Distributed Energy Management for Networked Microgrids
Using Online Alternating Direction Method of Multipliers with Regret

Wann-Jiu Ma, Student Member, IEEE, Jianhui Wang, Senior Member, IEEE, Vijay Gupta, Member, IEEE,
Chen Chen, Member, IEEE

Abstract—This paper proposes a distributed algorithm for online energy management in networked microgrids with a high penetration of distributed energy resources (DERs). The implementation of networked microgrids with high penetration of DERs brings power generation closer to electricity users, which has advantages of increasing power supply efficiency and reliability by shortening the power supply delivery distance between the main grid and the electricity users. However, a high penetration of DERs introduces high variability and uncertainty of the power generation into the microgrids. In general, state-of-the-art forecasting for non-dispatchable DERs such as wind and solar is not sufficiently accurate, which results in inaccurate energy scheduling. To address the high variability and uncertainty of DERs in networked microgrids, we propose a novel online energy management algorithm using the online Alternating Direction Method of Multipliers (ADMM) which does not require any a priori knowledge about the future power generation from the DERs to proceed. Despite the lack of the knowledge on DER forecasts, there exists an analytical guarantee that the energy scheduling solution generated by the algorithm converges to the optimal one and minimizes the regret. Furthermore, the proposed energy management algorithm does not require the inner loop subroutines of the typical ADMM algorithm to be fully executed, which further leads to the substantial reduction of the implementation time and the mitigation of the communication demand. The effectiveness of the proposed algorithm is verified by various numerical examples.

Index Terms—Networked microgrids, energy management systems, distributed energy resources, distributed optimization, alternating direction method of multipliers, online learning, regret minimization.

NOMENCLATURE

Numbers, and sets

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_s$</td>
<td>Set of utility microgrids</td>
</tr>
<tr>
<td>$M$</td>
<td>Number of utility microgrids</td>
</tr>
<tr>
<td>$E$</td>
<td>Set of power lines in the networked microgrids</td>
</tr>
<tr>
<td>$T$</td>
<td>Set of time slots considered</td>
</tr>
<tr>
<td>$T'$</td>
<td>Number of time slots considered</td>
</tr>
</tbody>
</table>

Parameters

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_{ij}$</td>
<td>Line resistance between microgrid $i$ and $j$</td>
</tr>
<tr>
<td>$x_{ij}$</td>
<td>Line reactance between microgrid $i$ and $j$</td>
</tr>
<tr>
<td>$p^i_r$</td>
<td>Minimum real power of micro turbine in microgrid $i$</td>
</tr>
<tr>
<td>$p^i_l$</td>
<td>Maximum real power of micro turbine in microgrid $i$</td>
</tr>
<tr>
<td>$p^i_{l'}$</td>
<td>Lower bound for $p^i_l$</td>
</tr>
<tr>
<td>$p^i_{u'}$</td>
<td>Upper bound for $p^i_l$</td>
</tr>
<tr>
<td>$p^i_l$</td>
<td>Minimum real power load in microgrid $i$</td>
</tr>
<tr>
<td>$p^i_{l'}$</td>
<td>Maximum real power load in microgrid $i$</td>
</tr>
<tr>
<td>$q^i_r$</td>
<td>Minimum reactive power load in microgrid $i$</td>
</tr>
<tr>
<td>$q^i_{l'}$</td>
<td>Maximum reactive power load in microgrid $i$</td>
</tr>
<tr>
<td>$q^i_{l'}$</td>
<td>Minimum reactive power of micro turbine in microgrid $i$</td>
</tr>
<tr>
<td>$q^i_{u'}$</td>
<td>Maximum reactive power of micro turbine in microgrid $i$</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Maximum allowed voltage deviation</td>
</tr>
<tr>
<td>$a_i$</td>
<td>The second order coefficient in the micro turbine power generation cost function for microgrid $i$</td>
</tr>
<tr>
<td>$b_i$</td>
<td>The first order coefficient in the micro turbine power generation cost function for microgrid $i$</td>
</tr>
<tr>
<td>$c_i$</td>
<td>The constant coefficient in the micro turbine power generation cost function for microgrid $i$</td>
</tr>
<tr>
<td>$\alpha_i$</td>
<td>Price for selling electricity from private microgrid $i$ to distribution network operator</td>
</tr>
<tr>
<td>$\beta_i$</td>
<td>Price for buying electricity from distribution network operator to private microgrid $i$</td>
</tr>
</tbody>
</table>

Uncertain quantities

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_i$</td>
<td>Power generation of PV panels in microgrid $i$</td>
</tr>
</tbody>
</table>

Decision variables

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p^i_r$</td>
<td>Power generation of micro turbine in microgrid $i$</td>
</tr>
<tr>
<td>$p^i_l$</td>
<td>Net power of power transaction and power generation of PV panels in microgrid $i$</td>
</tr>
<tr>
<td>$q^i_r$</td>
<td>Reactive power generation in microgrid $i$</td>
</tr>
<tr>
<td>$q^i_l$</td>
<td>Reactive power load in microgrid $i$</td>
</tr>
<tr>
<td>$P_{ij}/Q_{ij}$</td>
<td>Real/reactive power from microgrid $i$ to $j$</td>
</tr>
<tr>
<td>$V_i$</td>
<td>Voltage at microgrid $i$</td>
</tr>
</tbody>
</table>

The work of W.-J. Ma and V. Gupta is supported in part by NSF grant #1239224. The work of J. Wang and C. Chen is sponsored by the Office of Electricity of the U.S. Department of Energy.

W.-J. Ma and V. Gupta are with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA. {wma1, vgupta2}@nd.edu. J. Wang and C. Chen are with the Energy Systems Division, Argonne National Laboratory, Lemont, IL 60439, USA. {jianhui.wang, morningchen}@anl.gov.
I. INTRODUCTION

Microgrids are increasingly installed due to their improved reliability and resilience. Recent studies have shown the traditionally isolated microgrids can be interconnected to form a networked system to further improve reliability [6], [24], [26]. Networked microgrids are clusters of distributed energy resources (DERs), electricity loads, and storages in an distribution system where a distribution network operator coordinates the energy scheduling of the microgrids [24]. The ownership of the networked microgrids can be diverse ranging from utility owned microgrids, non-utility owned microgrids to public purpose microgrids which could be owned by a township but operated by a utility or a third party [3], [4]. The flexible physical structure and ownership of microgrids and their capabilities of being islanded when the utility supply fails greatly incent their wide deployment in practice.

The renewable nature of DERs in microgrids provides many advantages such as the reduction of fossil fuel consumption and the mitigation of carbon dioxide emissions [12]. Implementation of DERs by distributing electric vehicles, solar photovoltaic (PV) panels, and wind energy sources in a microgrid however requires fundamentally different considerations [12], [15]. Contrary to traditionally centralized energy sources, DERs are widely distributed over the underlying power network. The high variability and uncertainty of DERs complicate the design of the energy management in microgrids. For example, the effects of uncontrollable passing clouds on the power generation from the PV panels result in inaccurate power generation prediction from DERs [25]. In a traditional power grid, the unpredictable effects caused by the DERs on the total power generation may be small as the renewable energy sources may only contribute to a small portion of the total power generation. However, as the popularity of the microgrid is continuously growing with the ever increasing penetration of DERs, the uncertain and unpredictable features of DERs need to be taken into account to design more robust energy management systems specifically for microgrids.

Robust optimization is widely used in the design of energy management systems with the high penetration of renewable energy sources (see, e.g., [15], [28]). A probabilistic model is not needed to design an energy management algorithm based on robust optimization. However, to implement the algorithm, it requires a priori knowledge about the uncertainties by restricting them within a given set. In order to derive algorithms to take into account all possible outcomes of the uncertainties, robust optimization based energy management algorithms may provide over-conservative results [14].

In this paper, we design an online energy management algorithm for the distribution system operator to manage the energy scheduling of networked microgrids using regret minimization. The proposed online energy management algorithm does not require any a priori knowledge about the uncertainties from DERs to proceed, which is fundamentally different from the robust optimization based approaches. Meanwhile, regret minimization has recently gained tremendous popularity in the online optimization and machine learning community (see e.g., [8], [13] and references therein). In the regret minimization framework, a player has to make a decision first, and then obtains a loss corresponding to the decisions made. The environment in which the player makes a decision is assumed to be uncertain and changing over times. Prior to the decision made, the player is not aware of the loss incurred due to the decision made. The regret minimization algorithms provide, however, rules to update the decisions such that the average cumulative loss approaches the one of the best fixed decision discovered in hindsight. For our problem, the player is the distribution network operator. The decisions are those to be made on the scheduling of the power generation from the controllable sources (e.g., micro turbines) and the power consumption for various loads (e.g., EVs, batteries). The uncertainties come from the unpredictable nature of DER generation. Based on this regret minimization framework, the distribution network operator updates (in an online fashion) the schedules of various components without any a priori knowledge about the unpredictable parts of the generation from DERs.

It should be noted that online energy management in principle can be carried out by the distribution network operator in a centralized manner. However, the operator needs the private information about the energy usage/generation from all microgrids (utility owned and private). The demand of such kind of information may raise privacy concerns [17], [18]. Furthermore, the centralized algorithm causes a significant burden of the communication demand to coordinate the power scheduling of all components in the microgrids [9], [17]. This burden is becoming even more serious in the online implementation of the algorithms with real-time deadline constraints since the allowed time to accomplish the computation task is limited [9].

For the need of a distributed implementation of the energy management algorithm, we modify the alternating direction method of multipliers (ADMM) algorithm (see e.g., [7]) to coordinate the power scheduling of components in the microgrids to satisfy the underlying power network operation constraints. The ADMM algorithm is well suited for distributed optimization and large-scale distributed computing systems. By using ADMM, we do not need to have a centralized computation center and the computation can be distributed over the local computation devices (the reader is referred to the recent survey paper [7] and references therein for the details of ADMM).

ADMM consists of several inner loop subroutines. Each inner loop subroutine is an optimization (in a small-scale). Unlike the offline energy management algorithms based on ADMM [9], [17], the proposed online ADMM does not require the inner loop subroutines of the algorithm to be fully executed, which results in the speed-up of the runtime and the reduction of the communication demand.

Using regret minimization and ADMM as two building blocks, we propose an online distributed algorithm to allocate the electricity resources with the high penetration of DERs in the microgrids. The unique features of the proposed algorithm in this paper are twofold: 1) The online energy management algorithm does not require any a priori knowledge about the unpredictable parts of DERs to proceed. However, there exists an analytical guarantee about the performance of the
algorithm even with the lack of the knowledge; 2) The online energy management algorithm is implemented in a distributed manner and does not require the convergence of the inner loop subroutines of the distributed algorithm leading to the acceleration of the implementation and the reduction of the communication demand substantially. It has the advantage of being implemented in an online fashion without computation and communication interruptions.

Some recent works have been reported on online energy management for the microgrid [16], [21]. The work [16] studies online algorithms for the microgrid generation scheduling with intermittent renewable sources. However, rather than the notion of regret, a different metric, competitive ratio, is used to evaluate the performance of algorithms. The two metrics, regret and competitive ratio, are fundamentally incompatible. The algorithms designed to minimize the competitive ratio may not minimize regret [5]. [21] proposes an online optimization algorithm for a single microgrid based on regret minimization. However, the underlying physical power network is ignored in the design of the algorithm. In our work, we take into account the underlying power network and consider networked microgrids in the design of the online energy management.

The reminder of the paper is organized as follows. Section II describes the problem formulation. Section III presents the proposed online energy management algorithm. Section IV provides various numerical studies to verify the performance of the algorithm. Finally, Section V concludes the paper.

II. PROBLEM FORMULATION

Consider a scenario in which a distribution network operator owns some of the microgrids and intends to set up real-time power generation and consumption schedules for various components of microgrids in an online fashion. The components include controllable load and generation (e.g., micro turbine). There are uncontrollable generation sources (e.g., PV) in the microgrid that the distributed network operator does not have the capability to schedule. We also assume that there are microgrids which are privately owned. The distributed network operator can buy/sell power from/to those privately owned microgrids [24]. Figure 1 describes the interaction between the networked microgrids.

For planning purposes, we group utility microgrids and privately owned microgrids (see Figure 1). Notice that although we group the microgrids, the distribution network operator does not directly control the DGs and loads of the private microgrids. The utility microgrids and private microgrids still maintain different ownerships. We merely assume that the privately owned microgrids can buy/sell power from/to the assigned utility microgrids. The mechanism for doing so must be such that every private microgrid should deliver power with only its neighboring utility microgrids to reduce the feeder power routing complexity (e.g., breaker/switch operations).

We focus on the distribution network operator perspective and design an online energy management algorithm to minimize the generation cost of the utility microgrids. The scheduling is over T time slots, and each time slot represents a specific length of time (e.g., one hour). Let \( T := \{1,...,T\} \) denote the set of time slots. The power generation and consumption schedules remain constant in each time slot.

The networked microgrids are characterized as a connected and undirected graph \( G = (\mathcal{M}_s, \mathcal{E}) \). For our setup, the set of vertices \( \mathcal{M}_s := \{1,...,M\} \) denotes the set of microgrids and the set of edges \( \mathcal{E} \) represents the set of power lines connecting microgrids.

We use the DistFlow model (see e.g., [24], [27]) to model an AC distribution network. Following [24], [27], the DistFlow model is simplified to:

\[
\begin{align*}
    p_{ij} &= P_{ij} - \sum_{k:(i,j,k) \in \mathcal{E}, k \in \mathcal{M}_s} P_{jk}, \quad i, j \in \mathcal{M}_s, (i,j) \in \mathcal{E}, \quad (1) \\
    q_{ij} &= Q_{ij} - \sum_{k:(i,j,k) \in \mathcal{E}, k \in \mathcal{M}_s} Q_{jk}, \quad i, j \in \mathcal{M}_s, (i,j) \in \mathcal{E}, \quad (2) \\
    V_j &= V_i - \frac{(r_{ij}P_{ij} + x_{ij}Q_{ij})}{V_0}, \quad (3)
\end{align*}
\]

provided that the voltage is constrained by

\[
1 - \epsilon \leq V_j \leq 1 + \epsilon, \quad \forall j \in \mathcal{M}_s, \quad (4)
\]

where \( p_j = p_j^d - (p_j^d + p_j^s) \), \( q_j = q_j^d - q_j^s \), and \( \epsilon \) is a constant and set to 0.05.

In each time slot \( t \in \mathcal{T} \), the distribution network operator solves an optimization problem. The objective of the optimization problem is to minimize the cost of the micro turbine power generation and the cost of power transactions with the privately owned microgrids. The decision variables are the active/reactive power consumption, the active/reactive power generation of the micro turbines, the net power of the power transaction with privately owned microgrids and the power generation of the PV panels, and the voltage constraints of the utility microgrids.

The optimization problem is formulated as: for \( t \in \mathcal{T}, \)

\[
\begin{align*}
\text{minimize} & \quad C_t(p_i(t)) \\
& \quad + \alpha_t [p_i^l(t) - w_i(t)]^+ + \beta_t [p_i^u(t) - w_i(t)]^- \\
\text{subject to} & \quad p(t) = p^d(t) + p^s(t), \quad q(t) = q^d(t) + q^s(t), \quad w(t) = w^d(t) + w^s(t)
\end{align*}
\]
\[ p_i^2(t) \leq p_i^0(t) \leq \overline{p}_i(t), \quad t \in \mathcal{T}, \quad i \in \mathcal{M}_s, \quad (6) \]
\[ p_i^1(t) \leq p_i^0(t) \leq \overline{p}_i(t), \quad t \in \mathcal{T}, \quad i \in \mathcal{M}_s, \quad (7) \]
\[ p_i^1(t) \leq p_i^0(t) \leq \overline{p}_i(t), \quad t \in \mathcal{T}, \quad i \in \mathcal{M}_s, \quad (8) \]
\[ q_i^1(t) \leq q_i^0(t) \leq \overline{q}_i(t), \quad t \in \mathcal{T}, \quad i \in \mathcal{M}_s, \quad (9) \]
\[ q_i^2(t) \leq q_i^0(t) \leq \overline{q}_i(t), \quad t \in \mathcal{T}, \quad i \in \mathcal{M}_s, \quad (10) \]
\[ \sum_{k,(j,k) \in E} p_{jk}(t) = P_i(t) + p_j(t), \quad t \in \mathcal{T}, \quad \sum_{k,(j,k) \in E, k \in \mathcal{M}_s} \sum_{i,j \in \mathcal{E}} Q_{jk}(t) = Q_{ij}(t) + q_j(t), \quad t \in \mathcal{T}, \quad \sum_{k,(j,k) \in E} Q_{jk}(t) = Q_{ij}(t) + q_j(t), \quad t \in \mathcal{T}, \quad (11) \]
\[ V_j(t) = V_i(t) - \frac{(r_{ij} P_{ij}(t) + x_{ij} Q_{ij}(t))}{V_0}, \quad t \in \mathcal{T}, \quad 1 - \epsilon \leq V(t) \leq 1 + \epsilon, \quad t \in \mathcal{T}, \quad (12) \]
\[ \text{where the cost function } C_i(\cdot), \quad i \in \mathcal{M}_s \text{ is quadratic in } p_i^0, \text{ which encodes the micro turbine power generation cost [24], i.e.,} \]
\[ C_i(p_i^0(t)) := a_ip_i^0(t)^2 + b_ip_i^0(t) + c_i, \quad (15) \]
\[ \text{where } a_i \in \mathbb{R}_{>0}, b_i \in \mathbb{R}_{>0}, c_i \in \mathbb{R}_{>0} \text{ are positive real numbers. The transaction cost is defined as:} \]
\[ \alpha_i \left[ p_i^0(t) - w_i(t) \right]^+ + \beta_i \left[ p_i^0(t) - w_i(t) \right]^-, \quad (16) \]
\[ \text{where } [a]^+ := \max\{a, 0\}, \text{ and } [b]^- := \max\{-b, 0\}. \text{ We use } p_i^0, \quad i \in \mathcal{M}_s \text{ to denote the net power of the power transaction with the privately owned microgrids and the onsite PV power generation of the utility microgrid. Therefore the power that is bought from the privately owned microgrids is equal to } \alpha_i \left[ p_i^0(t) - w_i(t) \right]^+ \text{ and the power that is sold to the privately owned microgrids is equal to } \beta_i \left[ p_i^0(t) - w_i(t) \right]^-. \text{ Following [28], the transaction cost (16) is convex if the sale price does not exceed the purchase price, i.e., } \beta_i \leq \alpha_i, \quad i \in \mathcal{M}_s. \text{ The equations (6)-(10) are the capacity constraints for the corresponding components and (11)-(14) are the constraints in the DistFlow model.} \]

The active power generation of PVs is uncertain and unpredictable. More precisely, at the beginning of time slot \( t \in \mathcal{T} \), we assume that the distribution network operator only has the information about the power generation of PVs before time slot \( t \), i.e., \( w_i(s), \quad s = 1, ..., t - 1, \quad i \in \mathcal{M}_s \) and does not know the future power generation of PVs \( w_i(s), \quad s = t, ..., T, \quad i \in \mathcal{M}_s \). In the time slot \( t \), the distribution network operator implements the energy management algorithm. At the end of the time slot \( t \in \mathcal{T} \), the realized power generation of PVs \( w_i(t), \quad i \in \mathcal{M}_s \) is recorded and can be used by the operator for scheduling in the next time slot \( t+1 \).

The distribution network operator can, in principle, solve the problem (5)-(14) in a centralized manner. In this case, however, the communication demand between the the operator and the networked microgrids is high [1]. To reduce the communication demand, it is of interest to design a distributed algorithm to solve the problem [1]. The mechanism allows but limits every utility/private microgrid to exchange information with only its adjacent utility/private microgrids.

### III. On-line Distributed Algorithm for Energy Management

Our problem (5)-(14) involves coupled voltage and power flow constraints among the microgrids. The branch power flow and the voltage in the distribution network form the global variable denoted by \( z \); see the following example. For every microgrid \( i \in \mathcal{M}_s \), the corresponding branch power flows that go through microgrid \( i \) form the part of the local variable denoted by \( y_i \). Each branch power flow represents a specific part of the set of the branch power flows in the microgrids.

Thus, local variable \( y_i \) is a part of global variable \( z \). We denote this relationship by equation \( y_i = \tilde{y}_i \), where \( \tilde{y}_i \) is the specific part of \( z \) that contains local variable \( y_i \). Similarly, for microgrid \( i \in \mathcal{M}_s \), voltage \( V_i \), and \( V_j, \quad j \in \mathcal{M}_s \) form the part of local variable \( y_i \), where \( \mathcal{M}_i \) is the set of adjacent microgrids of microgrid \( i \). For microgrid \( i \), \( i \in \mathcal{M}_s \), the \( i \)-th microgrid’s power consumption and generation form the private variable denoted by \( x_i \). While private variable \( x_i \) is not shared with other microgrids, it is related to local variables \( y_i \) to satisfy the physical power network constraints. We let \( r_i := (x_i, y_i), \quad i \in \mathcal{M}_s \).

#### Example 3.1: Consider microgrid \( i, \quad i \in \mathcal{M}_s \) in Figure 2. We construct local variable \( y_i = (V_i, V_j, P_{i-1}, Q_{i-1}, P_i, Q_i) \) and global variable \( z = (V_1, ..., V_M, P_1, Q_1, ..., P_M, Q_M) \). The local variable represents a specific part of the global variable, i.e., \( y_i = B_z z \), where matrix \( B_z \) contains \( \{0, 1\} \). We construct private variable \( x_i = (p_i^0, q_i^0, p_i^1, q_i^1) \).

We define three types of sets. For each microgrid \( i \in \mathcal{M}_s \), set \( \mathcal{L}_i \) denotes the set of branch power flows that go through the \( i \)-th microgrid and the voltage \( V_i, V_j, \quad j \in \mathcal{M}_i \) that satisfies the physical power network constraints for the corresponding power lines; see the following example. For each microgrid \( i \in \mathcal{M}_s \), set \( \mathcal{H}_i \) simply encodes the set of constraints for the private variable \( x_i \) and the local variable \( y_i \). Denote the feasible set of power schedules by \( \mathcal{O} := \{ y_i \in \mathcal{L}_i, \quad x_i \in \mathcal{F}_i, \quad r_i \in \mathcal{H}_i, \quad i \in \mathcal{M}_s \} \).

#### Example 3.2: Consider microgrid \( i, \quad i \in \mathcal{M}_s \) in Figure 2.
We construct the sets as:
\[
\mathcal{L}_i = \left\{ V_i = V_{i-1} - \frac{r_{(i-1)}(t) P_{i-1} + x_{(i-1)} Q_{i-1}}{V_0} \right\},
\]
\[
\mathcal{F}_i = \left\{ P_i^0 \leq P_i^q \leq \overline{P}_i^q, ~ \overline{P}_i^q \leq P_i^f \leq \overline{P}_i^f, \quad \overline{P}_i^f \leq P_i^l \leq \overline{P}_i^l, \quad \overline{P}_i^l \leq P_i^q \leq \overline{P}_i^q \right\},
\]
\[
\mathcal{H}_i = \left\{ P_i = P_{i-1} + p_i, \quad Q_i = Q_{i-1} + q_i \right\},
\]
where local variable \( y_i = (V_{i-1}, V_i, P_{i-1}, Q_{i-1}, P_i, Q_i) \) and private variable \( x_i = (p_i^l, p_i^q, p_i^f, p_i^r, q_i^l, q_i^q) \).

### A. ADMM

Following the previous discussion, we are able to construct the corresponding set \( \mathcal{O} := \{ y_i \in \mathcal{L}_i, x_i \in \mathcal{F}_i, r_i \in \mathcal{H}_i, \quad i \in \mathcal{M}_s \}. \) Thus, problem (5)-(14) is transformed to the following optimization problem with the suitable defined sets \( \mathcal{L}_i, \mathcal{F}_i \), and \( \mathcal{H}_i, \quad i \in \mathcal{M}_s \). Consider the following optimization problem:
\[
\text{minimize } f^t(r) + g(r),
\]
where the equality \( Ar + Bz = 0 \) with suitable defined matrices \( A \) and \( B \) encodes the relationship between the local variable and the global variable. In (18), \( f^t(r) \) is the objective function\(^1\) in (5) and \( g(r) \) is the indicator function of the feasible set \( \mathcal{O} = \{ y_i \in \mathcal{L}_i, x_i \in \mathcal{F}_i, r_i \in \mathcal{H}_i, \quad i \in \mathcal{M}_s \}, \) i.e.,
\[
g(r) = \begin{cases} 
\text{constant, } \forall r \in \mathcal{O}, \\
\infty, \text{ otherwise.}
\end{cases}
\]

We formulate the augmented Lagrangian of (18):
\[
L = f^t(r) + g(r) + \lambda^T (Ar + Bz) + \frac{\rho}{2} \| Ar + Bz \|^2, \quad (20)
\]
where \( \lambda \) is the corresponding Lagrangian multiplier and the positive real number \( \rho \) is a given parameter.

The ADMM iteration (with iteration index \( k \)) uses two steps to update the primal variables \( (r, z) \) and one step to update the dual variable \( \lambda \), which leads to:
\[
r^{k+1} = \arg\min_r f^t(r) + g(r) + \lambda^T (Ar + Bz) + \frac{\rho}{2} \| Ar + Bz \|^2,
\]
\[
z^{k+1} = \arg\min_z \lambda^T (Ar^{k+1} + Bz) + \frac{\rho}{2} \| Ar^{k+1} + Bz \|^2,
\]
\[
\lambda^{k+1} = \lambda^k + \rho (Ar^{k+1} + Bz^{k+1}).
\]

The ADMM iteration (21) is distributed. The \( r \) and \( \lambda \) updates can be implemented at each microgrid and the \( z \) update can be implemented among neighboring microgrids [7]. We refer the reader to [7] and references therein for various convergence results of ADMM (21).

\(^1\)We use superscript \( t \) to indicate that the objective function in (5) is a function of time slot \( t \in \mathcal{T} \).

### B. Online ADMM

The online ADMM iteration [23] to solve (18) leads to
\[
r^{t+1} = \arg\min_r f^t(r) + g(r) + \lambda^T (Ar + Bz^t) + \frac{\rho}{2} \| Ar + Bz \|^2,
\]
\[
z^{t+1} = \arg\min_z \lambda^T (Ar^{t+1} + Bz) + \frac{\rho}{2} \| Ar^{t+1} + Bz \|^2,
\]
\[
\lambda^{t+1} = \lambda^t + \rho (Ar^{t+1} + Bz^{t+1}),
\]
(22)
where the iteration (22) is executed in each time slot \( t = 1, ..., T \).

Notice that in (22), in every time slot \( t \in \mathcal{T} \), there is only one iteration to update the decision \( (r^{t+1}, z^{t+1}, \lambda^{t+1}) \), i.e., iteration index \( k = 1 \), which is a standard formulation in the online convex optimization framework (see, e.g., [8], [13], [23]). The inner loop subroutines of online ADMM are not required to be fully executed (with only one iteration). This feature substantially reduces the implementation time and mitigates the communication burden.

We characterize the performance of the power schedule solutions generated by the online ADMM based on the regret minimization framework. In the regret minimization framework, the notion of regret is adopted to measure the performance of the decisions made. Regret after \( T \) time slots \( R(T) \) is defined as:
\[
R(T) := \sum_{t=1}^T f^t(r^t) + g(r^t) - \min_{Ar + Bz = 0} \sum_{t=1}^T f^t(r) + g(r).
\]
(23)

Clearly, regret \( R(T) \) measures the difference between the cumulative cost of decisions \( r^t \), \( t = 1, ..., T \) and that of the best fixed decision in hindsight. A fixed decision remains the same for all time slots \( t = 1, ..., T \). The best fixed decision can only be computed in hindsight (i.e., at the end of the \( T \)-th time slot). The goal of the online algorithm is to obtain an upper bound for \( R(T) \) such that the bound is sublinear in \( T \), e.g., \( R(T) = O(\sqrt{T}) \) [8]. In our problem, the decisions are variables \( r^t \), \( t = 1, ..., T \) which consist of private variables \( x_i^t \), \( i = 1, ..., T \), and local variables \( y_i^t \), \( i = 1, ..., T \) for all microgrids \( i \in \mathcal{M}_s \). Regret \( R(T) \) measures the difference between the cumulative cost of solutions \( r^t \), \( t = 1, ..., T \) and the cumulative cost of the best fixed solution in hindsight.

In general, the ADMM iteration does not generate feasible solutions before convergence, i.e., \( Ar^t + Bz^t \neq 0 \) for all \( t = 1, ..., T \) [7]. In the following, we define the regret for feasible solution \( r^t \). Consider \( r^t \) such that \( Ar^t + Bz^t = 0 \) for all \( t = 1, ..., T \). The regret for \( r^t \), \( t = 1, ..., T \) is defined as
\[
\tilde{R}(T) := \sum_{t=1}^T f^t(r^t) + g(r^t) - \min_{Ar + Bz = 0} \sum_{t=1}^T f^t(r) + g(r).
\]
(24)

In the following, we define the regret for the constraint violation [23]. The regret \( \tilde{R}(T) \) for the constraint violation
is defined as:
\[
R_c(T) := \sum_{t=1}^{T} \|A r^{t+1} + B z^{t+1}\| + \|B z^{t+1} - B z^t\|. \tag{25}
\]

The average primal residual is defined as:
\[
\frac{1}{T} \sum_{t=1}^{T} \|A r^{t+1} + B z^{t+1}\|. \tag{26}
\]

The average primal residual measures the feasibility of the solutions generated by the online ADMM iterations. The average primal residual approaching zero indicates that the primal variable approaches the feasible set of solutions.

We make the following assumptions which will facilitate the analysis for the regret.

**Assumption 1:**
\[
\|\nabla f^i(r)\| \leq E_f, \quad \forall t = 1, \ldots, T. \tag{27}
\]
\[
r^i = 0, \quad z^i = 0, \quad \lambda^i = 0, \quad g(z^i) = 0, \quad g(z) \geq 0. \tag{28}
\]

For any \(r^*\) and \(z^*\) such that \(Ar^* + Bz^* = 0\),
\[
\|z^* - z^i\| \leq E_z, \tag{29}
\]
\[
f^i(x^{t+1}) + g(r^{t+1}) - (f^i(x^*) + g(r^*)) \geq -G, \quad \forall t = 1, \ldots, T. \tag{30}
\]

In Assumption 1, (27) is usually required in the online learning framework [8], (28) and (29) are used to facilitate the analysis. (30) is used to prove the sub-linear regret bound for the constraint violation and it is satisfied if functions \(f^i\) and \(g\) are Lipschitz continuous in the convex set or bounded from below [23].

The results in [23] show that both regrets \(\bar{R}(T)\) and \(R_c(T)\) are in the form of \(O(\sqrt{T})\). The results characterize the performance of the online ADMM iteration (22). Thus, if we consider the average regret, i.e., \(\bar{R}(T)/T\) or \(R_c(T)/T\) and let the total number of time slots go to infinity, then the average regret goes to zero. The results characterize the asymptotic performance of the decisions generated by the online ADMM iterations, namely, asymptotically, the average cumulative generation cost obtained from the online ADMM iteration converges to the one with the optimal fixed decision in hindsight.

So far we have characterized the performance of the algorithm by comparing it with the one of the fixed decision in hindsight. In practice, due to the high variability of DERs, optimal decisions may vary between time slots \(t = 1, \ldots, T\). In comparison with the fixed decision, we can only provide a suboptimal performance guarantee of the decisions. In the following, to capture the high variability of DERs and to provide a comparison with the optimal performance, we relax the fixed nature of the decision that we compare with. We compare the performance of the solutions generated by the algorithm with the one generated by an optimal sequence of decisions which may vary between time slots \(t \in T\).

In order to distinguish the difference between comparing with the fixed decision and a sequence of slot-varying decisions, the regret for the later situation is named as tracking regret [11].

In the following, we define the tracking regret for the feasible solution \(\bar{r}\). Consider \(r^i\) such that \(Ar^i + Bz^i = 0\), for all \(t = 1, \ldots, T\). The tracking regret for \(r^i\), \(t = 1, \ldots, T\) is defined as:
\[
\bar{R}_{\text{tracking}}(T) := \sum_{t=1}^{T} f^i(r^i) + g(r^i) - \min_{Ar^i + Bz^i = 0, t \in T} \sum_{t=1}^{T} f^i(r^i) + g(r). \tag{31}
\]

Similarly, we define the regret for the constraint violation. The tracking regret for the constraint violation is defined as:
\[
R_{c,\text{tracking}}(T) := \sum_{t=1}^{T} \|A r^{t+1} + B z^{t+1}\| + \|B z^{t+1} - B z^t\|. \tag{32}
\]

The following lemma provides the bounds on the difference between the optimal cost function value and the cost function value of the decisions generated by the online ADMM iteration (22).

**Lemma 3.1:** Suppose that \(r^{*i}\), \(z^{*i}\), \(t \in T\) are the optimal slot-varying solutions, and \(r^i\), \(t \in T\) are the decisions generated by the iteration (22),
\[
f^i(r^{t+1}) + g(r^{t+1}) - (f^i(r^{*i}) + g(r^{*i})) \leq \frac{1}{2\rho} \left( \|\lambda^i\|^2 - \|\lambda^{t+1}\|^2 \right) - \frac{\rho}{2} \|Ar^t + Bz^t\|^2 + \frac{\rho}{2} \|B z^{*t} - B z^t\|^2 - \|B z^{*t+1} - B z^{t+1}\|^2 \tag{33}
\]
\[
+ \rho \|BT B\| \left( \|z^{*t+1}\|^2 - \|z^{*t}\|^2 \right) + \rho \|BT B\| Q \sum_{t=1}^{T} \|z^{*t+1} - z^{*t}\|,
\]
where \(Q := \max_{z \in L} \|z\|\).

The proof can be found in Appendix A.

**Theorem 3.2:** Suppose that Assumption 1 holds, and the parameter \(\rho\) is selected as \(\rho = \frac{E_f \sqrt{T}}{\lambda_{\min}^A \lambda_{\max} B}\),
\[
\bar{R}_{\text{tracking}}(T) \leq \frac{E_f \sqrt{T}}{\lambda_{\min}^A \lambda_{\max} B} \left( \|z^T\|^2 - \|z^1\|^2 \right) \tag{34}
\]
\[
+ \frac{E_f \sqrt{T} \|BT B\|}{2E_z \sqrt{\lambda_{\min}^A \lambda_{\max} B}} \sum_{t=1}^{T} \|z^{*t+1} - z^{*t}\|.
\]

Similarly, we have
\[
R_{c,\text{tracking}}(T) \leq \frac{E_f \sqrt{T} \|BT B\|}{2E_z \sqrt{\lambda_{\min}^A \lambda_{\max} B}} \sum_{t=1}^{T} \|z^{*t+1} - z^{*t}\| + \frac{2GE_z \sqrt{\lambda_{\min}^A \lambda_{\max} B}}{E_f} \sqrt{T} \tag{35}
\]
\[
+ \|BT B\| \left( \|z^{*T}\|^2 - \|z^{*1}\|^2 \right) \]
\[
+ 2Q \|BT B\| \sum_{t=1}^{T} \|z^{*t+1} - z^{*t}\|,
\]
where \(Q = \max_{z \in L} \|z\|\) and \(\lambda_{\max}^B\), \(\lambda_{\min}^A\) are the maximum eigenvalue of \(B\) and minimum eigenvalue of \(A\), respectively. The results are obtained by following [23], Lemma 3.1, and by summing over time slots.
IV. NUMERICAL RESULTS

We consider a 37-bus (M = 37) test system [22] in Figure 3. Each bus represents a group of microgrids as noted in Section II (Figure 1). We consider 24 time slots (T = 24). Each time slot represents one hour duration. We set the starting time to be 00:00. Therefore, the first time slot starts from 00:00 and ends at 01:00; the last time slot starts from 23:00 and ends at 00:00 (the next day). The base power is set to 10 MVA.

Table I shows the parameters of the microgrids. Table II shows the parameters of the cost function (5). We use the solar PV generation profile in Belgium [2] as shown in Figure 4. It is evident that the real-time measurements deviate substantially from the forecast. Since we want to prevent over-conservative energy scheduling due to the inaccurate forecast, the following simulation does not rely on the inaccurate forecast to proceed. More precisely, at the beginning of time slot \( t \in T \), the future PV power generation \( w_i(s), s = t, t + 1, ..., T \), \( i \in M_s \) is unknown to the distribution network operator. At the end of the time slot \( t \), data \( w_i(t), i \in M_s \) is measured and revealed to the distribution network operator for scheduling in the next time slot \( t + 1 \).

Figure 4 shows the cost of the optimal slot-invariant power schedule and the cost of the optimal slot-varying power schedule. The optimal slot-invariant and slot-varying costs can only be computed at the end of the last time slot (after all the power generation of the PV sources are realized). We can see in Figure 4 that the optimal slot-varying cost varies between the time slots. Notice that the optimal slot-varying cost is negative between 10:00 - 14:00, during which the PV power generation reaches the peak of the power generation. The utility microgrids sell the surplus power generation to the privately owned microgrids and obtain earnings resulting in a negative cost of power generation for the utility microgrids.

In the following, we will use the online ADMM algorithm to generate the power schedule and to track the optimal costs in Figure 4. We execute the online ADMM algorithm in the energy management system to schedule the power generation and consumption profiles for the micro turbines and the power loads. In each time slot \( t \in T \), we use CVX/CVXGEN [10], [20] to solve the corresponding optimization problem (5)-(14).

Each optimization problem is a convex program. Figure 5 shows the power schedules of the micro turbines and the power bought/sold from/to the privately owned microgrids. We can see in Figure 5 that the power generation of the micro turbines decreases as the power generation of PV energy sources increases. During the time slots when the PV power generation is around the peak of the power generation, the power generation of micro turbines is approaching zero. The micro turbines do not generate power during the peak of the PV power generation. Notice that when the online ADMM algorithm generates the power schedule in each time slot, we do not include the forecast of the PV power generation (since the forecast may not be accurate). However, the generated power schedule solutions have the ability to adjust according to the variation of the PV power generation. In Figure 5, we reduce the power generation price of the micro turbines. The generation price is changed from \( b_i = 0.2 \$/kWh \) to \( b_i = 0.1 \$/kWh \). We can see in Figure 5 that the power generation of the micro turbines increases as the power generation price decreases. A lower micro turbine generation price results in a larger amount of power generation from the micro turbines.

In order to verify the effectiveness of the online ADMM algorithm proposed in this work, we compare our online algorithm with the standard ADMM energy management algorithm including the inaccurate PV forecast (see, e.g., [9], [17]). The online ADMM algorithm does not require any
PV forecast to proceed. The standard ADMM algorithms [9], [17] require a PV forecast to proceed. Figure 6 shows the convergence (converging to zero) of the average regret of the online ADMM algorithm. It verifies that despite the fact that the PV power generation is slot-varying and uncertain, and the online ADMM algorithm does not have any a priori knowledge about the future PV power generation, the performance of the power schedule solution generated by the online ADMM algorithm approaches the performance of the best solution in hindsight. We can see in Figure 6 that in this case, at the end of the day, the average cumulative cost generated by the online ADMM algorithm is about $800 less than the cost generated by the standard ADMM algorithm with the inaccurate PV forecast. It verifies the effectiveness of the proposed online ADMM algorithm.

V. CONCLUSION

We propose a distributed online ADMM algorithm with regret for online energy management in networked microgrids. The algorithm does not require any a priori knowledge about the future power generation from the PV panels or the convergence of the inner loop subroutines of the algorithm to proceed; it overcomes the over-conservative issue of robust optimization based algorithms and substantially reduces the computation and communication burden. A performance guarantee of the power schedule solutions generated by the online ADMM iteration has been derived. The simulation results show the effectiveness of the proposed online ADMM algorithm.

APPENDIX A

PROOF OF LEMMA 3.1

Following [23], we have

\[
f^t(x^{t+1}) + g(r^{t+1}) - (f^t(x^*) + g(r^*)) \leq \frac{1}{2\rho} (||x^t||^2 - ||x^{t+1}||^2) - \frac{\rho}{2} ||Ar^{t+1} + Bz^t||^2 \leq \frac{\rho}{2} (||x^t||^2 - ||x^{t+1}||^2) - \frac{\rho}{2} ||Ar^{t+1} + Bz^t||^2 \leq \frac{\rho}{2} (||x^t||^2 - ||x^{t+1}||^2) - \frac{\rho}{2} ||Ar^{t+1} + Bz^t||^2 + \rho \left( \sum_{s=t+1}^{t+T} \|z^s - z^{t+1}\|. \right)
\]

Furthermore,

\[
\begin{align*}
\frac{\rho}{2} (||Bz^{s t} - Bz^t||^2 - ||Bz^{s t} - Bz^{t+1}||^2) \\
\leq \frac{\rho}{2} (||Bz^{s t} - Bz^t||^2 - ||Bz^{s t+1} - Bz^{t+1}||^2) \\
+ \frac{\rho}{2} (||Bz^{s t+1} - Bz^{t+1}||^2 - ||Bz^{s t} - Bz^{t+1}||^2) \\
\leq \frac{\rho}{2} (||Bz^{s t} - Bz^t||^2 - ||Bz^{s t+1} - Bz^{t+1}||^2) \\
+ \frac{\rho}{2} \left( \sum_{s=t+1}^{t+T} \|z^s - z^{t+1}\|. \right)
\end{align*}
\]

where \( Q = \max_{z \in \mathcal{Z}} \|z\| \). By combine (36) and (37), we complete the proof.

REFERENCES


