Coordination of Distributed Energy Storage Under Spatial and Temporal Data Asymmetry

Kyle Anderson, Ram Rajagopal, *Member, IEEE*, and Abbas El Gamal, *Life Fellow, IEEE*

***Abstract*—We consider the problem of controlling distributed storage in a network with renewable distributed generation to minimize operational cost while satisfying power quality constraints. We assume control is distributed between a global controller (GC) and local controllers (LCs) located at the nodes with storage units. Each LC has access to its most recent net load data and runs at every time step while the GC has delayed data due to smart meter infrastructure or communication network delay, hence runs less frequently. We describe three control schemes: 1) direct storage controller (DSC) in which the GC computes the storage control signals for an upcoming window and the LCs directly use these signals; 2) net load following controller (NLFC) in which the GC computes a net load profile for each node and each LC tries to track its set profile; and 3) nodal slack controller (NSC) in which the GC computes upper and lower bounds on the net load at each node and the LC optimizes the local control action constrained by these bounds. We use a radial network with real load data to compare the performance of these schemes based on arbitrage profit and maximum solar penetration relative to a perfect foresight controller. We find that NSC and NLFC increase the supported maximum solar penetration to 29% and 40%, respectively, as compared to 10% for DSC. Moreover, NSC is able to capture 90.7% of the available arbitrage profits which is significantly higher than that achievable with NLFC and DSC.**

***Index Terms*—Energy storage, distributed generation, control systems.**

I. Introduction

DISTRIBUTED energy storage (DES) can benefit the grid in several ways, including (i) shifting the load through energy arbitrage (EA) by charging during off-peak hours and discharging during peak hours; (ii) supporting renewable distributed generation (RDGs) by performing Volt/VAr control to help reduce the voltage violations, reactive power imbalance, and increased network losses caused by the bidirectional power flow introduced by RDGs, (iii) providing regulation service (RS) or other ancillary services that absorb short-term mismatches in supply and demand; (iv) avoiding future capital expenses for distribution infrastructure used to serve peak load; and (v) increasing grid security by providing emergency power during outages. In this paper, we focus on optimizing and quantifying the first two benefits of DES. The second benefit of supporting RDGs is especially important today as rooftop solar installations in the U.S. have been increasing by 7.5% per year [1] and are growing even faster in some regions. For example, in Hawaii, rooftop solar penetration has already reached 12% prompting grid operators to institute limits on net metering programs [2] to limit the aforementioned adverse effects of bidirectional power flow. Realizing the potential benefits of DES, however, requires the development of scalable distributed storage control schemes that operate within the power and data constraints of real world electric power distribution systems with potentially tens of thousands of customers.

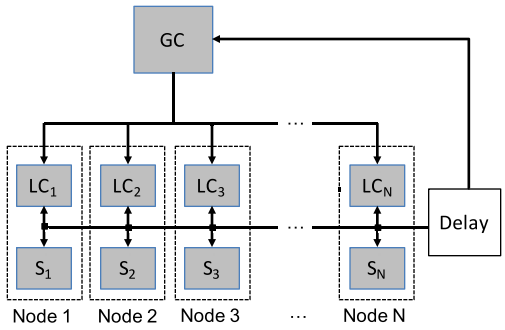


Fig. 1. Data flow in a distribution system. Shown are N customers with net loads $S_1, \ldots, S_N$ and local controllers $\mathrm{LC}_1, \ldots, \mathrm{LC}_N$. The local controllers communicate their data to a global controller GC, subject to a delay. The GC sends back control signals to the LCs based on the delayed information and its knowledge of the distribution network.

A primary challenge in developing such control schemes is the spatial and temporal asymmetry in the availability of load and generation data across the network due to: (i) the stochastic nature of loads and renewable generation, (ii) the distributed locations of the loads, renewable generation, and storage units, and (iii) the buffering delay in communicating the data from where it is generated to where it is used for the control. We capture this spatiotemporal asymmetry in data availability via the high level model of the communication/control plane of a distribution system depicted in Figure 1. The system comprises N nodes, each representing an aggregated collection of stochastic uncontrollable loads and

Manuscript received November 23, 2016; revised April 8, 2017 and June 20, 2017; accepted August 2, 2017. Date of publication August 15, 2017; date of current version February 18, 2019. This work was supported by the NSF CAREER under Grant EECS-1554178. Paper no. TSG-01652-2016. *(Corresponding author: Ram Rajagopal.)*

K. Anderson is with Rockset, Menlo Park, CA 94025 USA.

R. Rajagopal is with the Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: ram.rajagopal@stanford.edu).

A. El Gamal is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2017.2740430

renewable generators as well as controllable storage. The total net load from each of these sources (which is stochastic and only partially controllable) is denoted by s_i, $i = 1, 2, \ldots, N$. The nodes are connected via a communication network to a global controller GC, which is a software process that may be running at the substation or in the cloud. The GC has access to net load data from the nodes through a communication network subject to a delay. This delay may be the result of the smart meter infrastructure, which can introduce over 6 hours of delay in data propagation [3], or simply due to worst case packet delays in the communication network itself. We assume that the GC has perfect knowledge of the electric network model (line impedances, transformer settings, etc.), and can therefore solve a network power flow problem to compute voltages at each point in the network. However, the results of this computation are imperfect due to the delay and the intrinsic uncertainty about future net node loads.

Each node in our model has sensors that measure load consumption and renewable generation in real-time, and has the capability to communicate with the GC and to perform local computations. The node has a local controller (LC), which is a software process that may be running within a smart meter, within the firmware of the storage unit, or in a home automation appliance. The LC can control the net nodal load by setting the storage rate, subject to the physical constraints of the storage unit. The LC at node i makes its decisions based on perfect knowledge of past and present values of its load S_i as well as on the control signals from the GC. The LC does not have perfect knowledge of future S_i values due to the stochastic nature of the load and renewable generation. However, it has a better estimate of its future load than the GC because it has access to more recent past data of its own load. This is reflected in most common load models (e.g., ARIMA) in which the uncertainty increases with the amount of delay in the most recent observed load sample.

The problem we focus on in this paper is how to distribute the DES control between the GC and the LCs to simultaneously optimize energy arbitrage and support RDGs under the constraints and assumptions of the model in Figure 1. In addition to developing scalable distributed control schemes and quantifying their benefits in helping support increased DRG penetration, we aim to understand the dependence of the system performance on communication delay. This is important because tolerance to high delay can make the system less vulnerable to communication network failures, reduce the need to upgrade the smart meter infrastructure to provide faster data propagation, and make it feasible to simply use existing home broadband services to communicate with the GC. This framework can be easily extended to consider LCs asynchronously operating at a different time-scale than the GC.

The optimal control of distributed storage in power distribution networks involves the interaction between the cyber and physical system in the power network. Previous studies do not consider every constraint imposed by this interaction. More specifically, they do not consider at least one of the following factors: (i) the nonlinear network power flow constraints, (ii) the stochastic nature of loads and renewable generation, (iii) the spatial and temporal asymmetry of data (load/RDG forecasts), and (iv) the communication delay in the network. In a setting with deterministic loads and RDG outputs, the optimal control can be determined exactly by solving a global optimal power flow problem over a finite time horizon and with storage dynamics [4]. In a stochastic load setting, heuristic methods such as the rollout algorithms or model predictive control have been applied to approximate control problems of renewable generation and storage; see [5]–[8]. More recent papers, e.g., [9]–[11], have developed stochastic network control algorithms that approximate the optimal solution with different types of performance guarantees. These methods do not consider the spatial and temporal asymmetry of data. In [7] and [10], distributed optimization methods are described for solving the optimal power flow problem with storage over a finite time horizon. Each node in the network solves a subproblem and exchanges messages with its neighbors iteratively until the system finds a globally optimal solution. The methods assume no delay in communications and the availability of a reliable peer to peer connection for each LC capable of supporting a large number of iterations of the algorithms for each storage decision. In practice, existing utility networks do not support such real-time peer-to-peer communication. Experimental systems where the LC utilizes broadband Internet to communicate with the GC are unable to guarantee sufficient communication reliability to enable a large number of fast iterations.

In this paper, we develop DES control schemes in which the GC computes and sends output signals to the LCs based on the delayed net load, and each LC computes a control signal for the storage unit under its purview based on real-time local net load, state of charge, and the output signal received from the GC. This process is repeated continuously over an analysis horizon using Receding Horizon Control (RHC) [12], [13]. In the next section, we model the physical constraints of the network and storage as well as the communication constraints and formulate the DES control problem as a general stochastic optimization problem. We then present an optimal solution for this problem under perfect load and generation foresight. This provides an infeasible solution to the real problem but serves as a benchmark for the performance of implementable control schemes. In Section III, we present three heuristic distributed control schemes. The first scheme is the Direct Storage Control (DSC) in which the GC algorithm solves an optimal power flow using forecasts based on delayed net load data, and each LC passes through the GC output as the control signal. This method suffers in networks with large delays because it does not consider the more fresh past net load data available at the LCs. The second scheme is the Net Load Following Control (NLFC) in which the GC outputs a net load profile for each customer. Each LC generates a new forecast at each timestep and updates the charging schedule to minimize the deviation from the load profile sent by the GC. This method benefits from updating the storage decisions based on the more fresh load data at the LCs, but suffers from the requirement that each net load must follow a precomputed profile, which may reduce arbitrage profits. The last scheme is the Nodal Slack Controller (NSC) in which the GC outputs feasible

bounds on the net load at each node and each LC determines a control signal by minimizing its local operational cost while ensuring that the net load stays within these bounds. This technique avoids the overly restrictive nature of the NLFC method by loosening the constraints on the net load profile. In the last section, we characterize the performance of these three distributed control schemes under an array of assumptions regarding the communication delay, network model, and forecast error. We show that the NLFC and NSC schemes are able to support higher RDG penetrations than DSC, and that the NSC scheme is able to do so with higher arbitrage profits.

II. Problem Setup

We consider the steady state power flows in a radial distribution network modeled by a tree with $N+1$ nodes. We analyze this network over T timesteps each of length $\delta_{\min}$ and assume all quantities are constant over each timestep. The notation and variables used in the problem setup are summarized in Table I. The nodes in the network are indexed by $i \in \{0, 1, 2, \ldots, N\} = [0 : N]$. Node $i = 0$ is the root node typically corresponding to a substation. The remaining nodes $i \in \{1, \ldots, N\}$ model buses with the following:

Uncontrollable load: This includes a stochastic uncontrollable load and renewable generation, both of which are assumed to be real and independent of the power factor. The combined uncontrollable load at node i and timestep $t \in [1 : T]$ is denoted by d_{it}. This value may be positive or negative.

Energy Storage: A storage unit such as a battery with a maximum capacity $Q^{\max}$.[1] At time t, each storage unit has a net charging rate of u_{it} and a state of charge of q_{it}. We assume that each storage only charges and discharges real power. For simplicity, storage charges and discharges with an efficiency between 0 and 1 λ. Hence, the storage dynamics are $q_{it} = \lambda_i u_{it} \cdot \delta_{\min} + q_{it-1}$. In real-world implementations, we may also want to include constraints such as charging/discharging efficiency or battery cycle life as shown in [14]. We do not include these constraints here for simplicity and because they are not essential to the main goals of this paper.

Denote the complex net load at node i by s_{it}, where $\Re(s_{it}) = d_{it} + u_{it}$ and $\Im(s_{it})$ is determined by the network. The root node load s_{0t} represents the aggregate network load at time t. The real part of this value (denoted by $\Re(s_{0t})$) represents the power purchased or sold into the wholesale energy market at a time-varying price p_t. The relationship between the net load and the voltage at each node $i \in [0 : N]$ is governed by the AC powerflow equation,

$$s_{it} = v_{it} \sum_{j:(j->i)} \left(v_{jt}^* - v_{it}^*\right) y_{ij}^*,$$
$$v_i^{\max} \leq |v_{it}| \leq v_i^{\max},$$

[1]Note that the general formulation includes storage at each node. In practice, storage is likely to exist only at a subset of the nodes in which case the storage capacity at the nodes with no storage is set to zero.

TABLE I
Constants and Variables Used in the Formulation

Constants	
p	Energy price for 1kW of power over $\delta_{\min}$ interval ($\$/(kW \cdot \delta_{\min})$)
$u^{\min}$	Max discharging rate
$u^{\max}$	Max charging rate
$v^{\min}$	Min voltage
$v^{\max}$	Max voltage
y_{ij}	Admittance of line between connecting node i and node j
$q^{\max}$	Storage capacity
Δ_{GC}	Timesteps between GC runs
$\delta_{\min}$	Length of timestep
λ	Storage efficiency
Δ_F	Look ahead horizon for GC
A	Number of scenarios for GC optimization
G	Number of scenarios for LC optimization
Independent Variables	
d	Load (uncontrollable load and renewable generation)
Dependent Variables	
v	Voltage
w	Voltage after convex relaxation
q	State of charge
s	Net load (real part includes the load and charging)
u	Charging rate
Notation for any random variable X	
x_{it}	Variable x at node i at timestep t
$\hat{x}$	Future prediction for X
$\hat{x}^a$	Future decision for X in scenario a
$\bar{x}$	Realization of x
Timesteps	
$t(k)$	$k\Delta_{\mathrm{GC}}$ timestep at which the GC runs update k
T	Last timestep in the investment horizon
Other Notation	
$\Re(.)$	Real component of complex number
$\Im(.)$	Imaginary component of complex number
$*$	Complex conjugate

where v_{it} is the node voltage, v_{it}^* is its conjugate, y_{ij} is the admittance of the line connecting node i to node j, and $v_i^{\min}$ and $v_i^{\max}$ are the maximum and the minimum of the voltage magnitude, respectively.

If all d values are known and real-time communication is available to distribute charging decisions, we can determine

the storage charging rates that minimize the total operational cost of the network while ensuring that all network constraints are satisfied by solving the following program:

$$\begin{aligned}
\underset{u,s,q,V}{\text{minimize}} \quad & \sum_{t=1}^{T} p_t \cdot \Re(s_{0t}) \\
\text{subject to} \quad & \Re(s_{it}) = \bar{d}_{it} + u_{it}, && \text{(1a)} \\
& q_{it} = \lambda_i q_{it-1} + u_{it} \cdot \delta_{\min}, && \text{(1b)} \\
& u_i^{\min} \le u_{it} \le u_i^{\max}, && \text{(1c)} \\
& 0 \le q_{it} \le q_i^{\max}, && \text{(1d)} \\
& s_{it} = v_{it} \sum_{j:(j \leftarrow i)} (v_{jt}^* - v_{it}^*) y_{ij}^*, && \text{(1e)} \\
& v_i^{\min} \le |v_{it}| \le v_i^{\max}. && \text{(1f)}
\end{aligned}$$

In the above program, equations (1a) to (1f) are included for nodes $i \in \{1, \ldots, N\}$ and $t \in \{1, \ldots, T\}$. In addition, equation (1e) is also included for the root node $i = 0$ which is assumed to be the reference bus, and equation (1a) corresponds to the complex net load injection at node i in time t. For the storage at node i at time t, equation (1b) corresponds to its state of charge, equation (1c) to charging and discharging limits and equation (1d) to the charging capacity limit. Equation (1e) is the power flow equation and equation (1e) corresponds to the voltage magnitude constraints for node i. As written, the voltage constraints in the program are non-convex. Since the network is a tree, we can reformulate it as a convex problem using the convex relaxation technique in [15] by replacing the voltage decision variables with

$$w_{ij} = v_i v_j^*$$

for every node i connected to node j. Denote by W the matrix consisting of all w_{ij} values and by $W\{i,j\}$ the 2 x 2 submatrix

$$W\{i,j\} = \begin{bmatrix} w_{ii} & w_{ij} \\ w_{ji} & w_{jj} \end{bmatrix} \quad (j \leftarrow i).$$

We can now express (1) using the equivalent convex program,

$$\begin{aligned}
\underset{u,s,q,W}{\text{minimize}} \quad & \sum_{t} p_t \cdot \Re(s_{0,t}) \\
\text{subject to} \quad & \Re(s_{it}) = \bar{d}_{it} + u_{it}, \\
& q_{it} = \lambda_i q_{it-1} + u_{it} \cdot \delta_{\min}, \\
& u_i^{\min} \le u_{it} \le u_i^{\max}, \\
& 0 \le q_{it} \le q_i^{\max}, \\
& s_{it} = \sum_{j:(i,j) \in E} (w_{ijt} - w_{iit}) y_{ij}^*, \\
& (v_i^{\min})^2 \le w_{iit} \le (v_i^{\max})^2, \\
& W\{i,j\}_t \succeq 0. && \text{(2)}
\end{aligned}$$

We refer to the solution of this program as the *perfect foresight optimal storage controller* (PFOSC). This solution assumes that a global controller has access to information about loads in real-time. Of course this scenario is unrealistic, but this solution will serve as a benchmark on the performance of our control methods that operate under uncertainty and delay.

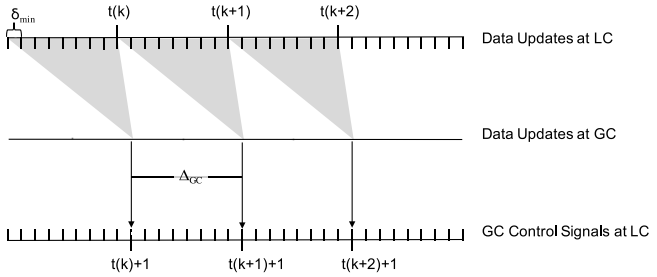


Fig. 2. Timing diagram of data availability in the distribution system. Load data from each LC is delayed and buffered before becoming available to the GC. The GC control signals that use this data are then sent to each LC.

In order to formulate our problem in a more realistic stochastic and distributed environment, we introduce a global controller (GC) and local controllers (LC) at each node. As shown in Figure 2, the GC receives data from the grid subject to a buffering delay Δ_{GC}. This buffering delay arises in real systems since data is collected at local aggregation points within the grid and it may take up to several hours before it is available for processing. Furthermore, although the amount of data sent from the GC to the LCs is relatively small, it may be communicated via unreliable networks such as consumer grade broadband. Therefore, the control scheme must assume worst-case communication delays. In this paper, we assume that Δ_{GC} is fixed and represents the worst-case time required to update all nodes. The GC solves a global optimization problem each time it receives new data. Hence, the GC runs at each timestep $t(k) = k \cdot \Delta_{\text{GC}}$ for each $k = 1, \ldots, T/\Delta_{\text{GC}}$. Each GC solution is available to the LCs at time $t(k) + \Delta_{\text{RTD}}$, where Δ_{RTD} is the propagation delay in the communication network. For simplicity, we assume that $\Delta_{\text{RTD}} = 1$ throughout this paper. Within each iteration k, the LCs perform optimizations at every timestep $t = t(k)+1, \ldots, t(k+1)$. Each of these optimizations depends on the last result received from the GC as well as on newly available data that was not available to the GC the last time it ran.

In practice, the GC and LCs do not know the future state of the system required to implement PFOSC. Instead, the GC and the LCs have the ability to utilize their information to generate estimates of future loads. We denote the forecast of the load for node i at future time τ as $\hat{d}_{i,\tau}$. In general the GC will utilize strategies that use the forecast in lieu of the actual load to determine its output. In the case study in Section IV, we use an ARIMA model to forecast future loads. In general, the load forecast error for a given timestep increases with delay in the available load data, which motivates distributing the control between the GC and the LCs.

Our goal is to find a distributed control method comprising both the GC and the LC processes that jointly minimizes the expected operational cost of the network under the aforementioned asymmetric data model, while ensuring that no network constraints are violated. In addition, distribution networks include voltage and VaR control devices that are utilized to manage the voltage profile in the network. Typically, these systems utilize discrete settings that change to predefined values a few times a day. The optimization program reflects the

Algorithm 1 GC Algorithm for the DSC Scheme; p Is the Price of Electricity, s Is the Net Load, d Is the Load, u Is Storage Charging Rate, and q Is the State of Charge. This Algorithm Runs at Each Timestep $t(k)$ for Each $k = 1, \ldots, T/\Delta_{\mathrm{GC}}$

Inputs:	$q_{it(k)}, \quad i \in [0:N]$
Output:	$u_{i\tau}\ , \quad \tau \in [t(k)+1 : t(k+1)+\Delta_{\mathrm{F}}]$
Forecast:	$\hat{d}^a_{i,\tau}, \quad a \in [1:A]$
Program	$\underset{u,s,q,W}{\text{minimize}} \frac{1}{A}\sum_a \sum_\tau p_\tau \cdot \Re(s^a_{0\tau})$ subject to: $\Re(s^a_{i\tau}) = \hat{d}^a_{i\tau} + u_{i\tau},$ $q_{it} = \lambda_i q_{i\tau-1} + u_{i\tau}\cdot\delta_{\min}$ $u_i^{\min} \le u_{i\tau} \le u_i^{\max}$ $0 \le q_{i\tau} \le q_i^{\max}$ $s^a_{i\tau} = \sum_{j:(i,j)\in E}(w^a_{ij\tau} - w^a_{ii\tau})y^*_{ij}$ $(v_i^{\min})^2 \le w^a_{ii\tau} \le (v_i^{\max})^2$ $W^a\{i,j\}_\tau \succeq 0$

effect of these devices by adjusting network parameters for each hour to reflect the optimal settings of the control devices. Dynamic Volt/VaR optimization could be included as part of the GC process and it would improve the performance of the proposed approaches by enabling additional power injection and extraction in the network buses without violation of constraints. This paper focuses on the GC/LC optimization for power, and leaves the topic of dynamic voltage management for future work.

III. Distributed Control Schemes

In this section, we develop three distributed control schemes that do not assume perfect load foresight. The schemes use the same forecasting model but differ in the control algorithms used.

A. Direct Storage Control Scheme

In the Direct Storage Control (DSC) scheme, the GC algorithm determines the charging schedule that minimizes the expected operational cost over set of A forecast scenarios, while ensuring that the network constraints are satisfied for all scenarios. As detailed in Algorithm 1, the GC runs at timesteps $t(k)$ for $k = 1, \ldots, T/\Delta_{\mathrm{GC}}$. For each solution, the forecast scenarios extend from the current hour $t(k)$ to $t(k+1)+\Delta_{\mathrm{F}}$, where Δ_{F} is included to discourage greedy solutions by considering the impact on subsequent iterations. The charging schedule over the window from $t(k)+1$ to $t(k+1)$ is applied directly to the storage (i.e., the LC simply applies the GC output signals directly).

The advantage of the DSC method is that the LC does not need to do any computation. The disadvantage is that the system does not leverage the more recent data that becomes available to the LC throughout the $t(k)+1$ to $t(k+1)$ window.

B. Net Load Following Control Scheme

In the Net Load Following Control (NLFC) scheme, the GC algorithm allows the storage action at each node to vary across scenarios, while fixing the real net load for all storage nodes. Denote the set of nodes with storage as Z. All nodes not in Z have a single scenario. The output of the GC is now the real net load targets $\Re(s)$ instead of the charging rates u. As before, the GC runs at timestep $t(k)$ for each $k = 1, \ldots, T/\Delta_{\mathrm{GC}}$.

The GC algorithm for NLFC is given in Algorithm 2 and its LC algorithm is given in Algorithm 3. Within each iteration of the GC algorithm, the LC runs at each timestep $t \in [t(k)+1, \ldots, t(k+1)]$, leveraging the output of the last GC execution at $t(k)$ as well as updated forecasts. In each execution, the LC generates $g \in [1, \ldots, G]$ new load forecasts beginning at the current time t and extending to $t(k+1)+\Delta_F$.

The advantage of the NLFC scheme is that the LCs are able to leverage data not available to the GC. Furthermore as long as all LCs can match their real net load to the GC target, the power flow constraints of the network are satisfied. The downside of this approach is that the target real net loads

Algorithm 2 GC Algorithm for NLFC Method; p Is the Price of Electricity, s Is Net Load Including Storage, d Is the Net Load Excluding Storage, u Is Storage Charging Rate, and q Is the State of Charge. This Program Runs at Timestep $t(k)$ for Each $k = 1, \ldots, T/\Delta_{\mathrm{GC}}$

Inputs:	$q_{i,t(k)}, \quad i \in [0:N]$
Output:	$s_{i\tau}, \quad \tau \in [t(k)+1 : t(k+1)+\Delta_F]$
Forecast:	$\hat{d}^a_{i,\tau}, \quad a \in [1:A]$
Program	$\underset{u,s,q,W}{\text{minimize}} \sum_\tau p_\tau \cdot \Re(s_{0\tau})$ subject to: $\Re(s_{i\tau}) = \hat{d}^a_{i\tau} + u^a_{i\tau}, \quad \forall i \in Z$ $\Re(s_{i\tau}) = \hat{d}_{i\tau} \quad \forall i \notin Z$ $q^a_{it} = \lambda_i q^a_{i\tau-1} + u^a_{i\tau}\cdot\delta_{\min} \quad \forall i \in Z$ $u_i^{\min} \le u^a_{i\tau} \le u_i^{\max} \quad \forall i \in Z$ $0 \le q^a_{i\tau} \le q_i^{\max}$ $s_{i\tau} = \sum_{j:(i,j)\in E}(w_{ij\tau} - w_{ii\tau})y^*_{ij}$ $(v_i^{\min})^2 \le w_{ii\tau} \le (v_i^{\max})^2$ $W\{i,j\}_\tau \succeq 0$

Algorithm 3 LC Algorithm for NLFC Method; p Is the Price of Electricity, s Is Net Load, d Is Load, u Is Storage Charging Rate, and q Is the State of Charge. Within Each GC Iteration, This Program Runs at Each Timestep t From $t(k)+1$ to $t(k+1)$ at Each LC i

Inputs:	$x_\tau \leftarrow \Re(s_{i\tau})$ (result from last GC program for node i), $\quad \tau \in [t(k)+1 : t(k+1)+\Delta_{\mathrm{F}}]$ q_{t-1} (from local reading)
Output:	u_t
Forecast:	$\hat{d}^g_\tau, \quad g \in [1:G]$
	$\underset{x,q,u}{\text{minimize}} \sum_\tau \sum_g ||y^g_\tau - x_\tau||_2$ subject to: $y^g_\tau = \hat{d}^g_\tau + u_\tau$ $q_\tau = \lambda_i q_{\tau-1} + u_\tau \cdot \delta_{\min}$ $u^{\min} \le u_\tau \le u^{\max}$ $0 \le q_\tau \le q^{\max}$

sent by the GC may not be the only feasible ones, potentially resulting in lower arbitrage profits.

C. Nodal Slack Control Scheme

Intuitively, the NLFC scheme makes sense when all network constraints in the GC optimization are tight, i.e., when small changes to the target real net loads result in voltage violations. In many scenarios, however, the LCs may be able to move significantly from the GC targets without causing any violations, which may provide additional benefits such as increased arbitrage profit. This motivates the Nodal Slack Controller (NSC) scheme, which places target bounds on the real net loads rather than a net load target and introduces arbitrage as an objective in the LC optimization algorithm.

The NSC GC algorithm is described in Algorithm 4. Denote by $x_{j\tau}^{+}$ and $x_{j\tau}^{-}$ the maximum and minimum feasible real net load at node j at time τ, respectively, assuming that the real net loads for all other nodes $i \neq j$ are set by the output of Algorithm 2. Let the real net loads of the root node and node j remain as variables and optimize the injection at node j to obtain upper and lower bounds on the net load at node j such that all voltage constraints in the network are satisfied. Storage dynamics and constraints are not included as the net load bounds are determined for each time period independently. These computations are setup as arbitrage optimizations for convenience. In each arbitrage optimization, the storage dynamics and constraints are dropped. The root node can purchase energy at the market price and node j can buy or sell energy at higher prices than the market (the magnitude of the higher price does not matter here). The solution increases the flow to node j until it reaches a network constraint. The new real net load is $x_{j\tau}^{+}$. Similarly, we find $x_{j\tau}^{-}$ by introducing an arbitrage in the opposite direction.

Note that in Algorithm 4 when we compute the bound for a node, we fix the loads of other nodes to the outputs of Algorithm 2. Although imperfect, this approach is quite reasonable because the "hot nodes" should already be at the minimum or maximum real power. Hence, we are effectively computing the bound for each node assuming that its interdependent nodes are operating at their boundary. Also note that computing these boundaries at each GC iteration can be computationally expensive. In practice, the process can be sped up by computing only the target bounds for nodes that can realistically introduce network violations. These nodes can be identified from previous GC outputs.

The NSC LC algorithm is described in Algorithm 5. At each timestep t, each LC generates $g = 1, \ldots, G$ new forecasts for their own loads. Each LC ensures that all G scenarios generate a feasible flow under the target bounds. The resulting charging decisions for the first timestep are executed, and the process is repeated until the next GC output is received.

Algorithm 4 GC Algorithm for NSC Scheme; p Is the Price of Electricity, s Is the Net Load, d Is Load, u Is Storage Charging Rate, and q Is the State of Charge. This Algorithm Is Run at Timestep $t(k)$ for Each $k = 1, \ldots, T/\Delta_{\mathrm{GC}}$

Inputs:	$q_{it(k)}, \ i \in [0:N]$
Output:	$x_{i\tau}^{-}$ and $x_{i\tau}^{+}, \quad \tau \in [t(k) : t(k+1) + \Delta_F]$
Forecast	$\hat{d}_{i\tau}^{a}, \ a \in [1:A]$
	$\bar{s}_{i\tau} \leftarrow$ Solve NLFC GC in Algorithm 2 For each node j: $\underset{s,W}{\text{minimize}} \sum_\tau (p_\tau \cdot \Re(s_{0\tau}) - 2p_\tau \cdot \Re(s_{j\tau}))$ subject to: $\Re(s_{i\tau}) = \Re(\bar{s}_{i\tau})$ for all $i \neq j$ and $i \neq 0$ $s_{i\tau} = \sum_{j:(i,j)\in E}(w_{ij\tau} - w_{ii\tau})y_{ij}^{*}$ $(v_i^{\min})^2 \leq w_{ii\tau} \leq (v_i^{\max})^2$ $W\{i,j\}_\tau \succeq 0$ $x_{j\tau}^{+} \leftarrow \Re(s_{j\tau})$ For each node j: $\underset{s,W}{\text{minimize}} \sum_\tau (p_\tau \cdot \Re(s_{0\tau}) - \frac{1}{2}p_\tau \cdot \Re(s_{j\tau}))$ subject to: $\Re(s_{i\tau}) = \Re(\bar{s}_{i\tau})$ for all $i \neq j$ and $i \neq 0$ $s_{i\tau} = \sum_{j:(i,j)\in E}(w_{ij\tau} - w_{ii\tau})y_{ij}^{*}$ $(v_i^{\min})^2 \leq w_{ii\tau} \leq (v_i^{\max})^2$ $W\{i,j\}_\tau \succeq 0$ $x_{j\tau}^{-} \leftarrow \Re(s_{j\tau})$

Algorithm 5 LC Algorithm for NSC Scheme. p Is the Price of Electricity, s Is Net Load, d Is Load, u Is Storage Charging Rate, and q Is the State of Charge. Within Each GC Iteration, This Algorithm Runs at Each Timestep t From $t(k)+1$ to $t(k+1)$ at Each LC i

Inputs:	q_t (local reading) For $\quad \tau \in [t(k)+1 : t(k+1) + \Delta_F]$: $x_\tau^{-} \leftarrow x_{i\tau}^{-}$ (from the GC output) $x_\tau^{+} \leftarrow x_{i\tau}^{+}$ (from the GC output)
Output:	u_t
Forecast:	$\hat{d}_\tau^{g}, \ g \in [1:G]$
Program	minimize $\sum_g \sum_\tau p_\tau \cdot x_\tau^{g}$ subject to: $x_\tau^{-} \leq x_\tau^{g} \leq x_\tau^{+}$ $x_\tau^{g} = \hat{d}_\tau^{g} + u_\tau$ $q_\tau = \lambda q_{\tau-1} + u_\tau \cdot \delta_{\min}$ $u^{\min} \leq u_\tau \leq u^{\max}$ $0 \leq q_\tau \leq q^{\max}$

IV. Case Study

In this section, we compare our control schemes and evaluate the effect of communication delay on their performance using the 47-bus radial distribution feeder model in Appendix A1. In Appendix A3, we describe the algorithm we use for RDG and storage placement in the network, and in Appendix B, we describe the load and RDG forecasting models we use to generate the loads. The wholesale prices we use are taken from the 2014 ISONE wholesale market data in [16].

Throughout the case study, we set the optimization parameters $\delta_{\min} = 1$ hour, $A = 25$, $G = 10$, $\Delta_F = 48$ hours. We set the storage parameter $\lambda = 0.9$ and we do not impose constraints on $u_i^{\min}$ or $u_i^{\max}$, that is, the storage is limited

only by its capacity constraints and not by charging or discharging rate limits. The capacity constraints are based on storage penetration as described in Appendix A3. We perform each simulation over an analysis horizon of 30 days ($T = 720$ hours). We create configurations of the radial network with different storage and RDGs penetrations using the placement method in Appendix A3.

Denote by x the network storage penetration, defined as the total capacity of all storage divided by the average daily energy use of the network, and denote by y the network RDG penetration, defined as the average RDG output divided by the average load in the network. The variable z will represent either the PFOSC benchmark or one of the control schemes DSC, LFLC, or NSC.

In planning a real world installation, we may vary x and y for a chosen control scheme z to minimize the aggregate capital and operational costs of the energy grid. In order to perform this analysis, however, we would need to know the capital costs of solar and storage, useable lifetime of devices, discount rates, and compensation rates for solar and storage back feeding (which likely differ from wholesale energy prices). Since our focus here is on comparing the efficacy of the control schemes and how their performance depends on communication delay, we introduce the two metrics which characterize the performance of the storage controllers independent of these factors:

Highest RDG Penetration: This is the highest achievable RDG penetration $\hat{y}(x, z, \Delta_{\mathrm{GC}})$ for a given storage penetration x, control scheme z, and communication delay Δ_{GC} such that Algorithm 6 returns True. The penetration measured by the algorithm is the highest RDG penetration where the network constraints are not violated over 3600 hours of operation.

Normalized Arbitrage Profit: In addition to supporting RDGs by alleviating local network constraints, the storage can also reduce the operational cost of the network by shifting the aggregate load from peak to off peak hours. Denote by $\mathrm{ARB}(x, y, z, \Delta_{\mathrm{GC}})$ the total realized battery operating profit (arbitrage) for a given network with storage penetration x, RDG penetration y, under control method z, with communication delay Δ_{GC}, i.e.,

$$\mathrm{ARB}(x, y, z, \Delta_{\mathrm{GC}}) = \sum_{(i,t)} p_t \cdot u_{it}. \tag{3}$$

The objective of maximizing ARB may conflict with achieving the highest RDG support because storage may need to charge during a peak price hour in order to avoid a constraint violation caused by RDG. Hence, we normalize the realized arbitrage by the arbitrage realized in the system with the same storage and RDG penetration operated using the PFOSC (Δ_{GC} is not relevant in PFOSC), i.e.,

$$\overline{\mathrm{ARB}}(x, y, z, \Delta_{\mathrm{GC}}) = \frac{\mathrm{ARB}(x, y, z, \Delta_{GC})}{\mathrm{ARB}(x, y, \mathrm{PFOSC}, 0)}. \tag{4}$$

We use Algorithm 6 to evaluate the distributed control schemes based on the above metrics.

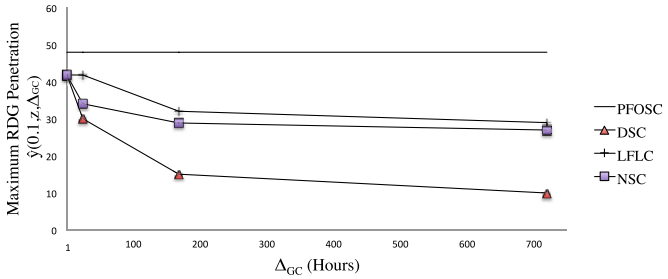


Fig. 3. RDG Support for $x = 10\%$.

Algorithm 6 Simulation of the Distributed Control Scheme Over 5 Iterations of a Finite Time Horizon of Length $T = 720$ Hours. The GC Runs at Multiples of Δ_{GC}, and the LC Runs Every Timestep. The Control Signals From the LCs Are Used to Compute the Overall Cost of Operating the System. This Algorithm Returns True if the Control Scheme Does Not Create Any Powerflow Violations in Any of the 5 Iterations

```
for m = 0, ..., 5 do
  for k = 0, ..., T/Δ_GC do
    Generate future net load forecast from delayed data
    Run the GC algorithm
    Send GC outputs to the LCs
    for τ = 1, ..., (Δ_GC) do
      for i = 1, ..., N do
        Generate future net load forecast for node i
        Run the LC algorithm on forecasted net loads and
        most recent GC output
        Execute LC output for current hour
      end for
    end for
  end for
  Verify that the power flow constraints are satisfied over
  the time horizon
  If any constraints failed, return False
  Else,
  ARB_m= total grid net energy cost
end for
return Average(ARB_m, ..., ARB_5)
```

First we explore the impact of communication delay on the highest RDG penetration for the three control schemes. Figure 3 plots the highest RDG penetration for storage penetration $x = 10\%$ as a function of the delay Δ_{GC} for the three control schemes. We see that the highest RDG penetration for all LC control schemes is the same when the GC updates every timestep, i.e., when $\Delta_{\mathrm{GC}} = 1$. As we increase Δ_{GC}, the highest RDG support degrades for all control schemes. This is expected as load forecasting errors increase, and thus network violations are more likely to occur. The degradation for the LFLC and NSC methods is significantly less than for the DSC since they use the local data in performing the control.

To quantify the value of coordination to RDG penetration, we compute the maximum RDG penetration at the highest amount of coordination corresponding to running the GC every hour ($\Delta_{\mathrm{GC}} = 1$) and compare it to the least

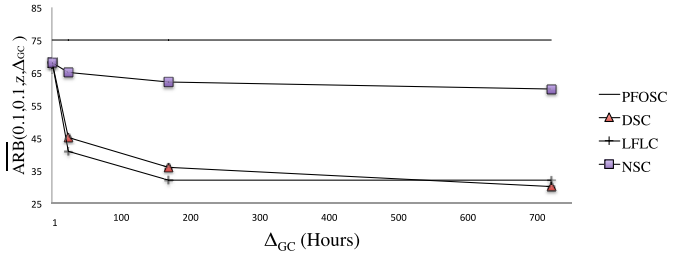


Fig. 4. Arbitrage profits for $x = 10\%$, $y = 10\%$.

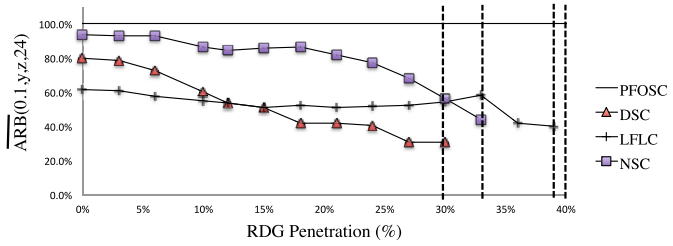


Fig. 5. Arbitrage profits for storage penetration $x = 10\%$, $\Delta_{GC} = 24$. The dashed lines indicate the maximum RDG penetration for each control scheme.

amount of coordination corresponding to running the GC once ($\Delta_{GC} = T = 720$). Utilizing the PFOSC controller, the network can support 48% RDG penetration with 10% storage penetration rate. Voltage violations occur if the penetration is higher. Note from Figure 3 that RDG penetration drops from 42% to 29% for the LFLC, to 27% for the NSC and to 10% for the DSC (with fixed schedule and no local control). Thus, different control schemes can achieve significantly different values for coordination and in particular NSC performs very similarly to LFLC.

Next, we explore the impact of communication delay on arbitrage profits. Figure 4 plots the normalized arbitrage profits for storage penetration $x = 10\%$ and RDG penetration $y = 10\%$. We see that the degradation in arbitrage profit with delay is much lower for the NSC scheme than the other two control schemes because the local objective of the NSC controller directly accounts for arbitrage profits. To quantify the value of coordination to arbitrage profit, note that from Figure 4 the maximum normalized arbitrage profit at $\Delta_{GC} = 1$ is 90.7%, while at $\Delta_{GC} = T = 720$ it drops to around 40% for DSC and LFLC but only to 80% for the NSC. In terms of the value of coordination to arbitrage profits, NSC significantly outperforms the DSC and LFLC.

To explore the impact of RDG penetration on arbitrage profits, in Figure 5 we plot the arbitrage profits versus the RDG penetration for the three control schemes for $\Delta_{GC} = 24$ and storage penetration $x = 10\%$. Each plot ends at the maximum RDG penetration achievable by the corresponding control scheme. We see that the arbitrage profit degrades as RDG penetration increases, and that the rate of degradation increases as we approach the maximum RDG penetration. We also see that the NSC scheme performs best in limiting the degradation for a given RDG penetration except at the highest penetration levels. The LFLC scheme is able to support higher RDG penetration levels for the chosen Δ_{GC}. For higher values of delay, the NSC scheme significantly outperforms LFLC in capturing arbitrate profits as noted earlier.

V. Conclusion

The optimal control of storage simultaneously considering realistic cyber and physical constraints has not yet been addressed in the literature. The main issue is that in practice load information is only available at the global controller with some time delay due to communications limitations in the metering infrastructure. Even in the absence of such limitations, it is unrealistic to expect real-time availability of local information in large distribution networks. In this paper, we demonstrate that control schemes that utilize local information can perform significantly better than those that do not. The benchmark we utilize is the DSC that prescribes the optimal control strategy from delayed information without accounting for the possibility of local updates. The DSC performs significantly worse than two strategies that utilize local information: the nodal slack controller (NSC) and the load following local controller (LFLC). LFLC assumes that following the strategy based on forecasted loads is optimal whereas NSC shifts more responsibility for control decisions to the local control scheme. In practice, NSC significantly outperforms LFLC when the information delay increases. LFLC supports up to 40% RDG penetration while NSC supports only up to 29% but captures a significantly higher percentage of arbitrage profits as compared to LFLC, especially when RDG penetrations are high. It would be straightforward to combine both the NSC and LFLC controllers into a controller that achieves a tradeoff between maximum RDG penetration and arbitrage profits not achievable by either controller alone.

In realistic scenarios there is a tradeoff between maximum RDG penetration and the achievable arbitrage profits for each proposed control scheme. An appropriate choice of the control scheme makes the system robust to delay. For example, for delay larger than delay is greater than 1 hour, the degradation in renewable RDG penetration is small if either LFLC or NSC is utilized. Furthermore, higher arbitrage profits require shifting more control flexibility to the local controller as the communication delay increases. The NSC scheme therefore achieves a better overall tradeoff as compared to LFLC and DSC.

One limitation of this work is that we assume all volt/var control equipment such as tap changing transformers, switchable shunt capacitors, and voltage regulators, are set to fixed operating points. These control devices are difficult to incorporate into our optimization methods because they introduce integer and combinatorial constraints. In a real system, these devices may be able to correct volt/var violations, increasing the amount of RDG a given amount of storage can support. In future work, we will consider both the role of various network control devices as well as imperfect knowledge of the networks. In such scenarios it will be important to compute the tradeoff between the probability of violating the network constraints and the arbitrage profits and maximum RDG penetration of the proposed schemes.

This paper focuses on optimizing storage operations for given solar and storage penetrations. By incorporating this optimization into a higher level planning tool, one could determine the most cost-effective penetration levels by comparing

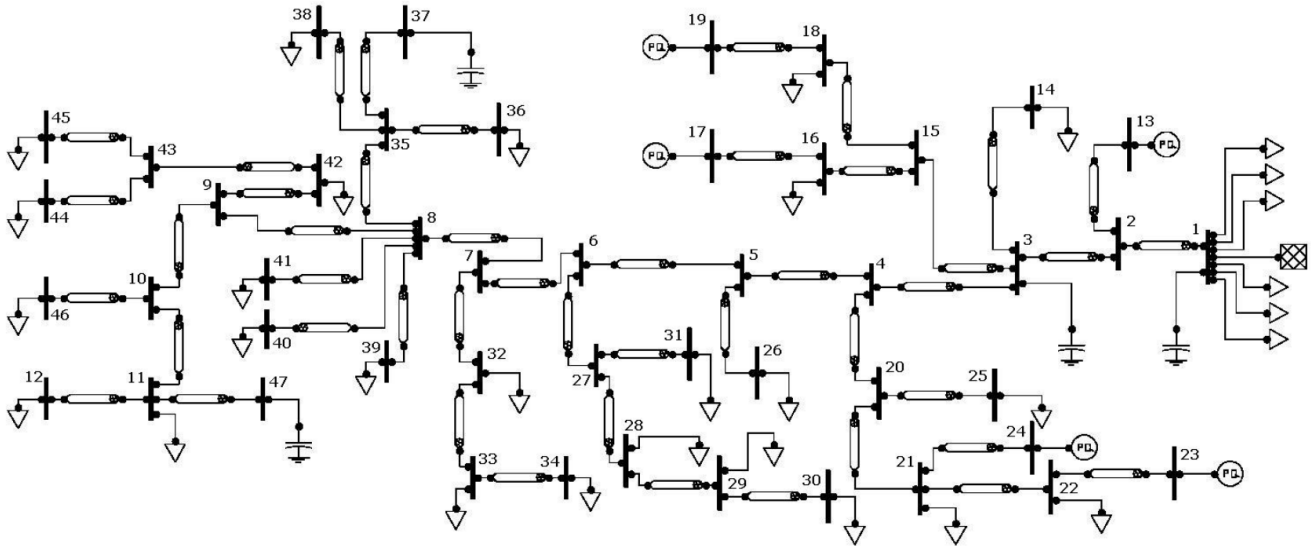


Fig. 6. Network model adapted from [17].

Network Data															
Line Data				Line Data				Line Data				Load Data		Load Data	
From Bus.	To Bus.	R (Ω)	X (Ω)	From Bus.	To Bus.	R (Ω)	X (Ω)	From Bus.	To Bus.	R (Ω)	X (Ω)	Bus No.	Peak MVA	Bus No.	Peak MVA
1	2	0.259	0.808	8	41	0.107	0.031	21	22	0.198	0.046	1	30	34	0.2
2	13	0	0	8	35	0.076	0.015	22	23	0	0	11	0.67	36	0.27
2	3	0.031	0.092	8	9	0.031	0.031	27	31	0.046	0.015	12	0.45	38	0.45
3	4	0.046	0.092	9	10	0.015	0.015	27	28	0.107	0.031	14	0.89	39	1.34
3	14	0.092	0.031	9	42	0.153	0.046	28	29	0.107	0.031	16	0.07	40	0.13
3	15	0.214	0.046	10	11	0.107	0.076	29	30	0.061	0.015	18	0.67	41	0.67
4	20	0.336	0.061	10	46	0.229	0.122	32	33	0.046	0.015	21	0.45	42	0.13
4	5	0.107	0.183	11	47	0.031	0.015	33	34	0.031	0	22	2.23	44	0.45
5	26	0.061	0.015	11	12	0.076	0.046	35	36	0.076	0.015	25	0.45	45	0.2
5	6	0.015	0.031	15	18	0.046	0.015	35	37	0.076	0.046	26	0.2	46	0.45
6	27	0.168	0.061	15	16	0.107	0.015	35	38	0.107	0.015	28	0.13		
6	7	0.031	0.046	16	17	0	0	42	43	0.061	0.015	29	0.13	Base Voltage (KV) = 12.35	
7	32	0.076	0.015	18	19	0	0	43	44	0.061	0.015	30	0.2	Base KVA = 1000	
7	8	0.015	0.015	20	21	0.122	0.092	43	45	0.061	0.015	31	0.07	Substation Voltage = 12.35	
8	40	0.046	0.015	20	25	0.214	0.046					32	0.13		
8	39	0.244	0.046	21	24	0	0					33	0.27		

Fig. 7. Network line impedances and rated net load, adapted from [17].

the upfront fixed cost of installing different levels of storage and solar to the overall savings under the control methods described in this paper.

APPENDIX

A. Network Configuration

1) Network Topology: The radial distribution model we use in the case study is depicted in Figure 6. It is based on the network in [17]. The resistance and reactance values for each line in the circuit are given in Figure 7. These values define the admittance values used in our controllers throughout the paper ($y_{ij} = 1/(R_{ij} + iX_{ij})$). The case study defines voltage control devices in the network. We select a static configuration for voltage regulators in this network. Volt/var control systems typically have discrete settings that change at most a couple of times per day. Hence, they do not need to be interdependent on the hourly storage control methods in this paper. They can be optimized in a separate system and their impact can be reflected in our control by changing the network parameters across hours to match the optimal settings of the control equipment. The full model specification also requires defining the load and solar PV time series at each bus in the network and is addressed in Section V-B.

2) Load Placement: To define a load time series for each bus, we utilize smart meter data for 55 residential customers collected over one year in a pilot program located in the Central Valley region of California. The original data comprises 15 minute measurements of power consumption, temperature, humidity, and the context of the reading (day of week, time of day, holidays). We down-sample the data to one hour intervals by averaging the data points within each hour.

Each bus in the network model used in the case study corresponds to aggregates of customers behind secondary transformers. We have made this assumption because of the model and data available to us. However, our method can be applied to a full network model in which buses correspond to customers at the network edge. This simplification

Algorithm 7 Place Storage and Solar

```
Network ← Base Network and Load (described in Appendix V-A1)
InstallationCount ←
      Round(SolarPenetration * Count(Network.Nodes()))
TargetNodes ←
      Network.getRandomNodes(InstallationCount)
TargetNodesTotalLoad =
      sum(node.AverageLoad() for node in TargetNodes)
RawSolar = SolarPenetration*Network.AverageLoad()
RawStorage =
     StoragePenetration*Network.AverageDailyEnergyUse()

for Node in TargetNodes do
  α ← Node.AverageLoad() / TargetNodesTotalLoad
  Node.addRDG( α*RawSolar )
  Node.addStorage( α*RawStorage)
end for
```

is reasonable and often used in practice, but requires that the reduced model be carefully constructed in order to sufficiently represent the real network.

We construct these aggregates by choosing customers uniformly at random with replacement and assigning them to buses. We continue adding customers to a given bus until the average daily peak load for that bus matches the peak loads given in Figure 7.

3) RDG and Storage Placement: The RDG dataset we use is obtained from the solar PV output data reported in a 2006 NREL study, which provides estimated time series of solar PV production based on solar irradiance and temperature.

RDG and storage are placed using Algorithm 7. A fraction of nodes y is selected within the network at random. Each selected bus receives solar and storage in a proportion weighted by the rated load of the bus, such that the total storage and solar on the network matches our target allocation. The number of nodes that is going to receive storage is InstallationCount and depends on the resource penetration rate and the number of nodes in the network. The set of nodes that receives storage is TargetNodes and is selected at random from all the nodes in the network.

B. Net Load Forecasting

The case study requires multiple scenarios for the behavior and forecasting of loads in order to compute a robust estimate of performance. We utilize the ARIMA model in [18] with the form $(3,0,3)(3,0,3)_{24}$ to fit the load data generated in the previous section and serve as a basis for scenario generation. We choose a seasonal differencing of 24 to represent the daily periodic trends that exist in the load data. The resulting ARIMA model is

$$\left(1-\phi_1 B-\phi_2 B^2-\phi_3 B^3\right)\left(1-\phi_{24}B^{24}-\phi_{48}B^{48}-\phi_{72}B^{72}\right)\left(1-B^{24}\right)x_t=\left(1+\theta_1 B+\theta_2 B^2+\theta_3 B^3\right)\left(1+\theta_{24}B^{24}+\theta_{48}B^{48}+\theta_{72}B^{72}\right)\epsilon_t,$$
$$\epsilon_t \sim \mathcal{N}(0,1), \tag{5}$$

with the variables defined in Table II.

We use the ϕ and θ values that maximize the likelihood of the load data. After the model is fitted, a single forecast estimate for a one hundred hour long interval is determined and assigned as the reference hourly load. We generate $A = 50000$ other sample scenarios by drawing random values for ϵ_t and sequentially applying them to the model in (5). The mean average percentage error over the forecasts is defined as

$$\text{MAPE}(t) = \frac{1}{A}\sum_{a=1}^{A}\frac{|\bar{x}_t - \hat{x}_t^a|}{\bar{x}_t}, \tag{6}$$

where $\bar{x}_t$ is the reference hourly load and $\hat{x}_t^a$ is a prediction in scenario a. Figure 8 plots the forecast error as a function of time. As expected, the forecast error increases as we forecast further into the future since less data available. The takeaway here is that the GC has a less accurate forecast of future load than the LCs, due to the delays in the network and the limitations on how frequently the GC runs.

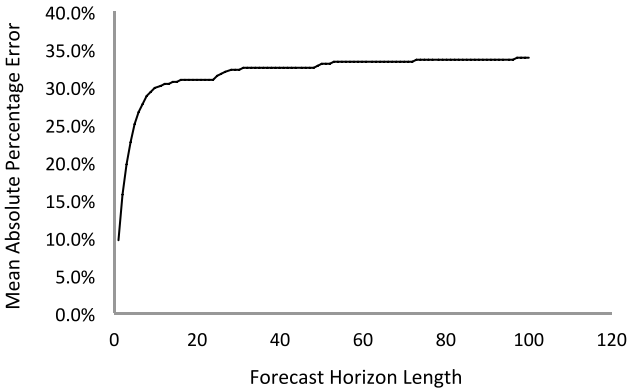


Fig. 8. MAPE of forecast over horizon.

TABLE II
CONSTANTS AND VARIABLES USED IN THE ARIMAX MODEL

Data	
x_t	load or RDG at time t
Model Coefficients	
ϕ	autoregressive coefficients
θ	moving average coefficients
Other	
ϵ_t	Gaussian error term
B	Lag operator. (i.e., $B^1 y_t = y_{t-1}$)

ACKNOWLEDGMENT

The authors would like to thank Robert Entriken and the anonymous reviewers for providing insightful comments that have helped improve the exposition of this paper, and

Thomas Navidi for proof reading the paper and providing several corrections.

REFERENCES

[1] "Annual energy outlook 2014 with projections to 2040," Energy Inf. Admin., U.S. Dept. Energy, Washington, DC, USA, Tech. Rep. DOE/EIA-0383(2014), 2014. [Online]. Available: http://www.eia.gov/forecasts/aeo/pdf/0383(2014).pdf

[2] H. Trabish. (Jan. 2015). *Hawaiian Electric's Plan to End Solar Net Metering, Explained*. [Online]. Available: http://www.utilitydive.com/news/hawaiian-electrics-plan-to-end-solar-net-metering-explained/356432

[3] "Smart meters and smart meter systems: A metering industry perspective a joint project of the EEI and AEIC meter committees," EEI-AEIC-UTC, Washington, DC, USA, Tech. Rep., Mar. 2011.

[4] D. Gayme and U. Topcu, "Optimal power flow with large-scale storage integration," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 709–717, May 2013.

[5] M. D. Galus, R. La Fauci, and G. Andersson, "Investigating PHEV wind balancing capabilities using heuristics and model predictive control," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Providence, RI, USA, 2010, pp. 1–8.

[6] M. D. Ilić, L. Xie, and J.-Y. Joo, "Efficient coordination of wind power and price-responsive demand—Part I: Theoretical foundations," *IEEE Trans. Power Syst.*, vol. 26, no. 4, pp. 1875–1884, Nov. 2011.

[7] M. Kraning, E. Chu, J. Lavaei, and S. P. Boyd, "Dynamic network energy management via proximal message passing," *Found. Trends Optim.*, vol. 1, no. 2, pp. 73–126, 2014.

[8] L. Xie, Y. Gu, A. Eskandari, and M. Ehsani, "Fast MPC-based coordination of wind power and battery energy storage systems," *J. Energy Eng.*, vol. 138, no. 2, pp. 43–53, 2012.

[9] S. Lakshminarayana, T. Q. S. Quek, and H. V. Poor, "Cooperation and storage tradeoffs in power grids with renewable energy resources," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 7, pp. 1386–1397, Jul. 2014.

[10] J. Qin, Y. Chow, J. Yang, and R. Rajagopal, "Online modified greedy algorithm for storage control under uncertainty," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 1729–1743, May 2015.

[11] J. Qin, Y. Chow, J. Yang, and R. Rajagopal, "Distributed online modified greedy algorithm for networked storage operation under uncertainty," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 1106–1118, Mar. 2015.

[12] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Robustness in Identification and Control*. London, U.K.: Springer, 1999, pp. 207–226.

[13] J. Mattingley, Y. Wang, and S. Boyd, "Receding horizon control," *IEEE Control Syst.*, vol. 31, no. 3, pp. 52–65, Jun. 2011.

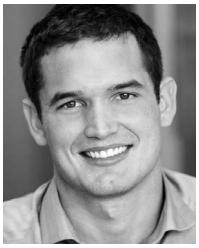
[14] K. Anderson and A. El Gamal, "Co-optimizing the value of storage in energy and regulation service markets," *Energy Syst.*, vol. 8, no. 2, pp. 369–387, 2017. [Online]. Available: http://dx.doi.org/10.1007/s12667-016-0201-0

[15] L. Gan, N. Li, U. Topcu, and S. H. Low, "Exact convex relaxation of optimal power flow in radial networks," *IEEE Trans. Autom. Control*, vol. 60, no. 1, pp. 72–87, Jan. 2015.

[16] (Sep. 2014). *Isone Market Operations*. [Online]. Available: http://www.iso-ne.com/markets-operations

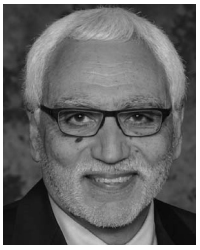
[17] L. Gan, N. Li, U. Topcu, and S. Low, "Branch flow model for radial networks: Convex relaxation," in *Proc. 51st IEEE Conf. Decis. Control (CDC)*, Piscataway, NJ, USA, 2012, pp. 1–8.

[18] J. Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, "Arima models to predict next-day electricity prices," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1014–1020, Aug. 2003.



Kyle Anderson received the B.S., M.S., and Ph.D. degrees in electrical engineering from Stanford University, where he graduated at the top of his class as the Henry Ford Scholar. He is a Founder of Rockset, a new company building distributed data systems. He was previously an early Engineer with Arista Networks (NYSE: ANET) and has worked with Google, Ericsson, and Greylock Partners. He was a recipient of Mayfield Fellow, Stanford Graduate Fellow, Frederick Emmons Terman Award Winner, and the 2010 Agilent Award Winner. He is a member of Tau Beta Pi.

Ram Rajagopal received the M.A. degree in statistics and the Ph.D. degree in electrical engineering and computer sciences from the University of California Berkeley. He is an Assistant Professor of civil and environmental engineering and electrical engineering (by Courtesy) with Stanford University, where he directs the Stanford Sustainable Systems Laboratory, focused on large scale monitoring, data analytics, and stochastic control for infrastructure networks, in particular power and transportation. He holds over 30 patents from his work. His current research interests in power systems are in integration of renewables, smart distribution systems, and demand-side data analytics. He has also built wireless sensing systems and large-scale analytics for transportation and power networks. He was a recipient of the Powell Foundation Fellowship, Berkeley Regents Fellowship, the NSF CAREER Award, the Makhoul Conjecture Challenge Award, and best paper awards in various conferences. He has advised or founded various companies in the fields of sensor networks, power systems, and data analytics.



Abbas El Gamal received the B.Sc. (Hons.) degree from Cairo University in 1972, and the M.S. degree in statistics and the Ph.D. degree in electrical engineering from Stanford University in 1977 and 1978, respectively. He is the Fortinet Founders Chair and a Professor with the Department of Electrical Engineering and the Hitachi America Professor with the School of Engineering, Stanford University. From 1978 to 1980, he was an Assistant Professor of electrical engineering with the University of Southern California. From 2003 to 2012, he was the Director of the Information Systems Laboratory, Stanford University. His research contributions have been in network information theory, field-programmable gate arrays, digital imaging devices and systems, and smart grid modeling and control. He has authored or co-authored over 230 papers and holds 35 patents in the above areas. He has co-authored the book entitled *Network Information Theory* (Cambridge Press, 2011). He was a recipient of several honors and awards for his research contributions, including the 2016 IEEE Richard Hamming Medal, the 2012 Claude E. Shannon Award, and the 2004 INFOCOM Paper Award. He is a member of the U.S. National Academy of Engineering. He served on the Board of Governors of the Information Theory Society from 2009 to 2016 and was President in 2014.