

Exploring Optimization Techniques for an Existing Microgrid and the Unit Commitment with AC Optimal Power Flow Problem

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Amelia Jane McIlvenna

December 2019

© by Amelia Jane McIlvenna, 2019
All Rights Reserved.

Acknowledgments

Many thanks to my advisor, Professor James Ostrowski, for all of your help and encouragement over the years. Special thanks to Ben Ollis for bringing me into the group at Oak Ridge National Lab. Thanks to Drew Herron for putting up with me, and of course for pushing the buttons. I would also like to thank Josh Hambrick, Dan King, and everyone else at Oak Ridge National Laboratory for making my time there so enjoyable. Thanks to Professors Anahita Khojandi, Hugh Medal, and Hector Pulgar for serving on my dissertation committee.

My deepest gratitude to my husband, Derek McIlvenna, for your unending support, friendship, and love. I couldn't have done this without you. I am also grateful for the wonderful friendships I've made at UT, including Jonathan Schrock, Whitney Forbes, Tony Rodriguez, Ben Knueven, and Ethan Deakins. Thanks to Francesca Moloney and Rebekah Ronan for being my best friends through it all, your friendship is so invaluable to me. And of course, thanks to my cats, Tigz and Captain, who always put the biggest smile on my face.

Abstract

The first three chapters of this dissertation will investigate mixed-integer optimization techniques for an existing microgrid system which was constructed in Hoover, Alabama in 2018.

Chapter 1 will introduce the details of the microgrid. As with any microgrid system, assets must be controlled and dispatched to provide sufficient power to cover any load. To facilitate optimal dispatch of the natural gas generator, lithium-ion battery, and photovoltaic power devices, a mixed-integer linear program was developed to optimize operation.

While operation of the real world system was being handled, the optimization was not solving in sufficient time due to open-source software constraints and limited hardware. Chapter 2 explores a solution to these constraints by implementing a novel relaxation for the rolling-time horizon optimization model developed in Chapter 1. The relaxation method reduces the number of integer variables in the formulation, taking advantage of the fact that on a rolling time horizon time periods close to the present are more important than those farther in the future.

Another issue when optimizing dispatch of a microgrid system is inherent uncertainty in the problems. Commonly uncertainty is handled using two-stage stochastic programs broken into stages by commitment variables vs. power variables. While this method is appropriate for day-ahead commitment models, second stage power variables complicate a rolling time horizon model's need to send power setpoints to the devices in the microgrid. We solve this problem by introducing a novel stage formulation in Chapter 3 broken by time interval instead of variable type. This enables incorporation of robustness while also retaining the ability to operate on a rolling time horizon.

The models from the first chapters are small enough that a convex relaxation of an AC optimal power flow formulation could be used with minimal adjustments to the solution in practice to maintain AC feasibility. Unfortunately the canonical unit commitment problem with AC optimal power flow constraints is nonconvex and not computationally tractable. In Chapter 4, we employ a Benders decomposition-like framework with an AC optimal power flow relaxation to find solutions to this problem which are closer to AC feasible than other techniques for large-scale networks such as DC optimal power flow.

Table of Contents

1	Real-world applications of modeling microgrid systems	1
1.1	Practice Summary	1
1.1.1	Introduction	2
1.1.2	Optimization Models	4
1.1.3	Pricing Algorithm	8
1.1.4	Software	9
1.1.5	Results	10
1.1.6	Conclusion	12
1.2	Economic Dispatch Model for Alabama Microgrid	14
1.3	Nomenclature:	14
1.3.1	Basic Model	15
1.3.2	AC Optimal Power Flow Formulation	18
1.4	Application Motivated Constraints	21
1.4.1	Microgrid Setup	21
1.4.2	Battery Cycling	21
1.4.3	Generator Minimum up-time	22
1.4.4	Reactive Power Production Limits	26
1.4.5	Off-grid Curtailment and VF Constraints	27
1.4.6	Full Model Discussion	28
1.5	Pricing Algorithm	29
1.5.1	Algorithm Description	32
1.5.2	Issues and fix	35

1.5.3	Simulation Results	38
1.6	Conclusion	38
2	Reducing the Computational Burden of a Microgrid Energy Management System	41
2.1	Motivation	41
2.2	Abstract	42
2.3	Nomenclature:	43
2.3.1	Variables	43
2.3.2	Parameters	43
2.3.3	Sets	43
2.4	Introduction	44
2.5	Model	47
2.5.1	Basic Economic Dispatch	47
2.5.2	Relaxation of Integrality	49
2.6	Description of Neighborhood	51
2.7	Computational Experiments	52
2.7.1	Description of Tests	52
2.7.2	Results	54
2.8	Conclusion	58
3	An uncertainty analysis for microgrid economic dispatch	61
3.1	Abstract	62
3.2	Nomenclature:	63
3.2.1	Variables	63
3.2.2	Parameters	63
3.2.3	Sets	63
3.3	Introduction	64
3.4	Model	66
3.4.1	Constraints	69
3.4.2	Scenario Construction	75

3.5	Computational Results	76
3.5.1	Test Cases	76
3.5.2	Results	80
3.6	Conclusion	83
4	Cut Generation for UC+ACOPF	84
4.1	Nomenclature:	85
4.1.1	Sets	85
4.1.2	Variables	85
4.1.3	Parameters	86
4.2	Introduction	86
4.3	Models	88
4.3.1	Unit Commitment Formulation	88
4.3.2	ACOPF Formulation	92
4.4	Solution Algorithm	95
4.5	Test Instances	98
4.5.1	Instance Formulation	98
4.5.2	Computational Results	98
4.6	Conclusion	100
	Bibliography	102
	Vita	112

List of Tables

1.1	Iterations to Convergence for 5 sample days	12
1.2	Device Parameters	21
2.1	Models Developed for Computational Testing	53
2.2	Average Computation Time (sec.)	55
2.3	Number of days which exceeded time limits	55
2.4	Number of days with cost within 10% of Normal	56
2.5	Number of days with cost less than Normal	56
3.1	Cost of operation in \$ for each test case	80
4.1	Benders Iterations and Computation Time	99
4.2	Cost results (\$/MWH) for all test instances	99
4.3	Real Power Slack on Feasibility Check (MW)	100

List of Figures

1.1	Overview of the Microgrid	5
1.2	Economic Dispatch for May 15, 2019 3:00pm-12:00am	11
1.3	Sample single home algorithm solution from August 20 test case	13
1.4	Battery real power dispatch on 06/28/18 when frequent cycling presented as an issue	23
1.5	Battery real power dispatch on 11/27/18 after battery cycling was penalized	23
1.6	Economic Dispatch January 30, 2019	25
1.7	Economic Dispatch December 19, 2018	30
1.8	Economic Dispatch March 4, 2019	31
1.9	PCC power March 4, 2019	31
1.10	Pricing Algorithm Overview	33
1.11	Two consecutive iterations of Algorithm 2	37
1.12	Sample single home solution from August 18 test case	39
1.13	Load Profiles August 20, 2017	39
2.1	Percent difference in cost for Table 2.1 models compared to Normal	57
2.2	Percent difference in cost for select models compared to Normal	59
3.1	Stage Construction for the stochastic program broken by time interval with important variables listed for each stage	68
3.2	PV forecast and measured values for 12/23/18	78
3.3	Forecast Horizons for the Rolling Time and Perfect Information Models	79
3.4	Generator power output for 01/04/19 for DM and SSE cases	82

Chapter 1

Real-world applications of modeling microgrid systems

Microgrid systems are small power networks which consist of local electricity generation [32]. They are often uniquely developed for their specific location [42], and can be utilized for university, industry, military, or residential purposes. The first chapter of this work will discuss optimization models and algorithms used in a real-world residential microgrid built in 2018. Section 1.1 is a practice summary developed for publication which details an overview of the microgrid system, its capabilities, and results from its operation over a 6 month period of time. Sections 1.2 and 1.4 describe the economic dispatch model introduced in Section 1.1 in explicit detail, while Section 1.5 expands on the pricing algorithm. Lastly, Section 1.6 wraps up the first chapter.

1.1 Practice Summary

Section 1.1 is based on a paper prepared for publication by the authors Amelia McIlvenna, Andrew Herron, Daniel King, Philip Irminger, Joshua Hambrick, Ben Ollis, and James Ostrowski.

McIlvenna, A., Herron, King, D., Irminger, P., A., Hambrick, J., Ollis, B., Ostrowski, J. (2019). Practice Summary: Improving Reliability Through Optimization in Residential Microgrids *Submitted*.

Author McIlvenna programmed the optimization routines listed in this section. Authors McIlvenna and Ostrowski developed the pricing algorithm strategy. Authors Herron, King, Irminger, Hambrick and Ollis facilitated testing of the assets within the physical microgrid system. Authors Hambrick, Herron, and McIlvenna programmed the communications architecture between devices in the microgrid. Author McIlvenna wrote the manuscript, and Authors Herron, Ollis, and Ostrowski provided feedback on the manuscript.

Southern Company is an electric utility that is embracing residential microgrids as a platform for distributed energy. The first residential microgrid in the southeastern United States was constructed in Hoover, Alabama as a testbed for development. To operate the microgrid, we developed an optimization model for unit commitment and economic dispatch. The optimization model was implemented in a microgrid central controller and tested under various operational scenarios. Challenges also include developing a pricing algorithm to deliver price forecasts to the residential homes in the microgrid, and open-source software restrictions.

1.1.1 Introduction

In recent years, renewable energy sources such as solar and wind power have seen a decrease in cost and an increase in uptake in the United States. Large-scale solar and wind farms, as well as single home solar and battery combinations are being utilized. On a residential level, renewable energy is especially attractive to homeowners who have an electric vehicle or are motivated to reduce their carbon footprint. Renewable resources can be installed within a microgrid, providing local generation to a residential neighborhood. Although other residential microgrids have been studied, such as [72], differences in layout and location can impact optimization and control strategies.

Notice: This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (<http://energy.gov/downloads/doe-public-access-plan>).

Differences between individual microgrids motivate case-specific solutions to optimal control. Studies such as [12] concentrate on university-owned microgrids which tend to be larger than residential microgrids and serve university operated buildings. Hypothetical studies are also conducted, for example [41], [58], but they may not capture all challenges associated with a physical microgrid system. Much of the applied research has focused on university-owned and government-owned microgrids, but now electric utilities are also beginning to adopt microgrids as an effective means of distributed generation. Transitioning from a highly-controlled system like a university-controlled microgrid to a residential neighborhood comes with considerable challenges. Residents will not tolerate unnecessary power outages and poor voltage control may cause damage to personal property. It is our understanding that this is the first practicing residential microgrid that is using operations research to optimize the microgrid's assets.

Residential sized microgrids can provide many benefits to electric utilities. Having a microgrid central controller (MCC) can allow the utility more control over the generation assets compared to a more distributed model, such as each home having its own solar panels. Another strong advantage of a residential sized microgrid is its ability to disconnect from the main electrical grid, called islanding. Islanding allows the microgrid to rely solely on localized electricity generation and storage capabilities. When the microgrid is islanded, the MCC can reconnect to the main grid via the point of common coupling (PCC), called resynchronization. Islanding and resynchronization improve reliability for customers by eliminating disruption during a planned or unplanned outage in the main electrical grid.

One of the utilities looking to explore the challenges associated with microgrids is Southern Company. Southern Company has built a smart neighborhood in Hoover, Alabama, with the intent of using its microgrid as a research testbed for challenges related to residential microgrids. The first of its kind in the southeastern United States, the neighborhood has 62 residential homes, each complete with smart appliances for their residents. Each home contains its own optimization architecture for its controllable assets, upon which load forecast information can be sent to the load aggregator to be communicated to the MCC. The load forecast optimization is separate from the MCC and not involved with the pricing algorithm beyond communicating forecasts and receiving a price signal. Generation devices controlled

by the MCC include a 330kW photovoltaic (PV) system, 400kW natural gas generator, and 300kW Li-ion battery with an energy rating of 680kWh. The PV forecasts are developed by the Clean Power Research company. Figure 1.1 outlines the microgrid system.

To facilitate daily operation of a microgrid system, a robust MCC is needed. Solving for optimal cost dispatch decisions often requires mixed-integer and nonlinear optimization problems, as well as technical communications between the MCC and devices in the microgrid. The need for these systems prompted collaboration with Oak Ridge National Laboratory and the University of Tennessee. The next section will detail the optimization model developed as a result of this collaboration.

1.1.2 Optimization Models

When the microgrid is running, managing resources is critical to avoiding outages and unnecessary costs. Optimization models are useful for making operational decisions intelligently. Managing the output of generation resources in an electrical grid while minimizing cost subject to operational constraints is known as the economic dispatch problem. Solving for economic dispatch also requires unit commitment decisions, which include when to turn generators on/off. To make the unit commitment and economic dispatch decisions for the Hoover microgrid, an optimization model for economic dispatch was developed, and will be referred to as (ED).

The optimization model (ED) solves for economic dispatch in the microgrid over a 24 hour rolling time horizon, with 5 minute time intervals. Each variable is included for each time interval in the rolling time horizon. Variables include real power and reactive power output of all generation devices in the microgrid, as well as commitment decisions such as when to turn the generator on/off. Real power represents energy transfer to a load; it is power which does work in an AC system. Reactive power is the energy which is bounced back from load to source, and does not transfer energy in the system (also known as wattless power). Additional variables also include power flows (both real and reactive) along the transmission lines.

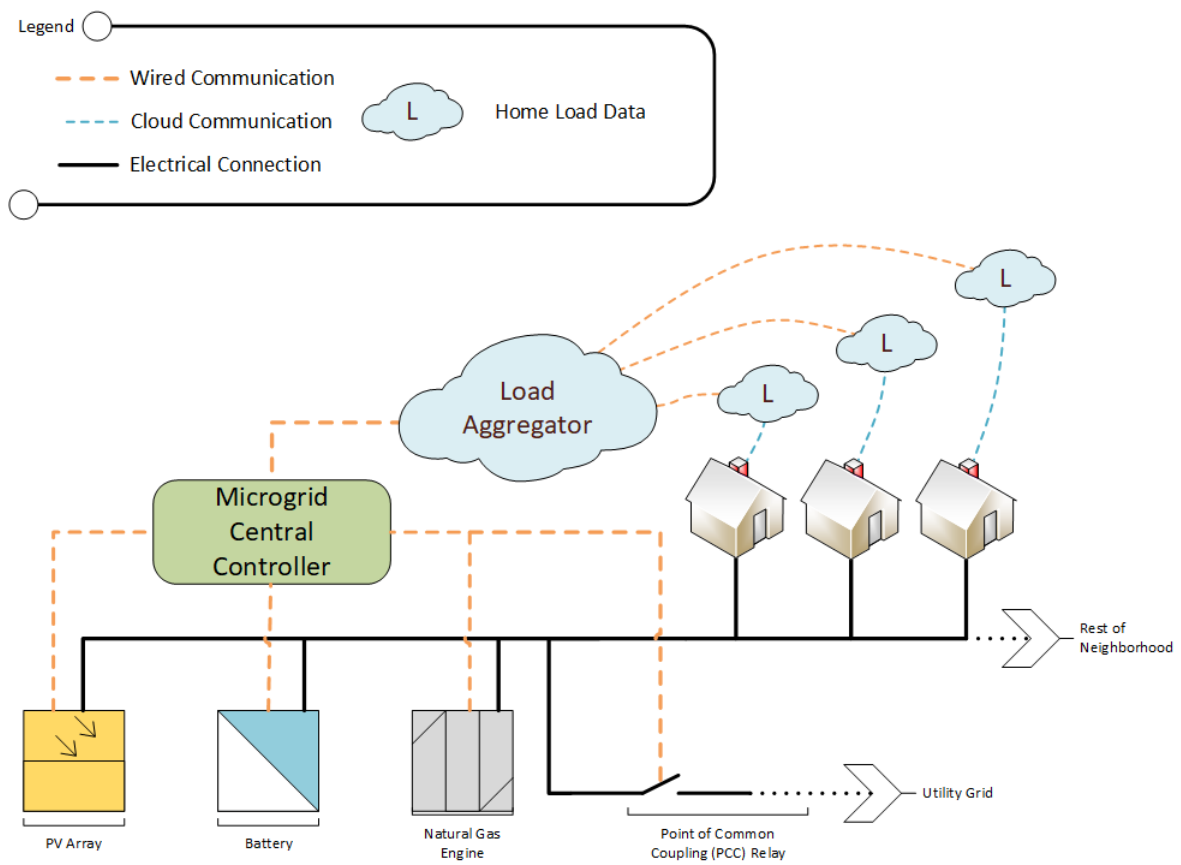


Figure 1.1: Overview of the Microgrid

$$\begin{aligned}
& \min \text{Generator and Grid Exchange Cost} && (\text{EDa}) \\
& \quad + \text{Constraint Violations} \\
& \textit{st.} \text{Generator Constraints} && (\text{EDb}) \\
& \quad \text{Battery Constraints} && (\text{EDc}) \\
& \quad \text{PV curtailment Constraints} && (\text{EDd}) \\
& \quad \text{Power Balance Constraints} && (\text{EDe}) \\
& \quad \text{AC Power Flow Constraints} && (\text{EDf})
\end{aligned}$$

Model (ED) shows the types of constraints in the full economic dispatch model included in the MCC. The full model will be detailed later in Section 1.2. The problem objective is to minimize the cost of operating the natural gas generator which is composed of its start-up cost, minimum up-cost, and a piecewise marginal cost curve, as well as cost of power exchange with the main grid. PV and battery operational costs are assumed to be negligible. Soft constraints penalize extreme behavior and are included in the constraint violations term. These include the battery reaching an extreme state of charge, for example greater than 80% state of charge (SOC), or less than 20% SOC. We also employ a penalty on line losses. Other soft constraints include excessive use of reactive power. Because customers are not charged for reactive power in a residential setting, its use is generally minimized to avoid power production which is unprofitable to the utility.

Hard constraints are also included for each generation device in the microgrid, which ensures each device stays within its operating envelope. For the generator, these constraints (EDb) include keeping track of generator start-up status, and ensuring if the generator is on its power stays above its minimum of 100kW and below its maximum output of 400kW. We also employ a generator minimum up-time to avoid frequent cycling of the generator. A minimum down-time can also be included, but for this application the down-time is 5 minutes, i.e. the length of one time interval. Also, ramping constraints are not necessary

because the generator is small and therefore able to ramp fully up or down within one 5-minute interval.

Battery constraints (EDc) include power charging/discharging limits, and limiting the battery to either charging or discharging at each time interval. We also track the energy stored in the battery at time t , with hard limits at 0% and 100% SOC. PV power can also be curtailed to less than its forecasted value via (EDd). Hard power balance constraints (EDe) are included to certify generation meets the forecasted demand of the neighborhood at every time interval.

When constructing an economic dispatch model, transmission constraints are often included in the form of an AC optimal power flow (ACOPF) formulation. AC optimal power flow is recognized as a notoriously difficult problem within the optimization community due to its nonconvexity and NP-hardness. There are some relaxation techniques developed for ACOPF using quadratic convex programming, semi-definite programming, second-order cone programming [14, 20, 7, 35]. These techniques, however are computationally intensive even for small networks, and will not solve within the 5 minute time limit required for our model. Therefore we use an ACOPF relaxation known as the dist-flow equations introduced in [8, 44] for constraints (EDf). It should be noted that there are concerns that an AC power flow relaxation may not return an AC feasible solution. The dist-flow equations, however, are exact under certain conditions [44]. Moreover devices can adjust power outputs due to slight inaccuracies in the relaxation as well as error in PV and load forecasts. The benefit of quick solution time in this case outweighs the risk of using an ACOPF relaxation instead of a more accurate and computationally intensive method. Note that even the ACOPF relaxation is a mixed-integer nonlinear program, due to discrete on/off decisions and nonlinear AC power flow constraints.

Because the microgrid needs the ability to transition from on-grid to off-grid and back, extra constraints are added to facilitate each mode of operation. For example, we require the power at the point of common coupling (PCC) to be zero when off-grid, because no exchange is allowed with the main grid. Furthermore, we may need to heavily curtail PV power when off-grid, for example when the battery is at maximum state of charge and there is excess PV power.

1.1.3 Pricing Algorithm

Another challenge of operating microgrids is developing a sufficient market framework for real-time electricity prices. Many future homes will have onboard energy management systems which seek to efficiently and independently manage each home’s controllable load in response to a price signal. The addition of smart appliances allowed Oak Ridge National Laboratory to develop mixed-integer linear programming models to control the HVAC and water heater systems for each home in the neighborhood [64]. For the purposes of this study, the home energy management system will be treated as a black box whereby the MCC has no knowledge of its components.

Pricing electricity would be easier if the MCC had full access to each home’s energy management system and smart appliances. However, many residential customers are uncomfortable with a utility having access to private data regarding their energy consumption. Therefore, the MCC only has access to the aggregate load forecast from each phase in the residential neighborhood. Because the neighborhood homes have autonomous control, a pricing scheme similar to a Lagrangian decomposition was developed wherein electricity price would be manipulated to influence load forecast changes in the neighborhood.

The pricing algorithm iterates between the MCC and the residential homes to decide an acceptable price forecast for the next 24 hours. An iterative framework was created to allow both the microgrid and individual homes to effectively manage their resources by minimizing cost, while still giving emphasis to reliability and privacy concerns. Although coordinating the neighborhood’s collective load response with the MCC adds a layer of complexity, it is necessary to ensure customer privacy. The algorithm updates the price based on a predetermined condition which uses the net change in load for each time period at each iteration.

A common problem in pricing algorithms is every load which receives the same price forecast will take similar actions in shifting their load. To avoid this, we take an individualized approach to the price forecast. The price signal sent to each home h at each time period t is manipulated with a small perturbation parameter ϵ_{ht} . When the aggregate load deviation between iterations is sufficiently small, the MCC will accept the load forecast

and the algorithm will terminate. Algorithm 1 details this process, where λ_{ht}^k represents the microgrid price for home h time t for iteration k , and D_t^k is the total demand at time t , iteration k . Note the initial price forecast can be set either by taking the dual variables of the balance constraint in problem (ED), or using a price provided by the utility. Although convergence of the algorithm is not guaranteed, a price can still be set either by accepting one of the intermediate price signals from any iteration, or simply rolling through the last converged price solution until the next algorithm convergence.

Algorithm 1 Pricing Algorithm

```

1: Initialize  $\lambda_{ht}$ ,  $k=0$ 
2: while Converge == False do
3:    $k = k + 1$ 
4:   run (ED) and send price to homes
5:   update aggregate load forecasts  $D_t^k$  (black box) and send to MCC
6:   if  $(D_t^k - D_t^{k-1}) < \beta$  then
7:     Converge = True (MCC accepts the load forecast)
8:   else if  $k \geq \text{max allowable iterations}$  then
9:     break
10:  else
11:    update Price:  $\lambda_{ht}^k = \lambda_{ht}^{k-1} + \epsilon_{ht} + \rho \cdot (D_t^k - D_t^{k-1})$ 

```

1.1.4 Software

Due to the nature of the project, all code was required to use open-source software. The model was coded in python using pyomo, an “open-source optimization modeling language” [30, 29]. The linear programming solver CBC was chosen, which is in the open-source Coin-OR optimization suite. Since the AC power flow equations are inherently nonlinear, both piecewise linearization and McCormick Envelopes were used to linearize the model. Although there are some nonlinear open-source solvers available within the Coin-OR optimization suite, we chose a linear solver for two reasons. The first is a guaranteed global optimal solution. Secondly, because each time interval is usually on a 5 minute scale, computation time should not exceed these limits. Oak Ridge National Laboratory has also developed software for microgrid controls called the Complete System-level Efficient and Interoperable Solution for Microgrid Integrated Controls (CSEISMIC), which was used as MCC in this

project. Microsoft Azure was also used to facilitate communications for logging data and communication between the MCC and the home load aggregator.

1.1.5 Results

The MCC was operated for approximately 95% of the time during a 6-month test period starting June, 2018. The majority of the testing consisted of normal on-grid operations, with the battery charging and discharging according to the optimization results from model (ED). Intentional islanding was also tested during this time. With CBC, solution time is generally less than the 5 minutes required, especially during on-grid operations, which solve in approximately 20-40s. Solutions take longer to achieve when the microgrid is islanded, due to the binary variables associated with generator commitment status, and extra constraints which need to be added. For cases when there is not an optimal solution within 5 minutes, we can use the best solution produced so far, and simply continue the rolling time horizon.

An example of the economic dispatch operations for the microgrid on May 15, 2019 from 3:00pm-12:00am is detailed in Figure 1.2. Dispatch for PV, battery (ES) and generator (GEN) are shown along with total load measurements recorded for the microgrid. Note the minimum operating power of the generator is 100kW.

As previously mentioned, a major advantage of a microgrid is maintaining power for customers during unintentional islanding events. During the test period, the main grid experienced an unintentional outage at 11:16am on October 22, 2018. Although the event caused the PV system to trip off, the battery was able to survive and maintain voltage in the microgrid. After the trip, the PV system came back on, and between the PV and battery the microgrid sustained power for the neighborhood for the duration of the outage. When the main grid came back on at approximately 11:44am, a successful resynchronization was performed.

The successful preservation of power in the neighborhood during the unintentional island was a major success for the project. A shortcoming, however, was that although the PV system was able to reconnect after 5 minutes, the MCC was unable to curtail PV power to the level given by the optimization's output. Therefore, the battery was left to absorb excess



Figure 1.2: Economic Dispatch for May 15, 2019 3:00pm-12:00am

Table 1.1: Iterations to Convergence for 5 sample days

Day:	Aug 16	Aug 17	Aug 18	Aug 19	Aug 20
Iterations:	3	5	2	2	3

PV power. Due to this event, regaining device control after an outage has been identified as a future area of improvement.

The pricing algorithm has been simulated without dispatch to investigate convergence properties. Approximately 80% of runs were converging in less than 10 iterations. Table 1.1 illustrates iterations to convergence for 5 test runs of the algorithm each on a different day of the week from August 16-20, 2018. The parameters used for these runs were $\rho = .0001$, and $\epsilon_{pt} = [-.05, .05]$. Figure 1.3 shows a sample home price forecast compared with the corresponding microgrid price for 24 hours with time 0 representing midnight. In the figure, the algorithm converged after 9 iterations, with $\rho = .00005$, $\epsilon_{pt} = [-.015, .015]$.

1.1.6 Conclusion

In conclusion, this work developed an economic dispatch model and pricing algorithm within a microgrid central controller (MCC) for a physical residential microgrid system. The MCC ran through a test period of 6 months, where successful control of the microgrid under normal and abnormal conditions was verified. Results from prolonged microgrid operation will reveal benefits and new challenges associated with distributed energy integration. Future work associated with this project will explore different pricing updates and acceptance criteria which can be implemented in the existing iterative algorithm structure. The project also provides a framework for Southern Company to continue expanding its distributed energy capabilities. In fact, a second microgrid is being constructed which will consider a more distributed market structure. These different microgrid systems will allow Southern Company to investigate challenges associated with residential markets, which are sure to become more commonplace as both residential customers and utilities adopt distributed renewable energy.

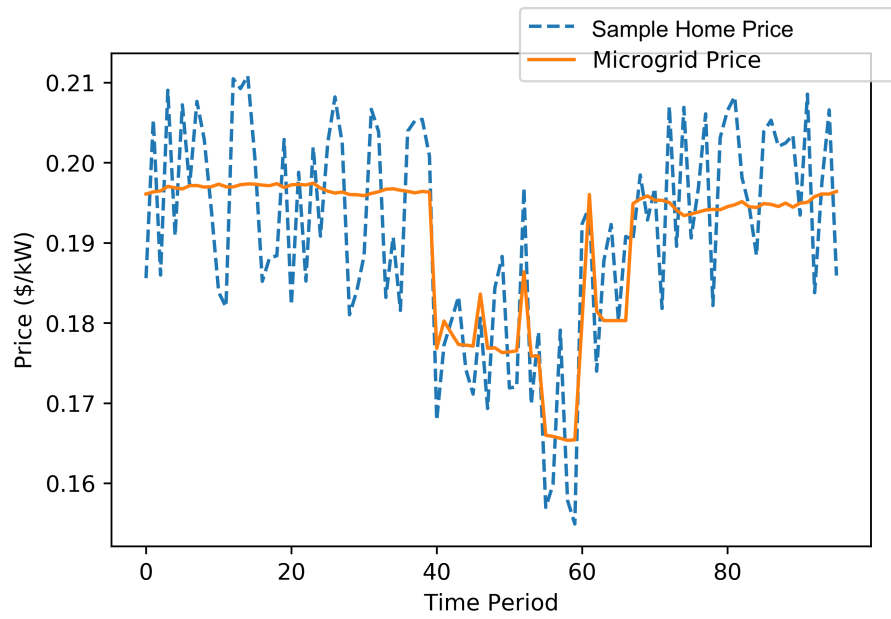


Figure 1.3: Sample single home algorithm solution from August 20 test case

1.2 Economic Dispatch Model for Alabama Microgrid

This section describes in detail the economic dispatch model outlined briefly in section 1.1, which was developed to control the real-world microgrid in Hoover, Alabama. The model will be contained within the CSEISMIC microgrid controller, and will determine operational setpoints for each device in the microgrid over a rolling 24-hour time horizon. Time periods are customarily 5 or 15 minute intervals. Subsection 1.3.1 describes the basic economic dispatch model for real power within the microgrid. Subsection 1.3.2 introduces reactive power and AC optimal power flow constraints to the model before motivating a linearization. Section 1.4 introduces constraints included as a consequence of the real-world application of the model.

1.3 Nomenclature:

VARIABLES:

P_{gt}	Total power at each generator
P_{bt}^C, P_{bt}^D	Battery charging/discharging power, respectively
p_{gt}^m	Power at m th generator interval
u_{gt}	Generator up/down status
v_{gt}	Generator start up indicator
u_{bt}^C, u_{bt}^D	Battery charging/discharging status, respectively
P_{st}	PV power which is not curtailed
E_{bt}	Energy in battery b at time t

PARAMETERS:

k_g	minimum up cost of generator g
C_{gt}^m	Marginal cost for m th generator interval
C_{gt}^v	Start up cost for generator g
C_{bt}	Battery operational cost
P_g^{min}	Generator min up power

D_t	Real demand at time t
P_{st}^F	Forecasted PV power at time t
UT_g	Minimum uptime for generator g

SETS:

$g \in \mathcal{G}$	Set of generators
$b \in \mathcal{B}$	Set of batteries
$s \in \mathcal{S}$	Set of PV panels
$i \in \mathcal{N}$	Set of nodes (buses)
$m \in \mathcal{I}$	Set of piecewise generator power components
$t \in \mathcal{T}$	Set of timeperiods

1.3.1 Basic Model

The basic economic dispatch model is formulated as a mixed-integer linear program (MILP). Equation 1.1 shows the objective for the basic model, which is to minimize cost of electricity generation for the microgrid. To formulate the objective, we need to consider cost for each generation source. Fuel cost for natural gas generators is typically a quadratic curve. However, it is common practice to approximate this cost curve as a piecewise linear function instead. To this end, let $m \in \mathcal{I}$ represent each piecewise section of the generator cost curve. Then C_g^m represents the cost of the m th interval for generator g , while p_{gt}^m is the power of the m th interval at time t . We let k_g be the cost of generator g at its minimum power and C_g^{SU} be the start up cost. Then 1.1a gives the total operation cost for the set $\mathcal{G}$ of natural gas generators. Because this microgrid has the option to be grid connected, we include cost of importing or exporting power to the grid as C_t^{grid} . Cost for the set of batteries $\mathcal{B}$ is represented as C_{bt} .

$$\min \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} \left[\sum_{m \in \mathcal{I}} C_g^m p_{gt}^m + k_g u_{gt} \right] + \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} C_g^{SU} v_{gt} \quad (1.1a)$$

$$+ \sum_{t \in \mathcal{T}} C_t^{grid} \cdot P_{PCC,t} + \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} C_{bt} (P_{bt}^D + P_{bt}^C) + \delta \quad (1.1b)$$

PV power generation is assumed to have negligible cost, therefore is not included in the objective function. A penalty function for constraint violations is represented with δ in 1.1b, and its terms will be described as they are introduced into the model.

Moving on to constraints, we start with the natural gas generators. Due to the piecewise cost function, we represent total generator power (1.2a) as the sum of each piecewise power interval plus minimum up power P_g^{min} . The binary variable u_{gt} expresses commitment status of generator g at time t . Constraint (1.2b) bounds each piecewise power interval and is further tightened by the binary variables u_{gt}^m which are off when interval p_{gt}^m is not producing power.

$$P_{gt} = \sum_{m \in \mathcal{I}} p_{gt}^m + u_{gt} P_g^{min} \quad (1.2a)$$

$$0 \leq p_{gt}^m \leq p_{gt}^{max,m} u_{gt}^m \quad \forall m \in \mathcal{I} \quad (1.2b)$$

$$u_{gt} - u_{g,t-1} \leq v_{gt} \quad (1.2c)$$

$$0 \leq P_{gt} \leq P_{gt}^{max} u_{gt}, \quad (1.2d)$$

$$\forall g \in \mathcal{G}, t \in \mathcal{T}$$

Constraint (1.2c) indicates time periods when generator g starts up. Lastly constraint (1.2d) ensures total generator power stays below the acceptable power rating for the generator when it is committed. Because this model is designed with intention of application to residential sized microgrids, we assume any generators used will be relatively small (ie, a few hundred kW). Generators of this size can ramp up/down sufficiently fast for ramping constraints to be ignored.

Next we will discuss constraints on battery storage systems in the microgrid. Constraints (1.3a) and (1.3b) bound the real power output when charging or discharging the battery. The battery is limited to either charging or discharging in (1.3c). Energy stored in the battery, E_{bt} , is modeled by 1.3d. Charging and discharging efficiencies are expressed as η_b^C , η_b^D represent respectively. Constraint (1.3e) ensures the energy in the battery does not exceed its storage capacity E_b^{cap} . Total battery power is represented in (1.3f).

$$0 \leq P_{bt}^C \leq P_{bt}^{C,max} u_{bt}^C \quad (1.3a)$$

$$0 \leq P_{bt}^D \leq P_{bt}^{D,max} u_{bt}^D \quad (1.3b)$$

$$u_{bt}^C + u_{bt}^D \leq 1 \quad (1.3c)$$

$$E_{bt} = E_{b,t-1} + P_{bt}^C \eta_b^C \Delta t - P_{bt}^D \frac{1}{\eta_b^D} \Delta t \quad (1.3d)$$

$$0 \leq E_{bt} \leq E_b^{cap} \quad (1.3e)$$

$$P_{bt} = P_{bt}^D - P_{bt}^C \quad (1.3f)$$

$$\forall b \in \mathcal{B}, t \in \mathcal{T}$$

To extend battery life, energy storage systems typically have a minimum and maximum state of charge which should not be exceeded under normal operating conditions. These limits, for example, could be a minimum of 20% and maximum of 80% of the battery's capacity. We will represent these SOC limits as E_b^{min} and E_b^{max} , ie. the minimum and maximum amounts of energy allowed in the battery. If the energy in the battery is below E_b^{min} or above E_b^{max} we consider it to be in an extreme state of charge. In order to prevent the battery from reaching extreme states of charge, a soft penalty was added to the battery model. This included introduction of an auxiliary variable for extreme state of charge, $E_{bt}^{EX} \in \mathbb{R}^+$. This variable is then penalized with the addition of (1.4) to the constraint violation expression δ .

$$\delta_E \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} E_{bt}^{EX} \quad (1.4)$$

Then, the following constraints were added to the model:

$$E_b^{min} - E_{bt} \leq E_{bt}^{EX} \quad (1.5a)$$

$$E_{bt} - E_b^{max} \leq E_{bt}^{EX} \quad \forall b \in \mathcal{B}, t \in \mathcal{T} \quad (1.5b)$$

Note that with constraints (1.5a) and (1.5b), our auxiliary variable E_{bt}^{EX} will only take a value other than 0 if the energy in the battery is above its maximum or lower than its minimum allowable limit. Because this model will be used to control a physical battery, soft

limits are preferable. This way, if the battery enters an extreme state of charge the model will not return infeasible and will instead direct the battery toward an acceptable state of charge.

We also require that overall real power demand D_t^p equals total generation, represented in the energy balance constraint (1.6). We assume that there is no forecasted reactive power for the microgrid because of its residential nature, ie. $D_t^q = 0 \quad \forall t \in \mathcal{T}$. Therefore, reactive power balance constraints are not included.

$$\sum_{g \in \mathcal{G}} P_{gt} + \sum_{b \in \mathcal{B}} P_{bt} + \sum_{s \in \mathcal{S}} P_{st} + P_{PCC,t} = D_t^p \quad (1.6)$$

$$\forall t \in \mathcal{T}$$

It is also assumed that the microgrid has the ability to curtail PV power. Therefore, we only require that the amount of PV power used is less than or equal to its forecasted power at each time interval, which cannot be higher than the PV power rating. This is represented in constraint (1.7), where P_{st}^F denotes forecasted PV power for PV system s at time t .

$$P_{st} \leq P_{st}^F \quad \forall s \in \mathcal{S}, t \in \mathcal{T} \quad (1.7)$$

The model with objective (1.1) and constraints (1.2)-(1.7) describes the complete ED model without transmission constraints.

1.3.2 AC Optimal Power Flow Formulation

This section will describe the AC optimal power flow (ACOPF) formulation which will extend the ED model.

The network constraints are based on a second order cone relaxation seen in [44], known as the branch flow model. Equations (1.8a)-(1.8c) represent the linear dist-flow equations for a radial distribution system, originally developed in [8]. Equations (1.8d) and (1.8e) sum up the total real and reactive power generation at each bus j . The inequality (1.8f) is a conic

relaxation to ensure convexity of the formulation.

$$v_{jt} = v_{it} - 2(r_f P_{f,t} + x_f Q_{f,t}) + (r_f^2 + x_f^2) l_{ft} \quad \forall (i, j) \in \mathcal{E}, t \in \mathcal{T} \quad (1.8a)$$

$$D_{jt}^p - P_{jt} = P_{ij,t} - r_{ij} l_{ij,t} - \sum_{k:(j,k) \in \mathcal{E}} P_{jk} \quad \forall j \in \mathcal{N}, t \in \mathcal{T} \quad (1.8b)$$

$$D_{jt}^q - Q_{jt} = Q_{ij,t} - x_{ij} l_{ij,t} - \sum_{k:(j,k) \in \mathcal{E}} Q_{jk} \quad \forall j \in \mathcal{N}, t \in \mathcal{T} \quad (1.8c)$$

$$P_{jt} = \sum_{b \in \mathcal{B}} P_{bt}^j + \sum_{g \in \mathcal{G}} P_{gt}^j + P_{PV,t}^j + P_{PCC,t}^j \quad \forall j \in \mathcal{N} \quad (1.8d)$$

$$Q_{jt} = \sum_{b \in \mathcal{B}} Q_{bt}^j + \sum_{g \in \mathcal{G}} Q_{gt}^j + Q_{PV,t}^j + Q_{PCC,t}^j \quad \forall j \in \mathcal{N} \quad (1.8e)$$

$$v_i \cdot l_{ij,t} \geq P_{ij,t}^2 + Q_{ij,t}^2 \quad \forall (i, j) \in \mathcal{E}, t \in \mathcal{T} \quad (1.8f)$$

$$l_{ij,t} \leq (I_{ij}^{max})^2 \quad \forall (i, j) \in \mathcal{E} \quad (1.8g)$$

$$(v_{min})^2 \leq v_{jt} \leq (v_{max})^2 \quad \forall j \in \mathcal{N}, t \in \mathcal{T} \quad (1.8h)$$

In an effort to minimize line losses we add (1.9) to the constraint violation expression δ , where δ_l represents the penalty parameter for the lines.

$$\delta_L \cdot \sum_{f \in \mathcal{E}} \sum_{t \in \mathcal{T}} r_f l_{f,t} \quad (1.9)$$

Constraints must also be added to bound apparent power on the batteries, generator, and phase angle for the generator.

$$P_{bt}^2 + Q_{bt}^2 \leq S_b^2 \quad \forall b \in \mathcal{B}, t \in \mathcal{T} \quad (1.10)$$

$$-\tan(\theta_i) P_{gt} \leq Q_{gt} \leq \tan(\theta_i) P_{gt} \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \quad (1.11)$$

$$P_{gt}^2 + Q_{gt}^2 \leq S_g^2 \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \quad (1.12)$$

We will now describe the linearization techniques used for the nonlinear AC power flow constraints. First, the squared terms in constraints (1.12), (1.10) and the right hand side of (1.8f) were piecewise linearized. The bilinear term $v_i \cdot l_{ij,t}$ in constraint (1.8f), which is nonconvex, was relaxed using the McCormick envelopes. To perform this relaxation, the

variable y_{it} was added, where $y_{it} = v_i \cdot l_{ij,t}$. The inequality (1.8f) becomes:

$$y_{it} \geq P_{ij,t}^2 + Q_{ij,t}^2 \quad \forall (i, j) \in \mathcal{E}, t \in \mathcal{T} \quad (1.13)$$

Then, let the notation $\underline{v}$ and $\bar{v}$ denote the lower and upper bound of variable v , respectively. Also note $\underline{l}_{ij,t} = 0$, causing these terms to cancel out in the McCormick estimators. Then we have constraints (1.14a)-(1.14d) added to the model.

$$\underline{v}_i l_{ij,t} \leq y_{it} \quad (1.14a)$$

$$\bar{v}_i l_{ij,t} + v_i \bar{l}_{ij,t} - \bar{v}_{it} \bar{l}_{ij,t} \leq y_{it} \quad (1.14b)$$

$$y_{it} \leq \bar{v}_i l_{ij,t} \quad (1.14c)$$

$$y_{it} \leq v_i \bar{l}_{ij,t} + \underline{v}_i l_{ij,t} - \underline{v}_i \bar{l}_{ij,t} \quad (1.14d)$$

$$\forall (i, j) \in \mathcal{E} \quad t \in \mathcal{T}$$

The model ED plus constraints (1.8a)- (1.8e), (1.8g), (1.8h), (1.10)-(1.14) will be known as ED+ACOPF. The linearized convex relaxation of the ACOPF allows for us to give reactive setpoints to devices in the microgrid while maintaining line limits. A concern of using this convex relaxation is that the result will not yield an AC feasible solution. In practice, however, due to forecasting errors and other unforeseen uncertainties in the system, the devices must already adjust for imperfect setpoints in order to maintain energy balance and voltage requirements within the microgrid. It is our experience that the devices in the microgrid are able to overcome errors in the ACOPF solution in the same manner as overcoming forecasting errors. During testing, the ED+ACOPF model has always produced a solution which can be implemented without incurring a fault in the microgrid. Nonetheless, we acknowledge that for larger systems or mesh networks, a more robust ACOPF algorithm may be necessary to achieve feasibility and avoid faults in the network.

1.4 Application Motivated Constraints

1.4.1 Microgrid Setup

The physical microgrid contains three devices for providing distributed generation. The first is a 300kW Li-ion battery with 680 kWh storage capacity. A total of 330kW of PV generation and a 400kW natural gas generator were also installed. An overview of the device parameters is shown in Table 1.2. For purposes of this study, all are assumed to be on one bus as they are more or less collocated. The PCC is assumed to be on its own bus. The third bus contains the 62 residential homes on three phases. The components of the microgrid are arranged so that the system can be islanded from the main grid, and subsequently resynchronized.

The ED+ACOPF model was first tested in a simulation configuration. In this configuration, the CSEISMIC microgrid controller receives the 24-hour ED+ACOPF solution, and uses the first time period as the real-time setpoints for the devices. These setpoints are then sent to the Real-Time Digital Simulator where an electrical simulation of the microgrid is run. Before deployment, simulations were run in a variety of test cases in order to identify possible issues with implementation. These issues and their solutions will be outlined in the rest of section 1.4.

1.4.2 Battery Cycling

Simulations were run with a variety of price curves representing cost of buying/selling at the PCC. The solutions for square wave prices returned large number of battery cycles. Because of durations which span multiple time periods with a flat price signal, there exist multiple

Table 1.2: Device Parameters

Device	Power Rating	Minimum Power	Energy Rating
Generator	400 kVA	100 kW	-
PV	330 kW	0 kW	-
Li-Ion Battery	300 kVA	0 kW	680 kWh

optimal solutions with slightly different battery charging/discharging schedules, yet identical objective values. As the MCC has no control over the price curve sent by the utility, we cannot guarantee whether or not this situation will occur. Additionally, because frequent cycling will shorten the lifetime of the battery, we need to take steps to limit battery cycling in the case of a relatively flat utility price forecast.

In order to limit battery cycling, battery setpoint deviation in consecutive time intervals was penalized. First, we added an auxiliary variable P_{bt}^V to represent the absolute value the deviation in battery power in from one time period to the next. This variable was then penalized by adding (1.15) to the constraint violation expression δ .

$$\delta_V \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} P_{bt}^V \quad (1.15)$$

Next, constraints of the form (1.16) were included in the model.

$$P_{bt} - P_{b,t-1} \leq P_{bt}^V \quad (1.16a)$$

$$P_{b,t-1} - P_{bt} \leq P_{bt}^V \quad (1.16b)$$

The result of adding constraints (1.16) was the battery started using the minimal number of cycles needed in its charge/discharge schedule, as the MCC is incentivized to group the battery's charging or discharging into consecutive time periods. The effects of incentivizing less battery cycling can be seen by comparing Figure 1.4 with Figure 1.5. Figure 1.4 shows the real power dispatch of the battery from 8 a.m. to just after 4 p.m. on June 28, 2018, when frequent cycling of the battery was observed. After introduction of the penalty, the battery was more likely to cluster the time periods in which it was charging or discharging, as in Figure 1.5

1.4.3 Generator Minimum up-time

So far, the generator does not have minimum up-time requirements. Lack of minimum up-time can cause frequent commitment changes thereby causing extra wear and increasing maintenance cost. During off-grid simulations with no up-time requirement, solutions were

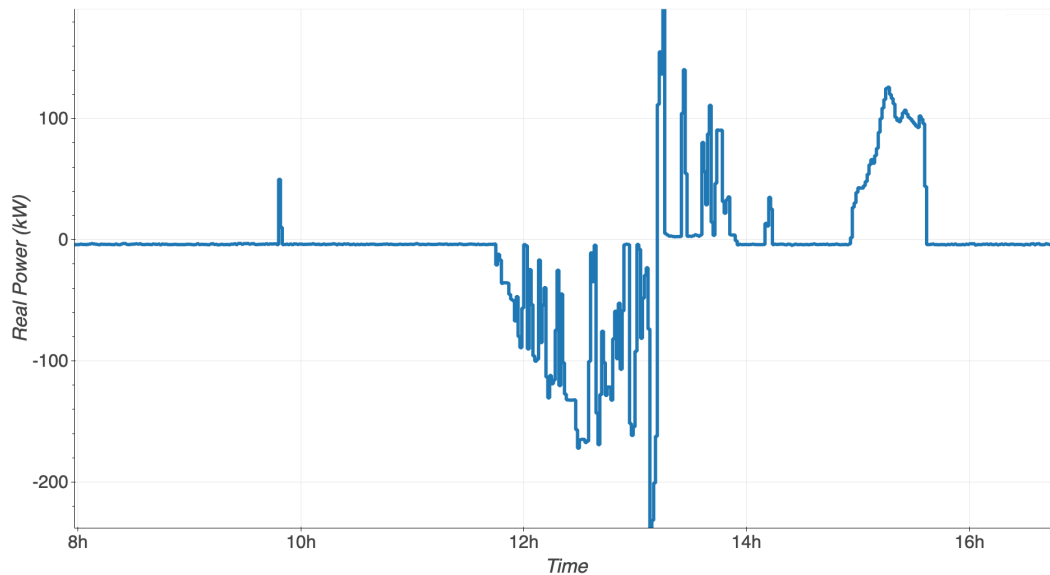


Figure 1.4: Battery real power dispatch on 06/28/18 when frequent cycling presented as an issue

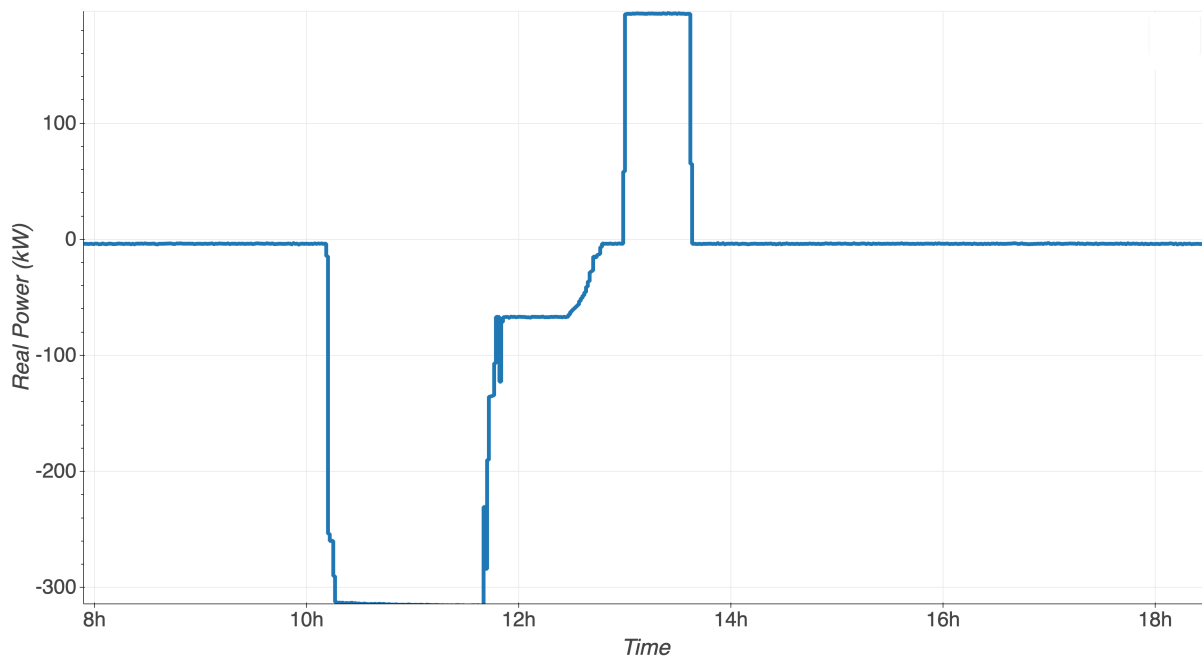


Figure 1.5: Battery real power dispatch on 11/27/18 after battery cycling was penalized

returned with frequent changes in generator commitment status. Because our time intervals are only 5 minutes long, these commitment changes can happen as many as 12 times in an hour. Addition of minimum up-time will eliminate these types of solutions.

Also, we want to exclude certain extreme cases that may happen while on-grid. In Figure 1.6, right before hour 16 (4:00pm), the generator turned on for one time interval, and then off. This is undesirable behavior, both because of the quick commitment change, and because usually it is not cost-effective to run the generator on-grid. Adding a minimum up-time would potentially avoid the generator being turned on due to added expense, or we can use the generator if necessary but avoid quick commitment changes.

Because of the simulation results, it was determined that minimum up-time requirements were necessary for the model. Let UT_g represent the number of time intervals which constitute the minimum up-time requirement for generator g . Then, we let R_g represent the number of time intervals remaining for minimum up-time if generator g was on before the first time interval. Equation (1.17) defines T_g^{break} as the time interval which is the sum of minimum up time plus intervals remaining from previous operation.

$$T_g^{break} = UT_g + R_g \quad \forall g \in \mathcal{G} \quad (1.17)$$

Next we add constraints (1.18) for all $t \in \mathcal{T}$. If generator g was not previously on, we have $R_g = 0$ and only conditions for constraints (1.18b) and (1.18c) are met. Otherwise, (1.18a) ensures the generator remains on at the beginning for as many time periods as necessary before it is permitted to shut down.

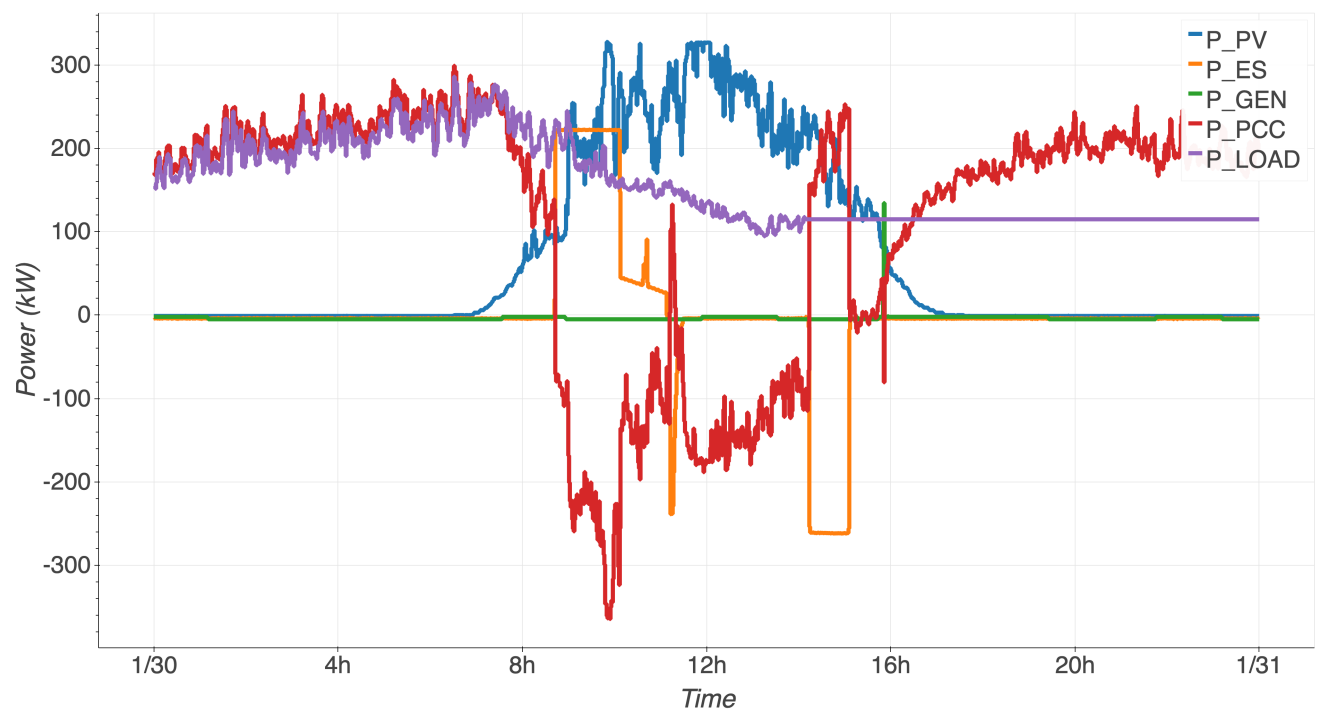


Figure 1.6: Economic Dispatch January 30, 2019

if $t < R_g$:

$$u_{gt} = 1 \quad (1.18a)$$

if $R_g \leq t \leq T_g^{break}$:

$$\sum_{i=R_g}^t v_{gi} \leq u_{gt} \quad (1.18b)$$

if $t > T_g^{break}$:

$$\sum_{i=t-UT_g}^t v_{gi} \leq u_{gt} \quad (1.18c)$$

1.4.4 Reactive Power Production Limits

During simulation, various cases returned optimal solutions with high reactive power setpoints on either the battery, generator, or PCC (recall PV only receives a real power setpoint). Because the microgrid is residential there is no reactive power demand and therefore the utility is not making any money off of reactive power. Thus, high reactive power setpoints are undesirable from a profit perspective.

We would prefer to arrive at a solution which meets the AC power flow constraints yet minimizes use of reactive power. The first step to implement this is adding variables $Q_{b,t}^{abs}$, $Q_{g,t}^{abs}$, $Q_{PCC,t}^{abs}$ to represent absolute value of reactive power for batteries, generators, and at

the PCC. Next, constraints (1.19) were added to the model.

$$Q_{b,t} \leq Q_{b,t}^{abs} \quad \forall b \in \mathcal{B}, t \in \mathcal{T} \quad (1.19a)$$

$$-Q_{b,t} \leq Q_{b,t}^{abs} \quad \forall b \in \mathcal{B}, t \in \mathcal{T} \quad (1.19b)$$

$$Q_{g,t} \leq Q_{g,t}^{abs} \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \quad (1.19c)$$

$$-Q_{g,t} \leq Q_{g,t}^{abs} \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \quad (1.19d)$$

$$Q_{PCC,t} \leq Q_{PCC,t}^{abs} \quad t \in \mathcal{T} \quad (1.19e)$$

$$-Q_{PCC,t} \leq Q_{PCC,t}^{abs} \quad t \in \mathcal{T} \quad (1.19f)$$

Then, to penalize use of reactive power, (1.20) was added to the penalty function δ . The values δ_B , δ_G , and δ_C are the penalty parameters for the battery, generator, and PCC respectively.

$$\delta_B \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} Q_{b,t}^{abs} + \delta_G \cdot \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} Q_{g,t}^{abs} + \delta_C \cdot \sum_{t \in \mathcal{T}} Q_{PCC,t}^{abs} \quad (1.20)$$

1.4.5 Off-grid Curtailment and VF Constraints

When operating in the off-grid mode, large transients in renewable energy output can put strain on maintaining voltage levels in the grid. This is especially a concern when a device with a slow ramp rate may not be fast enough to make an adjustment to its power output in time to cover demand or reduce excess generation. In order to decrease the likelihood of this scenario, we employ constraints to curtail PV power production in the off-grid state. Further, we employ a harsher curtailment if the battery is at its maximum SOC, because there will be no way to absorb excess PV production in this case as there is only one battery in the microgrid.

To formulate the curtailment constraints, the parameter α was added to represent the maximum allowable PV setpoint. Thus, if PV production is greater than α it must be curtailed to this level. Similarly, γ was added as the maximum allowable PV setpoint when the battery is unable to charge. Also introduced was the new binary variable x_{bt} , which takes the value 1 if the battery is unable to charge. The constraints (1.21) and (1.22) were then

added to the model. Constraint (1.21) sets the binary variable x_{bt} to 1 when the battery cannot charge, so that the PV curtailment upper bound will be set in constraint (1.22).

$$E_{bt} \leq E_{bt}^{max} + (1 - E_{bt}^{max}) \cdot x_{bt} \quad \forall t \in \mathcal{T} \quad (1.21)$$

$$P_{bt} \leq \alpha(1 - x_{bt}) + \gamma \cdot x_{bt} \quad \forall s \in \mathcal{S}, \quad \forall t \in \mathcal{T} \quad (1.22)$$

In the off-grid state, either the natural gas generator or battery is required to be the VF device, i.e. provide voltage support. If the generator is the VF device, constraint (1.23) must be enabled to ensure that the generator is operating for the entire forecast duration.

$$u_{gt} = 1 \quad \forall t \in \mathcal{T} \quad (1.23)$$

1.4.6 Full Model Discussion

We have now described the full economic dispatch model ED-FULL. The objective for ED-FULL is (1.1), which minimizes operational cost, and also includes the constraint violation term δ which is made up of (1.4), (1.9), and (1.20), ie:

$$\begin{aligned} \delta = & \delta_E \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} E_{bt}^{EX} + \delta_V \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} P_{bt}^V + \delta_L \cdot \sum_{f \in \mathcal{E}} \sum_{t \in \mathcal{T}} r_f l_{f,t} \\ & + \delta_B \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} Q_{b,t}^{abs} + \delta_G \cdot \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} Q_{g,t}^{abs} + \delta_C \cdot \sum_{t \in \mathcal{T}} Q_{PCC,t}^{abs} \end{aligned} \quad (1.24)$$

The constraints for ED-FULL consist of the model ED+ACOPF plus the application motivated constraints discussed in section 1.4. In total, the model ED-FULL is:

$$\begin{aligned} \min & (1.1) \text{ where } \delta = (1.24) \\ \text{st.} & (1.2), (1.3), (1.5) - (1.7), \\ & (1.8a) - (1.8e), (1.8g), (1.8h), (1.10) - (1.14) \\ & (1.16) - (1.19), (1.21) - (1.23) \end{aligned} \quad (1.25)$$

Figure 1.7 shows successful economic dispatch measurements sampled at 1 minute intervals for December 12, 2018. This day was a cloudy day, with PV power never above

150kW, which is less than half the energy rating of 330kW for the PV panels. We can observe that most of the load was served from the PCC for this day. If a low PV day was to occur during an islanding event, the generator would need to take over any of the power which cannot be served by the battery and would have been imported by the PCC otherwise.

March 4, 2019 had much higher PV production than December 12, which can be seen in Figure 1.8. We can observe from Figure 1.8 is that because we are grid connected the PCC was used to follow demand instead of the battery during times of low PV power production. The economic dispatch for this day also put an emphasis on selling PV power to the grid during times of high PV production (approx. hr 12-16 in Figure 1.8). This can be seen by comparing the PV production to the PCC power shown in Figure 1.9.

1.5 Pricing Algorithm

For islanded microgrids, determining electricity prices is crucial for operation in a residential setting with multiple consumers. Ideally, the microgrid central controller (MCC) would be able to have perfect information for the assets in the grid in order to settle the market. An issue when considering real world microgrid implementations, however, is that consumer privacy must be addressed. In some situations, customers may not be comfortable sharing individualized information about their load consumption with a Microgrid Central Controller. They may want to keep the configurations of their controllable devices a secret, possibly along with their individual load profile. When consumers are using their own optimization scheme to shift their load to periods of low price, the overall load profile of the system is affected, which puts a challenge on setting an appropriate price forecast for the system.

In the Hoover, Alabama microgrid, each home is equipped with their own energy management system (EMS) which optimizes their own load profile over 24 hours. This EMS will dispatch optimal HVAC and water heater setpoints while maintaining home comfort constraints. For the purposes of the MCC, we treat these home optimizations as a black box, where we do not have any information on individual models, nor access to individual

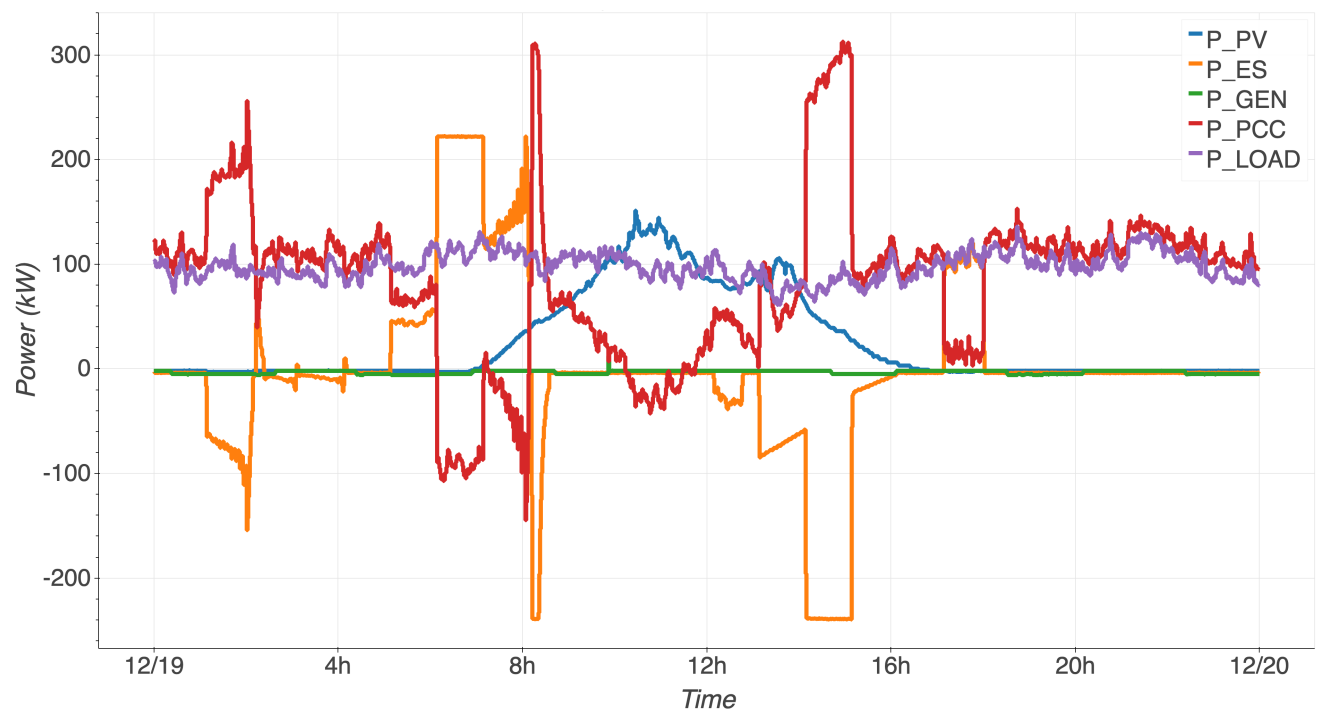


Figure 1.7: Economic Dispatch December 19, 2018

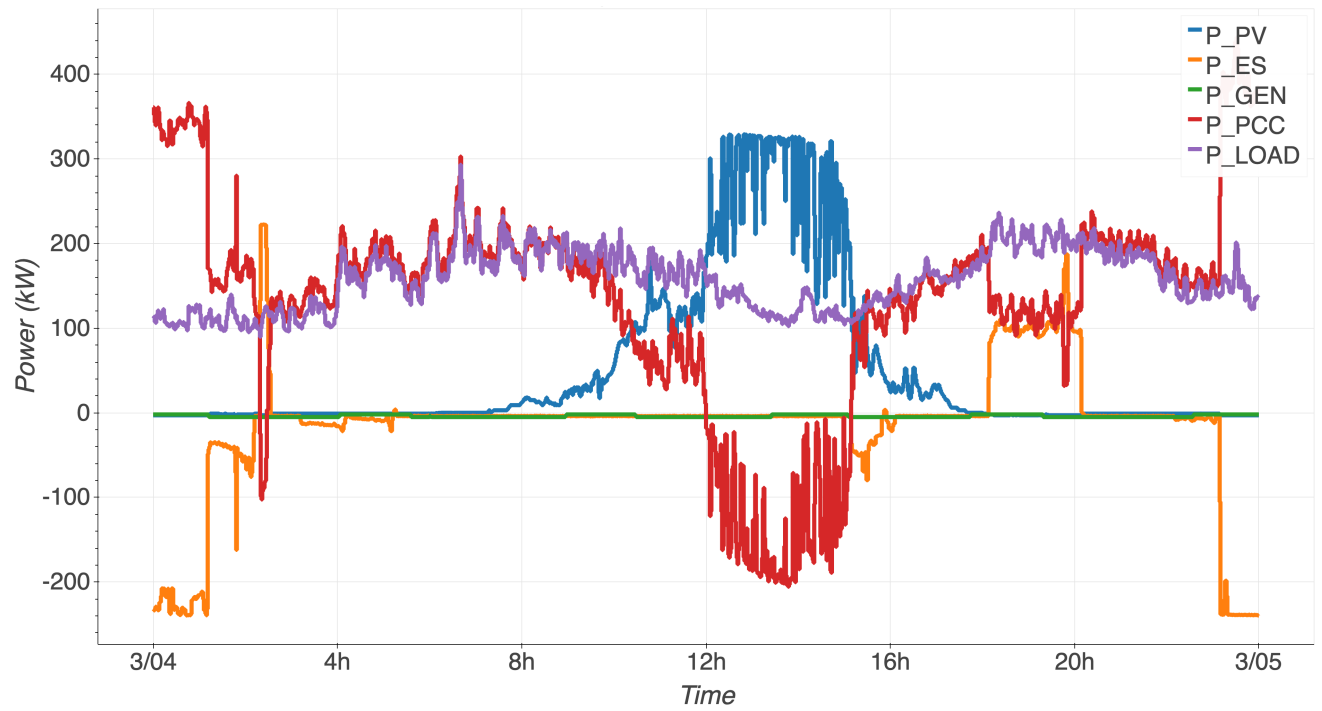


Figure 1.8: Economic Dispatch March 4, 2019

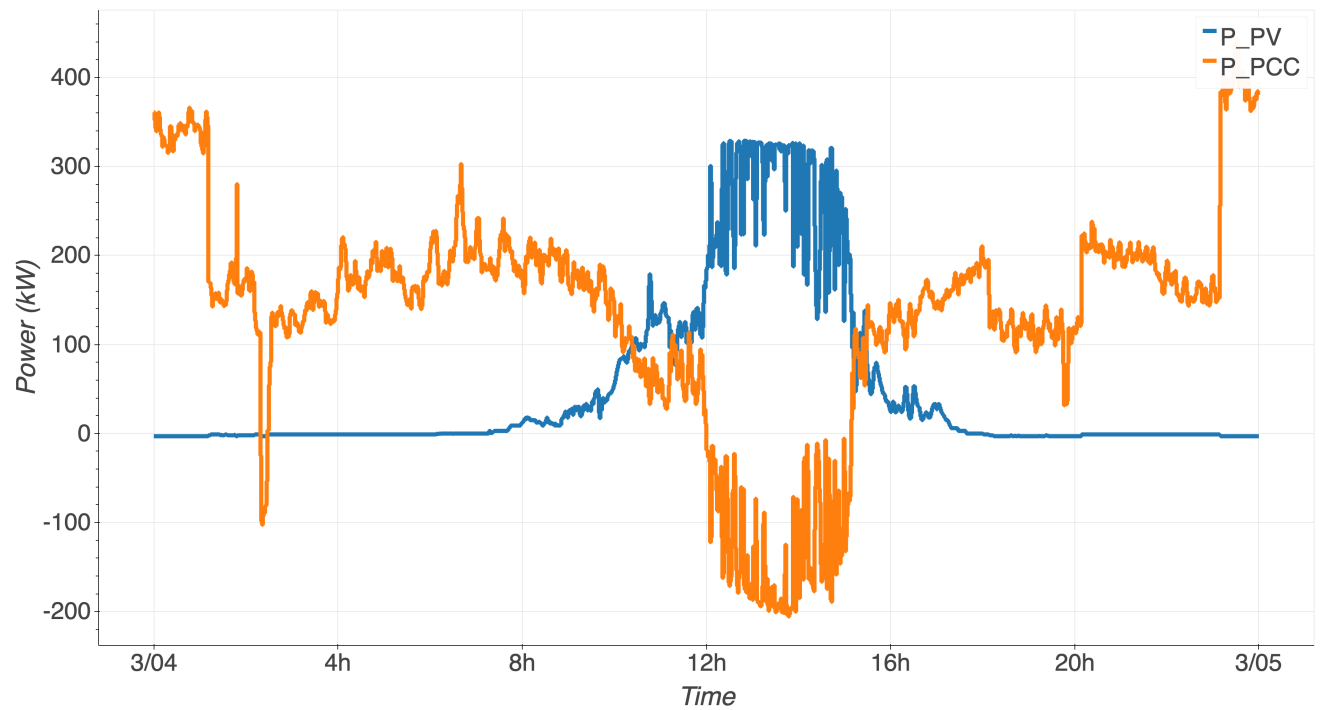


Figure 1.9: PCC power March 4, 2019

load profiles. Instead, home load is grouped by phase and each phase's aggregate 24 hour load forecast is sent to the CSEISMIC controller.

The pricing algorithm will use the per-phase load forecast to help decide an appropriate price forecast for the next 24 hours. Note that this optimization scheme allows the microgrid and homes to optimize separately. This method also allows for each home to independently schedule their HVAC control, rather than an MCC making this decision on their behalf. Therefore, because homes are allowed to act selfishly, forecasted load will not reflect the global optimal wherein the MCC had full knowledge of the microgrid. However, instead of the MCC accepting the first demand curve offered, an iterative algorithm will be implemented to influence demand. Also, the distributed nature allows for homes to individually add controllable devices to their optimization scheme in the future.

1.5.1 Algorithm Description

The goal of the algorithm is to determine an appropriate microgrid price forecast for the next 24 hours. To decide the price, the microgrid sends a price signal for each time period to the homes. We will call this price signal μ_t^k , where t represents the time period associated with the price, and k represents the iteration of the algorithm. The homes (load side) will use μ_t^k to update their load forecast and send the result back to the microgrid controller. The algorithm continues until a convergence criteria is met. An outline of the pricing algorithm is shown in Figure 1.10.

To describe the algorithm in more detail, we define Δ_t^k in (1.26) as the difference between forecasted load at iterations k and $k - 1$.

$$\Delta_t^k = D_t^k - D_t^{k-1} \forall t \in \mathcal{T} \quad (1.26)$$

To determine convergence we set a maximum value β_t for each $|\Delta_t^k|$ (in kW). The algorithm terminates when (1.27) is satisfied for all time periods. Since β_t can be unique for each time period, earlier time periods can be held to a more strict criterion than later time

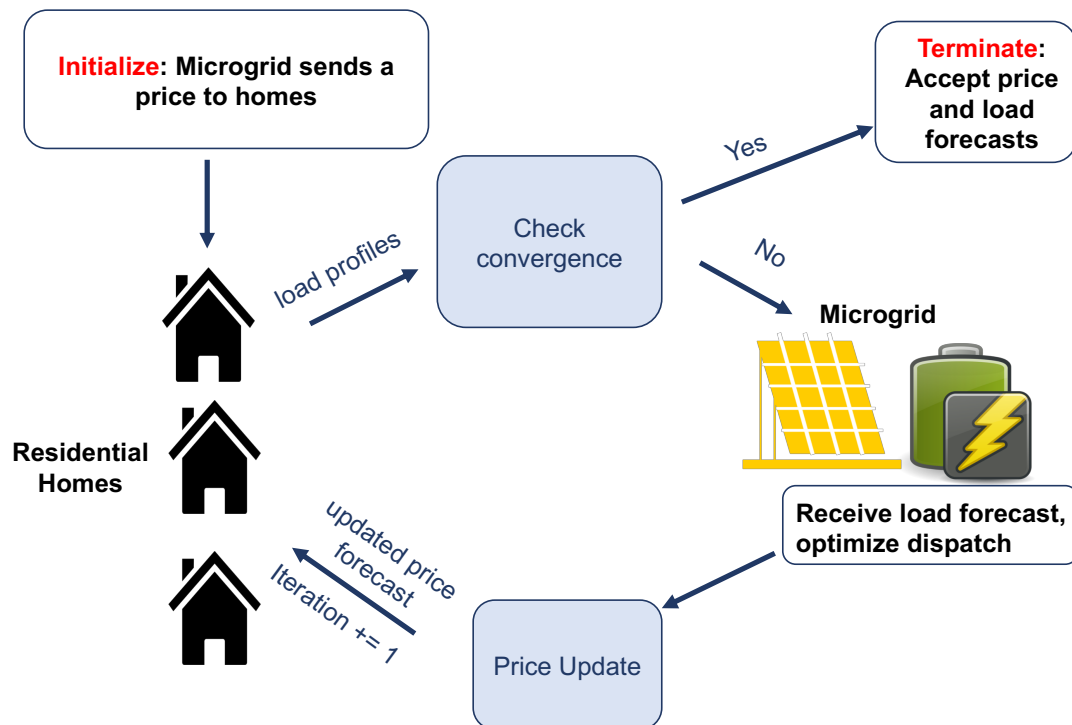


Figure 1.10: Pricing Algorithm Overview

periods in the forecast.

$$|\Delta_t^k| < \beta_t \forall t \in \mathcal{T} \quad (1.27)$$

As previously mentioned, each home's EMS will be a black box optimization wherein the MCC has only access to aggregate load forecasts for each phase. On the other hand, the microgrid side model used will be fully available to the MCC. The constraints will use real power only, therefore no AC power flow constraints are included. We will use the already described ED model in sections 1.3.1 and 1.4 to define the model ED-Price (1.28). In this model, the constraint violation term δ consists solely of (1.4) and (1.15).

$$\begin{aligned} \min \quad & (1.1) \\ \text{st.} \quad & (1.2), (1.3), (1.5) - (1.7), \\ & (1.16) - (1.18), (1.21) - (1.23) \end{aligned} \quad (1.28)$$

The pricing algorithm will begin with an initialization of all device parameters, as well as ambient temperature and PV forecasts. An initial microgrid price forecast $\mu_t^0 \forall t \in \mathcal{T}$ can be determined in two ways. The first way of determining price forecast is as follows:

1. Solve the MIP (1.28).
2. Fix all integer variables and solve the corresponding linear program.
3. Obtain the values of the corresponding dual variables for the power balance constraint (1.6), call them ν_t .
4. Let $\mu_t^0 = \nu_t \forall t \in \mathcal{T}$

Note that each ν_t should be the marginal cost of electricity for the microgrid at time t given initial demand D_t^0 . The second way to determine an initial price is simply to use a price curve provided by the utility. While this may not necessarily represent actual cost of electricity at each time period, the utility may prefer to have more control over factors like shape of the curve and range of prices. Additionally, if the microgrid has oversized

renewables, a marginal cost method may return $\mu_k^t = 0$ for multiple time periods t . A zero price is undesirable for the utility and can be mitigated by the second method.

Algorithm 1 describes the basic pricing algorithm with convergence criteria (1.27). We initialize u_t^0 , the initial price forecast. Let k represent the iteration number, and the quantity MAXITER be the maximum allowable iterations.

Algorithm 2 Pricing Algorithm

```

1: Initialize  $\mu_t^0 \forall \mathcal{T}, k = 0$ 
2: while Converge = False do
3:    $k = k + 1$ 
4:   Homes run their individual optimization with price  $\mu_t^k$  and return an updated forecast
5:   Model (1.28) performs a feasibility check on the load forecast  $D_t^p$ .
6:   for  $t \in \mathcal{T}$  do
7:      $\mu_t^k = \mu_t^{k-1} + \rho \cdot \Delta_t^k$ 
8:     if  $|\Delta_t^k| < \beta_t \forall t \in \mathcal{T}$  then
9:       Converge = True
10:    else if  $k \geq \text{MAXITER}$  then
11:      Break

```

1.5.2 Issues and fix

As one can imagine, in a pricing scheme all homes schedule their controllable load based on the time intervals with the lowest price. And in a pricing scheme like Algorithm 2, they all have the same price curve. Therefore homes all schedule this controllable load at very similar or even identical time intervals. Homes taking these similar or identical actions leads to load spikes at periods of low price, and low load at periods of high price. Moreover, due to the construction of the algorithm, the load spiking behavior will lead to divergence in the algorithm. Consider the example of two time intervals, 1 and 2, where time period 1 has low price and time period 2 has a high price. When each μ_t is updated, μ_1 will increase because the load will spike there, and μ_2 will decrease. Then at the next iteration, the homes will shift their load to time period 2 instead. Time period 2 will now have a load spike and need to increase the price, which is the opposite action of the previous iteration. Unfortunately this behavior will create a “flip-flop” type action, where the controllable load spikes will be passed around between time periods. A graphical example of this is seen in Figure 1.11,

which shows time periods 50-70 for the 5th and 6th iterations of Algorithm 2. We can see that iteration 5 and 6 look opposite of each other. The algorithm has started oscillating between two solutions which alternate load spikes.

In order to mitigate load spikes and “flip-flops”, we will instead send a different price signal to each home in the microgrid. The individualized pricing will incentivize homes to spread out their load instead of the aggregating effect which caused load spiking. On each iteration of the algorithm, the homes will now update their individualized price. The price signal for each home will be a slight perturbation of the microgrid price. Call this perturbation $\epsilon_{t,h}$. At each iteration k , every home’s unique price $\lambda_{t,h}$ will be defined as (1.29) for every time interval in the forecast horizon.

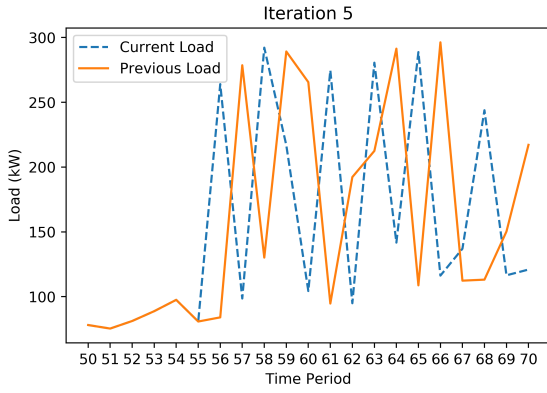
$$\lambda_{t,h}^k = \lambda_{t,h}^{k-1} + \epsilon_{t,h} + \rho \cdot \Delta_t^k \quad \forall t \in \mathcal{T}, h \in \mathcal{H} \quad (1.29)$$

From this new price update we can construct our novel individualized pricing algorithm, Algorithm 3, in which each home has a unique price update at each iteration. We let MAXITER be the number of maximum iterations before algorithm termination. The maximum number of iterations will be determined by length of each time interval and computation time for one iteration of the algorithm.

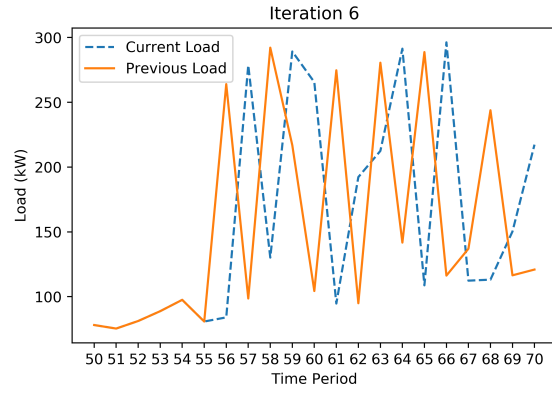
Algorithm 3 Individualized Pricing Algorithm

```

1: Initialize  $\mu_t^0, \lambda_{t,h}^0 \forall \mathcal{T}, h \in \mathcal{H}, k = 0$ 
2: while Converge = False do
3:    $k = k + 1$ 
4:   Homes run their individual optimization with price  $\mu_t^k$  and return an updated forecast
5:   Model (1.28) performs a feasibility check on the load forecast  $D_t^p$ .
6:   for  $t \in \mathcal{T}$  do
7:      $\mu_t^k = \mu_t^{k-1} + \rho \cdot \Delta_t^k$ 
8:      $\lambda_{t,h}^k = \lambda_{t,h}^{k-1} + \epsilon_{t,h} + \rho \cdot \Delta_t^k$ 
9:
10:  if  $|\Delta_t^k| < \beta_t \forall t \in \mathcal{T}$  then
11:    Converge = True
12:  else if  $k \geq \text{MAXITER}$  then
13:    Break
```



(a) Iteration 5



(b) Iteration 6

Figure 1.11: Two consecutive iterations of Algorithm 2

1.5.3 Simulation Results

To obtain PV and initial load forecasts for simulation, measurements were collected from the microgrid for the week of August 16-20, 2018. The data was averaged over 15-min intervals for a total of 96 time periods. Figure 1.12 shows a sample price for a single home compared to the microgrid price forecast after convergence of the algorithm for August 18, 2017. The algorithm converged after 3 iterations for this day, with $\epsilon \in [-.05, .05]$, $\rho = .0001$.

Figure 1.13 shows the load profiles from the pricing algorithm without individual price updates, with individual price updates, and a combined model for August 20, 2017. The individualized price for this day was shown in Section 1.1, Figure 1.3. The combined model in Figure 1.13 integrates the home optimization into the microgrid model as if the microgrid could control the neighborhood's assets. The combined model uses a one-time price which is the same for every home in the neighborhood. From Figure 1.13, we can see how iteration without individualized pricing (Algorithm 2) ends up with a spiky load that is worse off than the combined model. In fact, without individualized pricing the algorithm did not converge, and the figure simply shows the last iteration's result. The individualized pricing (Algorithm 3) outperforms even the combined model by reducing the peak load and providing a more uniform curve.

1.6 Conclusion

Chapter 1 has presented an economic dispatch model and pricing algorithm for controlling a physical microgrid constructed in Hoover, Alabama. Battery energy storage systems, PV power, and natural gas generators were considered as components in the microgrid. The model included an ACOPF relaxation to attempt to find AC feasible solutions, which provided reasonable estimates in practice. Constraints were included to mitigate problems which arose during testing, such as frequent battery cycling. The economic dispatch model has been able to provide a sufficient solution for the MCC over a period of months, during both grid-connected and islanded operation. The solutions have been used to successfully control the devices without causing any unintentional outages in the microgrid. The control

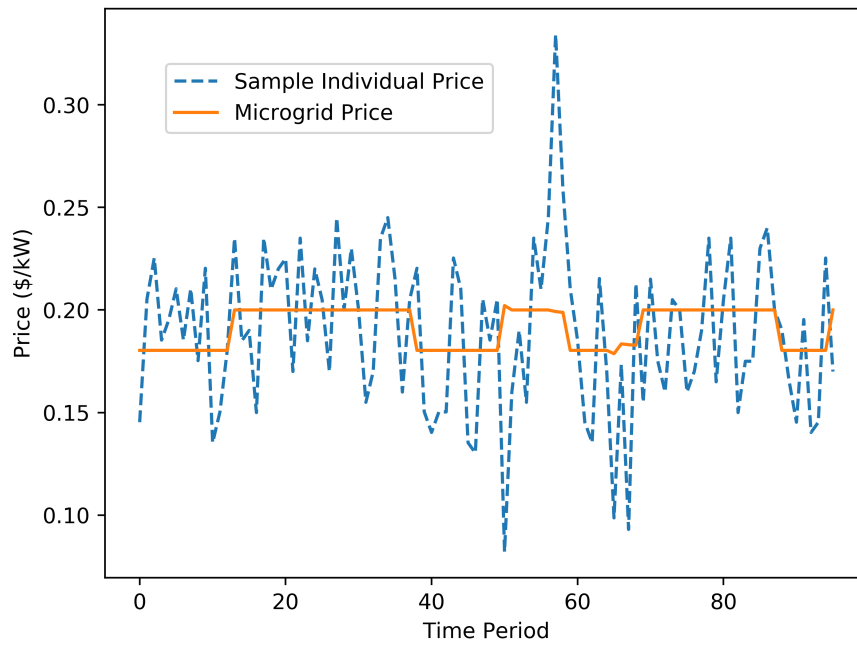


Figure 1.12: Sample single home solution from August 18 test case

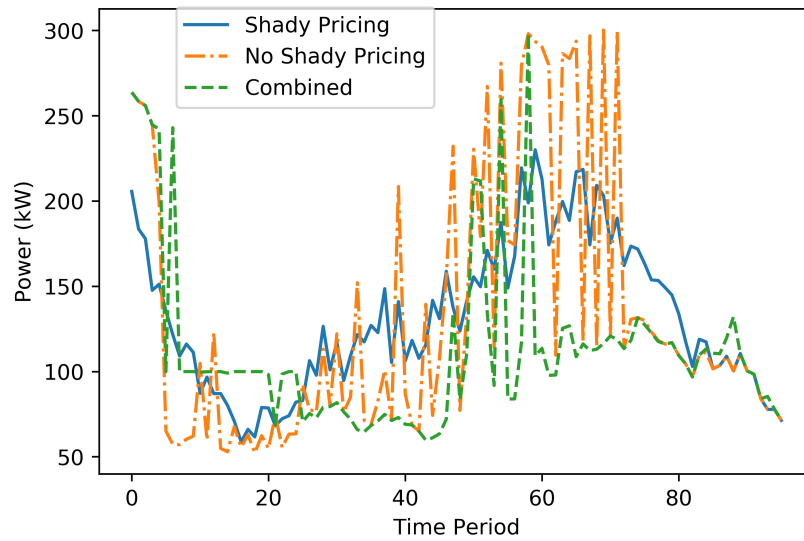


Figure 1.13: Load Profiles August 20, 2017

scheme with optimization was even able to mitigate an unintentional outage of the main grid on October 22, 2018, and successfully switch to island mode.

The pricing algorithm developed has provided a baseline for constructing iterative demand response algorithms. The baseline algorithm showed a need for at least a semi-individualized pricing scheme to avoid consumers making a similar reaction to a uniform price. To overcome this we designed an individual pricing scheme which would perturb prices slightly for each consumer in order to incentivize a more distributed controllable load. Test cases show proof of concept for individualized pricing schemes by successfully avoiding load spiking and reducing peak load.

Chapter 2

Reducing the Computational Burden of a Microgrid Energy Management System

Chapter 2 is based on a paper prepared for publication by the authors Amelia McIlvenna, Andrew Herron, Joshua Hambrick, Ben Ollis, and James Ostrowski.

McIlvenna, A., Herron, A., Hambrick, J., Ollis, B., Ostrowski, J. (2019). Reducing the Computational Burden of a Microgrid Energy Management System. *Submitted*.

Authors McIlvenna and Ostrowski developed the relaxation method based on a real-world need for improving computational time of the model. Author McIlvenna implemented the model, ran computational simulations which are included in the manuscript, and wrote the manuscript. Authors Herron, Hambrick, and Ollis assisted in real-time operation of the model at the physical microgrid location for which it was designed, as well as collecting data from the microgrid. Author Ostrowski edited the manuscript.

2.1 Motivation

Chapter 1 presented a real-world microgrid system which contained a microgrid central controller (MCC) with optimization capabilities. Each time interval in the optimization is 5 minutes in length, and therefore any optimization models must solve within this time

window. The microgrid in Chapter 1, however, required use of open-source software due to financial constraints. Open-source solvers tend to have longer computation time than standard commercial software. When the MCC was deployed, there were issues solving within the required 5-minute window. These real life issues lead to the development of a relaxation technique for rolling time horizon optimization systems developed in this chapter. The techniques add robustness and decrease computational burden by eliminating integer variables. Chapter 2 is a paper developed for publication which presents the relaxation developed, and also tests the model on severely limited computational hardware which might be used in a typical microgrid system.

2.2 Abstract

As renewable technology advances and decreases in cost, microgrids are becoming an appealing means of distributed generation both for isolated communities and integrated with existing electrical grid systems. Due to their small size, however, microgrids may have financial limitations which preclude them from using commercial software to optimize control of their assets. Open-source optimization solvers are a viable alternative, but increase computation time. This work expands on a rolling horizon optimization framework for economic dispatch within an existing residential microgrid located in Hoover, Alabama. The microgrid has an open-source solver requirement and a need for quick solution time on a rolling horizon as opposed to a day-ahead commitment. We present a method of reducing integer variables by relaxation which completes two goals: reduction in computation time for real-time operations, and reduction in daily operational cost for the microgrid. Seasonal data for load and photovoltaic (PV) power was also collected from the microgrid to facilitate simulation testing. Computation time was successfully reduced using multiple variations of the relaxation method, while obtaining solution quality with operational cost similar to or better than the original model.

2.3 Nomenclature:

2.3.1 Variables

P_{gt}	Total power at each generator
P_{bt}^C, P_{bt}^D	Battery charging/discharging power, respectively
p_{gt}^m	Power at m th generator interval
u_{gt}	Generator up/down status
v_{gt}	Generator start up indicator
u_{bt}^C, u_{bt}^D	Battery charging/discharging status, respectively
P_{st}	PV power which is not curtailed
E_{bt}	Energy in battery b at time t

2.3.2 Parameters

k_g	minimum up cost of generator g
C_{gt}^m	Marginal cost for m th generator interval
C_{gt}^v	Start up cost for generator g
C_{bt}	Battery operational cost
P_g^{min}	Generator min up power
D_t	Real demand at time t
P_{st}^F	Forecasted PV power at time t

2.3.3 Sets

$g \in \mathcal{G}$	Set of generators
$b \in \mathcal{B}$	Set of batteries
$s \in \mathcal{S}$	Set of PV panels
$i \in \mathcal{N}$	Set of nodes (buses)
$m \in \mathcal{I}$	Set of piecewise generator power components
$t \in \mathcal{T}$	Set of timeperiods

2.4 Introduction

With renewable energy adoption on the rise and a need for network expansion, modifications are being made to convert electrical grid infrastructure into smart technology. One technology which is receiving significant attention is the installation of microgrids. Microgrids are low-voltage networks which facilitate local generation on a relatively small scale, allowing electricity generation to be located close to its consumers [32]. An advantage of microgrids is that they can electrify remote locations for which grid connection is too costly due to geographical barriers [36]. Microgrids can also be connected to an existing network where they provide an additional layer of reliability and distributed generation. Microgrids have the ability to provide extra generation, load reduction, and frequency and voltage support for the main electrical grid. During severe outages caused by extreme weather, microgrids can help reduce the impact of outages, and provide support for service restoration [18]. While integrating microgrids with the existing network has the benefits of reliability and local generation, there are controls and optimization issues which can arise [60]. These issues are usually handled via either a microgrid central controller (MCC) to provide management for a decentralized layout or a centralized energy management system (EMS) [72, 55, 70]. Within the EMS is typically an optimization framework which can handle unit commitment (UC) and economic dispatch (ED) decisions on either a day-ahead schedule or updating on a regular basis throughout the day.

Nowadays large systems such as PJM [57] tend to rely on mixed integer programming formulations to determine day-ahead generator schedules, which rely on corrective actions for example when solutions become non-optimal or require ancillary services. An EMS can certainly take this approach in determining UC and ED schedules [22], but the day-ahead schedule may require drastic corrections due to inaccuracies in renewable energy source (RES) forecasting. This is especially important for microgrids since RES generation may be a significant portion of their overall generation capabilities. Because microgrids want to set power dispatch setpoints in addition to UC, many operate on schemes where optimization of either UC, ED, or both are recalculated throughout the day to better employ real-time optimization and control [76, 33, 66, 63, 53, 69, 68, 58]. This way, as uncertain parameters

are realized and updated forecasts become available, new information can be incorporated in an updated schedule.

Some of the techniques for microgrid ED include MILP, robust optimization, stochastic optimization, artificial intelligence, and model predictive control [77],[24]. Historically, non-predictive techniques such as load following and cycle charging have been used to control devices. But [49] shows that forecast based economic dispatch using MILP performs better than these techniques. Predictive MILP has also shown to perform better than heuristic algorithms for isolated microgrid systems that include expansion and rolling horizon dispatch [51].

Lagrangian relaxation has also been historically used on a receding horizon basis for the UC+ED problem, and [76] takes this approach for a 3-home isolated network with solar power capabilities. A dynamic programming approach is used in [33], which provides closed loop control for the microgrid and quick solution time. Shi et al. [66] propose a Lyapunov optimization for dispatch which operates greedily for each time interval instead of over a day-ahead forecast horizon.

To solve the full UC+ED problem for microgrids, it is common to break the solution into first solving a UC problem, then determining dispatch as uncertain values such as load or RES power are realized. In [63], an affine arithmetic UC problem is solved for a given time horizon, then intermediate dispatch with optimal power flow (OPF) is calculated using a secondary algorithm. A similar decomposition in [53] uses a 24-hour UC, but only 2-hour ED with transmission network. Variable length intervals are also used in [53] which help lessen computation time.

Another technique for determining device setpoints is to have the EMS operate entirely on a rolling time horizon, solving a full MILP for UC+ED multiple times every day as new information becomes available. Silvente et al. [69] developed a rolling horizon framework where the full model is solved over a prediction horizon, and dispatch is then fixed for a given control horizon, after which the model is iterated to the next control horizon. This model was extended to include stochastic uncertainty in wind penetration in [68]. A rolling horizon was also employed in [58], with a control horizon of one hour, with time intervals length of 15 minutes. This allows for up to 1 hour of computation time, in order to deliver

dispatch setpoints for the next four time intervals. Although this strategy is computationally tractable even for larger microgrids, there is a loss in model fidelity. The hourly solution may not stay optimal for all four time intervals, but the microgrid lacks the ability to provide cost-saving corrective actions until the next solve one hour later.

Unfortunately, many of these microgrid benchmark studies are conducted using state-of-the-art solvers. While useful for simulations, commercial solvers are not always practical for real-world systems where cost can pose a significant barrier. Microgrids have small generation assets, and therefore the profits made from using a commercial solver may not be enough to offset licensing costs, as it is with large-scale unit commitment problems. A solution to cost-effectiveness is use of open-source solvers, which are free and readily available. This work is primarily motivated by an existing residential microgrid located in Hoover, Alabama, which has this open-source software restriction. The microgrid contains an operational rolling horizon EMS using the model described in this paper. In practice, due to the open-source restriction, the EMS was not solving the UC+ED model within the 5 minute time window. To overcome this barrier, this work develops a modification to the UC+ED model inside the existing EMS with two goals in mind:

1. Reduce computation time to under the length of one time interval (5 min.)
2. Capitalize on the rolling horizon architecture to introduce robustness to the model which allows for a decrease computation cost.

As previously mentioned, variable time intervals have been successfully used to provide quick solutions for the microgrid UC+ED problem [53]. Instead of focusing on relaxing time interval granularity, we will relax integrality by reducing the number of integer variables in the MILP farther into the future in a 24-hour UC+ED problem. This will change the optimal solution, but it will be shown this does not affect solution quality in a detrimental way, and in fact provides robustness which can lead to an operational cost decrease. Because renewable resources are cheap and flexible, there are many near-optimal solutions which are similar, but have renewable resource operation shifted slightly. Having a MIP explore all of these near-optimal solutions at time periods which are 12-24 hours out is likely not time well spent, as the trade-off in computation time is great. Additionally, the EMS only broadcasts

real-time dispatch setpoints, so it is more important to focus on how much power devices will produce in the future rather than exactly when it will be produced. This work demonstrates that relaxing integer variables can yield sufficient computation time reduction without sacrificing solution quality as the microgrid rolls through daily operation of economic dispatch.

2.5 Model

2.5.1 Basic Economic Dispatch

The economic dispatch model minimizes cost of electricity generation for the microgrid, subject to operational and constraints of the devices, network constraints, and load balancing requirements. The objective is given by (2.1). Cost of operating the generator is (2.1a) and is comprised of its piecewise cost curve, minimum up cost k_g and start-up cost C_g^V . Battery operational cost also included in the objective function, though it is small compared to the cost of the natural gas generator. Photovoltaic (PV) power is assumed to have no cost. The penalty parameter δ contains constraint violations which are detailed in Chapter 1. The constraint violations include extreme state of charge of the battery, line losses, and excessive use of reactive power.

$$\min \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} \left[\sum_{m \in \mathcal{I}} C_g^m p_{gt}^m + k_g u_{gt} \right] + \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} C_g^v v_{gt} \quad (2.1a)$$

$$+ \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} C_{bt} (P_{bt}^D + P_{bt}^C) + \delta \quad (2.1b)$$

Constraints for the economic dispatch model include operational constraints for the natural gas generator which can be seen in (2.2) and (2.3). Constraint (2.2a) represents the total generator power P_{gt} as its minimum up power P_g^{min} plus the power above minimum for each piecewise interval m . Upper limits on generator power for marginal power intervals and total power are imposed via (2.2b) and (2.2c). Lastly, (2.2d) allows for the start-up

indicator v_{gt} to be 1 when generator g has turned on in time period t .

$$P_{gt} = \sum_{m \in \mathcal{I}} p_{gt}^m + u_{gt} P_g^{min} \quad (2.2a)$$

$$0 \leq p_{gt}^m \leq p_{gt}^{max,m} u_{gt} \quad \forall m \in \mathcal{I} \quad (2.2b)$$

$$0 \leq P_{gt} \leq P_{gt}^{max} u_{gt} \quad (2.2c)$$

$$u_{gt} - u_{g,t-1} \leq v_{gt} \quad (2.2d)$$

$$\forall g \in \mathcal{G}, t \in \mathcal{T}$$

For generator minimum up-time, we incorporate constraints (2.3) depending on the current status of the generator. Define the minimum up-time for the generator as TU , which is the minimum number of time intervals that the generator must be on for after started. Before forming the minimum up-time constraints, we need to consider whether the generator is currently on, and how long it has been operating for. This is important because the time horizon is rolling, so we cannot start from scratch each time the optimization is run. We let the quantity *uptimes* be equal to the number of time intervals left for which the generator must be on if it has been running for less than two hours at the present time. Then let $T = TU + \text{uptimes}$. Note that if the generator was not on for less than two hours before the first time interval, *uptimes* = 0. Then for each $t \in \mathcal{T}$, one of the constraints (2.3a), (2.3b), or (2.3c) will apply.

if $t < \text{uptimes}$:

$$u_{gt} = 1 \quad (2.3a)$$

elif $\text{uptimes} \leq t < T$:

$$\sum_{i=0}^t v_{gi} \leq u_{gt} \quad (2.3b)$$

elif $T < t$:

$$\sum_{i=t-TU+1}^t v_{gi} \leq u_{gt} \quad (2.3c)$$

Note that constraints may similarly be added for a generator downtime requirement, along with corresponding shutdown variables. However, the natural gas generator for the real-world microgrid described in section 2.6 does not have shutdown requirements. We also omit generator ramping requirements, as generators contained in microgrids are often small enough that they can ramp up or down in less time than the length of one time interval. Formulations which include ramping constraints and other more complex constraints can be found in [39], and could be employed in this formulation for generators which have these requirements.

Constraints on the battery are also in the model, with a detailed formulation given in Chapter 1. The battery constraints include tracking of energy in the battery and limits on state of charge and charging/discharging power. We also constrain the battery to either charging or discharging at every time interval. PV power is allowed to be curtailed below its forecasted value. In addition, a convex relaxation of the AC optimal power flow problem (ACOPF) is included in the model. The ACOPF formulation used was developed in [44] which includes the dist-flow equations from [8]. ACOPF constraints are also shown in detail in Chapter 1. Although there are concerns that using an ACOPF relaxation may not return an AC feasible solution, the dist-flow equations are shown to be exact for certain radial networks. Also, in practice, the devices in the microgrid are able to adjust output slightly as errors in the ACOPF relaxation are comparable to error in PV and load forecasting methods. The economic dispatch model also has constraints (2.4) to balance real power demand D_t with power generation at every time interval.

$$\sum_{g \in \mathcal{G}} P_{gt} + \sum_{b \in \mathcal{B}} P_{bt} + \sum_{s \in \mathcal{S}} P_{st} = D_t \quad \forall t \in \mathcal{T} \quad (2.4)$$

2.5.2 Relaxation of Integrality

As discussed in Section 2.4 the necessity of open-source software can cause an increased burden of computation time when solving. A microgrid may also be limited in computational hardware, which can also hamper solution time and/or quality. Therefore our main goal of this study will be reducing computation time without significantly impacting quality of the

solution through the daily operation of the microgrid. We will also benchmark computation time reduction on limited hardware, to motivate budget-friendly hardware solutions.

To reduce computation time we will take advantage of the rolling-time horizon offered by microgrid economic dispatch problems. When Independent System Operators (ISOs) set a day-ahead solution to unit commitment, the optimization model is only solved once. Therefore in a day-ahead situation, each time interval is equally important because the solution will not change as the day is realized. In contrast, time intervals are not equally important in the rolling-time model. We are frequently re-solving dispatch, so the time periods far into the future (say, 12-24 hours) are not as important as the ones close to the current time (say, 1-2 hours out). It may be enough to know that a generator will run sometime in the future, as opposed to knowing exactly when and how much the unit will dispatch. Accordingly, emphasis can be put feasibility of energy supply far in the future instead of specific dispatch decisions.

To lessen focus on time periods farther out in the forecast, binary variables associated with the natural gas generator will be relaxed after a certain cutoff time within the rolling 24-hour forecast. By reducing the number of integer variables, the complexity of the problem will be reduced, which in turn should lead to faster solution time. The model which relaxes binary variables after a certain cutoff time will be referred to as model TB (for time break). Note that binary variables associated with the battery, or any other potential generation devices could also be relaxed. The generator was chosen as the only device to relax for two reasons. The first is the generator is the device with the most complex constraints concerning the integer variables, and secondly the generator has the most impact on the operational cost of the microgrid.

To formulate model TB, we first choose an hour T_{hour} at which to relax the natural gas generator. Because our forecast is divided into 5-minute time intervals, $T_{break} = 12 \cdot T_{hour}$ will be the 5-minute interval at which the relaxation begins. Note that there are 288 total time periods when the day is divided into 5 minute intervals. The relaxation affects two sets of binary variables, namely the generator start-up indicator v_{gt} and the commitment indicator u_{gt} . These variables are created for every generator $g \in \mathcal{G}$ and every time period $t \in \mathcal{T}$. The bounds on these binary variables now change depending on the time period t

within the 24-hour forecast horizon. The new bounds are:

$$u_{gt}, v_{gt} \in \begin{cases} \{0, 1\} & \text{if } t < T_{break} \\ [0, 1], & \text{if } t \geq T_{break} \end{cases} \quad (2.5)$$

Model TB now allows for the generator schedule to include power setpoints below its minimum operating threshold P_g^{min} during time intervals further in the forecast than T_{break} . None of these setpoints will ever get sent to the generator, though, because the rolling time horizon will have imposed an integer restriction on these variables when they become less than T_{break} . For now, it is enough to know an approximation of the aggregate amount of power that will be needed for the generator in the time intervals greater than T_{break} .

2.6 Description of Neighborhood

Data used to construct test examples was gathered from a physical residential microgrid system located in Hoover, Alabama. The microgrid and neighborhood construction was completed in 2018. The residential neighborhood is comprised of 62 homes. Each home has a controllable HVAC and water heater which are governed by their own optimization architecture [64]. This enables them to individually make decisions based on a forecasted price of electricity, and send the aggregate load forecast to the microgrid central controller (MCC). We assume in this study that the MCC has no control over the residential EMS, and therefore it is not described in detail here.

Within the MCC is an energy management system (EMS), which coordinates the optimal operation of the devices in the microgrid. The EMS contains the optimization model discussed in section 2.5, which delivers economic dispatch (ED) and unit commitment (UC) setpoints to all devices. The EMS is required to use open-source software for the optimization architecture due to cost constraints. The optimization model is operated on a rolling 24-hour forecast horizon with 5 minute time intervals. The microgrid generation capabilities consist of a 330kW PV system, a 400kW natural gas generator, and a 300kW battery system with a storage capacity of 680kWh. The EMS logs all measurements for microgrid load and PV

generation, as well as the output of the economic dispatch model. The EMS also facilitates control of the generation devices in the microgrid. The microgrid has the ability to be grid-connected, allowing for energy exchange through a point of common coupling (PCC). The microgrid can also be isolated, or islanded, from the main grid and operate on its own using only local generation sources. For the purposes of this study, we assume the microgrid is in an islanded state and there is no PCC available for grid connection. This scenario puts the most restriction on the microgrid, and is also likely to occur when there is an unplanned outage in the main electrical grid.

2.7 Computational Experiments

2.7.1 Description of Tests

Computational experiments necessitate a comparison of models across different types of seasonal weather. Both PV and load forecasts are needed as inputs to the economic dispatch model. The Hoover microgrid samples the PV and load measurements twice a second throughout the day. Each day with complete data was averaged into 5 minute intervals and added into a pool of days for its respective season. Then, 10 days were randomly sampled from each of the three seasons (winter, spring, and fall) to create a total of 30 test days. At the time of writing, the microgrid has not had sufficient load and PV data collection through summer months, therefore summer was omitted. We will now discuss the two major tests which were run, namely a daily operational cost comparison, and a computation time comparison. Section 2.7.2 will detail the results of the computational experiments.

Test 1 was analyzing computation time for each model type. Because microgrids typically have limited computational resources available, an ideal situation would be to use as little computational hardware as possible. Accordingly, a raspberry pi 3 was used as the platform for tracking computation time. All models were coded using the pyomo modeling language [30, 29], and the open-source CBC solver Version 2.10.1 was used. With CBC on the raspberry pi 3, solutions to the economic dispatch problem often take more than 5 minutes to solve. Therefore a mip gap of .05 (i.e. 5%) was used. We also implement a time limit

Table 2.1: Models Developed for Computational Testing

Model	Description
Normal	Economic dispatch model with no modifications
TB5	TB model with $T_{hour} = 5$ (i.e. relaxation starts hour 5)
TB10	TB model with $T_{hour} = 10$
TB15	TB model with $T_{hour} = 15$
TB20	TB model with $T_{hour} = 20$

of 420s (7 mins). It should be noted that in order to have a usable solution for the rolling economic dispatch problem, computation time must be less than 5 minutes; i.e. the length of a time interval. Therefore we hope to reduce computation time to under 300s. To generate computation examples for Test 1, three time periods were randomly sampled from each of the 30 sample days. The test models were then run with each of the samples as the initial time period, with a 24-hour forecast horizon of 5 minute intervals. Initial conditions were used based on the state of the microgrid at the sampled time period. The initial conditions include generator commitment status, battery SOC, and PV/load measurements. Three sample time intervals for each of the 30 days yields 90 test cases total, comprised of 30 for each season (winter, spring, and fall).

Test 2 was determining the daily cost of operation for the microgrid under model TB compared to the normal economic dispatch model. Because we are only keeping track of the cost, the tests were run with Gurobi Version 8.1.0 on a 2x Intel E5-2670 CPU @ 2.6 ghz (16 cores, 32 threads total), with Ubuntu server 18.04 and 256GB ram. Each sample day was rolled through all 288 5-minute time intervals. When a solution was obtained, the time 0 solution cost was collected. The time 0 solution represents the economic dispatch setpoints which would be sent to the devices in the microgrid upon obtaining the optimal solution. It is assumed that the devices operate at this setpoint until the next 5-minute time interval, which will operate using setpoints from the next 'rolled' solution of the economic dispatch. A mip gap of .05 was set for the operation cost test, so that it is consistent with Test 1.

In total, 5 test models were used to get a baseline for the effectiveness of the TB formulation. We refer to the economic dispatch model with no modifications as Normal. Then, for model TB we use $T_{hour} = \{5, 10, 15, 20\}$ for four different starting hours of relaxation and hourly aggregation. Recall T_{hour} is the hour where relaxation of integer variables begins. Therefore test model TB5 has 19 hours of relaxation. All 5 test cases are labeled in Table 2.1.

2.7.2 Results

In section 2.7.1, Test 1 was created by using 30 sampled time intervals from each season, which were used as the starting interval for the 5 test models in Table 2.1 with respective initial conditions. Note the maximum time limit was set at 420s, or 7 minutes. Table 2.2 displays the average computation time for each season when the model was solved on a raspberry pi 3. Every TB test had a lower average computation time than the normal case. The lower computation times are confirmation that relaxing integer variables will have an impact on reducing computation time. TB5 had the overall lowest average computation times for all 3 seasons, with fall performing the best. It is expected that TB5 would perform best as it has the least amount of integer variables compared to all other test models. Given that the time limit is 300s, we can see that the normal case took on average longer than one time interval to solve. This long computation time is certainly inadequate, as solutions which take longer than 300s to optimal will either return no solution, or at best a sub-optimal solution, for the economic dispatch setpoints. In contrast, the TB5 and TB10 cases are on average solved to optimality in less than 300s.

Table 2.3 gives the total number of days for each test model which exceeded 300s and 420s of computation time. There were 90 test days in total, comprised of 30 for each season. Given that the length of a time interval is 300s, the most effective tests will be the ones which are able to find an optimal solution within 300s. Clearly TB5 out-performed every other model in this respect, and also solved every single case to optimality in under 420s. From Test 1 we also see that TB5 was able to solve 20 out of 30 full days within 10% of the Normal cost model, and 19 days for TB5 had cheaper cost than the Normal cost model. Table 2.3 also demonstrates that even relaxing a few hours can help solve more days under

Table 2.2: Average Computation Time (sec.)

Model	Winter	Spring	Fall	Overall
Normal	309.16	402.19	334.87	348.75
TB5	226.04	220.81	209.90	218.92
TB10	245.82	287.00	247.31	260.04
TB15	247.74	324.99	268.76	280.50
TB20	260.30	356.73	290.28	302.44

Table 2.3: Number of days which exceeded time limits

Test	$t > 300s$	$t > 420s$ (timeout)
Normal	63	24
TB5	11	0
TB10	26	2
TB15	36	6
TB20	44	9

the time limit, as TB20 solved 19 more days than the normal case within the 300s time interval length, with only 4 hours of integer relaxation.

Also of importance was that every test model found a feasible integer solution within the time limit of 420s. This fact is notable because in practice, a result of max computation time can still return a sub-optimal feasible solution rather than no solution at all. However, an optimal solution is still better so using a TB model for faster solution times would still be preferable, provided solution quality is not compromised. As we will see in the discussion of Test 2, this is indeed the case.

As stated in section 2.7.1, Test 2 consisted of rolling through 30 randomly sampled days from the days with complete PV and load data, and recording operational cost of the microgrid with each of the 9 test models in table 2.1. For each model, we then calculated each day's percent difference in cost compared to the Normal model. Table 2.4 shows how many days were within 10% of the cost of the normal model. It is notable that for every test model, at least half of the 30 days were within 10% of the normal operational cost.

Table 2.4: Number of days with cost within 10% of Normal

Test	Days
TB5	20
TB10	23
TB15	22
TB20	27

Table 2.5: Number of days with cost less than Normal

Test	Days
TB5	19
TB10	18
TB15	15
TB20	14

Figure 2.1 shows the percent difference in cost compared to the Normal model for every test day, in chronological order. Each day has 4 colored points on the plot, one for each test model. The dashed black lines outline 10% difference in cost from the Normal model. From the figure, we can see that the majority of the cases are within 10% of the normal cost. Most of the cases which did not fall within this range have found a cheaper solution than the Normal model. In fact, 66 out of the 120 total tests were cheaper (55%). This demonstrates that the flexibility offered by relaxation of the models leads to a better solution in most cases once the rolling time horizon has access to future forecasts, vs. preemptively anticipating strict integer requirements. Table 2.5 details how many of the 30 sample days were cheaper than the rolling normal cost for each of the test models. Model TB5 had the most, with 19/30 days coming up cheaper than with no relaxation.

To explore the cost differences more thoroughly, we collected the daily cost of Normal operation at time 0, as if the initial solution was a one-shot dispatch for the whole day. That is, we ran the optimization once, without rolling, and calculated the 24-hour expected cost of operation. Note that this one-shot dispatch assumes that the solution would have

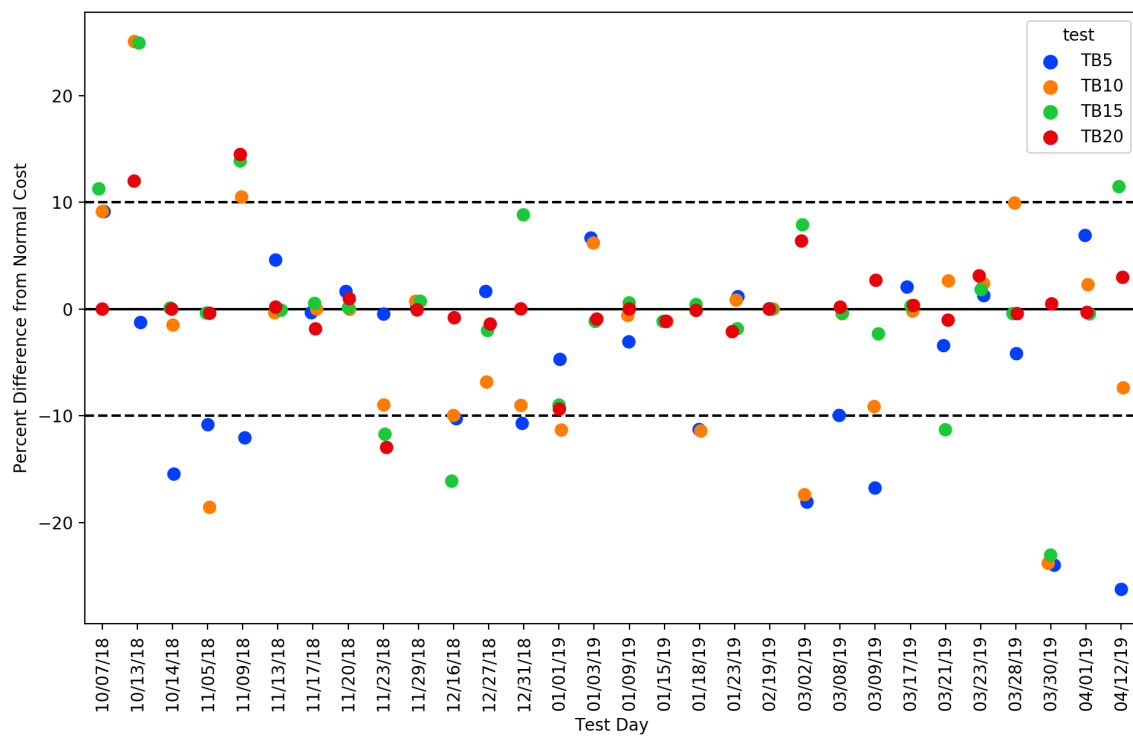


Figure 2.1: Percent difference in cost for Table 2.1 models compared to Normal

stayed optimal for the entire day. Therefore this solution gives a lower bound on daily cost of operation, because as more information becomes available in a rolling forecast horizon the solution given may not stay optimal, and will only increase in cost as corrective actions are taken. We will refer to the lower-bound expected cost Normal case as NLB. Note NLB is different from the rolling horizon normal model, which requires the maximal amount of integer variables in addition to the rolling horizon, whereas NLB is treated like a one-shot day ahead commitment without corrective actions (i.e. the lower bound case).

Figure 2.2 shows the percent deviation from rolling Normal cost for NLB, TB5, and TB20. The comparison cases were chosen because TB5 had the most days which performed cheaper than the rolling Normal, and TB20 had the least. Figure 2.2 clearly demonstrates that NLB is a lower-bound for the rolling-time tests. Figure 2.2 also shows that TB5 comes closest to reaching the lower bound cost, with minimal days which were worse than Normal. This can be explained by the fact that TB5 has the greatest amount of continuous variables compared to every other TB model. Because the continuous variables do not require TB5 to prematurely anticipate hard commitment schedules, the corrective actions needed as the day rolls through are typically less than those of TB20. Not only does TB5 offer faster computation time than any other model, the flexibility offered from relaxing time intervals offers the best cost reduction as well. Overall, every TB model performed better than Normal in terms of cost and computation time due to less integer variables, which in turn offered flexibility leading to less expensive corrections when the rolling horizon led to a change in optimal solutions.

2.8 Conclusion

This work investigated a relaxation method for a rolling horizon UC+ED optimization within an existing microgrid located in Hoover, Alabama, with open-source software restrictions. Due to the rolling horizon, the microgrid optimizations suffer from extreme time constraints, requiring an optimal solution in less than 5 minutes. A solution was sought to complete two goals: reduce computation time to under 5 minutes, and improve cost of operation for the microgrid.

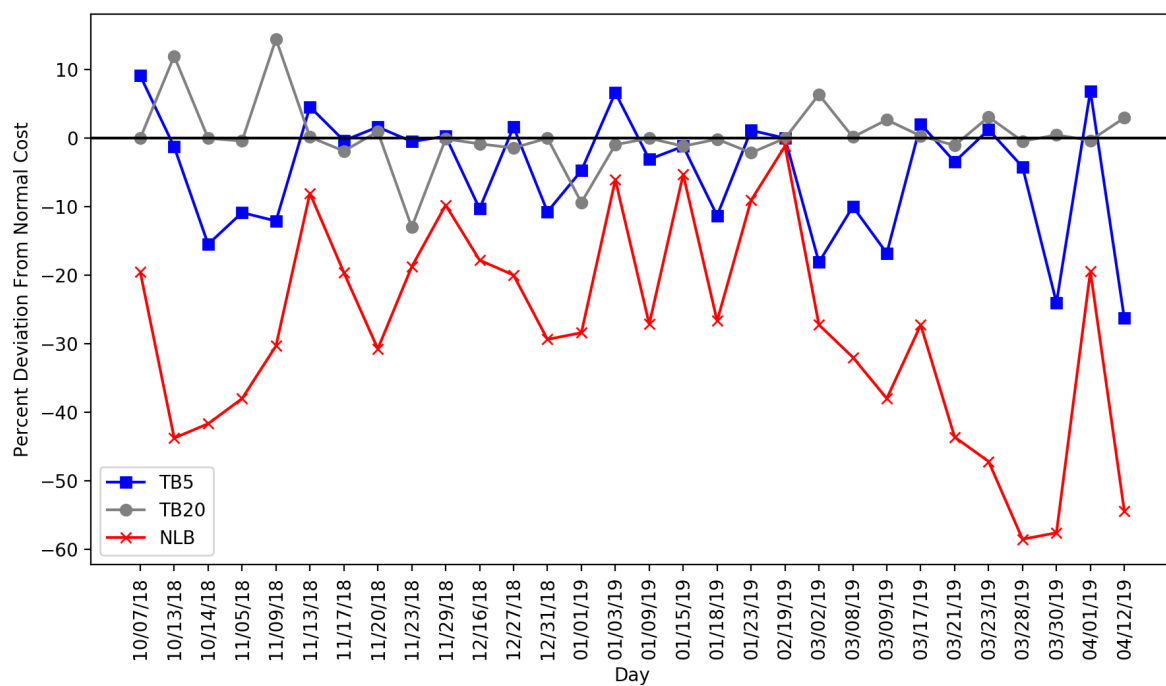


Figure 2.2: Percent difference in cost for select models compared to Normal

The solution method relaxes integer variables in the forecast horizon for the commitment and dispatch MILP within the EMS. Four different breakpoints were used (5,10,15, and 20 hours) to determine the impact of where the relaxation occurs, and computation time was tracked on a raspberry pi 3 using the open source solver CBC. In all four test models, computation time was successfully reduced compared to the normal operation model, indicating faster solution time due to less integer variables. The computation time reduction demonstrates the ability to solve small UC+ED problems on extremely limited computational hardware with open-source limitations.

For the second goal, the relaxation method TB found a cheaper cost solution than the Normal model in over half of the days tested, for all four models. The cheaper solutions can be explained by the effects of the rolling time horizon which are not present in a traditional day-ahead unit commitment scheme. As the 24-hour time window shifts new forecast information becomes available, and the rolling horizon is able to incorporate it into the next solution. The relaxation model TB offers more flexibility when compared to the normal model, due to containing less integer variables. The normal operation tries to lock down strict commitment requirements too early, resulting in a greater need for correctional actions as new information is added to the forecast. Therefore the TB model is able to adjust with cheaper corrective actions as new information becomes available while rolling through the day.

Future works may look at relaxing the model all the way up to the first time interval, to maximize the flexibility afforded by the continuous commitment variables. Because TB5 offered the greatest amount of days with cheaper cost and the fastest computation time, implementing a maximal relaxation in practice may yield the best results for a rolling horizon architecture. Note that the first time interval is the only interval which must have integer variables, as it needs to lock down commitment and dispatch setpoints to distribute to the devices. Additionally, now that computation time has been reduced, a more complex battery or generator formulation can be used with less worry of exceeding the 5 minute computation time window. This method can also be applied to other small-scale microgrids which may have more assets but still require fast open-source solutions for real-time operation.

Chapter 3

An uncertainty analysis for microgrid economic dispatch

Chapter 3 is based on a paper prepared for publication by the authors Amelia McIlvenna, Ben Ollis, and James Ostrowski.

McIlvenna, A., Ollis, B., Ostrowski, J. (2019). Stochastic Microgrid Energy Management With a Rolling Time Horizon

Authors McIlvenna and Ostrowski developed the stochastic formulation which provides the basis for the computational results in this chapter. Author McIlvenna programmed and ran the computational experiments, as well as wrote the manuscript. Author Ollis assisted in data collection from the existing microgrid which provided the motivation for the formulation. Author Ostrowski edited the manuscript.

In Chapters 1 and 2, all of the optimization models described have been deterministic. That is, they ignore uncertainty and assume quantities with inherent uncertainty are perfectly forecasted. The Hoover, Alabama microgrid contains a PV power system which falls under this category. Unfortunately when the each time interval in the optimization model is realized, the actual values for PV power output most likely do not match their forecasted values. Minimizing forecasting error has been a topic of concern in economic dispatch problems, particularly due to load balance constraints. For example, if the PV system produces much less power than forecasted, a more expensive generator may be utilized

to offset the error and balance load, resulting in a non-optimal and potentially undesirable solution.

For a simulated system, quantifying uncertainty is challenging because the perfect information under realization of uncertainty is only simulated. In this project, we have the advantage of obtaining forecasted and realized data for the physical microgrid in Hoover, Alabama. Chapter 3 takes advantage of the data available and develops a novel rolling horizon stochastic optimization for microgrid operation which can be compared to the cost of operation under perfect information.

3.1 Abstract

Various stochastic programming methods have been used to account for penetration of uncertain renewable energy generation in microgrids. While day-ahead unit commitment scheduling is common in microgrid stochastic optimization formulations, this strategy may not be necessary for small microgrids with fast ramping capabilities. This work developed a rolling horizon two-stage stochastic optimization for an existing residential microgrid located in Hoover, AL, with a novel stage construction. Because of the rolling horizon and fast generator ramping capabilities, stage construction was broken up by time instead of commitment variables in order to increase robustness of the formulation. The formulation accounts for uncertainty in photovoltaic (PV) power and operates on a rolling time horizon. PV forecasts along with measured load and PV data were collected from the existing Hoover, AL microgrid. The collected data was used to simulate the stochastic model with two different types of scenario sampling, along with the deterministic equivalent of the model for both measured values and forecasted values. These models were compared to a non-rolling horizon deterministic model with perfect information. The new two-stage decomposition techniques were found to reduce cost of operation for the existing microgrid over rolling time horizon deterministic models within the allotted solve time.

3.2 Nomenclature:

3.2.1 Variables

u_{gt}^s	Commitment status for each generator g in time t , for scenario s
v_{gt}^s	Generator start up indicator for each generator g in time t , scenario s .
u_{bt}^{Cs}, u_{bt}^{Ds}	Battery charging/discharging status, respectively in time t , scenario s .
P_{gt}^s	Total power at each generator g in time t , for scenario s .
P_{bt}^{Cs}, P_{bt}^{Ds}	Battery charging/discharging power, respectively
p_{gt}^{ms}	Power at m th generator interval
P_{vt}^s	PV power which is not curtailed at time t , scenario s
E_{bt}^s	Energy in battery b at time t for scenario s
$E_{bt}^{EX,s}$	Extreme battery state of charge in time t , scenario s .
L_t^s	Load shed at time t scenario s .

3.2.2 Parameters

k_g	minimum up cost of generator g
C_{gt}^m	Marginal cost for m th generator interval
C_{gt}^v	Start up cost for generator g
C_{bt}	Battery operational cost
P_g^{min}	Generator min up power
D_t	Real demand at time t
P_{vt}^{Fs}	Forecasted PV power at time t for scenario s
E_b^{min}	Minimum allowable energy for battery b
E_b^{max}	Maximum allowable energy for battery b

3.2.3 Sets

$g \in \mathcal{G}$	Set of generators
$b \in \mathcal{B}$	Set of batteries

$v \in \mathcal{V}$	Set of PV panels
$m \in \mathcal{I}$	Set of piecewise generator power components
$t \in \mathcal{T}$	Set of timeperiods
$s \in \mathcal{S}$	Set of scenarios

3.3 Introduction

Due to deregulation of electricity markets and increasing renewable energy adoption, distributed electricity generation is gaining traction. Microgrids have been introduced as a way to facilitate distributed energy and incorporate renewables while improving reliability [10]. Microgrids have already shown promise in delivering power to remote areas, as well as the ability to coexist with existing electrical infrastructure [32, 53].

Because of their distributed nature, microgrids require local control and optimization infrastructure to ensure successful operation, especially when isolated from any other electrical grid [31]. Many deterministic and stochastic optimization architectures have been developed to handle this need for local optimization of assets [51, 77]. The stochastic methods have the advantage of accounting for uncertain quantities in their operation such as wind turbines, photovoltaic (PV) power, and load. On the other hand, stochastic optimization often requires decomposition algorithms due to the large size of the models.

In [34], a two-stage stochastic program was formulated for a microgrid system with connection to the main grid, where the second stage included optimal power flow for both the main and micro grids. Another two-stage stochastic program for microgrid operation was developed in [50] and solved via an adaptive modified firefly algorithm.

An additional emphasis on reliability can be incorporated by adding measures of robustness or risk-aversion into the model [65]. A risk-averse two-stage stochastic UC model was presented in [4] which uses a CvaR assessment to quantify risk for isolated microgrids. A CvaR assesment is also analyzed in [23] for unexpected islanding events where the microgrid is rendered isolated.

Coupling stochastic generation with demand response has also been explored in the literature [48, 62]. A stochastic formulation for a microgrid with uncertain wind and PV was

simulated with and without demand response in [75]. Using a bender’s decomposition with second stage power flow constraints, it was found that demand response provided further benefits in cost reduction than the stochastic formulation alone.

A receding horizon model was examined in [54], which can be useful for isolated microgrids in the event that information may not always be available for a full forecast horizon. In [71] a two-stage stochastic model for microgrid energy scheduling with grid connection was developed. The model also showed reduction in operational cost and power losses when compared to its deterministic equivalent.

For microgrids which are operating under real-time conditions, rolling horizon models can help account for some uncertainty by extending the forecast horizon and re-solving for device setpoints throughout the day. A deterministic rolling horizon model for an existing microgrid in Chile was developed in [58]. This deterministic rolling horizon model showed overall lower cost compared to a day-ahead unit commitment model. Noticeably lacking in the literature is consideration of two-stage stochastic microgrid optimization under a rolling time horizon. There has been one attempt at such a model in [68], which modified a deterministic rolling time mixed-integer linear program (MILP) from [69] to account for uncertain heat and energy demand as well as wind power generation. The stochastic model incorporated a prediction horizon of k hours for which all scenarios were considered equal in order to extract a controls schedule for the generation devices and scheduling of tasks within residential homes. The time periods considered in [68], however, were 30 minutes long with forecasts considered deterministic during a lengthy prediction horizon. While this may be necessary for microgrids with ramping restrictions, for those with no restriction (such as the microgrid in this study) it is possible to solve for both commitment and dispatch decisions and deploy them time intervals much shorter than 30 minutes in length. Solving the model using shorter time intervals can make better use of new information as uncertainties are realized.

This study develops a two-stage rolling time horizon stochastic programming model with a novel stage decomposition for an existing microgrid located in Hoover, Alabama. As a result of the Hoover microgrid’s small size, we employ 5-minute time intervals for a 24-hour forecast horizon, and a unique stage formulation broken by time intervals which helps to maximize the

amount of uncertainty accounted for in the model. Currently the microgrid is operated using a 5-minute deterministic rolling time horizon optimization which delivers dispatch setpoints to the devices in the microgrid. These devices are comprised of a natural gas generator, PV system, and a lithium-ion battery. The Hoover microgrid has grid connection capabilities, however we will assume the microgrid is in island mode (i.e. isolated) for this study, as this is the most critical operational configuration.

First, we will describe what this rolling time stochastic model formulation looks like in Section 3.4, where the novel first and second stage decomposition will be described, along with our PV scenario construction and rolling horizon methodology. Then, Section 3.5 will detail computational results from a variety of test cases using real-world data from the existing Hoover microgrid. Both forecasted values and measured data for the PV system was collected under real-time operation of the microgrid. In addition, real-time measured values for residential neighborhood demand were collected and used for the demand values in the computational results. Lastly, section 3.6 will conclude the discussion.

3.4 Model

We will now construct our stochastic model formulation for the Hoover microgrid. For any two-stage stochastic problem, it must be determined which variables will belong to the first and second stage. Traditionally in power systems problems, generator commitment and start-up variables ($u_{g,t}, v_{g,t}$) would be included in the first stage, as commitment is often decided on a day-ahead basis. But this microgrid can ramp up any generation device within a 5-minute window, as the generator has a maximum power of only 400kW. Because of the fast ramping capabilities it is not necessary to decide commitment ahead of time. We can therefore decide these commitment variables as we roll through the day, meaning we only need to know the value of the commitment variables at present time. What becomes more important is to be able to extract valid setpoints for all devices in the microgrid from the solution of the optimization even though it is a stochastic formulation. Setting up the problem with multiple scenarios at every time period would make it difficult to determine

valid power setpoints, as the time period corresponding to present time would have variables for each scenario.

In [68] this issue of extracting device setpoints is handled by setting all scenarios equal for a certain prediction horizon near in the future. Instead of adopting a similar framework, we take a novel approach of breaking first and second stage variables by time interval instead of variable type. We will let all variables associated with the first time period $t = 0$ belong in the first stage. All variables not in the first time period (i.e. $t = 1, 2, \dots, 277$) will be the second stage. A breakdown of the stages and the most crucial variables in each is shown in Figure 3.1. With the stages constructed in this manner the first time period will have unique variables for each power setpoint and/or commitment status needed. Then because the first time period $t = 0$ will always represent the present time, these setpoint values can be delivered to the devices in the microgrid, and the current cost of operation can be calculated.

It is important to note that breaking stages by time period will leave the second stage with integer variables. For microgrids with a large number of assets these variables would necessitate advanced techniques for obtaining a solution. Fortunately the microgrid in this study is small, and with only one generator, battery, and PV system, the stochastic formulation is still small enough to be solved in its extensive form within the required 5-minute time frame for almost all instances (a fact we will revisit in Section 3.5). Therefore we do not need to worry about including commitment variables in the second stage.

To formulate the model, we start with the objective which will minimize cost of operation for the microgrid. Because the first stage includes variables which represent the present time, the first stage cost will be the generator and battery costs at $t = 0$, plus constraint violations β . PV is assumed to have negligible cost. Second stage cost includes cost for generator and battery operation in time periods $t = \{1, 2, \dots, 277\}$ along with constraint violations δ and

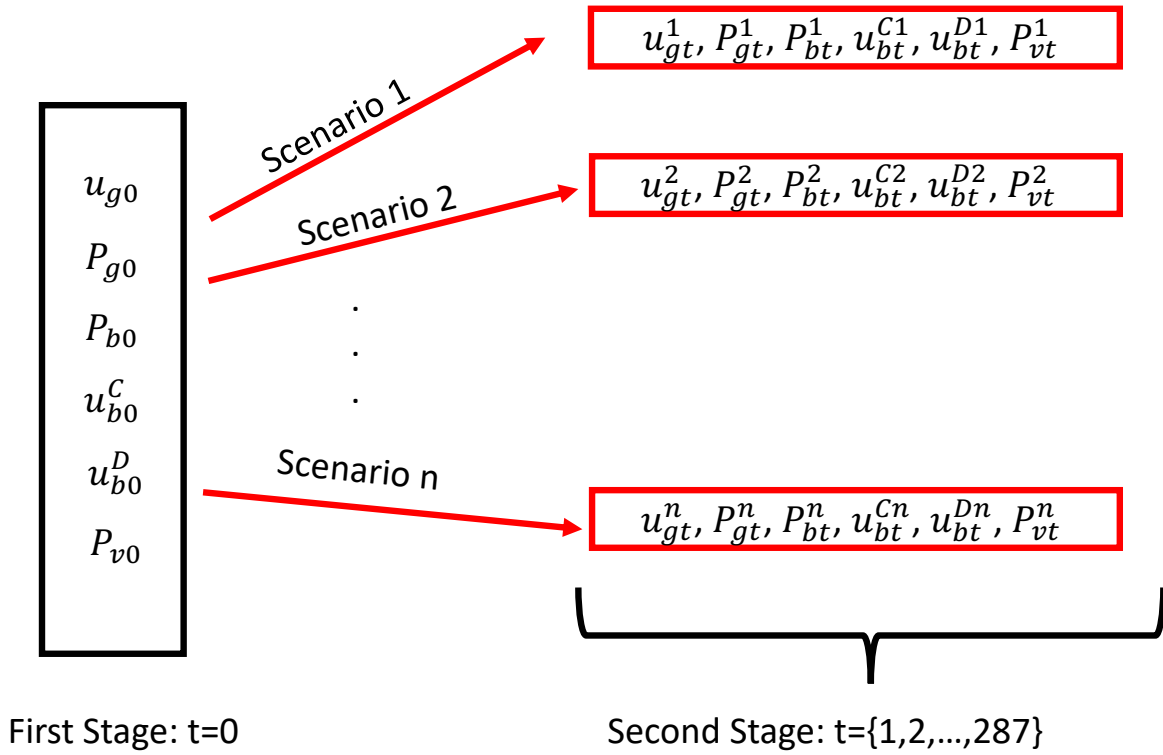


Figure 3.1: Stage Construction for the stochastic program broken by time interval with important variables listed for each stage

corresponding probabilities $prob_s$ for each scenario $s \in \mathcal{S}$.

$$\min \sum_{g \in \mathcal{G}} \left[k_g u_{g0} + C_g^{SU} v_{g0} \right] \quad (3.1)$$

$$+ \sum_{g \in \mathcal{G}} \sum_{m \in \mathcal{I}} C_g^m p_{g0}^m + \sum_{b \in \mathcal{B}} C_{b0} (P_{b0}^D + P_{b0}^C) + \beta \quad (3.2)$$

$$prob_s \left(\sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} \left[k_g u_{gt} + C_g^{SU} v_{gt} \right] \right) \quad (3.3)$$

$$+ \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} \sum_{m \in \mathcal{I}} C_g^m p_{gt}^m + \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} C_{bt} (P_{bt}^D + P_{bt}^C) + \delta \quad (3.4)$$

Constraint violation penalties for the first and second stage are detailed in (3.5) and (3.6), respectively.

$$\beta = \beta_E \cdot \sum_{b \in \mathcal{B}} E_{b0}^{EX} + \beta_V \cdot \sum_{b \in \mathcal{B}} P_{b0}^V \quad (3.5)$$

$$\delta = \delta_E \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} E_{bt}^{EX} + \delta_V \cdot \sum_{t \in \mathcal{T}} \sum_{b \in \mathcal{B}} P_{bt}^V \quad (3.6)$$

3.4.1 Constraints

First we will introduce the first-stage constraints, which represent all operational constraints for time period $t = 0$. Constraint (3.7a) sets total generator power at time 0, P_{g0} , equal to the sum of its piecewise power components, which are bounded in (3.7b). Let u_g^{INIT} be the initial commitment status of the generator. Then constraint (3.7c) represents the generator start up status at time $t = 0$, denoted v_{g0} , where u_{g0} will be the generator's commitment status. Constraint (3.7d) bounds the total power output of the generator.

$$P_{g0} = \sum_{g \in \mathcal{G}} p_{g0}^m + u_{g0} P_g^{min} \quad (3.7a)$$

$$0 \leq p_{g0}^m \leq p_{g0}^{max,m} u_{g0} \quad \forall m \in \mathcal{I} \quad (3.7b)$$

$$u_{g0} - u_{g,INIT} \leq v_{g0} \quad (3.7c)$$

$$0 \leq P_{g0} \leq P_{g0}^{max} u_{g0} \quad (3.7d)$$

$$\forall g \in \mathcal{G}$$

For the battery, we bound its charging rate P_{b0}^C , and discharging rate P_{b0}^D at time 0 in (3.8a),(3.8b). Energy capacity E_{b0} is bounded in (3.8d). The battery is restricted to be either charging or discharging at time 0 in (3.8c). Energy in the battery at time 0 is modeled by (3.8d), where E_b^{INIT} is the initial energy in the battery. For simplicity in notation, we set the total signed power in the battery as (3.8f).

Constraints (3.8g) and (3.8h) keep track of the difference in energy in the battery from the initial energy and the first time interval. The difference P_{b0}^V is penalized in the constraint violation term (3.5). While these constraints initially seem obscure, they are included as a means to minimize unnecessary cycling of the battery in the Hoover microgrid, by attempting to avoid drastic changes in the battery power setpoint. Constraints (3.8i) and (3.8j) will induce a penalty in the constraint violation term (3.5) if the battery enters an extreme state of charge (SOC) in the first stage, represented by E_{b0}^{EX} . This way, if an extreme SOC is necessary because of feasibility or an unexpected event occurs, the battery may assume this SOC for a penalty, and then will be motivated to either charge/discharge to a normal state

of operation.

$$0 \leq P_{b0}^C \leq P_{b0}^{C,max} u_{b0}^C \quad (3.8a)$$

$$0 \leq P_{b0}^D \leq P_{b0}^{D,max} u_{b0}^D \quad (3.8b)$$

$$u_{b0}^C + u_{b0}^D \leq 1 \quad (3.8c)$$

$$E_{b0} = E_b^{INIT} + P_{b0}^C \eta_b^C \Delta t - P_{b0}^D \frac{1}{\eta_b^D} \Delta t \quad (3.8d)$$

$$0 \leq E_{b0} \leq E_b^{cap} \quad (3.8e)$$

$$P_{b0} = P_{b0}^D - P_{b0}^C \quad (3.8f)$$

$$P_{b0} - P_{b,INIT} \leq P_{b0}^V \quad (3.8g)$$

$$P_{b,INIT} - P_{b0} \leq P_{b0}^V \quad (3.8h)$$

$$E_b^{min} - E_{b0} \leq E_{b0}^{EX} \quad (3.8i)$$

$$E_{b0} - E_b^{max} \leq E_{b0}^{EX} \quad (3.8j)$$

$$\forall b \in \mathcal{B}$$

The load/balance constraint is (3.9), with current demand D_0 and where L_0 accounts for load shedding at the current time. The constraint (3.10) allows for PV curtailment less than its measured value P_{vt}^{MEAS} . Note that we only use the measured PV power instead of the forecasted PV power because we are at the present time as this quantity is realized.

$$\sum_{g \in \mathcal{G}} P_{g0} + \sum_{b \in \mathcal{B}} P_{bt} + \sum_{v \in \mathcal{V}} P_{v0} = D_0 - L_0 \quad (3.9)$$

$$P_{v0} \leq P_{v0}^{MEAS} \quad \forall v \in \mathcal{V} \quad (3.10)$$

Now we will describe the second stage constraints, which account for each scenario $s \in \mathcal{S}$. Recall the second stage includes variables for all time periods $t \in \mathcal{T} \setminus 0$. Additionally, note if $t = 1$, then $u_{g,t-1}^s = u_{g0} \forall s \in \mathcal{S}$, because u_{g0} is a first stage variable. Then the generator

constraints (3.11) are similar to those of (3.7) except they are designated for each scenario.

$$P_{gt}^s = \sum_{g \in \mathcal{G}} p_{gt}^{sm} + u_{gt} P_g^{min} \quad (3.11a)$$

$$0 \leq p_{gt}^{ms} \leq p_{gt}^{max,m} u_{gt}^s \quad \forall m \in \mathcal{I} \quad (3.11b)$$

$$u_{gt}^s - u_{g,t-1}^s \leq v_{gt}^s \quad (3.11c)$$

$$0 \leq P_{gt}^s \leq P_{gt}^{max} u_{gt}^s \quad (3.11d)$$

$$\forall g \in \mathcal{G}, t \in \mathcal{T} \setminus 0, s \in \mathcal{S}$$

For generator minimum up-time, we incorporate constraints (3.12) depending on the current status of the generator. While these constraints may not be physically necessary, especially for a small generator such as the one in the Hoover microgrid, these constraints are included to discourage frequent generator cycling which can shorten the lifespan of the device. Define the minimum up-time for the generator as TU , which is the minimum number of time intervals that the generator must be on for after started. Before forming the minimum up-time constraints, we need to consider whether the generator is currently on, and how long it has been operating for. This is important because the time horizon is rolling, so we cannot start from scratch each time the optimization is run. We let the quantity *uptimes* be equal to the number of time intervals left for which the generator must be on if it has been running for less than two hours at the present time. Then let $T = TU + \text{uptimes}$. Note that if the generator was not on for less than its minimum up time before the first time interval, $\text{uptimes} = 0$. Then for each $t \in \mathcal{T}$, one of the constraints (3.12b), (3.12c), or (3.12d) will apply. Also note that for brevity constraints 3.12 contain variables denoted per scenario

scenario, but in the case $t = 0$ we have only u_{g0}, v_{g0} as these are first stage variables.

$$\text{For } t \in \mathcal{T} : \tag{3.12a}$$

if $t < uptimes$:

$$u_{gt}^s = 1 \tag{3.12b}$$

elif $uptimes \leq t < T$:

$$\sum_{i=0}^t v_{gi}^s \leq u_{gt}^s \tag{3.12c}$$

elif $T < t$:

$$\sum_{i=t-TU+1}^t v_{gi}^s \leq u_{gt}^s \tag{3.12d}$$

$$\forall s \in \mathcal{S}$$

Constraints (3.13) constrain the battery in the second stage for all scenarios $s \in \mathcal{S}$. They are similar to constraints (3.8) but are formed on a per-scenario basis for all time periods in

the second stage.

$$0 \leq P_{bt}^{Cs} \leq P_{bt}^{C,max} u_{bt}^C \quad (3.13a)$$

$$0 \leq P_{bt}^{Ds} \leq P_{bt}^{D,max} u_{bt}^D \quad (3.13b)$$

$$u_{bt}^C + u_{bt}^D \leq 1 \quad (3.13c)$$

$$E_{bt}^s = E_{b,t-1}^s + P_{bt}^{Cs} \eta_b^C \Delta t - P_{bt}^{Ds} \frac{1}{\eta_b^D} \Delta t \quad (3.13d)$$

$$0 \leq E_{bt}^s \leq E_b^{cap} \quad (3.13e)$$

$$P_{bt}^s = P_{bt}^{Ds} - P_{bt}^{Cs} \quad (3.13f)$$

$$P_{bt}^s - P_{b,t-1}^s \leq P_{bt}^{Vs} \quad (3.13g)$$

$$P_{b,t-1}^s - P_{bt}^s \leq P_{bt}^{Vs} \quad (3.13h)$$

$$E_b^{min} - E_{bt}^s \leq E_{bt}^{EX,s} \quad (3.13i)$$

$$E_{bt}^s - E_b^{max} \leq E_{bt}^{EX,s} \quad (3.13j)$$

$$\forall b \in \mathcal{B},$$

$$t \in \mathcal{T} \setminus 0, s \in \mathcal{S}$$

Load/balance constraints for all second stage variables are represented in (3.14). The constraint (1.7) allows for PV curtailment less than its forecasted value P_{vt}^{Fs} . Note that P_{vt}^{Fs} is the expected value of PV power for each scenario s for every time period t .

$$\sum_{g \in \mathcal{G}} P_{gt}^s + \sum_{b \in \mathcal{B}} P_{bt}^s + \sum_{v \in \mathcal{V}} P_{vt}^s = D_t - L_t^s \quad (3.14)$$

$$P_{vt}^s \leq P_{vt}^{Fs} \quad \forall v \in \mathcal{V} \quad (3.15)$$

$$\forall s \in \mathcal{S}, t \in \mathcal{T} \setminus 0$$

Finally, the total constraint set for our stochastic optimization combining the first and second stage is (3.7)-(3.15).

3.4.2 Scenario Construction

As with any two-stage stochastic program, we need scenarios which will represent some random vector ξ which represents our PV scenarios and takes values $\xi^s = P_v^s$ for each scenario $s \in \mathcal{S}$. For one scenario, this vector represents a possible PV forecast for all 287 second stage time intervals for the representative test day. Recall that the first time interval represents present time, is in the first stage, and as such will use the measured PV value instead of forecasted scenarios. To construct all of the scenarios, we need a method of generating these possible PV forecasts. Fortunately, we already have PV forecasts from the winter, spring, and fall seasons for the real-world microgrid located in Hoover, Alabama. There were 196 days in total which contained a full forecast, with 86 winter, 54 spring, and 56 for fall. We will now describe how this collection of data was used to create full day samples of PV power forecasts, which were then used to create scenarios for the stochastic program.

The forecasts for the Hoover microgrid were provided by the Clean Power Research company, and contain 48 data points, one for each 30 minutes. An updated forecast is also issued every 30 minutes throughout the day. Because the optimization runs in 5-minute time intervals, one 30 minute forecast point is representative of six 5-minute time intervals. Creating individual daily scenarios involves collecting the first point in the 30 minute forecast that was issued, and aggregating them into a full 5-minute forecast for each day with full data. Then, for a given test day, we use that day's original PV forecast plus 9 additional forecasts, each with probability $1/10$, as our scenario collection for the stochastic program. We settled on 10 scenarios as a sufficient balance between computational tractability and solution quality.

Two different methods of choosing the additional 9 forecasts were investigated. The first method was to randomly sample 9 days from the collection of forecasts in the same season as the test day. The second method involved taking each collection of forecasts for the three seasons, and grouping them into smaller clusters by the k-means clustering algorithm. The k-means clustering algorithm partitions data into k clusters for which the squared euclidean distance from each point to its corresponding cluster's centroid is minimized. There are

many initialization methods, but we chose a random assignment of k clusters for which the initial means were calculated. The algorithm then alternates between assigning all points to the cluster which minimizes their squared euclidean distance, and updating centroids, until cluster assignments stagnate. Our PV forecast data was broken into 2-3 clusters per season using the k-means clustering algorithm. Then, for a given test day, the 9 additional scenario forecasts will be chosen randomly from the k-means cluster containing the test day. This way, scenarios will forecasts which most closely resemble the forecast for the test day, in the hopes that this will group days with similar weather patterns together for the scenarios.

3.5 Computational Results

Next we will describe the computational test instances and results. We will consider deterministic versions of the model in order to form a comparison to the stochastic model developed in Section 3.4. The deterministic models contain the same constraints as the stochastic model, with the difference being there is only one scenario corresponding to the singular PV forecast given for the real-world microgrid under its normal operation.

3.5.1 Test Cases

Five days with complete PV forecasts, PV measurements, and load measurements were chosen as the test days. The load measurements from the microgrid were used as the load forecasts for the computation, which are reflected in the parameters D_t in constraints (3.9) and (1.6). The controllable generation devices in the Hoover microgrid are a 330kW PV system, a 400kW natural gas generator with a minimum power output of 100kW, and a battery storage system with a power output of 300kW and capacity of 680kWh. The battery minimum state of charge is set to 20%, while the maximum is 80%. This means the battery will only enter a state of charge of 0-20% or 80-100% in an emergency situation, and the optimization will incur a constraint violation penalty for entering this state. The generator minimum up-time was set to 30 minutes, or 6 time intervals, to avoid fast cycling of commitment status.

Four rolling time models were tested in total; two deterministic and two stochastic variants. The first is the deterministic model with the PV forecast completely replaced with measured PV data for the test day (DM). The other rolling deterministic model tested was simply using the singular PV forecast corresponding to the test day, and will be referred to as (DF). An example of the measured PV data versus deterministic forecast data for December, 23, 2018 is shown in Figure 3.2. It can be observed that December 23 was an overcast day, as the maximum measured PV power output was around 140kW out of a total capacity of 330 kW.

For the two stochastic models, we first considered the stochastic model with seasonal data sampling (SSE), where 9 randomly sampled days from the same season as the test day were chosen. The last model is the stochastic model with scenarios constructed by taking 9 days from the same k-means cluster as the test day (SKM). Recall that the PV forecast for the test day is used as the 10th scenario for each stochastic instance.

Also, we must recall that to simulate the real-world operation of the microgrid, the first stage will always represent the setpoints sent to the generation devices under realization of PV output. To accomplish this, the first time period in the PV forecast will always be replaced with the measured PV power value collected from the actual operation of the microgrid, as the optimization rolls through the day, per constraint (3.10).

To facilitate comparison with perfect information, we take use of the fact that measured PV values are available for the days we are testing. A fifth deterministic model was tested without rolling time, and with perfect information, referred to as PERF. When the rolling time models start, their forecast horizon is 24 hours, shown in figure 3.3. After rolling through all 24 hours, the forecast horizon is now 48 hours into the future from the original $t = 0$ at the initialization of the model. Therefore, we run the perfect information case PERF with a 48 hour horizon, also demonstrated in 3.3. However, in calculating operational cost, we take only the cost for the first 24 hours, as this will be comparable to the cost after the rolling horizon models have made it through 24 hours of realization.

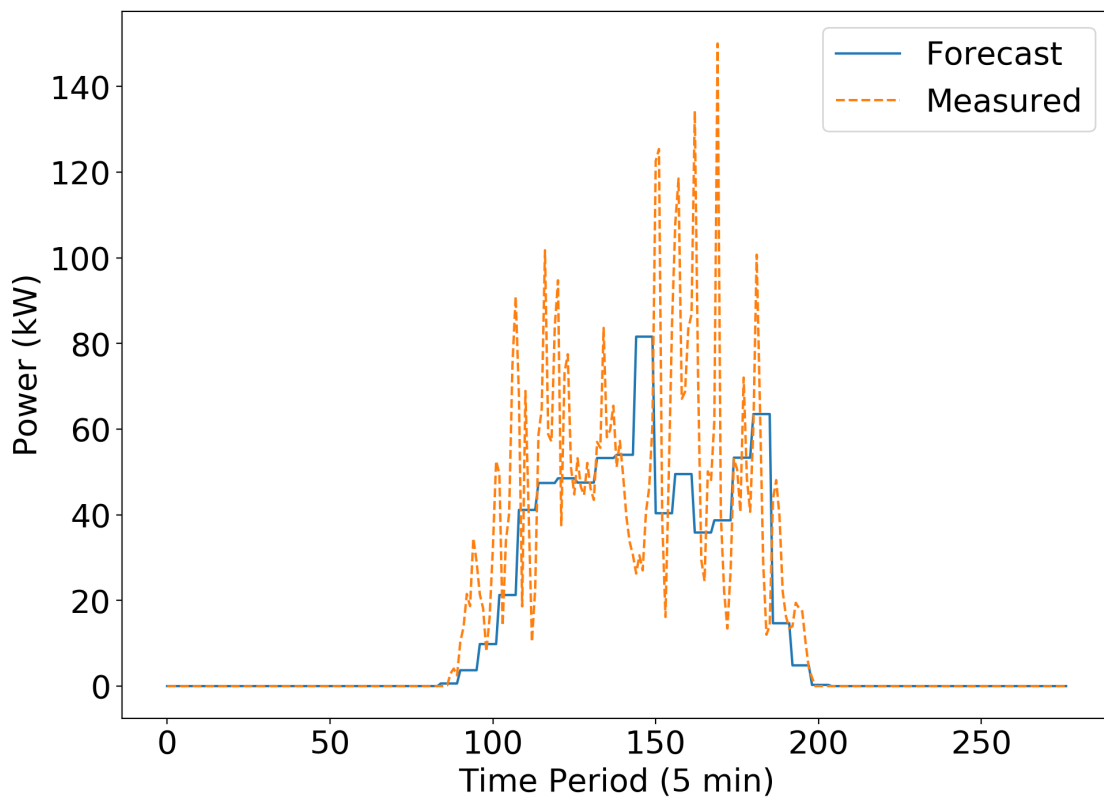


Figure 3.2: PV forecast and measured values for 12/23/18

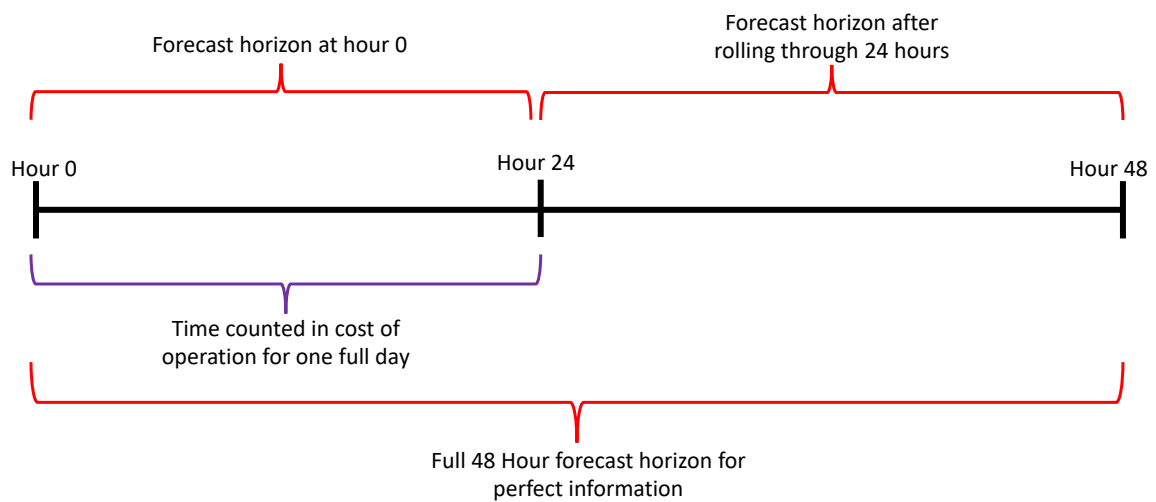


Figure 3.3: Forecast Horizons for the Rolling Time and Perfect Information Models

Table 3.1: Cost of operation in \$ for each test case

Date:	11/09/18	12/23/18	01/04/19	01/21/19	03/25/19
DM	235.27	366.64	277.12	565.85	109.96
DF	247.91	367.92	287.74	578.95	133.23
SSE	213.26	360.27	250.69	583.06	118.19
SKM	213.62	365.6	305.75	578.74	111.0
PERF	197.39	351.91	246.0	494.13	65.04

3.5.2 Results

The models were coded in python using the Pyomo modeling language [29],[30], and solved with Gurobi Version 8.1.0 on a 2x Intel E5-2670 CPU @ 2.6 ghz (16 cores, 32 threads total), with Ubuntu server 18.04 and 256GB ram. During simulation a mip gap of 2% was used in order to ensure adequate computation time in the stochastic cases. It is desirable for the models to solve within 5 minutes, as that is the length of the time intervals in this study. The extensive form solution for the stochastic models was found to solve in less than a minute for a majority of the solves, due to its relatively small size. In fact, only 27 of 1440 runs, less than 2%, exceeded the 5-minute window, and all were from the March 3rd 2019 test. If a solution hit max time, the best found solution at that time was used, and the computation was rolled to the next time interval. Because 98% of the the stochastic cases solved in under 5 minutes, advanced techniques such as progressive hedging were not necessary. The deterministic models solved generally in no more than a few seconds.

Table 3.1 displays the cost of operation for the microgrid for each model after rolling through a full day on the rolling horizon models, and a full day cost for the PERF model (perfect information). For the rolling horizon models, the cost of operation at the first stage ($t=0$) is added together after each solve, creating the daily cost of operation as each time interval is realized. Recall that the PERF case is a one-shot optimization where the cost of operation is collected for the first 24 hours. This is the same cost of operation window as the rolling time horizon models, with the exception that it is solved once instead of gathering one time period at a time.

One of the most notable results is that the stochastic seasonal scenario sampling (SSE) was cheaper than the rolling time deterministic model with the PV forecast (DF) in 4 out of 5 days. Additionally, the stochastic model with k-means scenario sampling (SKM) was also cheaper than DF in 4 of 5 days. These results are of particular note as the rolling time deterministic model DF is the current mode of operation for the Hoover, AL microgrid. Therefore the microgrid would have save cost by switching to a stochastic model from the deterministic version.

We can see also from Table 3.1 that the true perfect information case, PERF, indeed provides a lower bound on operational cost for the rolling time stochastic and deterministic models. Another interesting result is that the stochastic models SSE and SKM provided cheaper solutions than the deterministic result with measured data (DM) about half the time. Although DM uses the measured data, it only ever has access to the next 24 hours of measurement data, and as we roll through the day, the time periods are added on one by one. Therefore the DM case does not have access to complete perfect information like the PERF case, limiting its ability to provide a true optimal solution.

The tendency for rolling time deterministic models with a set forecast (or measurement values in the case of DM) is to over commit resources based on the 24-hour forecast. When more forecast information becomes available as the day rolls through, the optimal solution is likely to change, and will only increase in cost. The stochastic model allows for robustness, which is not inherent in any deterministic model. This means that it offers more flexibility in adjusting to new information when the forecast is extended in the rolling time horizon. Both deterministic models have difficulty responding to this new information as resources have only been committed based on a singular forecast. While DM clearly outperforms DF due to the deterministic nature of both models, overall the stochastic models show superiority in reducing operational cost of the microgrid. This superiority is evident in Figure 3.4, which shows the generator power output from January 4, 2019 in both the SSE (stochastic seasonal sampling) and DM (deterministic with measured values) cases. The deterministic model overcommits in the area outlined by the red arrow in the figure, as well as overproducing in more areas than the stochastic model. In fact, the deterministic run had the generator committed in 183/288 time periods, whereas the deterministic case commitment was only

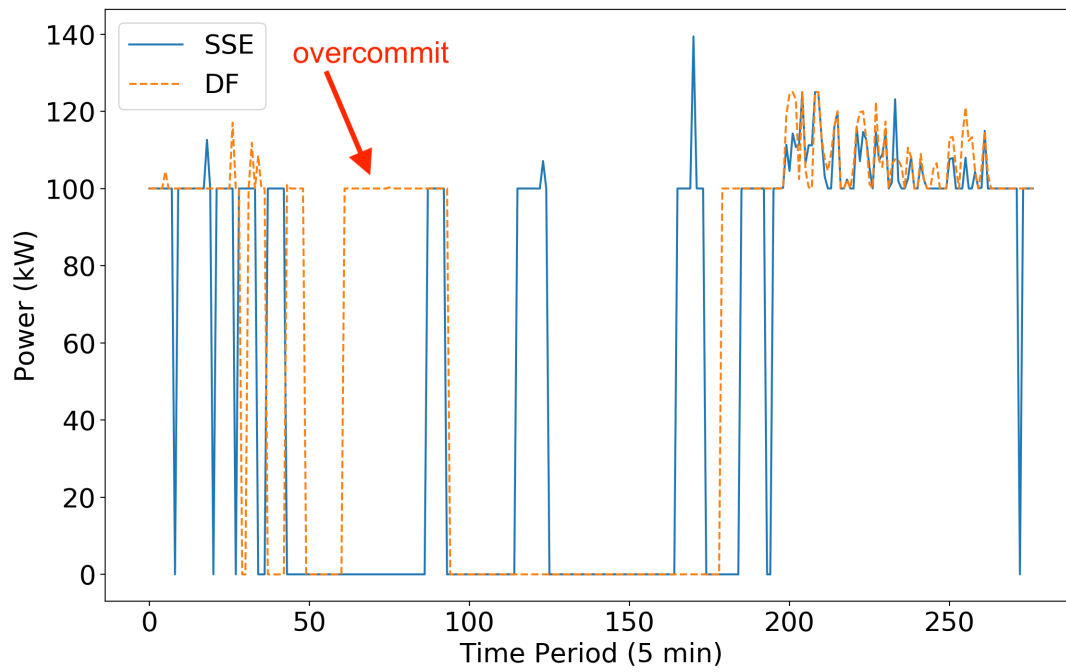


Figure 3.4: Generator power output for 01/04/19 for DM and SSE cases

160/288 time periods. This difference is almost two hours of commitment time which could be avoided with the stochastic model.

3.6 Conclusion

This work tested a novel two-stage stochastic optimization with a rolling time horizon, designed to be able to deliver device setpoints which are necessary for real-time operation of microgrid systems. Four rolling time models were tested, including a deterministic model with measured PV values, a PV forecast, and two stochastic scenario sampling methods. These methods were compared to a non-rolling deterministic model with perfect information. Overall, the stochastic models outperformed the deterministic in terms of delivering a lower cost solution after the realization of the measured PV power output. The rolling case allowed for the robustness of the stochastic model to be taken full advantage of, leading to cheaper solutions than even the measured PV data deterministic model. Due to the small number of assets in the microgrid, the extensive form of the stochastic models was able to solve within the required 5 minute window in approximately 98% of the time. These results show promise for using stochastic optimization in real-world cases where the models are solved in a rolling time manner instead of a day-ahead commitment. Indeed, the real-world microgrid used for data and forecasts in this study is operated under these conditions and could see a benefit in terms of cost from adopting a stochastic model with a current time first stage and future time second stage.

Future enhancements for this model could include transmission constraints, which at present were ignored. The possibility of adding transmission constraints to the first stage only may ensure current time operability without adding extensive computation time to the model. Nevertheless, the unique first and second stage breaking by time interval developed in this paper demonstrate potential in reducing microgrid operational cost by adding useful robustness into the formulation.

Chapter 4

Cut Generation for UC+ACOPF

For microgrids, an ACOPF relaxation such as the dist-flow/SOCP formulation used in chapter 1 may be close to exact or provably exact for certain radial networks. But in larger networks and mesh networks ACOPF relaxations rarely, if ever, give an AC feasible solution for the original problem. Additionally, because solving the original ACOPF outright is computationally intractable due to its nonconvexity and nonlinearity which render it NP-hard [11, 43], a robust solution algorithm for the ACOPF problem is of great need.

Our proposed algorithm will incorporate the UC problem with the linear copper plate relaxation developed in [19]. Because the copper plate relaxation ignores crucial transmission constraints, it is highly unlikely that this formulation alone will ever provide an acceptable solution for the AC power flow equations. Therefore to incorporate the AC power flow constraints, we will develop an algorithm which takes the preliminary solution and generates cuts from the SOCP formulation of ACOPF. Because transmission constraints are also independent over time, this leads to a natural decomposition for the problem which we can take advantage of in generating cuts. The developed algorithm will provide a scalable solution to provide tighter lower bounds for large networks which currently rely on approximations such as the DC power flow constraints.

The rest of the chapter is structured as follows: Section 4.1 will introduce relevant nomenclature. Then, Section 4.2 will provide an introduction and detail relevant literature. Section 4.3 will describe the models used in the Benders algorithm. The UC formulation is discussed in Section 4.3.1, followed by the ACOPF formulation and SOCP relaxation in

Section 4.3.2. Our Benders cut algorithm is in Section 4.4. Test instances are created in Section 4.5.1, and the computational results follow in Section 4.5.2. Finally, Section 4.6 concludes the discussion.

4.1 Nomenclature:

4.1.1 Sets

$g \in \mathcal{G}$	Set of generators
$i \in \mathcal{I}$	Set of piecewise generator power components
$b \in \mathcal{B}$	Set of buses
$(b, k) \in \mathcal{L}$	Set of lines
$t \in \mathcal{T}$	Set of hourly timeperiods
$s \in \mathcal{S}$	Set of hourly subproblems

4.1.2 Variables

p_t^g	Real power generated for generator g at time t , (MW)
$p_t^{g'}$	Real power generated above minimum for generator g at time t , (MW)
$\bar{p}_t^g$	Maximum real power available for generator g at time t , (MW)
$\bar{p}_t^{g'}$	Maximum power available above minimum for generator g at time t , (MW)
p_t^{gi}	Power at i th generator interval
u_t^g	Commitment status of generator g at time t , $\in \{0, 1\}$
v_t^g	Start up indicator for generator g in time t , $\in \{0, 1\}$
w_t^g	Shut down indicator for generator g in time t , $\in \{0, 1\}$
$x^g(t, t')$	Shutdown indicator arc: shut down at time t , start up at t' , $\in \{0, 1\}$
s_t^R	Reserve shortfall at time t
$p_{b,k}^f, p_{b,k}^t$	Real power at the from, to end of lines, respectively, (MW)
$q_{b,k}^t, q_{b,k}^f$	Reactive power at the from, to end of lines, respectively, (MW)
v_b	Complex voltage at bus b , $v_b^2 = (v_b^r)^2 + (v_b^j)^2$

$c_{b,k}, s_{b,k}$ Second order cone variables

4.1.3 Parameters

k_g	minimum up cost of generator g
c_{gt}^m	Marginal cost for m th generator interval
UT^g	Minimum up time for generator g , (h)
DT^g	Minimum down time for generator g , (h)
RU^g, RD^g	Ramp up/ Ramp down rate (respectively) for generator g , (MW/h)
SU^g, SD^g	Start up/ Shut Down rate (respectively) for generator g , (MW/h)
$\overline{P}^g, \underline{P}^g$	Maximum/Minimum real power output (respectively), for generator g
$\overline{Q}^g, \underline{Q}^g$	Maximum/Minimum reactive power output (respectively), for generator g
$\overline{P}^{gi}$	Maximum power output for piecewise interval i , generator g
R_t	Global spinning reserve at time t
P_{bt}^D	Real power demand at bus b time t
Q_{bt}^D	Reactive power demand at bus b time t
G_b^{sh}	Shunt conductance at bus b
B_b^{sh}	Shunt susceptance at bus b
$G_{b,k}$	Real part of the admittance matrix for branch (b, k)
$B_{b,k}$	Imaginary part of the admittance matrix for branch (b, k)

4.2 Introduction

Each day large energy markets must solve the unit commitment (UC) problem to schedule the operation of generators in the electrical grid. Because large generators usually have lengthy ramp up/down rates and minimum up/down time requirements, these commitment decisions are typically made on a day-ahead basis. However, ensuring feasibility of a commitment schedule necessitates inclusion of AC network constraints, turning the problem into a UC plus AC optimal power flow (ACOPF) formulation. But both the UC and ACOPF are NP-hard [9, 43, 11], and the ACOPF is nonlinear and nonconvex. Due to these properties, especially on large networks, the UC+ACOPF is considered computationally intractable [1].

Because of this intractability, imperfect solution methods must be employed by independent system operators (ISO's) to schedule thermal generators on a day-ahead basis. These solutions unfortunately require costly corrective actions when they are not AC feasible in order to adjust to the network constraints. As a result of intractability and need for a solution, solution techniques in the literature typically include network constraints as a convexification or relaxation of the ACOPF. The problem is then either rendered a more tractable formulation or solved through a solution algorithm. Algorithms which attempt to incorporate AC power flow constraints often rely convexification of the ACOPF problem through relaxation or other methods. These formulations are not guaranteed a globally optimal solution or in some cases even an AC feasible solution, and may require corrective actions to produce a usable commitment schedule.

A tri-level decomposition which uses a signomial transformation to convexify the AC subproblem was presented in [3]. A semi-definite programming (SDP) approach was used in both [6] and [47] to obtain a convex AC power flow formulation. Although these SDP techniques may lead to a non-integer solution, corrective actions can render the problem integer feasible.

Castillo et. al [16] provides a solution algorithm for the UC+ACOPF problem based on an outer approximation (OA) method. Similarly, a nested multitree approach is used in [46] to find a global solution for the UC+ACOPF, where the inner OA routine to solve the nonlinear nonconvex multiperiod ACOPF comes from [45]. While [46] may have provided the first solution algorithm with a global guarantee, this method is still computationally intensive compared to other techniques and therefore not tractable for large-scale instances.

Benders decomposition is also a popular technique for approaching a solution to the UC+ACOPF problem [25, 26, 2, 67, 52, 59], although it requires convexity in the subproblems to provide valid cuts. Fu et al. developed a solution algorithm using both benders decomposition and lagrangian relaxation techniques for the security constrained ACUC problem [25], which later added contingencies [26]. A Benders Cut approach has been used for a hydro scheduling problem with ACOPF subproblems in [67], which introduces application specific constraints in an attempt to overcome the nonconvexity issue. Nasri et. al, [52] also use properties of the problem application to overcome convexity for a stochastic

UC+AC problem with large wind penetration. Their formulation uses stochastic scenarios as subproblems, which leads to a convexification when a sufficient number of scenarios are used.

Benders decomposition approaches can be useful for specific applications where convexity issues can be mitigated. However, solution techniques with an inherently convex subproblem are more widely applicable. Paredes et al. [59] used a convex Semi-Definite program as a subproblem for a UC+ACOPF problem. Semi-Definite programs, however, have scalability issues due to computational complexity.

This work will also use a Benders decomposition method for the UC+ACOPF problem, but an SOCP relaxation introduced in [35] will be used to form convex nonlinear subproblems based on a 48-hour day-ahead UC problem. Although the SOCP relaxation may not be as tight as other methods, its relatively quicker solution times lend well to large-scale applications which may currently only be solvable by methods such as DC approximation. Furthermore, a comparison between the DCOPF and copperplate network formulations will be conducted to assess the value of the solution compared to these other large-scale solution methods.

4.3 Models

4.3.1 Unit Commitment Formulation

A Bender’s decomposition algorithm requires a master problem as well as subproblems which are used to provide feasibility cuts back to the master problem (described in detail in Section 4.4). For a UC+ACOPF formulation, it is natural for the UC problem to represent the master problem, as it is a mixed-integer linear program (MILP).

The constraints will use the 3-binary variables which were first introduced in [27]: namely commitment variables u_t^g , start-up variables v_t^g , and shut-down variables w_t^g . These three variables are related via constraint (4.1)

$$u_t^g - u_{t-1}^g = v_t^g - w_t^g \tag{4.1}$$

Next we will outline the relationships between some useful power variables. Note that by defining these relationships, different representations of the power generated can be used interchangeably in the constraints. We now introduce the power variables both for total power generated, denoted p_t^g , and power generated above minimum $p_t^{g'}$. These two variables are related via equation (4.2).

$$p_t^{g'} = p_t^g - \underline{P}^g \cdot u_t^g \quad \forall t \in \mathcal{T} \quad (4.2)$$

Let the maximum power available for generator g at time t be denoted $\bar{p}_t^g$. This quantity can be related to the maximum power available above minimum $\bar{p}^{g'}$ by equation (4.3).

$$\bar{p}_t^g = \bar{p}_t^{g'} + \underline{P}_g u_t^g \quad \forall t \in \mathcal{T} \quad (4.3)$$

And the total power generated by a generator g must be below the maximum power available (4.4)

$$p_t^g \leq \bar{p}_t^g \quad \forall t \in \mathcal{T} \quad (4.4)$$

Now the variables we just defined can be used to introduce constraints on generation limits which were proposed in [28]. We will present these in a slightly different format by bounding the maximum power available at time t , $\bar{p}_t^g$. For all generators which have $UT^g > 1$, constraints (4.5) represent their generation limits.

$$\bar{p}_t^g \leq \bar{P}^g \cdot u_t^g - (\bar{P}^g - SU^g)v_t^g - (\bar{P}^g - SD^g)w_{t+1}^g \quad \forall t \in \mathcal{T} \setminus T \quad (4.5a)$$

$$\bar{p}_T^g \leq \bar{P}^g \cdot u_T^g - (\bar{P}^g - SU^g)v_T^g \quad (4.5b)$$

In the event $UT^g = 1$, constraints (4.5) no longer apply, and we represent the generation limits as (4.6).

$$\bar{p}_t^g \leq \bar{P}^g \cdot u_t^g - (\bar{P}^g - SU^g)v_t^g \quad \forall t \in \mathcal{T} \quad (4.6a)$$

Spinning reserve requirements will be modeled with a global requirement, as in [15, 56], via constraint (4.7).

$$\sum_{g \in \mathcal{G}} \bar{p}_t^g + s_t^R \geq \sum_{b \in \mathcal{B}} P_{bt}^D + R_t \quad \forall t \in \mathcal{T} \quad (4.7)$$

Next we will include a variation of the two-period ramping constraints, which were originally introduced in Damci-Kurt et al. [21]. The ramping inequalities (4.8) are expressed in terms of the power generated above minimum $p_t^{g'}$ instead of total power generated. Note that although the ramp up inequalities (4.8a) include a substitution of $p_t^{g'}$ with $\bar{p}_t^{g'}$, they are still valid [39].

$$\bar{p}_t^{g'} - p_{t-1}^{g'} \leq (SU^g - \underline{P}^g - RU^g)v_t^g + RU^g u_t^g \quad (4.8a)$$

$$p_{t-1}^{g'} - p_t^{g'} \leq (SD^g - \underline{P}^g - RD^g)w_t^g + RD^g u_{t-1}^g \quad (4.8b)$$

We also need to ensure minimum uptime and downtime requirements are met for each generator. Each generator will have its own minimum uptime denoted UT^g , and downtime denoted DT^g . Then we add minimum uptime constraints (4.9a) and downtime constraints (4.9b) originally formulated in [61].

$$\sum_{i=t-UT^g+1}^t v_i^g \leq u_t^g \quad \forall t \in \{UT^g, \dots, T\} \quad (4.9a)$$

$$\sum_{i=t-DT^g+1}^t w_i^g \leq (1 - u_t^g) \quad \forall t \in \{DT^g, \dots, T\} \quad (4.9b)$$

The UC formulation also includes a copper plate relaxation of the AC power flow. Introduced in [19], the copper plate relaxation is a further relaxation of a network flow relaxation. The constraints (4.10) will only require that power generation be able to meet demand, ignoring network constraints altogether. Furthermore, we relax these to allow for load oversatisfaction.

$$\sum_{g \in \mathcal{G}} p_t^g - \sum_{b \in \mathcal{B}} P_{bt}^D - \sum_{b \in \mathcal{B}} G_b^{sh} \geq 0 \quad \forall t \in \mathcal{T} \quad (4.10)$$

Production costs will be modeled using piecewise generation intervals. We will use a set of constraints introduced in [37] to bound the production limits for the piecewise power intervals for each generator. For all generators g with $UT^g > 1$, constraints (4.11) apply.

$$p_t^{gi} \leq (\bar{P}^{gi} - \bar{P}^{g,i-1}) u_t^g - CV^{gi} v_t^g - CW^{gi} w_{t+1}^g \forall i \in \mathcal{I} \quad (4.11)$$

The coefficients CV^{gi} and CW^{gi} in (4.11) are determined in terms of SU and SD in (4.12a) and (4.12b).

$$CV^{gi} := \begin{cases} 0 & \bar{P}^{gi} \leq SU^g \\ \bar{P}^{gi} - SU^g & \bar{P}^{g,i-1} < SU^g < \bar{P}^{gi} \\ \bar{P}^{gi} - \bar{P}^{g,i-1} & \bar{P}^{g,i-1} \geq SU^g \end{cases} \quad (4.12a)$$

$$CW^{gi} := \begin{cases} 0 & \bar{P}^{gi} \leq SD^g \\ \bar{P}^{gi} - SD^g & \bar{P}^{g,i-1} < SD^g < \bar{P}^{gi} \\ \bar{P}^{gi} - \bar{P}^{g,i-1} & \bar{P}^{g,i-1} \geq SD^g \end{cases} \quad (4.12b)$$

For generators with $UT \leq 1$, constraints (4.13) now apply.

$$p_t^{gi} \leq (\bar{P}^{gi} - \bar{P}^{g,i-1}) u_t^g - CV^{gi} v_t^g \forall i \in \mathcal{I} \quad (4.13a)$$

$$p_t^{gi} \leq (\bar{P}^{gi} - \bar{P}^{g,i-1}) u_t^g - CW^{gi} w_{t+1}^g \forall i \in \mathcal{I} \quad (4.13b)$$

Now we can define the production cost above minimum $c_t^{P,g}$ for generator g in time t as (4.14).

$$c_t^{P,g} = \sum_{i \in \mathcal{I}} C^{g,i} \cdot p_t^{gi} \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \quad (4.14)$$

Start up costs will be modeled using the matching formulation developed in [38]. To include these, new variables for the downtime arcs $x(t, t')$ will be introduced for all $t \in \mathcal{T}$, and for all $t' \in \mathcal{T}$ where $t - TC^g < t' \leq t - DT^g$. The new variables are binary such that $x(t, t') = 1$ indicates a start-up in time t and a shutdown in time t' . Then constraints (4.15)

are added.

$$\sum_{t'=t-TC^g+1}^{t-DT^g} x(t', t) \leq v_t^g \quad \forall t \in \mathcal{T} \quad (4.15a)$$

$$\sum_{t'=t+DT^g}^{t+TC^g-1} x(t', t) \leq w_t^g \quad \forall t \in \mathcal{T} \quad (4.15b)$$

Then, we can define $c_t^{SU,g}$ as the start-up cost for generator g at time period t , which can be expressed in terms of the downtime arcs $x(t', t)$ by equation (4.16). We use the notation c^{sg} to denote a generator's cost in start-up category s .

$$c_t^{SU,g} = c^{sg} v_t^g - \sum_{s=1}^{S-1} (c^{sg} - c^{Sg}) \left(\sum_{t'=t-\underline{T}^{s+1}+1}^{t-\underline{T}^s} x(t', t) \right) \quad (4.16)$$

If a generator has a shutdown cost, then it will be formulated as (4.17).

$$c_t^{SD,g} = C^{SD,g} \cdot w_t^g \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \quad (4.17)$$

The objective for the UC model can then be expressed in terms of each generator's production cost $c_t^{P,g}$ start-up cost $c_t^{SU,g}$, minimum run cost $C^{R,g}$, and shutdown cost $c^{SD,g}$

$$\sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} \left(c_t^{P,g} + c_t^{SU,g} + C^{R,g} \cdot u_t^g + c_t^{SD,g} \right) \quad (4.18)$$

4.3.2 ACOPF Formulation

A Benders decomposition algorithm requires both a master problem and at least one subproblem to solve and add cuts to the master. In order to form cuts for our algorithm, we will utilize an ACOPF formulation and its corresponding standard SOCP relaxation. First we will describe the rectangular power voltage (RPQV) formulation for the ACOPF problem for one hourly time interval, which will represent one subproblem formulation. Although there are many equivalent formulations for the ACOPF problem [13, 17] the RPQV model provides a natural basis for describing the SOCP relaxation developed in [35].

To start off the RPQV formulation we introduce constraints (4.19a) and (4.19b) which ensure real and reactive power balance at each bus based on the generation p^g and q^g of all generators at the bus, along with real demand P_b^D , reactive demand Q_b^D , and power flows from and to the bus. Constraints (4.19c)-(4.19f) represent real and reactive power at both the “from” and “to” ends of the transmission lines.

$$\sum_{g \in \mathcal{G}^b} p^g = \sum_{(b,k) \in \mathcal{L}_b^{in}} p_{b,k}^t + \sum_{(b,k) \in \mathcal{L}_b^{out}} p_{b,k}^f + G_b^{sh} v_b^2 + P_b^D \quad \forall b \in \mathcal{B} \quad (4.19a)$$

$$\sum_{g \in \mathcal{G}^b} q^g = \sum_{(b,k) \in \mathcal{L}_b^{in}} q_{b,k}^t + \sum_{(b,k) \in \mathcal{L}_b^{out}} q_{b,k}^f - B_b^{sh} v_b^2 + Q_b^D \quad \forall b \in \mathcal{B} \quad (4.19b)$$

$$p_{b,k}^f = G_{b,k}^{ff} v_b^2 + G_{b,k}^{ft} (v_b^r v_k^r + v_b^j v_k^j) - B_{b,k}^{ft} (v_b^r v_k^j - v_b^j v_k^r) \quad \forall (b,k) \in \mathcal{L} \quad (4.19c)$$

$$q_{b,k}^f = -B_{b,k}^{ff} v_b^2 - B_{b,k}^{ft} (v_b^r v_k^r + v_b^j v_k^j) - G_{b,k}^{ft} (v_b^r v_k^j - v_b^j v_k^r) \quad \forall (b,k) \in \mathcal{L} \quad (4.19d)$$

$$p_{b,k}^t = G_{b,k}^{tt} v_k^2 + G_{b,k}^{tf} (v_k^r v_b^r + v_k^j v_b^j) - B_{b,k}^{tf} (v_k^r v_b^j - v_k^j v_b^r) \quad \forall (b,k) \in \mathcal{L} \quad (4.19e)$$

$$q_{b,k}^t = -B_{b,k}^{tt} v_k^2 - B_{b,k}^{tf} (v_k^r v_b^r + v_k^j v_b^j) - G_{b,k}^{tf} (v_k^r v_b^j - v_k^j v_b^r) \quad \forall (b,k) \in \mathcal{L} \quad (4.19f)$$

Constraints(4.19g) and (4.19h) enforce line thermal limits. We also constrain the squared voltage at each bus in the network with (4.19i).

$$(p_{b,k}^f)^2 + (p_{b,k}^t)^2 \leq (S_{b,k}^{max})^2 \quad \forall (b,k) \in \mathcal{L} \quad (4.19g)$$

$$(p_{b,k}^t)^2 + (p_{b,k}^f)^2 \leq (S_{b,k}^{max})^2 \quad \forall (b,k) \in \mathcal{L} \quad (4.19h)$$

$$(v_b^{min})^2 \leq (v_b)^2 \leq (v_b^{max})^2 \quad \forall b \in \mathcal{B} \quad (4.19i)$$

Finally, we also bound real and reactive power for each generator in the grid with constraints (4.20).

$$\underline{P}^g \cdot u^g \leq p^g \leq \overline{P}^g \cdot u^g \quad \forall g \in \mathcal{G} \quad (4.20a)$$

$$\underline{Q}^g \cdot u^g \leq q^g \leq \overline{Q}^g \cdot u^g \quad \forall g \in \mathcal{G} \quad (4.20b)$$

Constraints (4.19) and (4.20) fully describe the RPQV formulation for AC optimal power flow. We can observe that the RPQV formulation (4.19) and (4.20) is nonconvex and nonlinear due to the line flow equations containing bilinear voltage terms, and also constraints

containing squared terms. Because of the difficulty in solving nonlinear, nonconvex formulations, various convex relaxations have been developed for the AC optimal power flow problem. One of those is the second-order cone programming (SOCP) relaxation first introduced in [35]. We will now formulate the SOCP relaxation, in order to obtain a convex formulation which will render our problem more tractable. To facilitate the convex relaxation, three new sets of variables will now be defined as:

$$c_{b,b} \equiv (v_b^r)^2 + (v_b^j)^2 = v_b^2 \quad (4.21)$$

$$c_{b,k} \equiv v_b^r v_k^r + v_b^j v_k^j = |v_b| |v_k| \cos(\theta_{b,k}) \quad (4.22)$$

$$s_{b,k} \equiv v_b^r v_k^j - v_k^r v_b^j = -|v_b| |v_k| \sin(\theta_{b,k}) \quad (4.23)$$

Then we can perform a substitution of $c_{b,k}$ and $s_{b,k}$ for the constraints (4.19) using the newly defined variables. After the substitution, all bilinear terms and squared voltage terms are gone. The constraints which form the basis for our SOCP relaxation are now introduced:

$$\sum_{g \in \mathcal{G}^b} p^g = \sum_{(b,k) \in \mathcal{L}_b^{in}} p_{b,k}^t + \sum_{(b,k) \in \mathcal{L}_b^{out}} p_{b,k}^f + G_b^{sh} c_{b,b} + P_b^D \quad \forall b \in \mathcal{B} \quad (4.24a)$$

$$\sum_{g \in \mathcal{G}^b} q^g = \sum_{(b,k) \in \mathcal{L}_b^{in}} q_{b,k}^t + \sum_{(b,k) \in \mathcal{L}_b^{out}} q_{b,k}^f - B_b^{sh} c_{b,b} + Q_b^D \quad \forall b \in \mathcal{B} \quad (4.24b)$$

$$p_{b,k}^f = G_{b,k}^{ff} c_{b,b} + G_{b,k}^{ft} c_{b,k} - B_{b,k}^{ft} s_{b,k} \quad \forall (b,k) \in \mathcal{L} \quad (4.24c)$$

$$q_{b,k}^f = -B_{b,k}^{ff} c_{b,b} - B_{b,k}^{ft} c_{b,k} - G_{b,k}^{ft} s_{b,k} \quad \forall (b,k) \in \mathcal{L} \quad (4.24d)$$

$$p_{b,k}^t = G_{b,k}^{tt} c_{k,k} + G_{b,k}^{tf} c_{k,b} - B_{b,k}^{tf} s_{k,b} \quad \forall (b,k) \in \mathcal{L} \quad (4.24e)$$

$$q_{b,k}^t = -B_{b,k}^{tt} c_{k,k} - B_{b,k}^{tf} c_{k,b} - G_{b,k}^{tf} s_{k,b} \quad \forall (b,k) \in \mathcal{L} \quad (4.24f)$$

$$(p_{b,k}^f)^2 + (p_{b,k}^t)^2 \leq (S_{b,k}^{max})^2 \quad \forall (b,k) \in \mathcal{L} \quad (4.24g)$$

$$(p_{b,k}^t)^2 + (p_{b,k}^f)^2 \leq (S_{b,k}^{max})^2 \quad \forall (b,k) \in \mathcal{L} \quad (4.24h)$$

$$(v_b^{min})^2 \leq c_{b,b} \leq (v_b^{max})^2 \quad \forall b \in \mathcal{B} \quad (4.24i)$$

The SOCP relaxation also includes constraints (4.20) which bound real and reactive power generation. To complete the SOCP formulation, we must enforce additional constraints on our new variables. Constraint (4.25) is a result of the definitions of $c_{b,k}$ and $s_{b,k}$. From the pythagorean theorem and our variable definitions we also have constraint (4.26), which describes the surface of a rotated second order cone. Note that this means equation (4.26) is nonconvex.

$$c_{b,k} = c_{k,b} \quad s_{b,k} = -s_{k,b} \quad \forall (b, k) \in \mathcal{L} \quad (4.25)$$

$$c_{b,k}^2 + s_{b,k}^2 = c_{b,b}c_{k,k} \quad \forall (b, k) \in \mathcal{L} \quad (4.26)$$

The ACOPF formulation with constraints (4.20), (4.24), (4.25) and (4.26) is now equivalent to (4.20) and (4.19). In order to convexify the formulation, we perform a relaxation of constraint (4.26) to (4.27).

$$c_{b,k}^2 + s_{b,k}^2 \leq c_{b,b}c_{k,k} \quad \forall (b, k) \in \mathcal{L} \quad (4.27)$$

We have now introduced the full SOCP relaxation for the ACOPF subproblems in our algorithm, which consists of constraints (4.20), (4.24), (4.25), and (4.27). Now that the full UC and ACOPF models have been formulated, the decomposition and solution procedures can be introduced.

4.4 Solution Algorithm

The full UC+ACOPF problem has been defined in Sections 4.3.1 and 4.3.2. Due to the complexity and nonconvexity of the UC+ACOPF problem, solving large instances of this problem is intractable. Thus, solution algorithms such as Benders decomposition and outer approximation [16] are most commonly investigated as possible solution algorithms which can approximate a solution to this problem. The goal of these algorithms are usually to provide solutions which are either closer to being AC feasible than others, or which provide faster computation time. We will now explain a solution algorithm which uses Benders cuts

from the SOCP relaxation of the ACOPF problem to attempt a solution with less violation of the ACOPF constraints than its UC+DCOPF counterpart.

Because we have a 48-hour commitment window and the ACOPF problems can be formulated as time-independent, there is a natural decomposition by time which can be exploited. Therefore our solution algorithm can use a Benders decomposition where the UC problem is the master problem, and each hour can be decomposed into its own ACOPF subproblem. Then a Benders algorithm can be used to add feasibility cuts from the ACOPF subproblems.

To describe our algorithm in finer detail, we present the master problem which is UC formulation defined as (4.28).

$$\begin{aligned}
& \min (4.18) \\
& s.t. (4.1) - (4.11) \\
& \quad (4.13), (4.15)
\end{aligned} \tag{4.28}$$

The subproblems will be created for each time period $t \in \mathcal{T}$. Therefore, for a 48-hour timeframe we would generate 48 subproblems, one for each hour in the commitment horizon. The set of subproblems will be referred to as $\mathcal{S}$, with each individual subproblem referenced as $s \in \mathcal{S}$. In the subproblems, we have the variables p_t^g and u_t^g which appear in constraints (4.20) (4.24a) and (4.24b) also appear in the master problem. These variables are known as the complicating variables. After the master problem is solved, let the optimal values of our complicating variables in the master solution be denoted $((p^g)^*, (u^g)^*)$. Then, for each time period t we have a subproblem s formulated by (4.29). We create a "fake" objective of 0 for our subproblems, because we are only interested in putting up feasibility cuts for the master problem. When all subproblems are feasible we will have a 48-hour day ahead commitment schedule which is feasible for the SOCP relaxat.

$$\begin{aligned}
& \min 0 \\
& s.t. (4.24), (4.25), (4.27), (4.20) \\
& \quad p^g = (p^g)^* \\
& \quad u^g = (u^g)^*
\end{aligned} \tag{4.29}$$

To ensure feasibility of the subproblems, we reformulate by introducing slack on the constraints which belong to the subproblem. Therefore our finalized subproblem formulation is (4.30).

$$\begin{aligned}
& \min z \\
& s.t. (4.24), (4.25), (4.27), (4.20) - z \leq 0 \\
& p^g = (p^g)^* \\
& u^g = (u^g)^*
\end{aligned} \tag{4.30}$$

After solution of each subproblem, $z > 0$ is an indication that the subproblem without slack represented in (4.29) is infeasible. Therefore we will add a feasibility cut of the form (4.31), where μ^g and λ^g represent the dual variables for constraints $p^g = (p^g)^*$ and $u^g = (u^g)^*$, respectively. It is worth noting that the choice to use the SOCP relaxation for our AC power flow constraints ensures that the cuts (4.31) will be valid. The same cannot be said for a nonconvex subproblem.

$$\sum_{g \in \mathcal{G}} [\mu^g(p^g - p^{g*}) + \lambda^g(u^g - u^{g*})] - z \leq 0 \tag{4.31}$$

In the event $z = 0 \forall s \in \mathcal{S}$, the UC schedule given by the master problem is indeed feasible for all subproblems, and the algorithm terminates. See Algorithm 4 for a full outline of the cut generating procedure.

Algorithm 4 Benders-like decomposition algorithm

```

1: for  $i < MAXITERS$  do
2:   Optimize master problem
3:   Fix complicating variables  $u_t^g, p_t^g$ 
4:   for  $s \in \mathcal{S}$  do
5:     Solve subproblem  $s$  formulated as (4.30)
6:     if Objective is nonzero then
7:       Add cut of type (4.31)
8:   if  $i < MAXITERS$  and cuts were added then,
9:      $i = i + 1$ , go to 1
10:  else if No cuts added for any subproblem or  $i == MAXITERS$  then
11:    break

```

4.5 Test Instances

4.5.1 Instance Formulation

We have developed a small set of test instances for UC+ACOPF problems. All of the new cases were created using information from the Power Grid Library (PGLib). Our new instances are a modification of the PGLib optimal power flow (OPF) benchmarks [5]. Details for all networks used in this study can be found there.

Two modifications were made to create our new instances. The first was to replace generators from the PGLib OPF cases with generators from the PGLib unit commitment FERC benchmark cases[39, 40]. For each bus in the OPF case which had a generator on it, three generators were picked to replace the old generator. The new generators attempted to match the maximum power output $\bar{P}^g$ of the old generator as close as possible. The second modification was to scale the load from the FERC benchmark cases and distribute it across the buses in the OPF network which had load values. The FERC load is a 48 hour time series, which for each bus was scaled so that its maximum value matched the load value from the OPF instance. The test instances created are: PJM-5mod, IEEE-14mod, IEEE-30mod, IEEE-57mod. They each contain 15,15,18, and 21 thermal generators, respectively.

4.5.2 Computational Results

The UC+ACOPF Benders algorithm with SOCP subproblems was compared with a UC+DCOPF approximation, and with the UC + Copper Plate relaxation developed in [19]. The four test cases were run with each formulation. Coding was done in python using Pyomo version 5.6.7 [29, 30]. The UC master problem was solved using Gurobi version 8.1.1, while the nonlinear SOCP subproblems were solved with IPOPT [73, 74] version 3.12.13.

Table 4.1 details the number of iterations to convergence for each test instance, as well as computation time of the entire algorithm in seconds. Because the cases are so small, computation time was not an issue. In the future, a test of larger more robust instances will investigate impact of computation time as the complexity of the test instances grows.

Table 4.1: Benders Iterations and Computation Time

Test Case	Benders Iters	Total Time(s)
PJM-5	3	9
IEEE-14	3	10
IEEE-30	6	30
IEEE-57	8	209

Table 4.2: Cost results (\$/MWH) for all test instances

Test Case	Benders SOCP	UC+DCOPF	Copper Plate
PJM-5mod	1,046,063	1,041,153	1,027,710
IEEE-14mod	321,407	297,788	297,662
IEEE-30mod	450,721	425,689	392,248
IEEE-57mod	503,715	361,250	298,098

We can also see from Table 4.1 that each instance was able to solve in well under the 30 iteration limit for the Benders cut algorithm. However, there is still the chance that the algorithm will fail to converge in the allowable iteration limit. If this happens, there are two scenarios. The first is that the master problem becomes infeasible, which may signal infeasibility of the UC+ACOPF problem. The second is the algorithm simply wasn't able to find an acceptable solution in time. In this event, we can note that the first iteration of the Benders algorithm involves solving the master problem with copper plate network constraints. Therefore although this solution is less than desirable, in practice it can provide a low quality fallback solution if one is needed for an immediate UC scheduling.

Table 4.2 lists the cost of each instance with each formulation. The UC with copper plate relaxation has the lowest cost in each case, as it is the most relaxed of all formulations. The UC with Benders feasibility cuts offers the most expensive solution for each case. Because the Benders algorithm terminates with an SOCP feasible solution, we know that this solution is a lower bound for the cost of the true UC+ACOPF problem. Clearly even using the UC+DCOPF may lead to drastic underestimation of the true cost of the UC+ACOPF solution.

Table 4.3: Real Power Slack on Feasibility Check (MW)

Test Case	UC+DCOPF	Benders SOCP
PJM-5mod	2.04	0
IEEE-14mod	5.51	0
IEEE-30mod	6.07	10.00
IEEE-57mod	43.87	2.13

A second analysis was also conducted to investigate the potential AC feasibility of each solution method. To discern whether a solution is AC feasible, we can feed the solution to the standard rectangular ACOPF formulation in section 4.3.2 with the addition of feasibility slack variables for each bus. When the solution is fed in, we also fix all variables of the forms u_t^g and p_t^g . It follows that if the resulting feasibility check solution has all feasibility slack variables take value 0, the original solution was indeed AC feasible. If not, the amount of slack needed to obtain a solution gives a measure of how close the original solution was to being AC feasible. Table 4.3 shows the overall real power slack when each solution for the Benders SOCP and UC+DCOPF in Table 4.2 was run through the feasibility checker. In 3/4 cases, the SOCP solution was closer to ACOPF feasibility than the DCOPF, with two cases being outright AC feasible. These results give indication that on large-scale test instances the Benders SOCP formulation will outperform UC+DCOPF in terms of both cost estimation and AC feasibility.

4.6 Conclusion

This work developed an algorithm for the UC+ACOPF problem which adds Benders feasibility cuts from the SOCP relaxation to the master UC problem with copper plate flow constraints. The algorithm was shown to provide a better lower bound for the UC+ACOPF problem than either the UC+DCOPF or the UC with copper plate alone. Furthermore, the new algorithm with Benders cuts delivers solutions which are closer to AC feasible than the UC+DCOPF alone. Future work will develop a more robust set of large-scale test instances with corresponding AC feasibility check based off of the libraries used for these

relatively small instances. Because current state-of-the-art UC+ACOPF algorithms are too computationally intensive for large-scale instances, the Benders SOCP approach will provide higher quality solutions to problems which currently can only be solved via UC+DCOPF approximations.

Bibliography

- [1] Aghaei, J., Nikoobakht, A., Siano, P., Nayeripour, M., Heidari, A., and Mardaneh, M. (2016). Exploring the reliability effects on the short term ac security-constrained unit commitment: A stochastic evaluation. *Energy*, 114:1016–1032. [86](#)
- [2] Amjady, N. and Ansari, M. R. (2013). Hydrothermal unit commitment with ac constraints by a new solution method based on benders decomposition. *Energy Conversion and Management*, 65:57–65. [87](#)
- [3] Amjady, N., Dehghan, S., Attarha, A., and Conejo, A. J. (2016). Adaptive robust network-constrained ac unit commitment. *IEEE transactions on power systems*, 32(1):672–683. [87](#)
- [4] Asensio, M. and Contreras, J. (2015). Stochastic unit commitment in isolated systems with renewable penetration under cvar assessment. *IEEE Transactions on Smart Grid*, 7(3):1356–1367. [64](#)
- [5] Babaeinejadsarookolaee, S., Birchfield, A., Christie, R. D., Coffrin, C., DeMarco, C., Diao, R., Ferris, M., Fliscounakis, S., Greene, S., Huang, R., et al. (2019). The power grid library for benchmarking ac optimal power flow algorithms. *arXiv preprint arXiv:1908.02788*. [98](#)
- [6] Bai, X. and Wei, H. (2009). Semi-definite programming-based method for security-constrained unit commitment with operational and optimal power flow constraints. *IET Generation, Transmission & Distribution*, 3(2):182–197. [87](#)
- [7] Bai, X., Wei, H., Fujisawa, K., and Wang, Y. (2008). Semidefinite programming for optimal power flow problems. *International Journal of Electrical Power & Energy Systems*, 30(6-7):383–392. [7](#)
- [8] Baran, M. E. and Wu, F. F. (1989). Optimal capacitor placement on radial distribution systems. *IEEE Transactions on power Delivery*, 4(1):725–734. [7](#), [18](#), [49](#)
- [9] Bendotti, P., Fouilhoux, P., and Rottner, C. (2019). On the complexity of the unit commitment problem. *Annals of Operations Research*, 274(1-2):119–130. [86](#)

- [10] Bie, Z., Zhang, P., Li, G., Hua, B., Meehan, M., and Wang, X. (2012). Reliability evaluation of active distribution systems including microgrids. *IEEE Transactions on power systems*, 27(4):2342–2350. [64](#)
- [11] Bienstock, D. and Verma, A. (2015). Strong np-hardness of ac power flows feasibility. *arXiv preprint arXiv:1512.07315*. [84](#), [86](#)
- [12] Bracco, S., Delfino, F., Pampararo, F., Robba, M., and Rossi, M. (2014). A mathematical model for the optimal operation of the university of genoa smart polygeneration microgrid: Evaluation of technical, economic and environmental performance indicators. *Energy*, 64:912–922. [3](#)
- [13] Cain, M., O’Neill, R., and Castillo, A. (2013). Optimal power flow papers: Paper 1. history of optimal power flow and formulations. *Federal Energy Regulatory Commission, Tech. Rep.* [92](#)
- [14] Capitanescu, F. (2016). Critical review of recent advances and further developments needed in ac optimal power flow. *Electric Power Systems Research*, 136:57–68. [7](#)
- [15] Carrión, M. and Arroyo, J. M. (2006). A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. *IEEE Transactions on power systems*, 21(3):1371–1378. [90](#)
- [16] Castillo, A., Laird, C., Silva-Monroy, C. A., Watson, J.-P., and O’Neill, R. P. (2016). The unit commitment problem with ac optimal power flow constraints. *IEEE Transactions on Power Systems*, 31(6):4853–4866. [87](#), [95](#)
- [17] Castillo, A. and O’Neill, R. P. (2013). Survey of approaches to solving the acopf (opf paper 4). *US Federal Energy Regulatory Commission, Tech. Rep.* [92](#)
- [18] Che, L., Khodayar, M., and Shahidehpour, M. (2013). Only connect: Microgrids for distribution system restoration. *IEEE Power and Energy Magazine*, 12(1):70–81. [44](#)
- [19] Coffrin, C., Hijazi, H., and Van Hentenryck, P. (2016). Network flow and copper plate relaxations for ac transmission systems. In *2016 Power Systems Computation Conference (PSCC)*, pages 1–8. IEEE. [84](#), [90](#), [98](#)

- [20] Coffrin, C., Hijazi, H. L., and Van Hentenryck, P. (2015). The qc relaxation: A theoretical and computational study on optimal power flow. *IEEE Transactions on Power Systems*, 31(4):3008–3018. [7](#)
- [21] Damcı-Kurt, P., Küçükyavuz, S., Rajan, D., and Atamtürk, A. (2016). A polyhedral study of production ramping. *Mathematical Programming*, 158(1-2):175–205. [90](#)
- [22] Deckmyn, C., Van de Vyver, J., Vandoorn, T. L., Meersman, B., Desmet, J., and Vandeveld, L. (2017). Day-ahead unit commitment model for microgrids. *IET Generation, Transmission & Distribution*, 11(1):1–9. [44](#)
- [23] Farzin, H., Fotuhi-Firuzabad, M., and Moeini-Aghtaie, M. (2016). Stochastic energy management of microgrids during unscheduled islanding period. *IEEE Transactions on Industrial Informatics*, 13(3):1079–1087. [64](#)
- [24] Fathima, A. H. and Palanisamy, K. (2015). Optimization in microgrids with hybrid energy systems—a review. *Renewable and Sustainable Energy Reviews*, 45:431–446. [45](#)
- [25] Fu, Y., Shahidehpour, M., and Li, Z. (2005). Security-constrained unit commitment with ac constraints. *IEEE transactions on power systems*, 20(3):1538–1550. [87](#)
- [26] Fu, Y., Shahidehpour, M., and Li, Z. (2006). Ac contingency dispatch based on security-constrained unit commitment. *IEEE Transactions on Power Systems*, 21(2):897–908. [87](#)
- [27] Garver, L. L. (1962). Power generation scheduling by integer programming-development of theory. *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems*, 81(3):730–734. [88](#)
- [28] Gentile, C., Morales-España, G., and Ramos, A. (2017). A tight mip formulation of the unit commitment problem with start-up and shut-down constraints. *EURO Journal on Computational Optimization*, 5(1-2):177–201. [89](#)
- [29] Hart, W. E., Laird, C. D., Watson, J.-P., Woodruff, D. L., Hackebeil, G. A., Nicholson, B. L., and Siirola, J. D. (2017). *Pyomo—optimization modeling in python*, volume 67. Springer Science & Business Media, second edition. [9](#), [52](#), [80](#), [98](#)

- [30] Hart, W. E., Watson, J.-P., and Woodruff, D. L. (2011). Pyomo: modeling and solving mathematical programs in python. *Mathematical Programming Computation*, 3(3):219–260. [9](#), [52](#), [80](#), [98](#)
- [31] Hatziargyriou, N. (2014). *Microgrids: architectures and control*. John Wiley & Sons. [64](#)
- [32] Hatziargyriou, N., Asano, H., Iravani, R., and Marnay, C. (2007). Microgrids. *IEEE power and energy magazine*, 5(4):78–94. [1](#), [44](#), [64](#)
- [33] Heymann, B., Bonnans, J. F., Martinon, P., Silva, F. J., Lanas, F., and Jiménez-Estévez, G. (2018). Continuous optimal control approaches to microgrid energy management. *Energy Systems*, 9(1):59–77. [44](#), [45](#)
- [34] Hytowitz, R. B. and Hedman, K. W. (2015). Managing solar uncertainty in microgrid systems with stochastic unit commitment. *Electric Power Systems Research*, 119:111–118. [64](#)
- [35] Jabr, R. A. (2006). Radial distribution load flow using conic programming. *IEEE transactions on power systems*, 21(3):1458–1459. [7](#), [88](#), [92](#), [94](#)
- [36] Khodayar, M. E. (2017). Rural electrification and expansion planning of off-grid microgrids. *The Electricity Journal*, 30(4):68–74. [44](#)
- [37] Knueven, B., Ostrowski, J., and Watson, J.-P. (2017a). Exploiting identical generators in unit commitment. *IEEE Transactions on Power Systems*, 33(4):4496–4507. [91](#)
- [38] Knueven, B., Ostrowski, J., and Watson, J.-P. (2017b). A novel matching formulation for startup costs in unit commitment. [91](#)
- [39] Knueven, B., Ostrowski, J., and Watson, J.-P. (2018). On mixed integer programming formulations for the unit commitment problem. *Pre-print available at http://www.optimization-online.org/DB_HTML/2018/11/6930.html*. [49](#), [90](#), [98](#)
- [40] Krall, E., Higgins, M., and O’Neill, R. P. (2012). Rto unit commitment test system. *Federal Energy Regulatory Commission*. [98](#)

- [41] Kriett, P. O. and Salani, M. (2012). Optimal control of a residential microgrid. *Energy*, 42(1):321–330. [3](#)
- [42] Lasseter, R. H. (2002). Microgrids. In *2002 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No. 02CH37309)*, volume 1, pages 305–308. IEEE. [1](#)
- [43] Lehmann, K., Grastien, A., and Van Hentenryck, P. (2015). Ac-feasibility on tree networks is np-hard. *IEEE Transactions on Power Systems*, 31(1):798–801. [84](#), [86](#)
- [44] Li, N., Chen, L., and Low, S. H. (2012). Exact convex relaxation of opf for radial networks using branch flow model. In *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, pages 7–12. IEEE. [7](#), [18](#), [49](#)
- [45] Liu, J., Bynum, M., Castillo, A., Watson, J.-P., and Laird, C. D. (2018a). A multitree approach for global solution of acopf problems using piecewise outer approximations. *Computers & Chemical Engineering*, 114:145–157. [87](#)
- [46] Liu, J., Laird, C. D., Scott, J. K., Watson, J.-P., and Castillo, A. (2018b). Global solution strategies for the network-constrained unit commitment problem with ac transmission constraints. *IEEE Transactions on Power Systems*, 34(2):1139–1150. [87](#)
- [47] Madani, R., Atamturk, A., and Davoudi, A. (2017). A scalable semidefinite relaxation approach to grid scheduling. *arXiv preprint arXiv:1707.03541*. [87](#)
- [48] Mazidi, M., Zakariazadeh, A., Jadid, S., and Siano, P. (2014). Integrated scheduling of renewable generation and demand response programs in a microgrid. *Energy Conversion and Management*, 86:1118–1127. [64](#)
- [49] Mazzola, S., Vergara, C., Astolfi, M., Li, V., Perez-Arriaga, I., and Macchi, E. (2017). Assessing the value of forecast-based dispatch in the operation of off-grid rural microgrids. *Renewable Energy*, 108:116–125. [45](#)
- [50] Mohammadi, S., Soleymani, S., and Mozafari, B. (2014). Scenario-based stochastic operation management of microgrid including wind, photovoltaic, micro-turbine, fuel cell

- and energy storage devices. *International Journal of Electrical Power & Energy Systems*, 54:525–535. [64](#)
- [51] Moretti, L., Astolfi, M., Vergara, C., Macchi, E., Pérez-Arriaga, J. I., and Manzolini, G. (2019). A design and dispatch optimization algorithm based on mixed integer linear programming for rural electrification. *Applied energy*, 233:1104–1121. [45](#), [64](#)
- [52] Nasri, A., Kazempour, S. J., Conejo, A. J., and Ghandhari, M. (2015). Network-constrained ac unit commitment under uncertainty: a benders’ decomposition approach. *IEEE transactions on power systems*, 31(1):412–422. [87](#)
- [53] Olivares, D. E., Cañizares, C. A., and Kazerani, M. (2014a). A centralized energy management system for isolated microgrids. *IEEE Transactions on smart grid*, 5(4):1864–1875. [44](#), [45](#), [46](#), [64](#)
- [54] Olivares, D. E., Lara, J. D., Cañizares, C. A., and Kazerani, M. (2015). Stochastic-predictive energy management system for isolated microgrids. *IEEE Transactions on Smart Grid*, 6(6):2681–2693. [65](#)
- [55] Olivares, D. E., Mehrizi-Sani, A., Etemadi, A. H., Cañizares, C. A., Iravani, R., Kazerani, M., Hajimiragha, A. H., Gomis-Bellmunt, O., Saeedifard, M., Palma-Behnke, R., et al. (2014b). Trends in microgrid control. *IEEE Transactions on smart grid*, 5(4):1905–1919. [44](#)
- [56] Ostrowski, J., Anjos, M. F., and Vannelli, A. (2011). Tight mixed integer linear programming formulations for the unit commitment problem. *IEEE Transactions on Power Systems*, 27(1):39–46. [90](#)
- [57] Ott, A. L. (2010). Evolution of computing requirements in the pjm market: Past and future. In *IEEE PES General Meeting*, pages 1–4. IEEE. [44](#)
- [58] Palma-Behnke, R., Benavides, C., Lanas, F., Severino, B., Reyes, L., Llanos, J., and Sáez, D. (2013). A microgrid energy management system based on the rolling horizon strategy. *IEEE Transactions on smart grid*, 4(2):996–1006. [3](#), [44](#), [45](#), [65](#)

- [59] Paredes, M., Martins, L., Soares, S., and Ye, H. (2019). Benders’ decomposition of the unit commitment problem with semidefinite relaxation of ac power flow constraints. *arXiv preprint arXiv:1903.02628*. [87](#), [88](#)
- [60] Parhizi, S., Lotfi, H., Khodaei, A., and Bahramirad, S. (2015). State of the art in research on microgrids: A review. *Ieee Access*, 3:890–925. [44](#)
- [61] Rajan, D., Takriti, S., et al. (2005). Minimum up/down polytopes of the unit commitment problem with start-up costs. *IBM Res. Rep*, 23628:1–14. [90](#)
- [62] Rezaei, N. and Kalantar, M. (2015). Stochastic frequency-security constrained energy and reserve management of an inverter interfaced islanded microgrid considering demand response programs. *International Journal of Electrical Power & Energy Systems*, 69:273–286. [64](#)
- [63] Romero-Quete, D. and Cañizares, C. A. (2018). An affine arithmetic-based energy management system for isolated microgrids. *IEEE Transactions on Smart Grid*, 10(3):2989–2998. [44](#), [45](#)
- [64] Sharma, I., Dong, J., Malikopoulos, A. A., Street, M., Ostrowski, J., Kuruganti, T., and Jackson, R. (2016). A modeling framework for optimal energy management of a residential building. *Energy and Buildings*, 130:55–63. [8](#), [51](#)
- [65] Shen, J., Jiang, C., Liu, Y., and Wang, X. (2016). A microgrid energy management system and risk management under an electricity market environment. *IEEE Access*, 4:2349–2356. [64](#)
- [66] Shi, W., Li, N., Chu, C.-C., and Gadh, R. (2015). Real-time energy management in microgrids. *IEEE Transactions on Smart Grid*, 8(1):228–238. [44](#), [45](#)
- [67] Sifuentes, W. S. and Vargas, A. (2007). Hydrothermal scheduling using benders decomposition: accelerating techniques. *IEEE Transactions on Power Systems*, 22(3):1351–1359. [87](#)

- [68] Silvente, J., Kopanos, G. M., Dua, V., and Papageorgiou, L. G. (2018). A rolling horizon approach for optimal management of microgrids under stochastic uncertainty. *Chemical Engineering Research and Design*, 131:293–317. [44](#), [45](#), [65](#), [67](#)
- [69] Silvente, J., Kopanos, G. M., Pistikopoulos, E. N., and Espuña, A. (2015). A rolling horizon optimization framework for the simultaneous energy supply and demand planning in microgrids. *Applied Energy*, 155:485–501. [44](#), [45](#), [65](#)
- [70] Su, W. and Wang, J. (2012). Energy management systems in microgrid operations. *The Electricity Journal*, 25(8):45–60. [44](#)
- [71] Su, W., Wang, J., and Roh, J. (2013). Stochastic energy scheduling in microgrids with intermittent renewable energy resources. *IEEE Transactions on Smart Grid*, 5(4):1876–1883. [65](#)
- [72] Tsikalakis, A. G. and Hatziargyriou, N. D. (2011). Centralized control for optimizing microgrids operation. In *2011 IEEE power and energy society general meeting*, pages 1–8. IEEE. [2](#), [44](#)
- [73] Wächter, A. and Biegler, L. T. (2005). Line search filter methods for nonlinear programming: Motivation and global convergence. *SIAM Journal on Optimization*, 16(1):1–31. [98](#)
- [74] Wächter, A. and Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57. [98](#)
- [75] Zakariazadeh, A., Jadid, S., and Siano, P. (2014). Smart microgrid energy and reserve scheduling with demand response using stochastic optimization. *International Journal of Electrical Power & Energy Systems*, 63:523–533. [65](#)
- [76] Zelazo, D., Dai, R., and Mesbahi, M. (2012). An energy management system for off-grid power systems. *Energy Systems*, 3(2):153–179. [44](#), [45](#)

- [77] Zia, M. F., Elbouchikhi, E., and Benbouzid, M. (2018). Microgrids energy management systems: A critical review on methods, solutions, and prospects. *Applied energy*, 222:1033–1055. [45](#), [64](#)

Vita

Amelia Jane McIlvenna was born March 27, 1991 in Royal Oak, Michigan. She began her collegiate studies at Henry Ford Community College in Dearborn Michigan. After two years she transferred to Oakland University in Rochester, Michigan, obtaining her Bachelors Degree in Mathematics in 2013. She remained at Oakland University for her Masters Degree, graduating in 2015. At Oakland University she received the John Froemke Endowed Graduate Assistantship award, and also taught undergraduate mathematics as a Graduate Teaching Assistant.

In May, 2015 Amelia moved to Knoxville to begin the Ph.D program in Industrial Engineering at University of Tennessee. While a student at UT, she participated in both the Research Alliance in Math and Science internship at Oak Ridge National Laboratory in Oak Ridge, Tennessee, and the Mickey Leland Energy Fellowship at the National Energy Technology Laboratory in Morgantown, West Virginia. Her Ph.D in Industrial Engineering was completed in December, 2019.