(x_i(k))$ tracked by node i.

Noting that Algorithm 1 is based on exact communication, that is, in each communication step, node i can incept the exact $\lambda_j(k)$ and $y_j(k)$ and transmit exact $\lambda_i(k)$ and $y_i(k)$ to all its out-neighbors. However, the communication channels usually have limited channel capacities in real communication networks, exact communication is hard to achieve. Thus, in the next subsection, we will consider the case of digital communication among nodes based on LCDR, that is, before communication, node i needs to encode the information $\lambda_i(k)$ and $y_i(k)$ to be sent, and the information it receives from its in-neighbors is also encoded, and the estimates of the in-neighbors' states can only be obtained after decoding.

3.3 SCA-based DDGT Algorithm with Limited Communication Data Rate

In this subsection, we consider the case of digital communication among nodes based on LCDR, and propose a SCA-based DDGT algorithm with LCDR to solve (2.1). First, we introduce the communication mechanism of digital communication based on LCDR.

Let $\lambda_i(k) \in \mathbb{R}^m$ and $x_i(k) \in \mathbb{R}^{n_i}$ represent the estimations of the optimal solution for the problems (3.3) and (2.1) by the ith node, respectively. Let $y_i(k) \in \mathbb{R}^m$ and $\widehat{x}_i(k) \in \mathbb{R}^{n_i}$ be two intermediate variables. A key step of the distributed algorithms is the communication among neighboring nodes. And we consider the case that the exact state information is not available due to the fact that the communication channels have a finite bandwidth, and the nodes can only exchange symbolic data with each other. We consider a communication mechanism based on DED. The communication channels among different nodes are noiseless digital channels, which can be represented by a pair of encoders and decoders. The encoder of the jth node is denoted as Φ_j, which is given by

$$\begin{cases} \xi_j^\lambda(0) = \mathbf{0}_m, \\ \xi_j^y(0) = \mathbf{0}_m, \\ \Delta_j^\lambda(k) = Q\left(\dfrac{1}{g(k-1)}(\lambda_j(k) - \xi_j^\lambda(k-1))\right), \\ \Delta_j^y(k) = Q\left(\dfrac{1}{g(k-1)}(y_j(k) - \xi_j^y(k-1))\right), \\ \xi_j^\lambda(k) = g(k-1)\Delta_j^\lambda(k) + \xi_j^\lambda(k-1), \\ \xi_j^y(k) = g(k-1)\Delta_j^y(k) + \xi_j^y(k-1),\ k = 1, 2, \ldots \end{cases} \tag{3.7}$$

where $\xi_j^\lambda(k) \in \mathbb{R}^m$ and $\xi_j^y(k) \in \mathbb{R}^m$ are the internal states of Φ_j; $\Delta_j^\lambda(k) \in \mathbb{R}^m$ and $\Delta_j^y(k) \in \mathbb{R}^m$ are the outputs of Φ_j, and they will be transmitted to the out-neighbors of the jth node. $Q(\cdot) : \mathbb{R}^m \to \mathbb{R}^m$ is a uniform quantizer with finite quantization levels, and $g(k) = g_0\gamma^k$ is a decaying scaling function, where $g_0 > 0$, $\gamma \in (0,1)$.

The quantizer $Q(\cdot)$ with $2K+1$ quantization levels for a vector $z = [z_1, \ldots, z_m]^T$ is defined as $Q(z) = [q(z_1), \ldots, q(z_m)]^T$, where

$$q(x) = \begin{cases} 0,\ -1/2 < x < 1/2, \\ t,\ \dfrac{2t-1}{2} \le x < \dfrac{2t+1}{2},\ t = 1, \ldots, K-1, \\ K,\ x \ge \dfrac{2K-1}{2}, \\ -q(-x),\ \ x \le -1/2, \end{cases} \tag{3.8}$$

with $x \in \mathbb{R}$. Clearly, the quantizer $q(\cdot) : \mathbb{R} \to \Gamma$ maps a real number to a set Γ, where $\Gamma =$

$\{-K, \ldots, -1, 0, 1, \ldots, K\}$ is the set of quantized levels.

Remark 3. If the output of the quantizer is zero, then no information needs to be sent, therefore, for a $(2K+1)$-level quantizer, at each time step, the communication channel needs to be capable of transmitting inerrably $\lceil \log_2(2K) \rceil$ bits. When taking $K = 1$, it is clear that the minimal quantization level is 3, and in this case, the quantizer is a one-bit quantizer.

At time instant k, when the ith node receives the quantized outputs $\Delta_j^\lambda(k)$ and $\Delta_j^y(k)$ of its in-neighbor $j \in \mathcal{N}_i^{in}$, it needs to decode these quantized outputs using a decoder Ψ_{ji} to obtain the states of jth node, and the decoder Ψ_{ji} is defined as follows

$$\begin{cases} \widehat{\lambda}_{ji}(0) = \mathbf{0}_m, \\ \widehat{y}_{ji}(0) = \mathbf{0}_m, \\ \widehat{\lambda}_{ji}(k) = g(k-1)\Delta_j^\lambda(k) + \widehat{\lambda}_{ji}(k-1), \\ \widehat{y}_{ji}(k) = g(k-1)\Delta_j^y(k) + \widehat{y}_{ji}(k-1), \end{cases} \tag{3.9}$$

where $\widehat{\lambda}_{ji}(k) \in \mathbb{R}^m$ and $\widehat{y}_{ji}(k) \in \mathbb{R}^m$ are the outputs of Ψ_{ji} at time instant k. From (3.7) and (3.9), we have

$$\widehat{\lambda}_{ji}(k) = \xi_j^\lambda(k), \;\; \widehat{y}_{ji}(k) = \xi_j^y(k), \quad \forall j \in \mathcal{V}, i \in \mathcal{N}_j^{out}, k \geq 0. \tag{3.10}$$

Based on (3.5) as well as the encoder and decoder defined by (3.7) and (3.9), we give the following iterative formula for the ith node

$$\begin{aligned} \lambda_i(k+1) =& \lambda_i(k) + \eta \sum_{j \in \mathcal{N}_i^{in}} a_{ij} \left(\widehat{\lambda}_{ji}(k) - \xi_i^\lambda(k) \right) \\ & + \alpha y_i(k), \qquad \qquad (3.11a) \\ \widehat{x}_i(k+1) =& \underset{x \in \mathcal{X}_i}{\operatorname{argmin}} \Big\{ \widetilde{f}_i(x, x_i(k)) \\ & + \lambda_i^T(k+1) \nabla h_i(x_i(k))(x - x_i(k)) \Big\}, \qquad (3.11b) \\ x_i(k+1) =& x_i(k) + \beta \left(\widehat{x}_i(k+1) - x_i(k) \right), \qquad (3.11c) \\ y_i(k+1) =& y_i(k) + \eta \sum_{j \in \mathcal{N}_i^{in}} a_{ij} \left(\widehat{y}_{ji}(k) - \xi_i^y(k) \right) \\ & + h_i(x_i(k+1)) - h_i(x_i(k)), \qquad (3.11d) \end{aligned}$$

where $\alpha > 0$, $\beta > 0$ and $\eta > 0$ are the step sizes, $x_i(0)$ and $y_i(0)$ are initialized such that $y_i(0) = h_i(x_i(0))$, $\forall\, i \in \mathcal{V}$. $\widetilde{f}_i(x, x_i(k))$ is a convex approximation of $f_i(\cdot)$ at $x_i(k)$, and the specific selection methods of $\widetilde{f}_i(x, x_i(k))$ is the same as that in (3.5).

We call Algorithm 2 as the SCA-based DDGT algorithm with LCDR and now explain it in detail. After initialization, four vectors $\lambda_i(k)$, $\widehat{x}_i(k)$, $x_i(k)$ and $y_i(k)$ are iteratively updated by each node i. To be specific, at each iteration step k, firstly, each node i encodes its own information according to (3.7) to obtain its internal states $\xi_i^\lambda(k)$ and $\xi_i^y(k)$, and its outputs $\Delta_i^\lambda(k)$ and $\Delta_i^y(k)$, and sends the outputs to all its out-neighbors. In the same time, node i receives $\Delta_j^\lambda(k)$ and $\Delta_j^y(k)$ from all of its in-neighbors $j \in \mathcal{N}_i^{in}$. Then, node i decodes the received information according to (3.9) to obtain $\widehat{\lambda}_{ji}(k)$ and $\widehat{y}_{ji}(k)$. Finally, the vectors $\lambda_i(k+1)$, $\widehat{x}_i(k+1)$, $x_i(k+1)$ and $y_i(k+1)$ are updated according to (3.11). This process repeats until terminated.

Algorithm 2 SCA-based DDGT algorithm with LCDR for (2.1).

Initialization: $\lambda_i(0) = \mathbf{0}_m$, $x_i(0) \in \mathcal{X}_i$, $y_i(0) = h_i(x_i(0))$ for each $i \in \mathcal{V}$, $\alpha \in (0,1)$, $\beta \in (0,1]$, $\eta \in (0,1)$.
for $k = 0, 1, 2, \ldots$ **do**
  for $i = 1, \ldots, N$ **do**
    Encoder: calculate $\xi_i^\lambda(k)$, $\xi_i^y(k)$, $\Delta_i^\lambda(k)$ and $\Delta_i^y(k)$ via (3.7);
    Communication: receive $\Delta_j^\lambda(k)$ and $\Delta_j^y(k)$ from in-neighbor $j \in \mathcal{N}_i^{in}$, and broadcast $\Delta_i^\lambda(k)$ and $\Delta_i^y(k)$ to each of i's out-neighbors;
    Decoder: calculate $\widehat{\lambda}_{ji}(k)$ and $\widehat{y}_{ji}(k)$ via (3.9);
    Updation: update $\lambda_i(k+1)$, $\widehat{x}_i(k+1)$, $x_i(k+1)$ and $y_i(k+1)$ via (3.11) .
  end for
end for
Return $\{x_i(k)\}_{i=1}^N$.

4 Numerical Simulation

In this section, we test the SCA-based DDGT with LCDR on four examples of (2.1). The first example is the case of quartic local cost functions and quadratic equality constraints. In this case, the gradients are not Lipschitz continuous. We verify the linear convergence of the SCA-based DDGT algorithm with LCDR and compare it with the centralized algorithms in [Hours and Jones, 2014]. The second, third and fourth examples are distributed ED problems. In the second example, we compare the SCA-based DDGT algorithm with LCDR with the algorithms in [Chen and Li, 2020]. The third and fourth examples verify the linear convergence of the SCA-based DDGT algorithm with LCDR in solving the distributed ED problems with valve-points ([Walters and Sheble, 1993]) and cubic transmission losses ([Jiang and Ertem, 1995]), respectively.

4.1 The Distributed Resource Allocation Problems with Quartic Local Cost Functions and Quadratic Equality Constraints

We consider a network with 6 nodes, and the communication structure of this network is shown in Figure 1. The weights are chosen as follows: if $(j,i) \in \mathcal{E}_\mathcal{G}$, then $a_{ij} = 1$, otherwise $a_{ij} = 0$. The local cost function

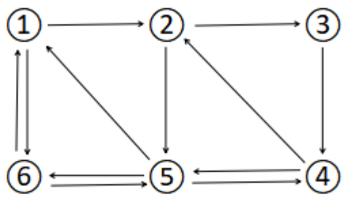

Figure 1. The communication network.

and local equality constraint of the ith node are a quartic function $f_i(x_i) = i(x_i - i)^4$ and a quadratic function $h_i(x_i) = ix_i^2 - d_i$, respectively, where $x_i \in \mathbb{R}$ and $d_i = 1, 4, 9, 16, 25, 36$, then we have $n_i = 1$, $m = 1$ in (2.1), where $i = 1, 2, 3, 4, 5, 6$. For convenience, denote $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)]^T \in \mathbb{R}^N$, $\lambda(k) = [\lambda_1(k), \ldots, \lambda_N(k)]^T \in \mathbb{R}^N$, $h(\mathbf{x}(k)) = \sum_{i=1}^N h_i(x_i(k))$, and $\mathbf{x}^* = [\mathbf{x}_1^*, \ldots, \mathbf{x}_N^*]^T \in \mathbb{R}^N$ denotes the optimal resource allocation scheme. We run Algorithm 2 with $\alpha = 0.1$, $\beta = 0.3$, $\eta = 0.3$ and $g(k) = 20(0.95)^k$ to solve this DNRAP. We use one-bit quantizer, i.e. $K = 1$. Take the initial value $\mathbf{x}(0) = [0, 1, 2, 3, 4, 5]^T$, $\lambda(0) = [19, 18, 17, 16, 20, 21]^T$, $y_i(0) = h_i(x_i(0))$. Then, we calculate by using Algorithm 2 and get Figure 2. It can be seen from Figure 2 (a) and Figure 2 (b) that Algorithm 2 is convergent, and Figure 2 (c) shows that the state of each node converges to the feasible solution of problem (2.1). Figure 2 (d) shows that the SCA-based DDGT algorithm with LCDR can converge linearly by using one-bit quantizer.

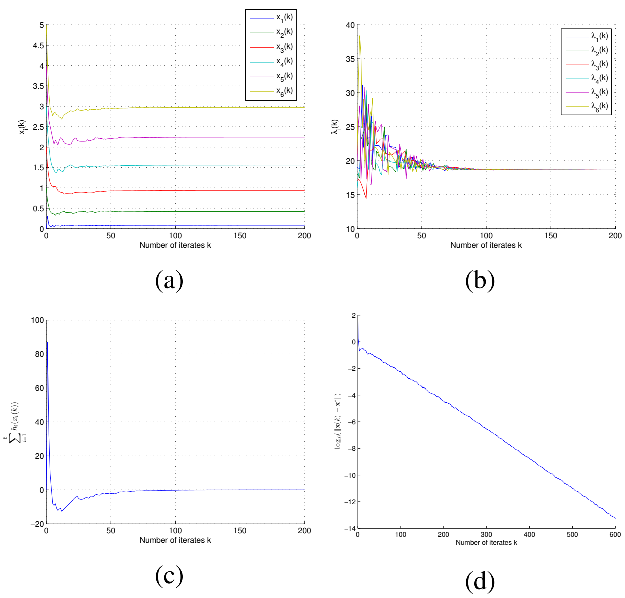

Figure 2. The convergence of Algorithm 2 for DRAPs with quartic cost functions and quadratic equality constraints. (a) Local resources allocated $x_i(k)$. (b) Dual variables $\lambda_i(k)$. (c) Global equality constraint $h(\mathbf{x}(k))$. (d) Trajectories of $\log_{10}(\|\mathbf{x}(k) - \mathbf{x}^*\|)$.

Let $\beta = 0.3$, $\eta = 0.2$, $K = 1$ and $g(k) = 20(0.99)^k$. We then implement Algorithm 2 with $\alpha = 0.017, 0.016, 0.013, 0.011$, respectively. Figure 3 displays the trajectories of $\log_{10}(\|\mathbf{x}(k) - \mathbf{x}^*\|)$ for different step sizes α. It shows that the SCA-based DDGT algorithm with LCDR is always convergent at a linear convergence rate with different α and larger α leads to a faster convergence. Figure 4 shows that Algorithm 2 converges faster than the centralized algorithm in [Hours and Jones, 2014].

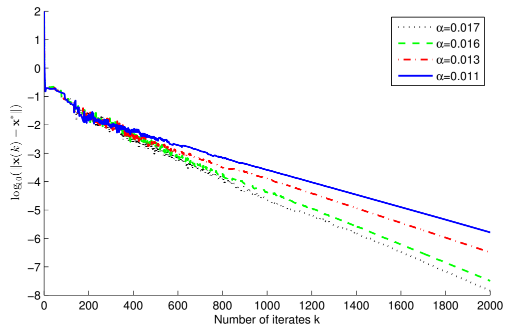

Figure 3. Trajectories of $\log_{10}(\|\mathbf{x}(k) - \mathbf{x}^*\|)$ for different step sizes α.

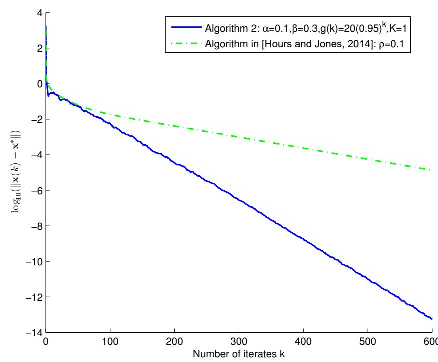

Figure 4. Trajectories of $\log_{10}(\|\mathbf{x}(k) - \mathbf{x}^*\|)$ for different algorithms.

4.2 Distributed Economic Dispatch in Power Systems

In this subsection, we use three different ED problems in the power system to test our Algorithm 2. Consider a microgrid system with N buses connected to a distributed system. All buses contain their own Distributed Generators (DGs) and loads. The DGs are equipped with Intelligent Control Units (ICUs) that serve as their local controllers and have generation costs associated with them. The ED problem involves minimizing the overall cost of electricity generation while ensuring that power

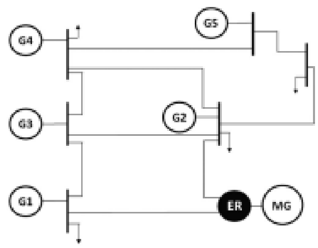

Figure 5. Test system of energy internet.

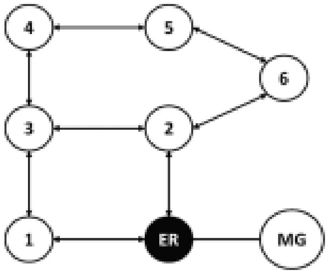

Figure 6. Communication topology of the test system.

supply meets demand and that the generation limits of DGs are adhered to. The formula of the ED problems is described as follows

$$\min_{P_1,\dots,P_N} \quad \sum_{i=1}^{N} F_i(P_i),$$
$$\text{subject to} \quad \sum_{i=1}^{N} P_i = P_L + P_{TL},$$
$$\underline{P}_i \le P_i \le \overline{P}_i, \qquad (4.1)$$

where the symbols P_i, $F_i(\cdot)$, P_L and P_{TL} represent the active power produced by the ith DG, the ith local cost function, the total current system load and the total system transmission loss, respectively. The positive constants $\underline{P}_i$ and $\overline{P}_i$ represent the lower and upper effective power limits of the ith DG, which are determined by its physical power constraints and maximum ramping rate. Obviously, ED problem (4.1) is a typical DRAP as described in (2.1). For convenience, let $P(k) = [P_1(k),\dots,P_N(k)]^T \in \mathbb{R}^N$, $P = [P_1,\dots,P_N]^T \in \mathbb{R}^N$, $\lambda(k) = [\lambda_1(k),\dots,\lambda_N(k)]^T \in \mathbb{R}^N$, $h(P(k)) = \sum_{i=1}^{N} h_i(P_i(k))$. Let $P^* = [P_1^*,\dots,P_N^*]^T \in \mathbb{R}^N$ be the optimal economic dispatch scheme of ED problem (4.1).

4.2.1 Distributed Economic Dispatch with Quadratic Local Cost Functions and Quadratic Transmission Losses

We study the ED problem within an energy internet that consists of Energy Routers (ERs), interconnected microgrids, and the main grid. And the ith local cost function is shown as

$$F_i(P_i) = \frac{(P_i - a_i)^2}{2b_i} + c_i, \;\; i = 1,\dots,N, \qquad (4.2)$$

where $a_i \le 0$, $b_i > 0$ and $c_i \in \mathbb{R}$ represent the cost coefficients. We give the total generation cost by $\sum_{i=1}^{N} F_i(P_i) + \lambda_0 P_0$, where λ_0 represents the electricity price for the distributed system that obtained by ER, and P_0 represents the power exchanged between the distributed system and the microgrid. We give the total current system load by $P_L = \sum_{i=1}^{N} P_{Li}$, where P_{Li} represents the ith bus's load. We give the total system transmission loss by

$$P_{TL}(P) = \sum_{i=1}^{N} B_i P_i^2, \qquad (4.3)$$

where $B_i > 0$ represents the loss factor. A distributed algorithm for problems (4.1)-(4.3) was proposed in [Chen and Li, 2020], but the algorithm in [Chen and Li, 2020] can only be used when the cost functions and the transmission loss functions are quadratic functions, and cannot be used when cost functions are nonconvex or transmission loss functions are cubic functions.

We consider a test electrical system containing five DGs, one ER and four loads, the microgrid is linked to the distributed system via ER. The electrical network of the test system is presented in Figure 5, while the communication network among ICUs is illustrated in Figure 6. It is important to note that the structures of the electrical network and the communication network are different here. Specifically, Bus 2 and Bus 4 in Figure 5 are neighbors in the electrical network, but not in Figure 6. The cost functions of the ER and the DGs are given by $F_0(P_0) = \lambda_0 P_0$ and (4.2), respectively. Let $h_i(P_i) = P_{Li} + B_i P_i^2 - P_i, i = 1,2,3,4,5$, $h_0(P_0) = -P_0$. Then, the ED problem (4.1)-(4.3) is a standard optimization problem as described in (2.1). Let $\widetilde{F}_0(P_0, y) = \lambda_0 P_0 + \frac{\tau}{2}\|P_0 - y\|^2$ with $\tau = 0.1$, $P_L = [50, 150, 0, 150, 200, 0]^T$. We give the parameters for DGs in Table 1 ([Chen and Li, 2020]), where the per unit of $\underline{P}_i$ and $\overline{P}_i$ is MW.

Table 1. The parameters of each DG ([Chen and Li, 2020]).

DG	a_i	b_i	c_i	$\underline{P}_i$	$\overline{P}_i$	B_i
G1	-7830.11	93.81	-326572	50	200	0.00021
G2	-4658.77	56.24	-192750	20	70	0.00017
G3	-5337.61	64.52	-220578	0	100	0.00016
G4	-6047.20	73.75	-247705	0	150	0.00020
G5	-5468.96	67.48	-221390	45	180	0.00019

Take the initial value $P(0) = [60, 30, 10, 10, 50, 163]^T$, $\lambda(0) = [79, 80, 81, 82, 83, 84]^T$, $y_i(0) = h_i(P_i(0))$. Then, we calculate by using Algorithm 2 with $\alpha = 0.01$,

$\beta = 0.01$, $\eta = 0.1$, $K = 1$ and $g(k) = 20(0.99)^k$, and Figure 7 shows the simulation results. It clearly shows that the SCA-based DDGT algorithm with LCDR converges to the optimal solution for ED problems (4.1)-(4.3) at a linear convergence rate by using one-bit quantizer.

We assess and compare the convergence performance for Algorithm 1, Algorithm 2 and the algorithm proposed in [Chen and Li, 2020]. The simulation results are shown in Figure 8. We observe that Algorithm 2 converges at a slower rate than Algorithm 1, which is to be expected considering that the performance is naturally impacted by information loss. And Figure 8 shows that proper step size ($\alpha = 0.011$) can be chosen to make Algorithm 2 converge faster than the algorithm in [Chen and Li, 2020].

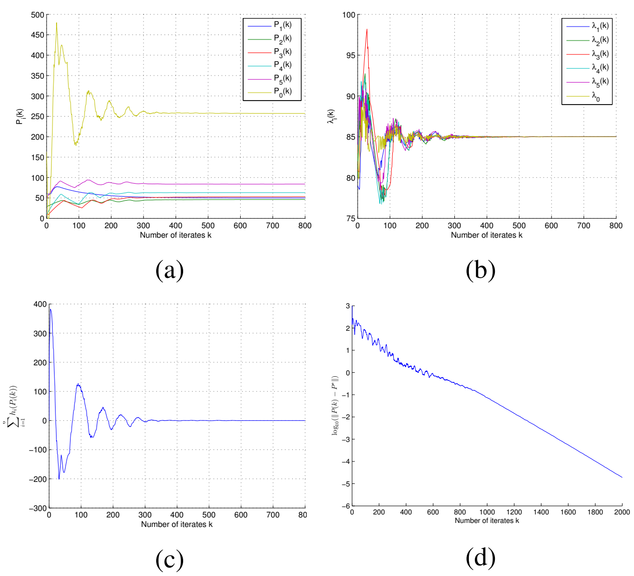

Figure 7. The convergence of Algorithm 2 for distributed ED with quadratic cost functions and transmission loss. (a) Active power generated by each DG and active power exchanged with the distribution system. (b) Dual variables $\lambda_i(k)$. (c) Global equality constraint $h(P(k))$. (d) Trajectories of $\log_{10}(\|P(k) - P^*\|)$.

4.2.2 Distributed Economic Dispatch with Valve-points Loading

The valve-points loading effects are universal in the ED problems of real power network. In this case, the local cost functions are nonconvex and nonsmooth. Different centralized genetic algorithms were proposed to solve this problem in [Chiang, 2005; Walters and Sheble, 1993]. Different from [Chiang, 2005; Walters and Sheble, 1993], the algorithms in this paper are distributed.

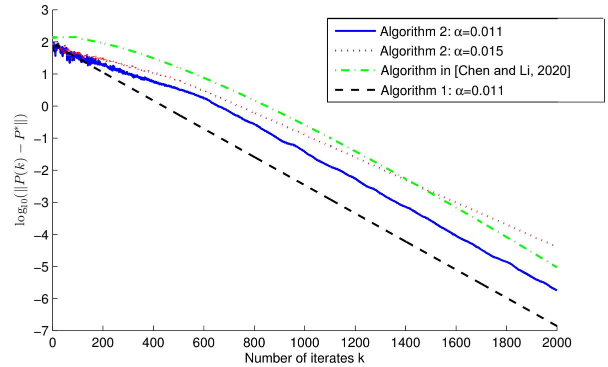

Figure 8. Trajectories of $\log_{10}(\|P(k) - P^*\|)$ for different algorithms.

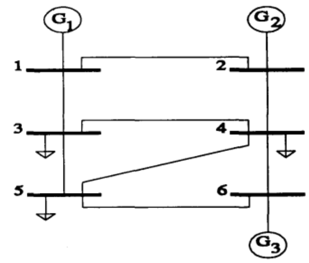

Figure 9. Test system.

In this subsection, we will test Algorithm 2 on the ED problems with valve-points loading. We consider a test system with three DGs, and the DGs can communicate with each other. The electrical network structure of the test system is illustrated in Figure 9. The local cost function for the ith DG is defined as

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i + |e_i \sin(\gamma_i(\underline{P}_i - P_i))|, \quad (4.4)$$

and the total system transmission loss is defined as

$$P_{TL}(P) = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B0_i P_i + B00. \quad (4.5)$$

Then, the local constraint function of the ith node is given by $h_i(P) = P_{Li} + \sum_{j=1}^{N} P_i B_{ij} P_j + B0_i P_i + B00/3 - P_i, i = 1, 2, 3$. Let $P_L = 210$ MW. We give the parameters for DGs in Table 2 ([Walters and Sheble, 1993]), where the per unit of $\underline{P}_i$ and $\overline{P}_i$ is 100 MW. The transmission loss coefficients are given by ([Walters and Sheble, 1993])

$$B = \begin{bmatrix} 0.06760 & 0.00953 & -0.00507 \\ 0.00953 & 0.05210 & 0.00901 \\ -0.00507 & 0.00901 & 0.02940 \end{bmatrix},$$

Table 2. The parameters of each DG ([Walters and Sheble, 1993]).

DG	a_i	b_i	c_i	e_i	γ_i	$\underline{P}_i$	$\overline{P}_i$
G1	0.001562	7.92	561	300	0.0315	0.5	2
G2	0.00194	7.85	310	200	0.042	0.375	1.5
G3	0.00482	7.97	78	150	0.063	0.45	1.8

$$B0 = \begin{bmatrix} -0.07660 \\ -0.00342 \\ 0.01890 \end{bmatrix}, \; B00 = 0.040357.$$

We take the convex approximation of the local cost functions according to (3.6) with $\tau = 0.1$. Let $\alpha = 0.1$, $\beta = 0.1$, $\eta = 0.1$, $K = 1$ and $g(k) = 20(0.94)^k$. Take the initial value $P(0) = [1, 1, 1]^T$, $\lambda(0) = [16, 17, 18]^T$, $y_i(0) = h_i(P_i(0))$. Then, we calculate by using Algorithm 2, and the simulation results are shown in Figure 10. It clearly shows that the SCA-based DDGT algorithm with LCDR can converge to the feasible solution of the ED problems (4.1), (4.4) and (4.5) linearly by using one-bit quantizer.

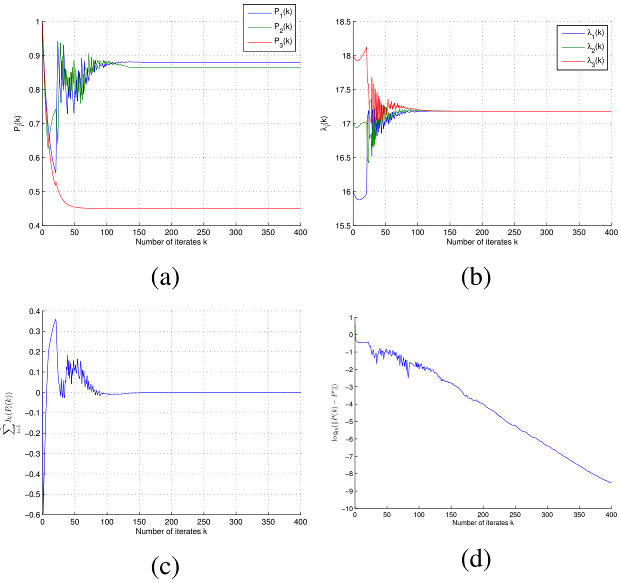

Figure 10. The convergence of SCA-based DDGT algorithm with limited communication data rate for distributed ED with valve-points loading. (a) Active power generated by each DG. (b) Dual variables $\lambda_i(k)$. (c) Global equality constraint $h(P(k))$. (d) Trajectories of $\log_{10}(\|P(k) - P^*\|)$.

4.2.3 Distributed Economic Dispatch Problems with Quadratic Local Cost Functions and Cubic Transmission Losses

In order to simulate the actual transmission losses more accurately in the ED problems of power network, the polynomial transmission loss model is introduced in [Jiang and Ertem, 1995]. In this subsection, we will test Algorithm 2 on the distributed ED problems with quadratic local cost functions and cubic transmission losses. We still consider the test system featuring the electrical network structure depicted in Figure 9. The local cost function of the ith DG is defined as

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i, \tag{4.6}$$

and the total system transmission loss is defined as

$$P_{TL} = C0 + \sum_{i=1}^{N} C1_i P_i + \sum_{i=1}^{N} C2_i P_i^2 + \sum_{i=1}^{N} C3_i P_i^3. \tag{4.7}$$

Table 3. The transmission loss coefficients ([Jiang and Ertem, 1995]).

DG	$C1_i$	$C2_i$	$C3_i$
G1	-0.02588	0.01274	0.00019
G2	-0.02517	0.01236	0.00016
G3	-0.01225	0.00696	0.00001

Then, the ith local constraint function is defined as $h_i(P_i) = P_{Li} + C0/3 + C1_i P_i + C2_i P_i^2 + C3_i P_i^3 - P_i, i = 1, 2, 3$. The parameters a_i, b_i, c_i of each DG are given in Table 2. The transmission loss coefficients are shown in Table 3 and $C0 = 0.065792$ ([Jiang and Ertem, 1995]).

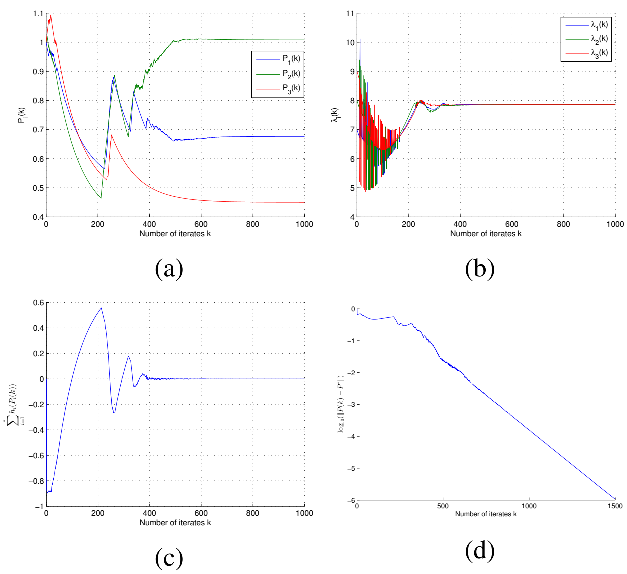

Figure 11. The convergence of SCA-based DDGT algorithm with limited communication data rate for distributed ED with quadratic local cost functions and cubic transmission losses. (a) Active power generated by each DG. (b) Dual variables $\lambda_i(k)$. (c) Global equality constraint $h(P(k))$. (d) Trajectories of $\log_{10}(\|P(k) - P^*\|)$.

Take $P_L = 210$ MW, $\alpha = 0.1$, $\beta = 0.01$, $\eta = 0.1$, $K = 1$ and $g(k) = 20(0.985)^k$. Take the initial value $P(0) = [1, 1, 1]^T$, $\lambda(0) = [7, 8, 9]^T$, $y_i(0) = h_i(P_i(0))$. Then, we calculate by using Algorithm 2, and the simulation results are shown in Figure 11. It clearly shows that the SCA-based DDGT algorithm with LCDR can converge linearly by using one-bit quantizer.

5 Conclusions

We have studied DNRAPs over a communication network. Every node in the network is associated with its own private local cost function. Each node needs to cooperatively minimize the total cost function, aiming to optimally allocate resources while maintaining a fixed total resource. Firstly, by using Lagrange dual methods, SCA-based methods and the DGT algorithms, we have proposed a SCA-based DDGT algorithm with exact communication to solve DNRAPs. Then, based on DED scheme and the SCA-based DDGT algorithms, we have proposed the SCA-based DDGT algorithm with LCDR. The information transmission among nodes is based on DED with finite-level uniform quantization, which can effectively solve the communication channel limitation problem and reduce the communication cost. The numerical simulations show that the SCA-based DDGT algorithm with LCDR can converge by using one-bit quantizer.

Future works are to provide a strict proof of the convergence for the SCA-based DDGT algorithm with LCDR and a tight bound for the convergence rate, and analyze the impact of quantization levels on the convergence rate. In addition, it is an interesting work to design the stochastic SCA-based DDGT algorithm with LCDR for distributed stochastic nonconvex optimization problems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 62261136550.

References

Alessandro, F., Ivano, N., Giuseppe, N., and Maria, P. (2020). Tracking-ADMM for distributed constraint-coupled optimization. *Automatica*, **117**, pp. 1–13.

Aybat, N. S. and Hamedani, E. Y. (2019). A distributed ADMM-like method for resource sharing over time-varying networks. *SIAM Journal on Optimization*, **29** (4), pp. 3036–3068.

Bertsekas, D. (2016). *Nonlinear Programming*. Athena Scientific, Belmont.

Chang, T. H. (2016). A proximal dual consensus ADMM method for multi-agent constrained optimization. *IEEE Transactions on Signal Processing*, **64** (14), pp. 3719–3734.

Chang, T. H., Hong, M., and Wang, X. (2015). Multi-agent distributed optimization via inexact consensus ADMM. *IEEE Transactions on Signal Processing*, **63** (2), pp. 482–497.

Chang, T. H., Nedić, A., and Scaglione, A. (2014). Distributed constrained optimization by consensus-based primal-dual perturbation method. *IEEE Transactions on Automatic Control*, **59** (6), pp. 1524–1538.

Chen, W. and Li, T. (2020). Distributed economic dispatch for energy internet based on multi-agent consensus control. *IEEE Transactions on Automatic Control*, **66** (1), pp. 137–152.

Chiang, C. L. (2005). Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels. *IEEE Transactions on Power Systems*, **20** (4), pp. 1690–1699.

Fan, Z., Parag, K., Sedat, G., Costas, E., Georgios, K., Mahesh, S., Zhu, Z., Sangarapillai, L., and Chin, W. (2013). Smart grid communications: Overview of research challenges, solutions, and standardization activities. *IEEE Communications Surveys and Tutorials*, **15** (1), pp. 21–38.

Georgiadis, L., Neely, M. J., and Tassiulas, L. (2006). Resource allocation and cross-layer control in wireless networks. *Foundations and Trends in Networks*, **1** (1), pp. 1–144.

Giulio, B., Ali, D., David, N., Biagio, T., and Frank, L. L. (2014). A distributed auction-based algorithm for the nonconvex economic dispatch problem. *IEEE Transactions on Industrial Informatics*, **10** (2), pp. 1124–1132.

Hariharan, L. and Daniela, P. F. (2008). Decentralized resource allocation in dynamic networks of agents. *SIAM Journal on Optimization*, **19** (2), pp. 911–940.

Ho, Y. C., Servi, L., and Suri, R. (1980). A class of center-free resource allocation algorithms. *IFAC Proceedings Volumes*, **13** (6), pp. 475–482.

Hours, J. and Jones, C. (2014). An augmented Lagrangian coordination-decomposition algorithm for solving distributed non-convex programs. In *Proceedings of the 2014 American Control Conference*, Portland, OR, Dec. 04-06,, pp. 4312–4317.

Jiang, A. and Ertem, S. (1995). Polynomial loss models for economic dispatch and error estimation. *IEEE Transactions on Power Systems*, **10** (3), pp. 1546–1552.

Li, H., Lü, Q., and Huang, T. (2019a). Convergence analysis of a distributed optimization algorithm with a general unbalanced directed communication network. *IEEE Transactions on Network Science and Engineering*, **6** (3), pp. 237–248.

Li, K., Liu, Q., and Zeng, Z. (2021). Quantized event-triggered communication based multi-agent system for distributed resource allocation optimization. *Information Sciences*, **577**, pp. 336–352.

Li, P. and Hu, J. (2018). An ADMM based distributed finite-time algorithm for economic dispatch problems. *IEEE Access*, **6**, pp. 30969–30976.

Li, Q., Gao, D., Zhang, H., Wu, Z., and Wang, F.

(2019b). Consensus-based distributed economic dispatch control method in power systems. *IEEE Transactions Smart Grid*, **10** (1), pp. 941–954.

Li, T., Fu, M., Xie, L., and Zhang, J. F. (2011). Distributed consensus with limited communication data rate. *IEEE Transactions on Automatic Control*, **56** (2), pp. 279–292.

Liang, S., Zeng, X., and Hong, Y. (2018). Distributed sub-optimal resource allocation over weight-balanced graph via singular perturbation. *Automatica*, **95**, pp. 222–228.

Ma, X., Yi, P., and Chen, J. (2021). Distributed gradient tracking methods with finite data rates. *Journal of Systems Science and Complexity*, **34**, pp. 1927–1952.

Mohammadreza, D., Alireza, A., Mohammad, P., Ehsan, N., Usman, A. K., and Themistoklis, C. (2022). Fast-convergent anytime-feasible dynamics for distributed allocation of resources over switching sparse networks with quantized communication links. In *Proceedings of the 2022 European Control Conference*, London, United Kingdom, Jul. 12-15,, pp. 84–89.

Nedić, A., Olshevsky, A., and Shi, W. (2017). Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, **27** (4), pp. 2597–2633.

Pham, C. T., Tran, T. T. T., Dang, H. P., and Mai, V. H. (2023). A non-convex total generalized variation model for image denoising. *Cybernetics and Physics*, **12** (1), pp. 70–81.

Richard, O. A., Aresh, D., and Kiseon, K. (2013). Multicast scheduling and resource allocation algorithms for OFDMA-Based systems: A survey. *IEEE Communications Surveys and Tutorials*, **15** (1), pp. 240–254.

Saini, N. and Ohri, J. (2023). Load frequency control in three-area single unit power system considering non-linearities effect. *Cybernetics and Physics*, **12** (1), pp. 60–69.

Scutari, G., Facchinei, F., and Lampariello, L. (2017). Parallel and distributed methods for constrained nonconvex optimization–part I: Theory. *IEEE Transactions on Signal Processing*, **65** (8), pp. 1929–1944.

Scutari, G. and Sun, Y. (2019). Distributed nonconvex constrained optimization over time-varying digraphs. *Mathematical Programming*, **176**, pp. 497–544.

Soliman, S. A. H. and Mantawy, A. A. H. (2012). *Modern Optimization Techniques With Applications in Electric Power Systems*. Springer-Verlag, New York.

Yi, P., Hong, Y., and Liu, F. (2016). Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems. *Automatica*, **74**, pp. 259–269.

Walters, D. C. and Sheble, G. B. (1993). Genetic algorithm solution of economic dispatch with valve point loading. *IEEE Transactions on Power Systems*, **8** (3), pp. 1325–1332.

Wollenberg, A. J. and Bruce, F. (1996). *Power Peneration, Operation, and Control*. Fuel and Energy Abstracts, London, U.K.

Xiao, L. and Boyd, S. (2006). Optimal scaling of a gradient method for distributed resource allocation. *Journal of Optimization Theory and Applications*, **129** (3), pp. 469–488.

Xiong, Y., Wu, L., You, K., and Xie, L. (2022). Quantized distributed gradient tracking algorithm with linear convergence in directed networks. *IEEE Transactions on Automatic Control*, **68** (9), pp. 5638–5645.

Xu, Y., Cai, K., Han, T., and Lin, Z. (2015). A fully distributed approach to resource allocation problem under directed and switching topologies. In *Proceedings of the 10th Asian Control Conference*, Kota Kinabalu, Malaysia, May 31-June 3,, pp. 1–6.

Xu, Y., Han, T., Cai, K., Lin, Z., Yan, G., and Fu, M. (2017a). A distributed algorithm for resource allocation over dynamic digraphs. *IEEE Transactions on Signal Processing*, **65** (10), pp. 2600–2612.

Xu, Y., Han, T., Cai, K., Lin, Z., Yan, G., and Fu, M. (2017b). A distributed algorithm for resource allocation over dynamic digraphs. *IEEE Transactions on Signal Processing*, **65** (10), pp. 2600–2612.

Yalcinoz, T. and Short, M. (1996). Neural networks approach for solving economic dispatch problem with transmission capacity constraints. *IEEE Transactions on Power Systems*, **13** (2), pp. 307–313.

Yang, S., Tan, S., and Xu, J. (2013). Consensus based approach for economic dispatch problem in a smart grid. *IEEE Transactions on Power Systems*, **28** (4), pp. 4416–4426.

Yang, T., Lu, J., Wu, D., Wu, J., Shi, G., Meng, Z., and Johansson, K. H. (2017). A distributed algorithm for economic dispatch over time-varying directed networkswith delays. *IEEE Transactions on Industrial Electronics*, **64** (6), pp. 5095–5106.

Yi, P., Hong, Y., and Liu, F. (2015). Distributed gradient algorithm for constrained optimization with application to load sharing in power systems. *Systems and Control Letters*, **83** (711), pp. 45–52.

Zhang, J., You, K., and Cai, K. (2020). Distributed dual gradient tracking for resource allocation in unbalanced networks. *IEEE Transactions on Signal Processing*, **68**, pp. 2186–2198.

Zhao, C., Chen, J., He, J., and Cheng, P. (2018). Privacy-preserving consensus-based energy management in smart grids. *IEEE Transactions on Signal Processing*, **66** (23), pp. 6162–6176.

Zhou, H., Yu, W., Yi, P., and Hong, Y. (2019). Quantized gradient-descent algorithm for distributed resource allocation. *Unmanned Systems*, **7** (2), pp. 119–136.