Distributed Multi-Period Optimal Power Flow for Demand Response in Microgrids

Paul Scott[1] and Sylvie Thiébaux[1]

The Australian National University,
NICTA,
{first.lastname}@anu.edu.au

Abstract. With the arrival of electric vehicles, battery storage and smart appliances, households now have the opportunity to actively participate in balancing supply and demand in electricity networks. We propose to coordinate this multi-agent system using distributed optimisation, in order to scale to large systems whilst preserving agent privacy. However, the practical applicability of distributed optimisation remains an open question in this context, as AC power flows are inherently nonconvex and households often make discrete decisions about how to schedule their loads. In this paper we show that one such method, the alternating direction method of multipliers (ADMM), can be adapted to remain practical in this challenging microgrid setting. We formulate and solve a multi-period optimal power flow (OPF) problem featuring household agents with shiftable loads, and study the results obtained with a range of power flow models and approaches to managing discrete decisions. Our experiments on a suburb-sized microgrid show that the AC power flows and a simple two-stage approach to handling discrete decisions do not appear to cause convergence issues, and provide near optimal results in a time that is practical for receding horizon control. This brings distributed control of microgrids several steps closer to reality.

1 Introduction

The role of electricity market operators is to supply low-cost and reliable electricity to customers, which typically involves centrally solving unit commitment (UC) and/or optimal power flow (OPF) problems at regular intervals. This is only possible because consumers are assumed to be inflexible, and are therefore ignored. This assumption will no longer hold true when smart appliances, distributed generation, storage, and electric vehicles (EVs) are adopted on mass. The traditional centralised approaches were not designed to operate where every consumer is an active agent, or to handle their unique time-coupled behaviours.

Microgrids are another technology that challenges the traditional approach to operating electricity networks. They require more intelligent control to ensure that demand and supply are balanced in each instance, but they also offer the opportunity for more efficient network configurations and operations compared to our existing passive high-capacity distribution networks.

In this new regime, demand response (DR) techniques will be invaluable for coordinating thousands of customers and providing network support. The line between who is a generator and who is a load is blurring as households have the opportunity to feed power back to the grid from rooftop solar and batteries. DR can be naturally extended to not just oversee the operation of loads, but also distributed generators and other network support devices in a comprehensive way. This can be modelled as a constrained optimisation problem, with the goal of serving power at the lowest cost on average over the long term, whilst preserving safe network operating conditions.

In order to solve this problem for many agents, several works [19,13,21] have adopted distributed solving techniques. These algorithms greatly parallelise the problem and help to preserve the privacy of agents. As a byproduct, they provide a natural market mechanism for fairly allocating payments between consumers and producers. Theoretically these algorithms require the problem to be convex in order to guarantee convergence to a globally optimal solution. However, the behaviour of many loads within households are discrete in nature [24], and the equations that govern how power physically flows on the network are non-convex.

In this paper, we show that in the context of a microgrid, these theoretical problems can be dealt with in practice. We show that for a distributed DR algorithm in a microgrid, exact non-convex power flow models perform well compared to inexact convex models. Secondly, we identify that the non-convex nature of the most common form of discrete household loads is a non-issue, and that in practice simple approaches to handling these discrete loads are effective at the microgrid level. By solving these problems, we show that the use of distributed algorithms for managing the balance of power on a microgrid is in practice not only possible, but also highly effective.

We formulate the problem as a multi-period optimal power flow (OPF) problem to account for multiple time steps over a day, which can be used as part of a day-ahead pricing scheme or, as we propose, a receding horizon control algorithm. We solve the multi-period OPF problems in a distributed manner by adapting the alternating direction method of multipliers (ADMM) approach presented in [19]. This enables agents to negotiate their power consumptions in a cooperative manner. We experiment with a range of power flow models of varying degrees of accuracy, to compare their relative behaviour in a distributed algorithm. We then introduce and compare several approaches layered on top of ADMM which manage the introduction of discrete variables into the problem. Technically, our contributions can be summarised as:

- A comprehensive experimental comparison of the convergence of five commonly used power flow models when used for distributed DR in a microgrid.
- The identification that the exact non-convex power flow model in practice not only converges in this context, but also finds near-optimal solutions in a timely fashion relative to other models.
- The introduction and comparison of three simple but effective approaches to managing the discrete shiftable loads that are found within households.

Combined, these results show that distributed DR using ADMM can achieve near optimal solutions in a time frame that is practical for receding horizon control in this challenging microgrid setting, regardless of the theoretical limitations. This brings distributed DR closer to the point where it can be deployed in a real microgrid.

In the next section we discuss the related work and how our contribution is unique. In Sections 3–4 we formulate the problem and present the distributed algorithm we use to solve it. The test microgrid is introduced in Section 6 before presenting our results in Sections 7–8 on power flows and discrete decisions.

2 Related Work

Much of the existing work on demand response (DR) has focused on using real-time pricing (RTP) as a control signal [24,22,26]. In these methods, participants receive a RTP signal and individually optimise their own behaviour, so as to minimise a combination of monetary and discomfort costs. Other approaches have utilised simple non-pricing control signals [28] and cooperative games to improve the predictability and purchasing power of collectives [30,29].

These approaches implement a form of open loop control, i.e. the process that sets the control signal (RTP or otherwise) at best can only estimate how agents will respond to it. A closed loop approach to RTP was presented by Gatsis et al. [13] in order to improve the quality of solutions. In this scheme, RTP is iteratively updated by a central agent, with consumer agents communicating their best responses to the price before actually acting. Mohsenian-Rad et al. [21] introduce an alternative iterative procedure not based on RTP, where consumer agents cooperate to reduce total generation costs in a distributed manner.

The approaches discussed so far do not model the electricity network, so we cannot be sure that the DR outcome is efficient, safe or even possible. Many of the works on distributed algorithms which explicitly model the network have used ADMM as a solving technique, due to its decomposition ability and its convergence guarantees on a wide range of problems [5]. However, most of these works have focused on more traditional OPF problems rather than demand response in a microgrid context.

One of the first authors to apply ADMM to power networks was Kim et al. [18], who decomposed a convex approximation of the OPF problem into regions, and compared the results to two other approaches. They found it to have a significant speed improvement over a centralised approach, and that it preserved privacy between regions. Erseghe [10] also performed region-based decomposition of the network and found exact local solutions to the OPF problem. The recent work by Magnússon et al. [20] decomposes the network to a greater degree and uses sequential convex approximations to solve the problem. One thing all these works have in common is that they are focused on the more traditional OPF problem, whereas we consider a microgrid where independent agents are distributed throughout the network.

Region-based decomposition was also used by Dall'Anese et al. [7] to control distributed generation on radial feeders using a semidefinite programming (SDP) relaxation. In our work we consider each customer to be independent, for privacy reasons, and we also allow for meshed microgrid topologies. Šulc et al. [31] use the relaxed DF (SOCP) equations to perform reactive power control on radial networks. For a similar problem, Peng et al. [23] provide closed-form solutions for ADMM subproblems, greatly reducing the computational requirements. Again these works focus on radial networks.

The work that is closest to ours is that presented by Kraning et al. [19], and indeed we build on their approach. They decompose all network components for a multi-period OPF problem using a quadratic power flow approximation. Their experiments showed that very large problems can be solved efficiently in a parallel environment.

All these works have taken different approaches to modelling the power flows on the network. There is no comparison of the performance of the distributed algorithm for the different power flow models, which is what we achieve in this paper for five different models. Our results in this area indicate that exact local methods can produce close to optimal solutions in a competitive number of iterations relative to other models. In addition, to the best of our knowledge, we are the first to incorporate discrete decisions into a distributed demand response mechanism that models the network. Our work brings ADMM to the point where it can be considered a practical approach for efficiently balancing power in a microgrid setting.

3 Problem Formulation

The objective of the demand response problem is to minimise the cost of supplying electricity. We formulate this as a multi-period OPF problem over a finite horizon of $n \in \mathbb{N}$ time steps. This can be embedded within a receding horizon control process in order to manage uncertainty, but for the purposes of this paper we just focus on a single horizon.

We adopt a more general, and what we believe is a more intuitive, formulation from that presented in [19]. In our model, a network N consists of a set of components C, terminals T and connections L. Each component $c \in C$ (e.g., bus, line, generator, load) has a set of terminals $T_c \subseteq T$ which can be connected to the terminals of other components, where the T_c sets partition T. Each connection $l \in L$ is a pair of terminals, i.e. $L \subseteq T \times T$. We assume there is a different agent that looks after each component, but it would also be possible for agents to oversee the operation of multiple components.

3.1 Connections

Connections exist between the terminals of two different components. We use the quantities of real power, reactive power, voltage and voltage phase angle ($p, q, v, \theta \in \mathbb{R}^n$ respectively) to model the flow of power into a component through a terminal. These are vectors in order to capture each time step in the horizon. For convenience, we use a parent vector $y_i \in \mathbb{R}^{4n}$ to represent all variables for a terminal $i \in T$, where $y_i := (p_i, q_i, v_i, \theta_i)^\intercal$. When two terminals are connected together, $(i,j) \in L$, we pose the following constraints:

$$p_i + p_j = 0, \quad q_i + q_j = 0, \quad v_i - v_j = 0, \quad \theta_i - \theta_j = 0$$

The first two constraints ensure that for a connected pair of terminals, at each time step, any power that leaves one terminal must enter the other. The second two constraints ensure that the connected terminals have the same voltage and phase angle. This duplication of variables is necessary in order to decompose the problem for our distributed algorithm.

We rewrite these constraints as $y_i + Ay_j = 0$ for y, where A is the appropriate $4n \times 4n$ diagonal matrix. Further, we define the function $h : \mathbb{R}^{4n} \times \mathbb{R}^{4n} \mapsto \mathbb{R}^{4n}$ as the LHS of this constraint for convenience: $h(y_i, y_j) := y_i + Ay_j$.

3.2 Components

At a high level, each component $c \in C$ has a variable vector $x_c \in \mathbb{R}^{a_c}$, an objective function $f_c : \mathbb{R}^{a_c} \mapsto \mathbb{R}$, and a constraint function $g_c : \mathbb{R}^{a_c} \mapsto \mathbb{R}^{b_c}$, where $g_c(x_c) \leq 0$. For a component $c \in C$, the vector x_c includes all terminal variables for that component: $y_i, \forall i \in T_c$.

In the following sections we describe at a lower level the models used for the components in our experiments. When necessary, we use $t \in \{0, \dots, n\}$ to index vectors by time, otherwise we imply standard vector operations. The index where $t = 0$ is used to represent the value of the variable at the beginning of the current horizon, which we assume is known.

Bus A bus has a variable number of terminals which depends on how many other components connect to it. For example, a bus might be connected to a generator, a load and 3 lines for a total of 5 terminals. Regardless of the number of terminals, the constraints take the form:

$$\sum_{i \in T_c} p_i = 0 \quad \sum_{i \in T_c} q_i = 0$$
$$\forall i, j \in T_c : v_i = v_j, \ \theta_i = \theta_j$$

The first two constraints are an expression of Kirchhoff's current law (KCL) in terms of power flows. The remaining constraints ensure that all terminal voltages and phase angles are the same.

Line A line is a two terminal component which transports power from one bus to another. We model a line as having a constant conductance $g \in \mathbb{R}_+$, susceptance $b \in \mathbb{R}$ and maximum apparent power $s \in \mathbb{R}_+$. The AC power flow equations are derived from Ohm's law, where $\forall i, j \in T_c, i \neq j$:

$$p_i = gv_i^2 - gv_iv_j \cos(\theta_i - \theta_j) - bv_iv_j \sin(\theta_i - \theta_j) \tag{1}$$

$$q_i = -bv_i^2 + bv_iv_j\cos(\theta_i - \theta_j) - gv_iv_j\sin(\theta_i - \theta_j) \quad (2)$$

$$s^2 \geq p_i^2 + q_i^2, \quad \underline{v} \leq v_i \leq \bar{v}, \quad \theta_i - \theta_j \leq \bar{\theta} \quad (3)$$

These constraints are identical for each time step, so we have left out the indexing by time to improve clarity. These equations are non-convex, so they are often either approximated or relaxed, as we will discuss further in Section 7.

Generator A generator is a single terminal component which produces real and reactive power. In our formulation the generator has a floating phase angle and voltage. A generator has lower and upper real and reactive power limits such that $p_{i,t} \in [\underline{p}, \bar{p}]$ and $q_{i,t} \in [\underline{q}, \bar{q}]$, and a quadratic cost function f for generation costs with price parameters including the diagonal matrix $\varPsi \in \mathbb{R}_+^{n\times n}$ and vector $\psi \in \mathbb{R}_+^n$: $f(x) = p_i^\intercal \varPsi p_i - \psi^\intercal p_i$

Shiftable Load A shiftable load is a single terminal component used to model electrical loads like dish washers and clothes dryers. A household has some flexibility on when these loads can run, and will schedule them to minimise the costs they pay for the electricity. These loads must start running between an earliest and a latest start time: $t^e, t^l \in \mathbb{N}$. To model this we introduce binary variables $u \in \{0,1\}^n$ for the horizon. A value of 1 indicates that the component starts at the given time. A component runs for a duration of $d \in \mathbb{N}$ consecutive time steps, during which it consumes a load of $p^{\text{nom}} \in \mathbb{R}$. A convex relaxation of this component can be obtained by relaxing the integrality requirement: $u \in [0,1]^n$.

$$p_{i,t} = p^{\text{nom}} \sum_{t'=t-d+1}^{t} u_{t'} \quad \sum_{t=t^e}^{t^l} u_t = 1$$

$$\forall t \notin \{t^e, \ldots, t^l\} : u_t = 0$$

Other Components A range of other components can be modelled within this framework, for example, batteries, inverters, solar PV, electric vehicles, HVACs and voltage regulators (see [26] for additional models). Indeed we have experimented with batteries and solar PV using our distributed algorithm, but in this paper we focus on the more difficult to handle shiftable loads.

3.3 Optimisation Problem

Now that we have the component models and the relations between them, we can write down the multi-period OPF problem for one horizon. The objective is to minimise the sum of all component cost functions, subject to component and terminal connection constraints.

$$\min_x \sum_{c\in C} f_c(x_c) \quad (4)$$

$$\text{s.t. } \forall c \in C : g_c(x_c) \leq 0 \quad (5)$$

$$\forall (i,j) \in L : h(y_i, y_j) = 0 \quad (6)$$

4 Distributed Algorithm

The next step is to show how we can solve this problem in a distributed manner using the alternating direction method of multipliers (ADMM). ADMM is a variation of the standard augmented Lagrangian method that enables problem decomposition [5,9,12]. The augmented Lagrangian relaxation applied to the connection constraints (6) is:

$$\begin{aligned}\mathcal{L}(y, z, \lambda, \rho) := &\sum_{c \in C} f_c(x_c) \\ &+ \sum_{(i,j) \in L} \left[\frac{\rho}{2} \|h(y_i, z_j)\|_2^2 + \lambda_{i,j}^{\mathsf{T}} h(y_i, z_j) \right]\end{aligned}$$

where $\rho \in (0, \infty)$ is a penalty parameter and $\lambda_{i,j} \in \mathbb{R}^{4n}$ are the dual variables for the connection constraints. These dual variables represent the locational marginal prices in our problem, or put another way, connection dependent RTPs. They are important because they indicate what each component should be paying for the power that they receive through their terminals. These prices are based on not just the cost of generation, but also account for line losses and adjust to prevent congestion. They provide a natural mechanism for the fair distribution of payments from consumers to producers.

4.1 Algorithm

A single iteration of the ADMM algorithm consists of two optimisation phases followed by a dual update. Components are each allocated to one of the two phases. We allocate all buses to the second phase and all other components to the first phase in order to fully decompose the problem. The component sets C_1 and C_2, and the variable vectors x_1 and x_2 represent this allocation.

The superscript $k \in \mathbb{N}$ is used to indicate the k-th iteration. At the start of the algorithm all terminal and dual variables are initialised to some values $y_i^{(0)}$ and $\lambda_{i,j}^{(0)}$. For the k-th iteration ADMM proceeds as follows:

1. Optimise $\mathcal{L}$ over x_1, holding x_2 constant at its $k-1$ value
2. Optimise $\mathcal{L}$ over x_2, holding x_1 constant at its k value
3. Update the dual variables λ

For our optimisation problem this becomes:

$$x_1^{(k)} = \arg\min_{x_1} \; \mathcal{L}(y, y^{(k-1)}, \lambda^{(k-1)}, \rho^k) \tag{7}$$

$$\text{s.t. } \forall c \in C_1 : g_c(x_c) \leq 0$$

$$x_2^{(k)} = \arg\min_{x_2} \; \mathcal{L}(y^{(k)}, y, \lambda^{(k-1)}, \rho^k) \tag{8}$$

$$\text{s.t. } \forall c \in C_2 : g_c(x_c) \leq 0$$

$$\forall (i,j) \in L_{1,2} : \lambda_{i,j}^{(k)} = \lambda_{i,j}^{(k-1)} + \rho^{(k)} h(y_i^{(k)}, y_j^{(k)}) \tag{9}$$

In the simple case when ρ is constant, f_c and g_c are convex, and h is affine, ADMM converges to a global optimum [5].

Within each phase, each agent can solve the sub-problem for its component independently of the components of other agents. This is because we have placed the buses and non-bus components into separate phases. As an additional benefit, some sub-problems are simple enough when separated that they have closed-form solutions [23]. We adopt a closed-form solution for buses as proposed in [19].

5 Implementation

We developed an implementation of the above approach in C++ using Gurobi [14] for mixed-integer quadratic sub-problems, and Ipopt [32,16] for more general non-linear sub-problems. CasADi [1] is used as a modelling and automatic differentiation front end to Ipopt. This implementation was designed with flexibility in mind, so that a wide range of experiments could be conducted. The experiments were run on machines with 2 AMD 6-Core Opteron 4184, 2.8GHz, 3M L2/6M L3 Cache CPUs and 64GB of memory.

6 Test Microgrid

Our experiments are based around a modified 70 bus 11kV benchmark distribution network [8], which was chosen because it has a comparable size to that of a suburb. We close all tie lines in the network in order to mesh it and replace the substations with microgenerators. We expect microgrids to take on more of a meshed network structure to improve reliability and efficiency, and to better utilise distributed generation. We proportionally replace the benchmark static PQ bus loads with houses, which results in a total of 3674 houses.

A house is an independent agent that manages sub-components. For our experiments these include an uncontrollable background power draw (based on aggregate data for an Australian Autumn day) and two shiftable loads. A house has a single terminal through which it can exchange real and reactive power with the rest of the network, and an apparent power limit of $s = 10$kVA.

The time horizon spans 24hrs with 15min time steps, which produces a problem instance with over 2 million variables. We used a primal and dual stopping tolerance of $\epsilon = 10^{-4}$ and a fixed penalty parameter of $\rho = 0.5$ (we adopt stopping criteria similar to that in [19]). In terms of the real power at a connection, this residual translates to 10W, or about 1% of the average household load.

We randomise generator and household load parameters to produce different problem instances. The starting values for the algorithm are zero except for the voltages $v_t = 1$ and the real power duals $\lambda_t^p = 5$. This is a naive (or cold) starting point as it uses no information about the particular network instance.

In addition to the 70 bus microgrid, we also ran experiments on 20 to 2000 bus randomly generated networks, similar to those described in [19]. These results will not be discussed here, however they are much the same as the 70 bus test microgrid results.

7 Impact of Power Flow Models

In this section we investigate how the ADMM algorithm performs with 5 different power flow models in order to establish their relative trade-offs.

7.1 Power Flow Models

Due to their non-convex nature, the AC power flow equations (1–3) are often either relaxed or approximated. Convex relaxations include a quadratic constraint (QC) model [15], a semi-definite program (SDP) [2], the dist-flow (DF) relaxation [11,3] and an equivalent SOCP relaxation [17,4]. Approximations include the linear DC (DC) model [25,27], the LPAC model [6] and the quadratic formulation (K) proposed by Kraning et al. [19].

As shown in Section 2, some of these models have been used with the ADMM algorithm, but what is lacking is a comparison of the relative strengths and weaknesses between the different models when used in distributed context. In this section we compare the algorithm performance when using the AC, QC, DF, DC, and K line models. We compare the differences in solution quality, feasibility, processing time and number of iterations for our test network. What we find is that even though the AC equations are non-convex, in practice they converge and perform well compared to the other approaches.

We generate 60 random instances of our test microgrid with the binary variables for the shiftable devices relaxed. These are then solved using the distributed algorithm described in Section 4, for each of the 5 different power flow models.

7.2 Convergence

The algorithm successfully converged for all 60 instances and all 5 power flow models. This was expected for all the convex models, but we had no guarantee for the non-convex AC model. This gives us confidence that the exact AC model, even though non-convex, can in practice be used within distributed algorithms.

Table 1 provides the number of iterations and time taken to converge in the form of means and standard deviations. The parallel solve time is the amount of time required to solve the problem in a fully distributed implementation. This was measured by summing together the time of the slowest component at each iteration. The K model is significantly faster than the other models, but in absolute terms the solve times are modest for all models.

Fig. 1 shows an example of the primal residual convergence for the different line models. The AC, QC and DF models overlap. One unintuitive result is that the DC model converges poorly when it is in fact a very simple linear model. Large oscillations build up across the network which slows progress.

Warm Starting As described in Section 6, we are giving the algorithm a naive starting point for both the primal and dual variables . In practice, the receding horizon control scheme will provide an excellent warm starting point, because

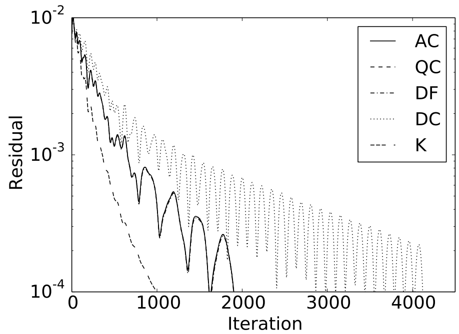

Fig. 1: Convergence of primal residuals.

Table 1: Iterations and parallel solve time for line models.

	Iterations (std.)	Time in sec (std.)
AC	1945 (17)	148 (12)
QC	1951 (14)	546 (33)
DF	1933 (26)	110 (8)
DC	4140 (50)	244 (8)
K	1027 (52)	15 (1)

the solution from the previous horizon can be used for all but one time step. We performed warm starting experiments for the AC model. Similar to what was done in [19], we duplicate a problem instance and then randomly resample the household background power and shiftable device power parameters according to the rule: $p \sim p\mathcal{N}(1, \sigma^2)$. We used the solution of the original instance as a starting point for the modified instance. For $\sigma = 0.2$ the warm started run only needed 11% of the original iterations on average. When the resampling step is fully correlated between all agents, this number increases to 29%.

7.3 Solution Quality

For each model we calculate the percentage difference in objective value relative to the best known AC solution: $100 \cdot (f - f_{best})/f_{best}$. The means and standard deviations of the 60 instances are:

QC -0.031% (0.008%)	DF 0.039% (0.018%)
DC -3.541% (0.072%)	K 4.726% (0.090%)

Because the AC equations are non-convex, we don't have a guarantee that the solutions they produce are globally optimal. However, they provide a feasible upper bound. The QC and DF models are convex relaxations of the AC equations, so they provide a lower bound on the global optimal. This means the global optimal solution resides somewhere between the values of the AC solution and the QC and DF solutions.

With this in mind, we find that the AC, QC and DF models all produce solutions which are very close to each other. The difference is within the margin of error of the objective function afforded by our stopping criteria, which we estimate to be 1% (see Section 6). This indicates that the AC, QC and DF models produce solutions that are within 1% of the global optimal. Experiments were run on a limited number of instances with tighter tolerances, and the results show that this gap further shrinks into insignificance for our network.

These results show the feasibility of using the non-convex AC power flow equations for solving a distributed OPF problem in a microgrid context. The K model adopted in [19] converges much faster, but it is unlikely to be usable in a realistic setting, as it produces unrealistically high costs.

8 Discrete Decisions

We now want to solve the original problem where the shiftable load binary variables u_t are not relaxed. In order to do this we extend the algorithm so that it can manage discrete decisions.

8.1 Methods

We investigate 3 tractable methods that have no global optimality guarantees. Just as we did for the AC equations, we will compare our result to a lower bound in order to get an understanding of the optimality gap. They are called:

- Relax and price (RP)
- Relax and decide (RD)
- Unrelaxed (UR)

The RP and RD approaches are broken up into 2 and 3 stages respectively. The first stage, called the negotiation stage, is common to both methods. All integer variables are relaxed and the distributed algorithm is run until convergence, just like what was done for the power flow experiments. At this point the integer variables may take on fractional values, and this solution gives a lower bound on the global optimal solution. In the second stage each component makes a local decision in order to force any fractional values to integers.

Relax and Price In the second stage of this method, each house performs a local optimisation to determine how to enforce integer feasibility of u_t. We designed a range of cost functions which penalise a component if it changes its terminal values from those that were negotiated in the first stage. For a given cost function, each house solves a MIP to obtain an integer-feasible solution. The two most effective cost functions that we identified are:

$$f_0(y, \hat{y}, \hat{\lambda}) = \hat{\lambda}^\mathsf{T} y + \alpha h(y, \hat{y})^\mathsf{T} h(y, \hat{y}) \tag{10}$$

$$f_3(y, \hat{y}, \hat{\lambda}) = \hat{\lambda}^\mathsf{T} A\hat{y} + \alpha h(y, \hat{y})^\mathsf{T} \Lambda h(y, \hat{y}) \tag{11}$$

where, for a given house to bus connection, $\hat{y}$ is the negotiated terminal values for the bus and $\hat{\lambda}$ the negotiated dual variables. We use Λ to represent the diagonal matrix where $\Lambda_{i,i} := |\hat{\lambda}_i|$ and α is a penalty parameter.

The first function charges households at the negotiated price for what they *actually* consume and a quadratic penalty for operating away from the negotiated consumption. The second function requires the household to pay for all power that was negotiated in the first stage, and it has a price-scaled penalty for operating away from the negotiated operating point.

After this local optimisation step, we check that the solution is feasible and what the overall cost is by using the generators to balance the load.

Relax and Decide In the second stage of the RD method, the largest u_t value of each shiftable component is chosen to be fixed at 1 and the rest set to 0. In the third stage the distributed algorithm is restarted in order to converge to a new solution that is integer feasible.

Unrelaxed The final approach, UR, consists of a single stage where it attempts to enforce integrality satisfaction at each iteration of the distributed algorithm. We have already foregone theoretical convergence guarantees by our adoption of the non-convex AC equations. Here we push the ADMM algorithm even further by allowing discrete variables into the algorithm (7–9), where Gurobi solves MIPs for houses, and Ipopt NLPs for lines.

We ran experiments on 60 random instances of our test microgrid for each of the three approaches. We use the AC line model for each experiment and a penalty of $\alpha = 10$ for the RP approach. In the following sections we discuss the convergence of the methods and the quality of the solutions.

8.2 Convergence

None of the approaches are guaranteed to find an integer feasible solution if one exists, however, for all experiments on our test microgrid they converged to feasible solutions. The RP method only marginally increases the solve time above the results in Section 7, and the RD method only requires a small amount of extra time in order to finish solving the problem from a warm starting position. The UR method takes 1.7 times longer on average.

8.3 Solution Quality

We compare the change in objective value relative to the relaxed version of the problem. The results are shown in Fig. 2, where we have separated the objective into terms for the cost of generation and the charge to households. For the RP methods the charge is given by the cost functions in the previous section. For the RD and UR methods the charge is simply the final $\lambda^\intercal y$ for each house.

All methods produce costs that are within 1% of the relaxed problem, and hence also the global optimum. There is no significant difference between the methods as they reside within our estimated margin of error based on our stopping tolerance. This suggests that we have a tight relaxation of the integer problem. One reason for this is that each shiftable load only contributes a tiny amount to total demand.

By artificially increasing the size of the shiftable loads by more than an order of magnitude, and heavily congesting the network, we do find instances where there is a significant gap between the relaxed solution and the candidate. However, for the realistically sized residential shiftable loads as utilised in our test microgrid, the relaxation was tight.

The charges to households are higher for the RP method without gaining any benefit in terms of reduced costs. We ran the same experiments with a

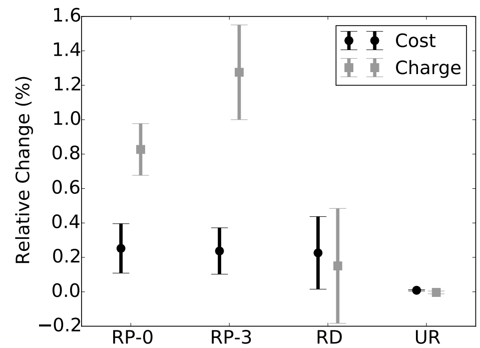

Fig. 2: Generator cost and household charge relative to relaxed solution.

much smaller α, which all but eliminated charges without any increase to costs. This suggests that for the sole purpose of managing shiftable loads, only a small penalty is required. However, the penalty may serve other purposes, for example, in deterring strategising agents.

All of the methods we have presented provide an efficient means for dealing with the discrete decisions in a household. Other factors such as the way they can handle uncertainty and the possible need to discourage agents gaming the system will influence the choice between these methods.

9 Conclusion and Future Work

We have presented a distributed demand response mechanism for operating a microgrid. It can coordinate a whole range of distributed agents with time coupled behaviours, whilst preserving network constraints and providing locational prices in the network. Using this mechanism we have successfully compared the performance of a range of power flow models in a meshed microgrid, and introduced simple but effective approaches to handling the shiftable loads within households. We developed a suburb-sized test microgrid, and found that the full non-convex AC equations produce close to optimal solutions in short solve times. All three of our methods for handling household shiftable loads produce close to optimal solutions with only a moderate increase in solve times.

Our work has shown that in practice distributed algorithms are not only feasible, but also highly effective at performing DR within a microgrid.

Future research will investigate alternative distributed solving techniques with the aim of improving convergence rates. There are also opportunities for finding closed-form solutions for the exact AC equations, and it might be possible to build a frequency regulation market into the distributed algorithm.

We need further experiments to investigate if our results carry over to larger discrete decisions, for example, those related to large industrial plant, generator start-up costs, and line switching. We also plan to answer the important question of how susceptible this mechanism is to gaming in practice, and if this is a problem, what can be done about it.

References

1. J. Andersson. *A General-Purpose Software Framework for Dynamic Optimization.* PhD thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, October 2013.
2. X. Bai, H. Wei, K. Fujisawa, and Y. Wang. Semidefinite programming for optimal power flow problems. *International Journal of Electrical Power & Energy Systems*, 30(67):383 – 392, 2008.
3. M. Baran and F. Wu. Optimal sizing of capacitors placed on a radial distribution system. *Power Delivery, IEEE Transactions on*, 4(1):735–743, 1989.
4. S. Bose, S. H. Low, T. Teeraratkul, and B. Hassibi. Equivalent relaxations of optimal power flow. *CoRR*, abs/1401.1876, 2014.
5. S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, Jan 2011.
6. C. Coffrin and P. V. Hentenryck. A linear-programming approximation of ac power flows. *Informs Journal on Computing*, May 2014.
7. E. Dall'Anese, H. Zhu, and G. B. Giannakis. Distributed optimal power flow for smart microgrids. *Smart Grid, IEEE Transactions on*, 4(3):1464–1475, 2013.
8. D. Das. Reconfiguration of distribution system using fuzzy multi-objective approach. *International Journal of Electrical Power & Energy Systems*, 28(5):331 – 338, 2006.
9. J. Douglas, Jim and J. Rachford, H. H. On the numerical solution of heat conduction problems in two and three space variables. *Transactions of the American Mathematical Society*, 82(2):pp. 421–439, 1956.
10. T. Erseghe. Distributed optimal power flow using admm. *Power Systems, IEEE Transactions on*, 29(5):2370–2380, Sept 2014.
11. M. Farivar, C. R. Clarke, S. H. Low, and K. M. Chandy. Inverter var control for distribution systems with renewables. In *SmartGridComm*, pages 457–462. IEEE, 2011.
12. D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17 – 40, 1976.
13. N. Gatsis and G. Giannakis. Residential load control: Distributed scheduling and convergence with lost ami messages. *Smart Grid, IEEE Transactions on*, 3(2):770–786, June 2012.
14. Gurobi Optimization, Inc. Gurobi optimizer reference manual, 2014.
15. H. L. Hijazi, C. Coffrin, and P. Van Hentenryck. Convex quadratic relaxations for mixed-integer nonlinear programs in power systems. *NICTA Technical Report http://www.optimization-online.org/DB_HTML/2013/09/4057.html*, March 2014.
16. HSL Archive. A collection of fortran codes for large scale scientific computation, 2014.
17. R. Jabr. Radial distribution load flow using conic programming. *IEEE Transactions on Power Systems*, 21(3):1458–1459, Aug 2006.

18. B. Kim and R. Baldick. A comparison of distributed optimal power flow algorithms. *Power Systems, IEEE Transactions on*, 15(2):599–604, 2000.
19. M. Kraning, E. Chu, J. Lavaei, and S. Boyd. Dynamic network energy management via proximal message passing. *Foundations and Trends in Optimization*, 1(2), 2014.
20. S. Magnússon, P. C. Weeraddana, and C. Fischione. A Distributed Approach for the Optimal Power Flow Problem Based on ADMM and Sequential Convex Approximations. *ArXiv e-prints*, Jan. 2014.
21. A. Mohsenian-Rad, V. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *Smart Grid, IEEE Transactions on*, 1(3):320 –331, dec. 2010.
22. A.-H. Mohsenian-Rad and A. Leon-Garcia. Optimal residential load control with price prediction in real-time electricity pricing environments. *Smart Grid, IEEE Transactions on*, 1(2):120 –133, sept. 2010.
23. Q. Peng and S. H. Low. Distributed Algorithm for Optimal Power Flow on a Radial Network. *ArXiv e-prints*, Apr. 2014.
24. S. D. Ramchurn, P. Vytelingum, A. Rogers, and N. R. Jennings. Agent-based control for decentralised demand side management in the smart grid. In Tumer, Yolum, Sonenberg, and Stone, editors, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems - Innovative Applications Track (AAMAS 2011)*, pages 330–331, Taipei, Taiwan, May 2011.
25. F. Schweppe and D. Rom. Power system static-state estimation, part ii: Approximate model. *power apparatus and systems, IEEE transactions on*, (1):125–130, 1970.
26. P. Scott, S. Thiébaux, M. van den Briel, and P. Van Hentenryck. Residential demand response under uncertainty. In *International Conference on Principles and Practice of Constraint Programming (CP)*, pages 645–660, Uppsala Sweden, sep. 2013.
27. B. Stott, J. Jardim, and O. Alsac. Dc power flow revisited. *Power Systems, IEEE Transactions on*, 24(3):1290–1300, Aug 2009.
28. M. van den Briel, P. Scott, and S. Thiebaux. Randomized load control: A simple distributed approach for scheduling smart appliances. In *International Joint Conference on Artificial Intelligence (IJCAI)*, Beijing, China, August 2013.
29. M. Vinyals, F. Bistaffa, A. Farinelli, and A. Rogers. Coalitional energy purchasing in the smart grid. In *Energy Conference and Exhibition (ENERGYCON), 2012 IEEE International*, pages 848–853, Sept 2012.
30. M. Vinyals, V. Robu, A. Rogers, and N. R. Jennings. Prediction-of-use games: A cooperative game theoryapproach to sustainable energy tariffs. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, AAMAS '14, pages 829–836, Richland, SC, 2014. International Foundation for Autonomous Agents and Multiagent Systems.
31. P. Šulc, S. Backhaus, and M. Chertkov. Optimal Distributed Control of Reactive Power via the Alternating Direction Method of Multipliers. *ArXiv e-prints*, 2014.
32. A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.