# Privacy-Preserving Distributed Coordination of Distributed Energy Resources

Madi Zholbaryssov, Christoforos N. Hadjicostis, *Fellow, IEEE*,
Alejandro D. Domínguez-García, *Senior Member, IEEE*

***Abstract*— In this paper, we consider the problem of optimally coordinating the response of a set of distributed energy resources (DERs) for serving the needs of a set of electrical loads while protecting the privacy of consumer usage data. The DERs are coordinated via a distributed approach that relies on an averaging step for guiding DERs towards the optimal operating point. Since naive exchanges of information during the averaging step might reveal sensitive information about personal energy consumption, we incorporate homomorphic encryption into the averaging step to enforce electricity usage privacy. To carry out such a procedure, power consumption data collected at electrical loads are first quantized and encrypted using the Paillier homomorphic cryptosystem. The averaging step is then executed using the homomorphically encrypted version of the so-called ratio consensus algorithm that operates exclusively on integer values. We further analytically show that the use of homomorphic encryption does not significantly affect the performance of the distributed scheme, and prove that the resulting homomorphically encrypted distributed algorithm achieves geometric convergence speed over directed communication graphs with packet losses. We showcase the proposed algorithm using the standard IEEE 14-bus test system.**

## I. Introduction

The modernization of the power grid controls aims to address the growing integration of distributed energy resources (DERs). With increasing penetration of DERs, energy production is expected to rely less on large centralized power plants and become more decentralized. Since the output of distributed generation is largely based on intermittent energy sources, like solar or wind, and can vary significantly over a short period of time, coordination of DERs requires the design of fast and scalable control algorithms. To effectively operate a large population of DERs, grid controls will increasingly rely on smart meters that provide high-resolution data containing detailed information about energy consumption of electricity end-users. This, however, raises serious concerns about the consequences of highly plausible data breaches that can reveal lifestyle features of households. To protect privacy, grid controls must be equipped with additional protection mechanisms that can reduce the risk of exposure of personal data [1].

Due to the communication overhead and bandwidth constraints, the traditional centralized control architecture has practical limitations that may hinder further integration of DERs. To overcome some of these limitations, a vast body of works proposed control strategies for managing DERs in a distributed way (see, e.g., [2]–[12]). One of the many challenges impeding the practical usefulness of the distributed control architecture for operating power grids is the potentially detrimental impact of communication data losses on their stable operation. Nevertheless, a number of works attempted to address these challenges by designing robust distributed control strategies (see, e.g., [13], [14]) based on recent advances in the field of distributed control and optimization (see, e.g., [15]–[19]).

Another issue associated with the distributed approach is the high risk of exposure of personal data to an intruder when DERs and loads exchange information during the controller execution. Privacy-preserving methods grounded on recent advances in cryptography were developed for distributed optimization and estimation (see, e.g., [20]–[23]). However, great computational complexity and significant communication overhead may limit the practicality of some of these tools at designing effective distributed control strategies for power systems. Non-cryptographic tools (e.g., differential privacy and transformation methods) have also attracted the attention of the research community (see, e.g., [24]–[27]). Differential privacy methods operate by randomly perturbing the publicly shared control signals to make it difficult for an intruder to infer sensitive information. Despite the obvious implementation simplicity, these methods achieve increased privacy at the expense of lower accuracy of the obtained solution and slower convergence rate [27]. On the other hand, transformation methods operate by transforming a given optimization problem into another form whose solution allows generating an exact solution to the original problem without revealing privately held data. Nevertheless, the applicability of such transformation methods is limited to a small subset of problems [28].

In this paper, we propose a privacy-preserving distributed approach for optimally managing DERs based on their generation cost and capacity. This approach performs a certain averaging step for guiding power outputs of the DERs towards the optimal operating point. During the averaging step, neighboring electrical devices (DERs or loads) exchange information, which substantially increases the risk of privacy breach. To preserve privacy, we carefully perform the averaging step using homomorphic encryption based on the Paillier cryptosystem [29]. The averaging step relies on the so-called *running-sum ratio-consensus* algorithm [18], which allows us to make the proposed distributed approach resilient to random packet losses in the communication channels and

M. Zholbaryssov and A. D. Domínguez-García are with the ECE Department at the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. E-mail: {zholbar1, aledan}@ILLINOIS.EDU.

C. N. Hadjicostis is with the ECE Department at the University of Cyprus, Nicosia, Cyprus, and also with the ECE Department at the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. E-mail: chadjic@UCY.AC.CY.

capable of coordinating DERs via unidirectional communication channels. Despite the additional complexity due to the use of encryption, the convergence rate is geometric. It remains to be seen how other types of privacy-preserving consensus algorithms (see, e.g., [30]–[32]) would perform when applied to distributed constrained optimization, and in particular, distributed energy management.

## II. Preliminaries

In this section, we formulate the problem of coordinating a collection of DERs over a directed communication network with packet drops while preserving energy consumption data privacy. We then give a brief overview of the nominal distributed approach for solving this problem, which is susceptible to data privacy violation. Later, we apply an encryption scheme to protect data privacy in the nominal approach.

### A. DER Coordination Problem

We consider a collection of DERs and loads physically interconnected by an electrical network. Assume the network has $n$ buses indexed by the elements in the set $\mathcal{V} = \{1, 2, \ldots, n\}$. We let $p_i$ and $\ell_i$ denote the power output of the DER and the power consumed by the load, respectively, at bus $i \in \mathcal{V}$. We assume that power demand at every bus is quantized, i.e., $\ell_i$ is an integer multiple of $1/Q$ for some positive integer $Q$. Each DER can only be operated within its lower and upper capacity limits denoted by $\underline{p}_i$ and $\overline{p}_i$, respectively. We let $f_i(\cdot)$ denote the generation cost associated with the DER at bus $i$. Our main objective is to coordinate the DERs to collectively satisfy the total electric power demand, $\sum_{i\in\mathcal{V}} \ell_i$, while minimizing the total generation cost, $\sum_{i\in\mathcal{V}} f_i(p_i)$.

The DER coordination problem (see, e.g., [2], [3], [5], [6], [9], [10]) can be stated as follows:

$$\min_{p\in\mathbb{R}^n} \sum_{i\in\mathcal{V}} f_i(p_i) \tag{1a}$$

$$\text{subject to } \sum_{i\in\mathcal{V}} p_i = \sum_{i\in\mathcal{V}} \ell_i, \tag{1b}$$

$$\underline{p} \le p \le \overline{p}, \tag{1c}$$

where $p = [p_1, \ldots, p_n]^\top$, $\underline{p} = [\underline{p}_1, \ldots, \underline{p}_n]^\top$, $\overline{p} = [\overline{p}_1, \ldots, \overline{p}_n]^\top$. We assume that $\sum_{i\in\mathcal{V}} \underline{p}_i \le \sum_{i\in\mathcal{V}} \ell_i \le \sum_{i\in\mathcal{V}} \overline{p}_i$, which makes (1) feasible. We let $p^*$ denote the solution of (1).

We make the following standard assumption regarding the objective function [19].

**Assumption 1.** *$f_i(\cdot)$, $\forall i \in \mathcal{V}$, is twice differentiable and strongly convex with parameter $m > 0$, i.e., $\nabla^2 f_i(x) \ge m$, $\forall x \in [\underline{p}_i, \overline{p}_i]$, $\forall i \in \mathcal{V}$.*

The main objective of our work in this paper is to design a distributed algorithm for solving (1) geometrically fast over a directed communication network with packet losses without revealing information about power consumption of electrical loads to an intruder.

### B. Cyber Layer

We outline the model for representing the directed communication network that enables the exchange of information between DERs and electrical loads. Let $\mathcal{G}^{(0)} = (\mathcal{V}, \mathcal{E}^{(0)})$ denote the nominal directed communication graph, where $\mathcal{E}^{(0)}$ is the set of all available communication links. During any time interval $(t_k, t_{k+1})$, successful data transmissions among the DERs can be captured by the graph $\mathcal{G}^{(c)}[k] = (\mathcal{V}, \mathcal{E}^{(c)}[k])$, where $\mathcal{E}^{(c)}[k] \subseteq \mathcal{E}^{(0)}$ is the set of active communication links, with $(i, j) \in \mathcal{E}^{(c)}[k]$ if node $j$ receives information from node $i$ during time interval $(t_k, t_{k+1})$ but not necessarily vice versa. Let $\mathcal{N}_i^+[k]$ and $\mathcal{N}_i^-[k]$ denote the sets of out-neighbors and in-neighbors of node $i$, respectively, during time interval $(t_k, t_{k+1})$, i.e., $\mathcal{N}_i^+[k] \coloneqq \{j \in \mathcal{V} : (i, j) \in \mathcal{E}^{(c)}[k]\}$ and $\mathcal{N}_i^-[k] \coloneqq \{\ell \in \mathcal{V} : (\ell, i) \in \mathcal{E}^{(c)}[k]\}$. We define node $i$'s instantaneous (communication) out-degree (including itself) to be $D_i^+[k] \coloneqq |\mathcal{N}_i^+[k]| + 1$. Let $\mathcal{N}_i^+ \coloneqq \{j \in \mathcal{V} : (i, j) \in \mathcal{E}^{(0)}\}$ denote the nominal set of out-neighbors of node $i$, i.e., $\mathcal{N}_i^+[k] \subseteq \mathcal{N}_i^+$, and $d_i^+ \coloneqq |\mathcal{N}_i^+| + 1$ denote the nominal out-degree.

Regarding the communication model, we make the following standard assumption (see, e.g., [17], [19]).

**Assumption 2.** *There exists some positive integer $B$ such that the graph with node set $\mathcal{V}$ and edge set $\bigcup_{l=kB}^{(k+1)B-1} \mathcal{E}^{(c)}[l]$ is strongly connected for $k = 0, 1, \ldots$.*

We also assume that each node knows its nominal out-degree.

**Assumption 3.** *The value of $d_i^+$ is known to node $i$.*

Notice that Assumption 3 is weaker than the standard assumption (see, e.g., [19]) that instantaneous out-degree $D_i^+[k]$ is known to node $i$ at every time instant $k$.

### C. Primal-Dual Algorithm

Below, we provide a brief overview of the standard algorithm that serves as a primitive for designing its distributed counterpart. Consider a minor variation of the first-order Lagrangian method (see, e.g., [33, Chapter 4.4]) given below:

$$p[k+1] = \Big[p[k] - s\nabla f(p[k]) + s\xi \mathbf{1}\overline{\lambda}[k]\Big]_{\underline{p}}^{\overline{p}}, \tag{2a}$$

$$\overline{\lambda}[k+1] = \overline{\lambda}[k] - s\mathbf{1}^\top(p[k] - \ell), \tag{2b}$$

where $p[k] = [p_1[k], p_2[k], \ldots, p_n[k]]^\top$, $\ell = [\ell_1, \ell_2, \ldots, \ell_n]^\top$, $[\cdot]_{\underline{p}}^{\overline{p}}$ denotes the projection onto the interval $[\underline{p}, \overline{p}]$, $s > 0$ is a constant stepsize, $\mathbf{1}$ denotes the all-ones vector of length $n$, $\xi \in (0, 1]$ is a constant parameter, and $\overline{\lambda}[k]$ is the dual variable associated with the power balance constraint, $\mathbf{1}^\top p = \mathbf{1}^\top \ell$. Notice that Algorithm (2) requires knowledge of the total power imbalance, $\mathbf{1}^\top p[k] - \mathbf{1}^\top \ell$, which is global information, to update $\overline{\lambda}[k]$.

In the distributed version of (2) (to be presented later), we let each node $i$ update $p_i[k]$ and $\lambda_i[k]$ (a local version of $\overline{\lambda}[k]$) to estimate $p_i^*$ and $\lambda^*$, respectively. Neighboring

nodes share their local information via the averaging step that guides local estimates $\lambda_i[k]$ to $\lambda^*$ and $p_i[k]$ to $p_i^*$, $i \in \mathcal{V}$.

### D. Running-Sum Ratio Consensus

One of the fundamental goals of the neighbor-to-neighbor information exchange is to estimate the average power imbalance (or total power imbalance if $n$ is known) needed to update the local variables $\lambda_i[k]$'s in a way that is as close as possible to the update of $\overline{\lambda}[k]$ in (2). To this end, we utilize the *running-sum ratio consensus* algorithm [18] capable of computing the average power imbalance over a directed communication network, even in the presence of packet drops. Below, we give a brief overview of the running-sum ratio consensus algorithm.

Consider a group of nodes indexed by the set $\mathcal{V}$, each with some real initial value, i.e., $v_i$ at node $i$. Each node aims to obtain the average of the initial values via exchange of information over the graph $\mathcal{G}^{(c)}[k]$ that satisfies Assumption 2. To this end, we let node $i$ maintain two variables, $\mu_i[k]$ and $\nu_i[k]$ such that $\mu_i[0] = v_i$ and $\nu_i[0] = 1$. Before we introduce the running-sum ratio-consensus algorithm, we first consider the following simpler updates performed by node $i$:

$$\mu_i[k+1] = \sum_{j \in \mathcal{N}_i^-[k] \cup \{i\}} \frac{\mu_j[k]}{d_j^+}, \tag{3a}$$

$$\nu_i[k+1] = \sum_{j \in \mathcal{N}_i^-[k] \cup \{i\}} \frac{\nu_j[k]}{d_j^+}, \tag{3b}$$

$$r_i[k+1] = \frac{\mu_i[k+1]}{\nu_i[k+1]}. \tag{3c}$$

Iterations (3a) – (3b) can be written in a matrix-vector form more compactly as follows:

$$\mu[k+1] = P[k]\mu[k], \tag{4a}$$

$$\nu[k+1] = P[k]\nu[k], \tag{4b}$$

where $P[k]$ is an $(n \times n)$-dimensional matrix with $P_{ij}[k] = 1/d_j^+$, $j \in \mathcal{N}_i^-[k] \cup \{i\}$, and $P_{ij}[k] = 0$, otherwise. We notice that if $\mathcal{G}^{(c)}[k] = \mathcal{G}^{(0)}$, $P[k]$ is column-stochastic, which is a primary reason for achieving the following convergence result: $\lim_{k\to\infty} r_i[k] = \frac{\sum_j v_j}{n}$ [18]. If $\mathcal{G}^{(c)}[k]$ is time-varying and satisfies Assumption 2, $P[k]$ is no longer a column-stochastic matrix. The issue however can be resolved by augmenting the original network of nodes with additional virtual nodes and links so that if node $i$ does not receive a packet from node $j$, we let a virtual node receive the packet via a virtual link [18]. This allows us to augment (4) with additional states corresponding to the virtual nodes so that the augmented system becomes

$$\mu'[k+1] = P'[k]\mu'[k], \tag{5a}$$

$$\nu'[k+1] = P'[k]\nu'[k], \tag{5b}$$

where $\mu'[k]$ and $\nu'[k]$ are the augmented states that contain $\mu[k]$ and $\nu[k]$, and $P'[k]$ is column-stochastic, which allows $r_i[k]$ to converge to the average of the initial values. Effectively, the running-sum ratio consensus algorithm is executing (5) by requiring some additional computations at each transmitting/receiving node so as to account for the absence of the virtual nodes [18]. We now formally describe the running-sum ratio consensus algorithm; the reader is referred to [18] for full details. We let node $j$ broadcast the running sums $\sum_{t=0}^k \mu_j[t]/d_j^+$ and $\sum_{t=0}^k \nu_j[t]/d_j^+$ at time instant $k$. Then, $\mu_i[k]$ and $\nu_i[k]$ are updated by node $i$ as follows:

$$\mu_i[k+1] = \sum_{j \in \mathcal{N}_i^-[k] \cup \{i\}} \mu_{ij}[k+1] - \mu_{ij}[k], \tag{6a}$$

$$\nu_i[k+1] = \sum_{j \in \mathcal{N}_i^-[k] \cup \{i\}} \nu_{ij}[k+1] - \nu_{ij}[k], \tag{6b}$$

$$r_i[k+1] = \frac{\mu_i[k+1]}{\nu_i[k+1]}, \tag{6c}$$

where $\mu_{ij}[k]$ and $\nu_{ij}[k]$ are updated using the running sums received at node $i$ from node $j$ and given by

$$\mu_{ij}[k+1] = \begin{cases} \sum_{t=0}^k \frac{\mu_j[t]}{d_j^+} & \text{if } j \in \mathcal{N}_i^-[k], \\ \mu_{ij}[k] + \frac{\mu_j[k]}{d_j^+} & \text{if } j = i, \\ \mu_{ij}[k] & \text{otherwise,} \end{cases}$$

$$\nu_{ij}[k+1] = \begin{cases} \sum_{t=0}^k \frac{\nu_j[t]}{d_j^+} & \text{if } j \in \mathcal{N}_i^-[k], \\ \nu_{ij}[k] + \frac{\nu_j[k]}{d_j^+} & \text{if } j = i, \\ \nu_{ij}[k] & \text{otherwise.} \end{cases}$$

As shown in [18], $r_i[k]$ asymptotically converges to the average of the initial values, namely, $\lim_{k\to\infty} r_i[k] = \frac{\sum_j v_j}{n}$.

### E. Robust Distributed Primal-Dual Algorithm

To solve (1), we utilize a distributed primal-dual algorithm proposed in [34], which emulates the first-order Lagrangian method (2) by using the running-sum ratio consensus algorithm (6). Below, we give its formal description.

We let node $j$ broadcast the running sums $\sum_{t=0}^k \mu_j[t]/d_j^+$, $\sum_{t=0}^k \nu_j[t]/d_j^+$, and $\sum_{t=0}^k y_j[t]/d_j^+$, where we recall that $d_i^+ := |\mathcal{N}_i^+| + 1$ is the nominal out-degree. For each $j$, node $i$ performs the following updates:

$$\mu_{ij}[k+1] = \begin{cases} (1-\gamma)\mu_{ij}[k] + \gamma \sum_{t=0}^k \frac{\mu_j[t]}{d_j^+} & \text{if } j \in \mathcal{N}_i^-[k], \\ \mu_{ij}[k] + \frac{\mu_j[k]}{d_j^+} & \text{if } j = i, \\ \mu_{ij}[k] & \text{otherwise,} \end{cases} \tag{7a}$$

$$\nu_{ij}[k+1] = \begin{cases} (1-\gamma)\nu_{ij}[k] + \gamma \sum_{t=0}^k \frac{\nu_j[t]}{d_j^+} & \text{if } j \in \mathcal{N}_i^-[k], \\ \nu_{ij}[k] + \frac{\nu_j[k]}{d_j^+} & \text{if } j = i, \\ \nu_{ij}[k] & \text{otherwise,} \end{cases} \tag{7b}$$

$$y_{ij}[k+1] = \begin{cases} (1-\gamma)y_{ij}[k] + \gamma \sum_{t=0}^k \frac{y_j[t]}{d_j^+} & \text{if } j \in \mathcal{N}_i^-[k], \\ y_{ij}[k] + \frac{y_j[k]}{d_j^+} & \text{if } j = i, \\ y_{ij}[k] & \text{otherwise,} \end{cases} \tag{7c}$$

$$\mu_i[k+1] = \sum_{j\in\mathcal{N}_i^-[k]\cup\{i\}} \Big(\mu_{ij}[k+1] - \mu_{ij}[k] - sy_{ij}[k+1] + sy_{ij}[k]\Big), \tag{7d}$$

$$\nu_i[k+1] = \sum_{j\in\mathcal{N}_i^-[k]\cup\{i\}} \nu_{ij}[k+1] - \nu_{ij}[k], \tag{7e}$$

$$\lambda_i[k+1] = \frac{\mu_i[k+1]}{\nu_i[k+1]}, \tag{7f}$$

$$p_i[k+1] = \Big[p_i[k] - s\nabla f_i(p_i[k]) + s\xi\lambda_i[k]\Big]_{\underline{p}_i}^{\overline{p}_i}, \tag{7g}$$

$$y_i[k+1] = \sum_{j\in\mathcal{N}_i^-[k]\cup\{i\}} y_{ij}[k+1] - y_{ij}[k] + \hat{n}(p_i[k+1] - p_i[k]). \tag{7h}$$

where $0 < \gamma < 1$, $\lambda_i[k]$ is the local version of $\overline{\lambda}[k]$ maintained at node $i$, $y_i[k]$ is the node $i$ estimate of the total power imbalance, $\mathbf{1}^\top p[k] - \mathbf{1}^\top \ell$, and $\hat{n}$ is the estimate of the total number of nodes, $n$, that each node has. Algorithm (7) is initialized with $\mu_i[0] = 0$, $\nu_i[0] = 1$, $y_i[0] = \hat{n}(p_i[0] - \ell_i)$, $\mu_{ij}[0] = 0$, $\nu_{ij}[0] = 0$, and $y_{ij}[0] = 0$.

### F. Privacy Concerns, Attacker Model, and Trusted Node

Notice that at the initial step of the algorithm described by (7) every node $j$ is required to send its running sum, $y_j[0]/d_j^+ = \hat{n}(p_j[0] - \ell_j)/d_j^+$, to its neighbors. By estimating $p_j[0]$ and $d_j^+$, its neighbors may determine $\ell_j$, the amount of power consumed at node $j$. This motivates us to develop a privacy-preserving distributed algorithm that makes use of homomorphic encryption to prevent power demands from being revealed. At the same time, our goal is to retain the geometric convergence rate of (7) despite the encryption. We assume that there might be multiple adversaries correctly executing the algorithm described by (7) and using the information received over the communication channels to infer the amount of power consumed at other nodes. We do not exclude the possibility that these adversaries might collaborate by sharing information with each other. We additionally assume that there is a trusted node that holds a private key not shared with the rest of the network and used for decryption purposes.

## III. Homomorphically Encrypted Distributed Coordination of DERs

In this section, we give a brief overview of homomorphic encryption, introduce the homomorphically encrypted running-sum ratio consensus algorithm, and present the distributed approach for DER coordination while preserving privacy.

### A. Homomorphic Encryption

Encryption schemes operate by converting a plaintext message $m$ into a ciphertext $c$ using an encryption mechanism $E$, i.e., $E(m) = c$. The ciphertext $c$ is decrypted into the original plaintext $m$ using a decryption mechanism $D$, i.e., $D(c) = m$.

In our approach, we utilize the Paillier cryptosystem [29], in which the encryption mechanism is constructed using a public key provided by some trustworthy entity, and the decryption mechanism is constructed by the trustworthy entity using a private key. Below, we only provide a brief description of the scheme, and the reader is referred to [29] for full details.

More formally, to devise a Paillier cryptosystem, we need to select two prime numbers $v$ and $w$, and calculate the product $N = vw$. We then choose a random integer $g$ such that $\gcd(L(g^\chi \bmod N^2), N) = 1$, where $\gcd(a, b)$ denotes the greatest common divisor of integers $a$ and $b$, $L(t) = \lfloor (t-1)/N \rfloor$, $\lfloor x \rfloor$ denotes the largest integer smaller or equal to $x$, $\chi = \mathrm{lcm}(v-1, w-1)$, and $\mathrm{lcm}(a, b)$ denotes the least common multiple of integers $a$ and $b$. The public key is composed of $(g, N)$, and the private key is composed of $\chi$. Then, the encryption mechanism is constructed using the public key $(g, N)$ as follows:

$$E(m) = g^m r^N \bmod N^2,$$

where $r$ is a random integer such that $0 < r < N$. The decryption mechanism is constructed using the private key $\chi$ as follows:

$$D(c) = \frac{L(c^\chi \bmod N^2)}{L(g^\chi \bmod N^2)} \bmod N,$$

where the modular multiplicative inverse of $L(g^\chi \bmod N^2)$ computed modulo $N$ exists since $\gcd(L(g^\chi \bmod N^2), N) = 1$, by the choice of $g$. The Paillier cryptosystem allows us to perform several algebraic operations on encrypted values without decrypting them. Such properties are referred to as homomorphic properties. Given two plaintexts $m_1$ and $m_2$ and an integer $c$, the Paillier cryptosystem satisfies the following additive homomorphic properties:

$$(m_1 + m_2) \bmod N = D(E(m_1)E(m_2) \bmod N^2), \tag{8}$$

and

$$cm_1 \bmod N = D(E(m_1)^c \bmod N^2). \tag{9}$$

In the subsequent developments, we also need the following homomorphic property for subtraction:

$$(m_1 - m_2) \bmod N = D(E(m_1)E(m_2)^{-1} \bmod N^2), \tag{10}$$

if $m_1 > m_2$, where $E(x)^{-1}$ denotes the modular multiplicative inverse of $E(x)$ computed modulo $N^2$, which exists if $\gcd(E(x), N^2) = 1$, that is, if $\gcd(g, N) = 1$ and $\gcd(r, N) = 1$.

### B. Homomorphically Encrypted Running-Sum Ratio Consensus

In the following, we present the encrypted version of the running-sum ratio consensus algorithm. To this end, we first describe the variation of the running-sum ratio consensus algorithm that exclusively operates on integer values and allows us to compute the average of the integer initial values. Effectively, the integer variation is obtained from (6)

by multiplying its weights by an integer so that the state variables, $\mu_i[k]$, $\mu_{ij}[k]$, $\nu_i[k]$, and $\nu_{ij}[k]$ become integer-valued. In particular, based on (5) with column-stochastic $P'[k]$, the integer variation is executing

$$\mu'[k+1] = W'[k]\mu'[k], \tag{11a}$$
$$\nu'[k+1] = W'[k]\nu'[k], \tag{11b}$$

where $W'[k]$ is an $(n \times n)$-dimensional matrix with integer-valued entries such that $W'[k] = c[k]P'[k]$ for some positive integer $c[k]$, and the augmented state variables $\mu'[k]$ and $\nu'[k]$ are integer-valued. If $c[k]$ is strictly positive and bounded, we have that [23, Lemma 1]

$$\lim_{k\to\infty} \frac{\mu_i[k]}{\nu_i[k]} = \frac{\sum_j v_j}{n}.$$

Below, we provide the pseudocode for the proposed algorithm, where $c[k] = \alpha$ for some strictly positive integer $\alpha$ for all $k$, $s_i^\mu[k]$ and $s_i^\nu[k]$ denote the integer running sums, and $w_i$ and $\beta_i$ are positive integer weights such that $\beta_i + w_i(d_i^+ - 1) = \alpha$ for all $i$.

Algorithm 1. *Integer Running-Sum Ratio Consensus Algorithm:*
Initialize: $\mu_i[0] = v_i$, $\nu_i[0] = 1$, $s_i^\mu[0] = 0$, $s_i^\nu[0] = 0$, $\mu_{ij}[0] = 0$, $\nu_{ij}[0] = 0$
Update:

$$s_i^\mu[k+1] = \alpha s_i^\mu[k] + w_i\mu_i[k]$$
$$s_i^\nu[k+1] = \alpha s_i^\nu[k] + w_i\nu_i[k]$$
$$\mu_{ij}[k+1] = \begin{cases} s_j^\mu[k+1] & \text{if } j \in \mathcal{N}_i^-[k] \\ \alpha\mu_{ij}[k] & \text{otherwise} \end{cases}$$
$$\nu_{ij}[k+1] = \begin{cases} s_j^\nu[k+1] & \text{if } j \in \mathcal{N}_i^-[k] \\ \alpha\nu_{ij}[k] & \text{otherwise} \end{cases}$$
$$\mu_i[k+1] = \beta_i\mu_i[k] + \sum_{j\in\mathcal{N}_i^-[k]} \mu_{ij}[k+1] - \alpha\mu_{ij}[k]$$
$$\nu_i[k+1] = \beta_i\nu_i[k] + \sum_{j\in\mathcal{N}_i^-[k]} \nu_{ij}[k+1] - \alpha\nu_{ij}[k]$$
$$r_i[k+1] = \frac{\mu_i[k+1]}{\nu_i[k+1]}$$

Next, we establish the following convergence results for Algorithm 1.

**Proposition 1.** *Let Assumptions 2 – 3 hold. Then, under Algorithm 1, we have that*

$$\lim_{k\to\infty} r_i[k] = \frac{\sum_j v_j}{n},\ i \in \mathcal{V}.$$

*Proof.* The proof can be easily established by using the results in [18, Theorem 1] and [23, Lemma 1]. □

Next, we give a description of the encrypted version of the running-sum ratio consensus algorithm. Here, we only need to encrypt $\mu_i[k]$, $\mu_{ij}[k]$, and the running sum, $s_i^\mu[k]$, whereas the remaining variables are updated without encryption. More formally, let $\tilde{s}_i^\mu[k] = E(s_i^\mu[k])$, $\tilde{\mu}_{ij}[k] = E(\mu_{ij}[k])$, and $\tilde{\mu}_i[k] = E(\mu_i[k])$. By using the homomorphic properties in (8) – (10), we obtain that

$$\tilde{s}_i^\mu[k+1] = \tilde{s}_i^\mu[k]^\alpha \tilde{\mu}_i[k]^{w_i} \bmod N^2,$$
$$\tilde{\mu}_{ij}[k+1] = \begin{cases} \tilde{s}_j^\mu[k+1] & \text{if } j \in \mathcal{N}_i^-[k], \\ \tilde{\mu}_{ij}[k]^\alpha \bmod N^2 & \text{otherwise,} \end{cases}$$
$$\tilde{\mu}_i[k+1] = \tilde{\mu}_i[k]^{\beta_i} \prod_{j\in\mathcal{N}_i^-[k]} E(\mu_{ij}[k+1] - \alpha\mu_{ij}[k]) \bmod N^2$$
$$= \tilde{\mu}_i[k]^{\beta_i} \prod_{j\in\mathcal{N}_i^-[k]} \tilde{\mu}_{ij}[k+1](\tilde{\mu}_{ij}[k]^\alpha)^{-1} \bmod N^2.$$

Algorithm 2. *Homomorphically Encrypted Running-Sum Ratio Consensus Algorithm:*
Initialize: $\tilde{\mu}_i[0] = E(v_i)$, $\nu_i[0] = 1$, $\tilde{s}_i^\mu[0] = 1$, $s_i^\nu[0] = 0$, $\tilde{\mu}_{ij}[0] = 1$, $\nu_{ij}[0] = 0$
Update:

$$\tilde{s}_i^\mu[k+1] = \tilde{s}_i^\mu[k]^\alpha \tilde{\mu}_i[k]^{w_i} \bmod N^2$$
$$s_i^\nu[k+1] = \alpha s_i^\nu[k] + w_i\nu_i[k]$$
$$\tilde{\mu}_{ij}[k+1] = \begin{cases} \tilde{s}_j^\mu[k+1] & \text{if } j \in \mathcal{N}_i^-[k] \\ \tilde{\mu}_{ij}[k]^\alpha \bmod N^2 & \text{otherwise} \end{cases}$$
$$\nu_{ij}[k+1] = \begin{cases} s_j^\nu[k+1] & \text{if } j \in \mathcal{N}_i^-[k] \\ \alpha\nu_{ij}[k] & \text{otherwise} \end{cases}$$
$$\tilde{\mu}_i[k+1] = \tilde{\mu}_i[k]^{\beta_i} \prod_{j\in\mathcal{N}_i^-[k]} \tilde{\mu}_{ij}[k+1](\tilde{\mu}_{ij}[k]^\alpha)^{-1} \bmod N^2$$
$$\nu_i[k+1] = \beta_i\nu_i[k] + \sum_{j\in\mathcal{N}_i^-[k]} \nu_{ij}[k+1] - \alpha\nu_{ij}[k]$$

For completeness, in Algorithm 2 we provide the pseudocode for the homomorphically encrypted running-sum ratio consensus algorithm. Once Algorithm 2 is run sufficiently long enough,[1] the trustworthy node decrypts $\tilde{\mu}_i[k]$ into $\mu_i[k]$, computes $r_i[k] = \mu_i[k]/\nu_i[k]$, and sends this value to all other nodes. However, such complete reliance on the trusted node somewhat negates the resilience of the distributed control architecture. One approach to reduce the reliance on a single trusted node is to run several instances of Algorithm 2 concurrently and let a group of nodes perform decryption and sum their results to obtain the average. If at least one node from the group is trustworthy, the initial values will not be revealed [23].

### *C. Robust Distributed Primal-Dual Algorithm with Homomorphic Encryption*

To coordinate DERs, we propose the following privacy-preserving strategy, which comprises two stages. In the first stage, we find the initial average power imbalance across the network, namely, $\frac{1}{n}\sum_{i\in\mathcal{V}}(p_i[0] - \ell_i)$, via the encrypted ratio consensus Algorithm 2. Notice that Algorithm 2 operates exclusively on integer values, whereas $p_i[0]$ and $\ell_i$ are real-valued. Recall that, by assumption, power demands $\ell_i$, $i =$

[1]The execution of Algorithm 2 should be terminated before the values of $\mu_i[k]$ and $s_i^\mu[k]$, $i = 1, 2, \ldots, n$, become equal to or greater than $N$; the reader is referred to [23, Section IV] for a more in-depth discussion.

$1, 2, \ldots, n$, are integer multiples of $1/Q$ for some positive integer $Q$. We also assume that all initial $p_i[0]$, $i = 1, 2, \ldots, n$, are integer multiples of $1/Q$. This quantization of $\ell_i$ and $p_i[0]$ allows us to operate on integer values and apply Algorithm 2 to estimate $\frac{1}{n}\sum_{i\in\mathcal{V}}(p_i[0]Q - \ell_i Q)$. Since $p_i[0]Q - \ell_i Q$ can be negative, we separately compute $\frac{1}{n}\sum_{i\in\mathcal{V}} p_i[0]Q$ and $\frac{1}{n}\sum_{i\in\mathcal{V}} \ell_i Q$ by using Algorithm 2. Once Algorithm 2 is run for a sufficiently large number of iterations, we let the trusted node perform the decryption and obtain the estimates of $\frac{1}{n}\sum_{i\in\mathcal{V}} p_i[0]Q$ and $\frac{1}{n}\sum_{i\in\mathcal{V}} \ell_i Q$. Then, the estimate of $\frac{1}{n}\sum_{i\in\mathcal{V}}(p_i[0]-\ell_i)$, denoted by $\overline{b}$, is computed by the trusted node and communicated to all other nodes. Since Algorithm 2 only guarantees asymptotic convergence, after a finite number of iterations, the result will only be accurate within some error denoted by $\epsilon$, namely, $\overline{b} = \frac{1}{n}\sum_{i\in\mathcal{V}}(p_i[0]-\ell_i)-\epsilon$. In the second stage, each node uses the estimate $\overline{b}$ to initialize $y_i[0]$, namely, $y_i[0] = \hat{n}\overline{b}$, and proceeds to executing (7). The pseudocode for the proposed approach is provided below.

Algorithm 3. *Robust Distributed Primal-Dual Algorithm with Homomorphic Encryption:*

1) Initialize (7):
   a) Set $\tilde{\mu}_i[0] = E(p_i[0]Q)$, execute Algorithm 2 to estimate $\frac{1}{n}\sum_{i\in\mathcal{V}} p_i[0]Q$
   b) Set $\tilde{\mu}_i[0] = E(\ell_i Q)$, execute Algorithm 2 to estimate $\frac{1}{n}\sum_{i\in\mathcal{V}} \ell_i Q$
   c) Set $y_i[0] = \hat{n}\overline{b}$
2) Execute (7)

We notice that homomorphic encryption is only utilized in the first stage of Algorithm 3 for estimating $\overline{b}$ and initializing the $y_i[0]$'s, and that the second stage of Algorithm 3, which executes (7), does not apply any encryption. Because of the error $\epsilon$ in the estimate of $\frac{1}{n}\sum_{i\in\mathcal{V}} p_i[0] - \ell_i$ (incurred after running Algorithm 2 for a finite number of iterations), the solution computed by Algorithm 3 will be different from the solution of (1). In fact, below we show that Algorithm 3 solves the following problem, which is slightly different from (1):

$$\min_{p\in\mathbb{R}^n} \sum_{i\in\mathcal{V}} f_i(p_i) \tag{12a}$$

$$\text{subject to } \sum_{i\in\mathcal{V}} p_i = n\epsilon + \sum_{i\in\mathcal{V}} \ell_i, \tag{12b}$$

$$\underline{p} \le p \le \overline{p}, \tag{12c}$$

The only difference from (1) is that the total power demand that DERs need to collectively satisfy is $n\epsilon+\sum_{i\in\mathcal{V}} \ell_i$, which differs from the total power demand in (1) by an amount $n\epsilon$, which is negligible for sufficiently small $\epsilon$. Next, we establish the following convergence results for Algorithm 3.

**Proposition 2.** *Let Assumptions 1 – 3 hold. Then, under Algorithm 3, $p[k]$ converges to $p^*_\epsilon$ at a geometric rate $\mathcal{O}(a^k)$, for some $a$, $0 < a < 1$, sufficiently small $s > 0$, and any $\xi \in (0, \frac{n}{\hat{n}}]$, where $p^*_\epsilon$ is a solution of* (12).

*Proof.* The proof easily follows from the analysis in [34, Proposition 9]. □

In the next result, we show that under some mild conditions the adversaries will not be able to determine amounts of power consumed at non-adversarial nodes.

**Proposition 3.** *Let $\mathcal{A} \subset \mathcal{V}$ denote the set of all adversarial nodes. We assume that all adversarial nodes collaborate by sharing information with each other. We assume that there is a trusted node that holds a private key not shared with the rest of the network and used for performing the decryption in Algorithm 3. We also assume that there are at least three non-adversarial nodes that do not share information with the adversarial nodes, namely, $|\mathcal{V}| \ge |\mathcal{A}| + 3$. Then, under Algorithm 3, the amounts of power consumed at non-adversarial nodes in the set $\mathcal{V} \setminus \mathcal{A}$ will remain unknown to the adversaries.*

*Proof.* Let node 1 denote the trusted node. Let $K$ denote the total number of iterations of Algorithm 2 executed by the trusted node during the initialization stage of Algorithm 3 to estimate $\frac{1}{n}\sum_{i\in\mathcal{V}} \ell_i Q$. To simplify the analysis, we assume that $\mathcal{G}^{(c)}[k] = \mathcal{G}^{(0)}$, for $k \ge 0$, during the operation of Algorithm 2, which can then be reduced to (4) rewritten as follows: $\mu[K] = P^K\mu[0]$; then, we have that

$$\mu_1[K] = \sum_{i=1}^{n} a_{K,i}\mu_i[0], \tag{13}$$

for some weights $a_{K,i}$, $i = 1, 2, \ldots, n$. Since $\mathcal{G}^{(0)}$ is strongly connected, $a_{K,i} > 0$, $i = 1, 2, \ldots, n$, for sufficiently large $K$. Let $\overline{\ell}$ denote the estimate of $\frac{1}{n}\sum_{i\in\mathcal{V}} \ell_i Q$ after $K$ iterations. Then, by using (13), we have that

$$\overline{\ell} = \sum_{i=1}^{n} a_{K,i}\ell_i Q. \tag{14}$$

Now, consider $\ell' = [\ell'_1, \ell'_2, \ldots, \ell'_n]^\top$ such that $\ell'_j \ne \ell_j$, $j \in \mathcal{V} \setminus \mathcal{A}$, $\ell'_j = \ell_j$, $j \in \mathcal{A}$, and

$$\sum_{i=1}^{n} \ell'_i = \sum_{i=1}^{n} \ell_i, \tag{15a}$$

$$\overline{\ell} = \sum_{i=1}^{n} a_{K,i}\ell'_i Q. \tag{15b}$$

Next, we show that such $\ell'$ exists. By using (14) and (15), we obtain that $\ell'$ satisfies the following two equations:

$$0 = \sum_{i\in\mathcal{V}\setminus\mathcal{A}} a_{K,i}(\ell'_i - \ell_i), \tag{16a}$$

$$0 = \sum_{i\in\mathcal{V}\setminus\mathcal{A}} (\ell'_i - \ell_i). \tag{16b}$$

Then, there exists $x = [x_1, x_2, \ldots, x_n]^\top \ne 0$, with $x_i = 0$, $i \in \mathcal{A}$, such that

$$0 = \sum_{i\in\mathcal{V}\setminus\mathcal{A}} a_{K,i}x_i, \tag{17a}$$

$$0 = \sum_{i\in\mathcal{V}\setminus\mathcal{A}} x_i, \tag{17b}$$

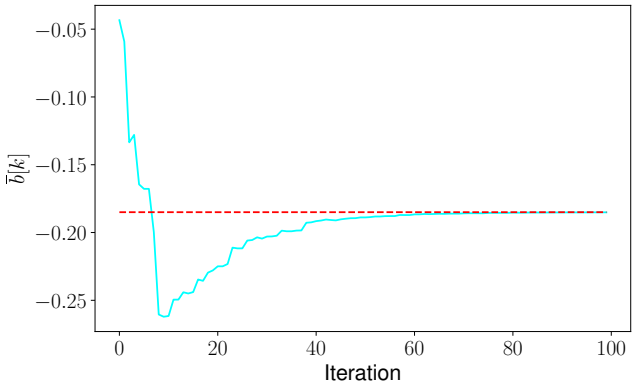


Fig. 1: Trajectory of $\bar{b}[k]$ for Algorithm 2.

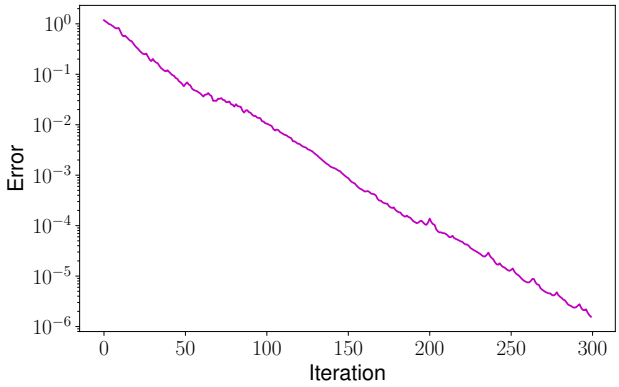


Fig. 2: Trajectory of $\|p[k] - p_\epsilon^*\|_2$ for Algorithm 3.

since (17) is an underdetermined system of equations in view of the fact that $|\mathcal{V}\backslash\mathcal{A}| \geq 3$. Then, note that $\ell' = \ell + \alpha x$, with any $\alpha \neq 0$, solves (16). Now, consider the same dispatch problem in (1) but with a different set of electrical loads, $\ell'_1, \ell'_2, \ldots, \ell'_n$, at nodes $1, 2, \ldots, n$, respectively. Because of the properties of $\ell'$ captured by (15), the initialization stage of Algorithm 3 will yield the same $y_i[0] = \hat{n}\bar{b}$ for both sets of loads, $\ell$ and $\ell'$. Thus, the second stage of Algorithm 3 will be identical for both sets of loads. Hence, the adversaries will not be able to distinguish between $\ell'_j$ and $\ell_j$ at $j \in \mathcal{V}\backslash\mathcal{A}$. □

**Remark 1.** *In Proposition 3, we allow the possibility that the adversarial nodes may know the coefficients $a_{K,i}$ in the state transition matrix corresponding to the trusted node. This is a main reason for having the additional condition on the cardinality of the set of non-adversarial nodes, namely, $|\mathcal{V}| \geq |\mathcal{A}| + 3$. However, if the coefficients $a_{K,i}$ are unknown to the adversarial nodes, then, it is sufficient to have $|\mathcal{V}| \geq |\mathcal{A}| + 2$.*

## IV. Numerical Simulations

In this section, we present numerical results that illustrate the performance of the proposed privacy-preserving approach, Algorithm 3. We consider the standard IEEE 14-bus test system. We utilize the python library for the Paillier cryptosystem written by CSIRO's Data61 [35]. For each node $i$ with a generating unit, we choose $f_i(p_i) = a_i p_i^2$, where $a_i > 0$ is randomly selected. For each node $i$ with a load, we choose $\ell_i$ using the test data [36]. We set $Q = 1000$ such that $\ell_i$ is an integer multiple of $1/Q$ for all $i$. The topology of the nominal communication graph coincides with that of the power network such that any two nodes that are connected by an electrical line are also connected by a single undirected link (or two opposite unidirectional communication links) making the nominal graph strongly connected. Each link becomes inactive, i.e., a data packet is lost, with probability 0.4. We initialize $p_i[0] = 0$ for all $i$ so that $p_i[0]$ is an integer multiple of $1/Q$. We begin with the execution of the first stage of Algorithm 3, where we estimate $\frac{1}{n}\sum_{i\in\mathcal{V}} p_i[0]Q$ and $\frac{1}{n}\sum_{i\in\mathcal{V}} \ell_i Q$ by running Algorithm 2 and obtain $\bar{b}[k]$, an estimate of the average power imbalance, $\frac{1}{n}\sum_{i\in\mathcal{V}}(p_i[0] - \ell_i)$. Figure 1 shows the evolution of $\bar{b}[k]$ decrypted by the trustworthy entity at time $k$. After 100 iterations, $\bar{b}[k] = -0.1851$ is within $1 \times 10^{-4}$ of the average power imbalance, $-0.185$, i.e., the error $\epsilon = 1 \times 10^{-4}$. For all nodes, we choose the same value for the estimate of the total number of nodes in the network, namely, $\hat{n} = 10$, and set $y_i[0] = \hat{n}\bar{b}[100]$ for all $i$. Based on this initialization, we further proceed to the execution of the second stage of Algorithm 3, where we run the robust distributed primal-dual algorithm (7). We set the parameters in (7) as follows: $\gamma = 0.9$, $s = 0.1$ and $\xi = 1.1$. Figure 2 shows the evolution of the Euclidean norm of the error between $p[k]$ and the optimal solution $p_\epsilon^*$ (note that $\epsilon = 1 \times 10^{-4}$), which demonstrates the geometric convergence rate of Algorithm 3.

## V. Concluding Remarks and Future Work

In this paper, we considered the problem of optimally coordinating DERs while protecting the confidentiality of the energy consumption information. We described how homomorphic encryption can be applied to preserve privacy while retaining the geometric convergence rate of the robust distributed primal-dual algorithm over directed communication graphs with packet drops. An interesting extension would be to also consider information leakage when some knowledge of the optimization function (e.g., the $a_i$'s) is available to the intruder.

## References

[1] S. Sridhar, A. Hahn, and M. Govindarasu, "Cyber–physical system security for the electric power grid," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 210–224, Jan. 2012.

[2] Z. Zhang and M. Y. Chow, "Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid," *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1761–1768, Nov. 2012.

[3] A. D. Domínguez-García, S. T. Cady, and C. N. Hadjicostis, "Decentralized optimal dispatch of distributed energy resources," in *Proc. IEEE Conference on Decision and Control*, Dec. 2012, pp. 3688–3693.

[4] S. Yang, S. Tan, and J. Xu, "Consensus based approach for economic dispatch problem in a smart grid," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4416–4426, Nov. 2013.

[5] S. Kar and G. Hug, "Distributed robust economic dispatch in power systems: a consensus + innovations approach," in *Proc. IEEE Power Energy Society General Meeting*, July 2012, pp. 1–8.

[6] X. Zhang and A. Papachristodoulou, "Redesigning generation control in power systems: methodology, stability and delay robustness," in *Proc. IEEE Conference on Decision and Control*, Dec. 2014, pp. 953–958.

[7] S. T. Cady, A. D. Domínguez-García, and C. N. Hadjicostis, "A distributed generation control architecture for islanded AC microgrids," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 1717–1735, Sept. 2015.

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

[9] G. Chen and Z. Zhao, "Distributed optimal active power control in microgrid with communication delays," in *Proc. Chinese Control Conference*, July 2016, pp. 7515–7520.

[10] J. Wu, T. Yang, D. Wu, K. Kalsi, and K. H. Johansson, "Distributed optimal dispatch of distributed energy resources over lossy communication networks," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 3125–3137, Nov. 2017.

[11] W. Du, L. Yao, D. Wu, X. Li, G. Liu, and T. Yang, "Accelerated distributed energy management for microgrids," in *Proc. IEEE Power Energy Society General Meeting*, Aug. 2018, pp. 1–5.

[12] T. Yang, D. Wu, H. Fang, W. Ren, H. Wang, Y. Hong, and K. H. Johansson, "Distributed energy resource coordination over time-varying directed communication networks," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 3, pp. 1124–1134, Sep. 2019.

[13] A. D. Domínguez-García and C. N. Hadjicostis, "Distributed algorithms for control of demand response and distributed energy resources," in *Proc. IEEE Conference on Decision and Control and European Control Conference*, Dec. 2011, pp. 27–32.

[14] O. Azofeifa, S. Nigam, O. Ajala, C. Sain, S. Utomi, A. D. Domínguez-García, and P. W. Sauer, "Controller hardware-in-the-loop testbed for distributed coordination and control architectures," in *Proc. North American Power Symposium*, Oct. 2019, pp. 1–6.

[15] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. IEEE Symposium on Foundations of Computer Science*, Oct. 2003, pp. 482–491.

[16] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[17] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 601–615, Mar. 2015.

[18] C. N. Hadjicostis, N. H. Vaidya, and A. D. Domínguez-García, "Robust distributed average consensus via exchange of running sums," *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1492–1507, June 2016.

[19] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017.

[20] D. Brettschneider, D. Hölker, A. Scheerhorn, and R. Tönjes, "Preserving privacy in distributed energy management," *Computer Science - Research and Development*, vol. 32, no. 1, pp. 159–171, 2017.

[21] Y. Lu and M. Zhu, "Privacy preserving distributed optimization using homomorphic encryption," *Automatica*, vol. 96, pp. 314–325, 2018.

[22] C. N. Hadjicostis, "Privary preserving distributed average consensus via homomorphic encryption," in *Proc. IEEE Conference on Decision and Control*, Dec. 2018, pp. 1258–1263.

[23] C. N. Hadjicostis and A. D. Domínguez-García, "Privacy-preserving distributed averaging via homomorphically encrypted ratio consensus," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3887–3894, 2020.

[24] O. L. Mangasarian, "Privacy-preserving linear programming," *Optimization Letters*, vol. 5, no. 1, pp. 165–172, 2011.

[25] G. M. Fung and O. L. Mangasarian, "Privacy-preserving linear and nonlinear approximation via linear programming," *Optimization Methods and Software*, vol. 28, no. 1, pp. 207–216, 2013.

[26] P. C. Weeraddana, G. Athanasiou, C. Fischione, and J. S. Baras, "Perse privacy preserving solution methods based on optimization," in *Proc. IEEE Conference on Decision and Control*, 2013, pp. 206–211.

[27] S. Han, U. Topcu, and G. J. Pappas, "Differentially private distributed constrained optimization," *IEEE Transactions on Automatic Control*, vol. 62, no. 1, pp. 50–64, Jan. 2017.

[28] S. Gade and N. H. Vaidya, "Private optimization on networks," in *Proc. American Control Conference*, June 2018, pp. 1402–1409.

[29] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. International Conference on the Theory and Applications of Cryptographic Techniques*, J. Stern, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 223–238.

[30] N. E. Manitara and C. N. Hadjicostis, "Privacy-preserving asymptotic average consensus," in *Proc. European Control Conference*, 2013, pp. 760–765.

[31] Y. Mo and R. M. Murray, "Privacy preserving average consensus," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 753–765, Feb. 2017.

[32] H. Gao, C. Zhang, M. Ahmad, and Y. Wang, "Privacy-preserving average consensus on directed graphs using push-sum," in *Proc. IEEE Conference on Communications and Network Security*, May 2018, pp. 1–9.

[33] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Scientific, 1999.

[34] M. Zholbaryssov, C. N. Hadjicostis, and A. D. Domínguez-García, "Fast distributed coordination of distributed energy resources over time-varying communication networks," *arXiv e-prints*, arXiv:1907.07600, 2019.

[35] C. Data61, "Python paillier library," https://github.com/data61/python-paillier, 2013.

[36] "Power Systems Test Case Archive." [Online]. Available: https://labs.ece.uw.edu/pstca/pf14/pg_tca14bus.htm