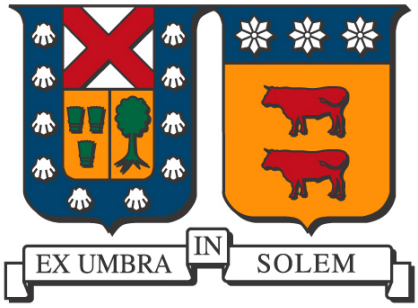

Universidad Técnica Federico Santa María

Department of Electrical Engineering

Robust Co–optimization of Droop and Affine Policy Parameters in Active Distribution Systems with High Penetration of Photovoltaic Generation

Author:
Juan Pablo
Sepúlveda Adriazola

Supervisor:
Dr. Alejandro Alberto
Angulo Cárdenas

A thesis submitted in partial fulfillment of the requirements for the degree of

MSc Electrical Engineering

October 03, 2021

Abstract

Legacy distribution networks need to adapt quickly to upcoming operating scenarios. The expected growth of electromobility and distributed generation will bring forth many challenges regarding efficient operation. Additionally, the limited communication infrastructure of these networks hinders the applicability of heavily centralized control strategies. In this context, this work proposes a distributed control scheme for photovoltaic distributed generators. The scheme comprises a local real–time control layer for the synthesis of active and reactive power references, using traditional voltage droop control and affine policies to handle the uncertainty associated to solar irradiance and demand levels. The characterization and periodic adaptation of such control scheme is achieved via a global, computationally tractable robust model able to co–optimize both droop and affine policy parameters. Performance is benchmarked via extensive computational experiments against other state–of–the–art distributed and local strategies. Results suggest the proposed scheme is able to outperform all competitive frameworks. Its performance is also evaluated as a strictly local control scheme obtaining superior voltage regulation as compared to the recommendations in the Standard IEEE 1547.8.

Acknowledgements

A mi familia por su incondicional apoyo y amor, a mis amigos por su cariño, y al profesor Alejandro Angulo por su paciencia y dedicación.
A la Agencia Nacional de Investigación y Desarrollo (ANID), que por medio del proyecto basal FB0008 "Advanced Center for Electrical and Electronic Engineering, AC3E" y el proyecto Fondecyt Regular Nº1210625, colaboraron con el desarrollo de este trabajo.

Contents

List of Figures

List of Tables

Chapter 1

Motivation

1.1 Motivation

Introducing renewable energy sources has been presented as a solution to meet the ever-increasing demand in power systems. It has many desirable outcomes, such as reducing the environmental impact by displacing other unsustainable energy sources. However, it also presents a significant difficulty: the reliable delivery of electrical power of acceptable quality. The term renewable energy conveys different resources [4]. Between them, we focus on the ones suitable for the distribution level such as wind and solar generation. Each of these renewable options affects reliability and power quality in different and often multiple ways. Here we focus on photovoltaic (PV) solar generation.

Increasing penetration of PV distributed generation poses challenges regarding voltage control and efficient operation of distribution networks. Traditional equipment such as transformers equipped with tap changers and switched capacitors have been rendered insufficient to deal with the uncertain and fast–changing nature of renewable energy supply due to their mechanical limitations [5]. Alternative power electronic devices such as SVC and STATCOMS respond fast enough but are too expensive to be widely deployed [6]. On the other hand, voltage source converters (VSCs) interfacing controlled distributed generators (DGs) can control their active and reactive output fast enough and at a low cost, enabling real–time distributed control.

VSC voltage regulation has been studied extensively in the past decade and is regarded as a viable alternative for the future operation of active distribution networks (ADNs). Based on [7], we classify the works into three main categories: (i) centralized control where a central controller defines the set–points of DGs by solving an optimal power flow (OPF) problem with global information, such as in [1,8–10]; (ii) local control where DG controllers apply easy–to–evaluate functions on local information, e.g., [2, 7, 11–14]; and (iii) Distributed control, based on distributed computation algorithms, either sharing information between neighbors [15–18] or in a hierarchical way between central and local controllers, e.g. [6, 19, 20].

The assumption behind centralized and some distributed approaches is the availability of a robust communication infrastructure, as they rely heavily on sharing information across the system. However, this assumption is not valid for legacy distribution networks with poor communication infrastructure. Moreover, issues regarding communication networks such as heterogeneous communication delays, link failures, and packet losses encourage the application of control schemes that do not overly rely on communication [21]. Cheaper and more easy–to–implement communication technologies such as satellite communication and cellular systems are affected by "considerable" delays [22]. Therefore, our concern is to design a control scheme that is "immune" to delays.

Our work lies in the category of distributed strategies, structured under a two-layer control hierarchy: a layer of local real-time control and an adaptivity layer in charge of periodically updating the local controllers. The adaptivity layer is composed of a central controller that utilizes global system information to calculate optimal control parameters for the local controllers. However, the interval of time between each periodic update is one day. Meanwhile, the local controllers define the DGs' active and reactive power output in real-time by continuously evaluating real-time decision rules based only on local measurements. They simultaneously apply voltage proportional (droop) control and affine policies. The former determines the active and reactive power output of DGs proportionally to voltage variations. Meanwhile, the latter calculates the output of DGs

as a polynomial function of local measurements of the uncertainty, namely, demand and solar irradiance.

Two main problems arise in tuning proportional voltage controllers: system stability and characterization of quasistationary equilibrium. Stability has been handled by imposing bounds on the magnitude of the proportional gains, explicitly as in [18], or, more precisely, by constraining the maximum singular value of the linearized closed–loop system matrix as in [7]. However, the latter condition cannot be handled by off–the–shelf solvers. In [20], the authors proposed a second–order cone inner approximation of the condition in [7], improving computational tractability at the cost of constraining the feasible set. In addition, extra difficulties come from the characterization of quasistationary equilibrium, which naturally introduces products of continuous variables in the control tuning model. Some approaches circumvent the issue by selecting arbitrary stable fixed slopes a priori such as [18] and [6]; and others apply ex–post linear regressions on OPF solutions [15]. In [14], the authors approximate the bilinear terms applying a binary expansion of the feasible set of droop slopes and the Big–M method for the product of binary and continuous variables. In [20], the Neuman series approximation models the controlled system equilibrium without bilinear terms but is not flexible enough to incorporate affine policies. More general approaches to deal with bilinear terms, such as McCormick envelopes [23], provide enough versatility to co–optimize both real–time control rules.

Alternative approaches for the operation of DGs consider control rules that are linear combinations of functions applied to the uncertain data. These control rules are known as linear decision rules or affine policies. Practical applications require sticking to a chosen basis of functions in advance. A basis with too many elements may lead to intractable control tuning models or exhibit in–sample overfitting [24]. At the same time, a basis with low cardinality may not be complex enough to approximate the actual optimal decision rule [25]. Affine policies with a polynomial basis of functions—hereon called polynomial policies— seem to provide enough flexibility for the dispatch of generators in power systems. For instance, the work in [26] applies polynomial policies to the real–time dispatch of generators, but the control tuning model suffers from scalability issues. The work in [12] proposed a tractable approach, but did not consider higher degree polynomials nor active power control. The work in [24] applies polynomial policies to the long–term hydrothermal dispatch problem. They considered higher degree polynomials and utilized AdaLASSO as a regularization technique to reduce overfitting by selecting relevant basis elements [27].

To achieve reliable solutions, the control tuning model must properly characterize the uncertainty associated with demand and solar power availability. In the context of power system operation, three approaches are typically used: stochastic optimization [28], chance constrained optimization [29], and robust optimization [30]. The control tuning model has to be flexible enough to represent control parameters and other operational variables that model the system behaviour. Two–stage optimization is a viable modeling alternative [28]. Robust approaches for DOPF studies applying affine policies have been investigated in [12] under a two–stage affinely adjustable framework. Similar two–stage adjustable robust models have been proposed in [31], [32], and [33] and references therein. Between them, most of the reported works rely on budget–constrained uncertainty sets with a combinatorial number of vertexes as defined in [34], leading to untractable formulations because of the large number of vertices. The works in [1] and [35] apply adaptive data–driven polyhedral uncertainty sets (DDUS), defined as the convex hull of a set of directly observed scenarios, capturing spatial and temporal correlations in actual data. Moreover, the number of vertices of the DDUS is a user–defined parameter, thus resulting in tractable formulations.

In this work, we design local real–time control policies to calculate the power injection of DGs in response to local changes in voltage, demand, and solar irradiance. We define those control policies as parameterizations of the most recent observed values of the quantities mentioned above. In particular, we choose a functional form that conveys linear terms on the voltages and polynomial terms on the demand and solar irradiance observations. A two–stage robust control tuning model is leveraged to obtain the "best" control parameters, considering well–defined operational performance metrics. A central entity periodically generates the DDUS with global information and solves the control tuning problem. The updated control parameters are sent to the local controllers through a non–dedicated wide area network (WAN). In this approach, centralized communication is not a critical task as real–time control policies remain local, and the system can still efficiently operate under a strictly local operation.

1.2 Hypothesis

Consider an active distribution network, equipped with controllable DGs. Let there be controllers associated with the DGs that can determine their active and reactive power output in real–time. Additionally, take into account standard operational performance measures on distribution networks to define efficiency. We claim the following:

It is possible to operate the distribution network, applying robust two-stage optimization to synthesize and tune distributed controllers, achieving more efficient performance than the local network-agnostic control settings recomended by the IEEE-1547 standard.

1.3 Objectives

Main objective

- Design a distributed control scheme for DGs, applied to distribution networks, to achieve better performance than traditional local voltage control, considering high penetration of solar energy.

Specific objectives

- In the context of distribution networks, review the current literature on:
  - Optimal power flow
  - Voltage control strategies
  - Optimization under uncertainty
- Define a stationary model of DGs suitable for optimal power flow formulation.
- Define and implement an optimal power flow model, suitable for optimization using off-the-shelf solvers, considering the equilibrium equation for the closed–loop controlled distribution network.
- Design real–time local control policies for the operation of DGs.
- Implement the required software to solve IEEE instances utilizing the aforementioned scheme.
- Design and develop out-of-sample computational experiments using a set of IEEE instances, under different scenarios of uncontrolled distributed generation, and load.
- Analyze and benchmark the performance of the proposed control strategy by applying statistical methods to the out–of–sample simulation results.

1.4 Thesis outline

This document is structured as follows: Chapter 2 provides the modeling background utilized during all subsequent developments. Chapter 3 characterizes the proposed control design and leverages the control tuning model as a well-defined optimization problem. Then, we describe in detail the solution approach taken to solve the problem mentioned above. Later in chapter 4, we characterize the computational experiments performed to define hyperparameters of the model and evaluate the proposed control strategy's operational performance and alternative approaches. Additionally, we discuss the experimental results and give insight into the properties of the proposed approach. Finally, in chapter 5, we give insight into the most relevant features of the proposed control strategy and raise further questioning for future developments.

Chapter 2

Background

This chapter provides the fundamental modeling techniques utilized throughout this document. First of all, the mathematical notation is stated to disambiguate non-standard symbols. Second, some background on the optimal power flow problem is presented, and two linear models are built for later use. Then, a standard model for the feasible dispatch region of distributed generators is characterized. After that, the concept of robust adjustable optimization is defined, which naturally leads to the notion of two-stage optimization modeling at the end of the section. Finally, the McCormick relaxation is described to deal with bilineary constrained optimization problems such as the proposed control tuning model.

2.1 Mathematical notation

Linear algebra

Operator $(\cdot)^\intercal$ means transposition, $(\cdot)^*$ means complex conjugate, $|\cdot|$ denotes the absolute value of a number or the cardinality of a set. For a given $N \times 1$ vector $\mathbf{x}$: x_i represents the i^{th} element; $\mathbf{x}^m$ denotes its m^{th} element–wise power; and $(\mathbf{x})^+ := [\max\{x_i, 0\}]_{i=1,\dots,N}^\intercal$ denotes the positive part of $\mathbf{x}$. Let $\mathbf{y}$ be another $N \times 1$ vector, then $\mathbf{x} \odot \mathbf{y}$ and $\mathbf{x}/\mathbf{y}$ denote element–wise multiplication and quotient respectively (same notation applies for matrices). For an $N \times N$ matrix $\mathbf{A}$, with eigenvalues $\lambda_1, ..., \lambda_N$, the spectral radius $\rho(\cdot)$ is defined as $\rho(\mathbf{A}) := \max(\{|\lambda_1|, ..., |\lambda_N|\})$. For an $M \times N$ matrix $\mathbf{A}$, the Frobenius norm is $||\mathbf{A}||_F = \sqrt{\text{Tr}(\mathbf{A}^\mathsf{H}\mathbf{A})}$, where $\text{Tr}(\cdot)$ is the trace.

Graphs

Consider a graph $G = (\mathcal{N}, \mathcal{E})$ consisting of a set $\mathcal{N}$ of nodes and a set $\mathcal{E}$ of edges. For undirected graphs, let $i \sim j \in \mathcal{E}$ denote the arc formed by nodes i and j. For directed graphs, let $i \leftarrow j \in \mathcal{E}$, and $j \rightarrow i \in \mathcal{E}$ denote the arc that goes from j towards i. The set of nodes connected to a node i is given by $\mathcal{N}_i := \{j \in \mathcal{N} \mid (i,j) \in \mathcal{E}\}$.

2.2 Optimal power flow

The OPF is a mathematical optimization problem that seeks to minimize a certain function, such as total power loss, generation cost or user disutility, subject to the Kirchoff's laws, as well as capacity, stability, and security constraints. It is a fundamental problem in power systems operation and it is embeded as a subproblem in many standard applications such as economic dispatch, unit commitment, state estimation, stability assesment, volt/var control, demand response, etc [36].

The OPF is NP–Hard in the general case; see [37] for a hardness proof based on a reduction from an instance of the optimal power flow to a complex quadratic optimization problem. Tractable and accurate representations of the OPF problem are desired in practical applications. Consequently, plenty of works leverage solution algorithms and approximations. Some approaches approximate the OPF using linear programming (LP), like the conventional DC OPF [38], or more precise ones such as the LinDistFlow [39] and first–order Taylor (FOT) approximations [40]. Others apply

semidefinite relaxations based on second–order cone programming (SOCP), proposed by [41] for radial networks, and semidefinite programming (SDP), proposed by [42] for general networks.

In this section, we will provide some insights into the OPF modeling and the available solution approaches. Then we will leverage the linear approximations that will be utilized later in this work.

2.2.1 OPF formulations

The fundamental building block used to leverage the OPF is a model of the AC power flow equations which describe the steady state of the power system. Based on [36], we provide some structural properties regarding two alternative models: the bus injection model (BIM) and the branch flow model (BFM). To start, we need the following definitions.

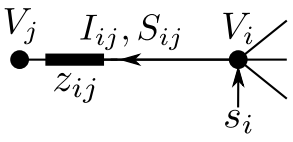

Figure 2.1: Notation summary.

Let $G = (\mathcal{N}^+, \mathcal{E})$ be a connected undirected graph representing the power network, where $\mathcal{N}^+$ is the set of buses and $\mathcal{E} \subseteq \mathcal{N}^+ \times \mathcal{N}^+$ is the set of branches, and let $\mathcal{N} = \mathcal{N}^+ \setminus \{0\}$ be the set of buses excluding the *slack bus*. Let $V_i \in \mathbb{C}$ be the voltage of bus $i \in \mathcal{N}^+$ and assume, without loss of generality, that $V_0 = 1\angle 0$. Let $s_i \in \mathbb{C}$ be the complex power injected at bus $i \in \mathcal{N}$. Let $I_i \in \mathbb{C}$ be the injected current at bus $i \in \mathcal{N}$. Let $I_{ij}, S_{ij} \in \mathbb{C}$ be the complex current and power flowing from bus i to bus j, $(i,j) \in \mathcal{E}$. And, let z_{ij} be the impedance of the branch (i,j), and $y_{ij} = 1/z_{ij}$ be the admittance. See Figure 2.1 for notation reference.

The physics of the network define the following relations:

1. Ohm's law:
$$I_{ij} = y_{ij}\left(V_i - V_j\right), \ \forall\ (i,j) \in \mathcal{E} \tag{2.1}$$

2. Current balance:
$$I_i = \sum_{j:i\sim j} I_{ij}, \ \forall\ i \in \mathcal{N} \tag{2.2}$$

3. Power injection:
$$s_i = V_i I_i^{\mathsf{H}}, \ \forall\ i \in \mathcal{N} \tag{2.3}$$

Bus injection model (BIM)

Substituting I_i in (2.3) as a function of the voltages gives the canonical bus injection model (BIM)

$$s_i = \sum_{j:i\sim j} V_i (V_i - V_j)^* y_{ij}^* \tag{2.4}$$

Let define the solution set of (2.4) as

$$\mathcal{V}(\mathbf{s}) := \left\{V \in \mathbb{C}^{n+1} | V \text{ satisfies } (2.4)\right\} \tag{2.5}$$

The BIM OPF given by

$$\min_V \ \ c(V) \tag{2.6a}$$
$$\text{s.t.} \ \ \check{s}_i \le \sum_{j:i\sim j} V_i (V_i - V_j)^* y_{ij}^* \le \hat{s}_i, \quad \forall i \in \mathcal{N}^+ \tag{2.6b}$$
$$\check{V}_i \le |V_i| \le \hat{V}_i, \quad \forall i \in \mathcal{N}^+ \tag{2.6c}$$

where $c(\cdot)$ is the cost function. Typical costs include the cost of generating real power at each generator bus or line loss over the network. All these costs can be expressed as functions of V.

Branch flow model (BFM)

Let $\tilde{G} = (\mathcal{N}^+, \tilde{\mathcal{E}})$ be a connected directed graph. Where $\mathcal{N}^+$ is the set of nodes and $\tilde{\mathcal{E}} \subseteq \mathcal{N}^+ \times \mathcal{N}^+$ is the set of edges. Let $\tilde{G}$ have an arbitrary orientation and let $i \to j$ denote an edge that goes

from i to j. The branch flow model (BFM) is

$$\sum_{k:j\to k} S_{jk} = \sum_{i:i\to j} \left(S_{ij} - z_{ij}|I_{ij}|^2\right) + s_j,\ j \in \mathcal{N}^+ \tag{2.7a}$$

$$I_{jk} = y_{jk}(V_j - V_k),\ (j \to k) \in \tilde{\mathcal{E}} \tag{2.7b}$$

$$S_{jk} = V_j I_{kj}^* \tag{2.7c}$$

where notation is based on Figure 2.1. For each $\mathbf{s}$, let $\tilde{x} := [S, I, V]^\intercal$ be a solution to (2.7), then define the solution set of the BFM as

$$\tilde{\mathcal{X}}(\mathbf{s}) := \left\{\tilde{x} \in \mathbb{C}^{2m+n+1} | \tilde{x} \text{ satisfies (2.7)}\right\} \tag{2.8}$$

where $m = |\tilde{\mathcal{E}}|$.

The BFM OPF is given by

$$\min_{\tilde{x}}\ c(\tilde{x}) \tag{2.9a}$$

$$\text{s.t. } \tilde{x} \in \tilde{\mathcal{X}}(\mathbf{s}) \tag{2.9b}$$

$$\check{V}_i \le |V_i| \le \hat{V}_i,\ \ \forall i \in \mathcal{N}^+ \tag{2.9c}$$

$$\check{s}_i \le s_i \le \hat{s}_i,\ \ \forall i \in \mathcal{N}^+ \tag{2.9d}$$

where we have abused on notation to write $c(\cdot)$ as a function of $\tilde{x}$. Also, for convenience, we include in the vector V_0 as a variable with equal lower and upper bounds.

Equivalence between BIM and BFM

Eliminating I_{ij} and S_{ij} in (2.7) we obtain the BIM. In the opposite direction, if we consider the definitions (2.7b) and (2.7c) and manipulate (2.4), we obtain the BFM. Formally we say that BIM and BFM are equivalent in the sense that there is a bijection between their solution sets $\mathcal{V}(\mathbf{s})$ and $\mathcal{Y}(\mathbf{y})$. This fact is proven in [43].

2.2.2 OPF convex relaxations

Solving the OPF applying convex relaxations provides some useful guarantees. If an optimal solution of a convex relaxation is feasible for the OPF, then it is the global optimum of the OPF. If it is not feasible for the OPF, it provides a lower bound on the objective function, so if the relaxation is unfeasible, then the OPF is also unfeasible [44]. Naturally, questions arise about tightness, efficiency, and numerical stability guarantees of these formulations. High quality convex relaxations are desirable because the OPF appears as a sub–problem of more complex optimal control problems [1]. In the last years, researchers have been focusing their attention on semidefinite programming (SDP) and second–order cone programming (SOCP) relaxations. Refer to [36, 45] for an introduction on this topic.

The OPF is a quadratically constrained quadratic problem (QCQP) [46]. SDP relaxations of QCQP problems can be derived in a standard way [47]. The core idea behind the SDP relaxations is to shift the non–convexity of the OPF to a rank constraint and then relax it. The hope is the rank condition to be satisfied even without forcing it. If so, the relaxed solution is tight and corresponds to a global optimum of the original problem. Conditions under which the SDP relaxation is exact can be found in [48] and references therein. Nevertheless, it is known that the SDP relaxation is not exact in the general case [49]. Suppose the SDP relaxation solution is not tight. In that case, a feasible solution to the original OPF is not easy to recover and poses extra computational difficulties that make the SDP approach impractical [48, 50].

SOCP relaxations provide a practical approach to solve the OPF with excellent accuracy [51]. The work in [52] presents three different SOCP relaxations, two of them being incomparable[1] to the SDP relaxation. The models result from applying standard strengthening techniques: adding valid inequalities to existing SOCP relaxations based on arctangent envelopes, McCormick envelopes, and a separator algorithm that cuts SOCP optimal solutions that lie outside the SDP relaxation's

[1] Two different relaxations are said to be incomparable when the feasible set of one of them is neither a subset nor a superset of the other.

feasible set. The models are compared against the quadratic cone (QC) relaxation present in [53], which is also incomparable to the SDP relaxation.

For radial networks the recomendation is solving the OPF SOCP relaxations in either BIM or BFM; although, there is evidence that the BFM is more numerically stable [36]. It is important to mention that SOCP relaxations do not offer exactness guarantees when explicit voltage upper bounds bind at optimality [54]. This issue is problematic under high solar penetration scenarios as the voltage rises with the active power injections because of the significant resistance to reactance ratio of distribution networks. For this reason we chose linear models as the ones developed in the next section. However, voltage upper bounds in branch flow SOCP–OPF formulations can be enforced by suitably shaping the power injection feasible set, potentially offering more precise solutions than linear models, although it is not in the scope of this work.

2.2.3 Linear models for radial networks

Define $\mathbf{v} := [|V_i|]_{i\in\mathcal{N}}$ and let $p_i, q_i \in \mathbb{R}$ be the real and imaginary parts of s_i such that $s_i = p_i + \jmath q_i$ and stack them in column vectors $\mathbf{p} := [p_i]_{i\in\mathcal{N}}$, $\mathbf{q} := [q_i]_{i\in\mathcal{N}}$. Consider a function $\mathbf{F} : \mathbb{R}^{2n} \to \mathbb{R}^n$ such that the solution to

$$\mathbf{v} = \mathbf{F}(\mathbf{p}, \mathbf{q}) \tag{2.10}$$

lies in (2.5). Function (2.10) do not exist for all $\mathbf{p}$ and $\mathbf{q}$ because $\mathcal{V}(\mathbf{s})$ may contain more than one element. In general, the existence of (2.10) is related to the question of existence and uniqueness of the power flow solutions, for instance, explicit conditions under which (2.10) exists are investigated in [55]. In this section, we assume that those conditions are met and we focus is on the construction of linear approximations of (2.10) for radial networks, such that

$$\mathbf{v} = \mathbf{B}\tilde{\mathbf{u}} + \mathbf{b} \tag{2.11}$$

where $\tilde{\mathbf{u}} := [\mathbf{p}^\intercal, \mathbf{q}^\intercal]^\intercal$ and the column vector $\mathbf{b}$ represents an intercept.

LinDistFlow

First, here is presented a non–convex relaxation of the branch flow model (2.7) called DistFlow equations, firstly presented in [56] and given by

$$S_{ij} = s_i + \sum_{k:k\to i} (S_{ki} - z_{ki}\ell_{ki}) \qquad \forall (i \to j) \in \mathcal{E} \tag{2.12a}$$

$$V_i = V_j + 2\Re\{z_{ij} S_{ij}\} - |z_{ij}|^2 \ell_{ij} \qquad \forall (i \to j) \in \mathcal{E} \tag{2.12b}$$

$$\ell_{ij} = \frac{|S_{ij}|^2}{v_i} \qquad \forall (i \to j) \in \mathcal{E} \tag{2.12c}$$

Then, following [39], we neglect the terms involving ℓ_{ij}, i.e., $\ell_{ij} \approx 0 \; \forall \; (i \to j) \in \mathcal{E}$. Thus, obtaining the LinDistFlow equations

$$P_{ij} = p_i + \sum_{k:k\to i} P_{ki}, \quad \forall (i \to j) \in \mathcal{E} \tag{2.13a}$$

$$Q_{ij} = q_i + \sum_{k:k\to i} Q_{ki}, \quad \forall (i \to j) \in \mathcal{E} \tag{2.13b}$$

$$|V_i| = |V_j| + r_{ij} P_{ij} + x_{ij} Q_{ij}, \quad \forall (i \to j) \in \mathcal{E} \tag{2.13c}$$

Considering fixed arbitrary enumeration, let stack the variables and parameters into column vectors $\mathbf{v} := [|V_i|]^\intercal_{i\in\mathcal{N}}$, $\mathbf{p} := [p_i]^\intercal_{i\in\mathcal{N}}$, $\mathbf{q} := [q_i]^\intercal_{i\in\mathcal{N}}$, $\mathbf{r} := [r_{ij}]^\intercal_{(i\to j)\in\mathcal{E}}$, $\mathbf{x} := [x_{ij}]^\intercal_{(i\to j)\in\mathcal{E}}$, $\mathbf{P} := [P_{ij}]^\intercal_{(i\to j)\in\mathcal{E}}$, and $\mathbf{Q} := [Q_{ij}]^\intercal_{(i\to j)\in\mathcal{E}}$. Let $\hat{\mathbf{A}} := [\mathbf{a_0}\ \mathbf{A}] \in \mathbb{R}^{N\times(N+1)}$ be the element to node incidence matrix, where a_0 is its first column, and $\mathbf{A} \in \mathbb{R}^{N\times N}$ the bus incidence matrix. Therefore

$$\mathbf{p} = \mathbf{A}^\intercal \mathbf{P} \tag{2.14a}$$

$$\mathbf{q} = \mathbf{A}^\intercal \mathbf{Q} \tag{2.14b}$$

$$\mathbf{A}\mathbf{v} + |V_0|\mathbf{a}_0 = \mathrm{diag}(\mathbf{r})\mathbf{P} + \mathrm{diag}(\mathbf{x})\mathbf{Q} \tag{2.14c}$$

Recall that our goal is to formulate (2.14) in compact form as (2.11). To do so, define $\mathbf{C} := \mathbf{A}^{-1}$ and note that the tree structure of the network allows us to write

$$\mathbf{C}\mathbf{a_0} + \mathbf{1} = \mathbf{0} \tag{2.15}$$

Then we write (2.14) in compact form becoming

$$\mathbf{v} = |V_0|\mathbf{1} + \mathbf{R}\mathbf{p} + \mathbf{X}\mathbf{q} \tag{2.16}$$

where $\mathbf{R} := \mathbf{C}\text{diag}(\mathbf{r})\mathbf{C}^\intercal$ and $\mathbf{X} := \mathbf{C}\text{diag}(\mathbf{x})\mathbf{C}^\intercal$ are symmetric positive definite and have positive entries [7]. Now set $\mathbf{B} = [\mathbf{R}, \mathbf{X}]$ and $\mathbf{b} = |V_0|\mathbf{1}$ and we obtain a linear model as (2.11).

First Order Taylor (FOT)

FOT linearization is a well–known general approach that has been traditionally applied to all sorts of smooth systems of equations. Here, we apply the FOT linearization on (2.17) to instantiate the linear model (2.11), i.e., to calculate some values for $\mathbf{B}$ and $\mathbf{b}$ similar to what was done in Section 2.2.3. Let $\mathbf{s} := [s_i]_{i\in\mathcal{N}}^\intercal \in \mathbb{C}^N$ and $\mathbf{Y} \in \mathbb{C}^{N\times N}$ be the bus admitance matrix. Then (2.4) can be written as

$$V = V_0\mathbf{1} + \mathbf{Y}^{-1}\text{diag}(V)^*\mathbf{s} \tag{2.17}$$

First, we need to calculate the Jacobian $\frac{\partial V}{\partial \tilde{\mathbf{u}}}$. To do so, take partial derivatives of (2.17) with respect to $\tilde{\mathbf{u}}$ to obtain the following

$$\mathbf{U} = \text{diag}(V)\mathbf{Y}^*\frac{\partial V}{\partial \tilde{\mathbf{u}}} + \text{diag}\left(V_0^*\mathbf{1} + \mathbf{Y}^*V^*\right)\frac{\partial V}{\partial \tilde{\mathbf{u}}} \tag{2.18}$$

where $\mathbf{U} := [\mathbf{I}_N, \jmath\mathbf{I}_N] \in \mathbb{C}^{N\times 2N}$ and $\mathbf{I}_N \in \mathbb{R}^{N\times N}$ is the identity matrix. Let $(\overline{V}, \overline{\mathbf{s}})$ be a point satisfying (2.17) and set $V = \overline{V}$ in (2.18) to form a system of linear equations where $\frac{\partial V}{\partial \tilde{\mathbf{u}}} \in \mathbb{C}^{N\times 2N}$ is the unknown matrix. Observe that in rectangular coordinates (2.18) defines a system of $(2N)^2$ real–valued linear equations and variables. To obtain the system in (2.17) we utilize the following derivation rule

$$\frac{\partial|f(x)|}{\partial x} = \frac{1}{|f(x)|}\Re\left\{\frac{\partial f(x)}{\partial x}f(x)^*\right\} \tag{2.19}$$

and set

$$\mathbf{B} = \left.\frac{\partial V}{\partial \tilde{\mathbf{u}}}\right|_{V=\overline{V}} = \text{diag}\left(\overline{V}\right)^{-1}\Re\left\{\text{diag}(\overline{V}^*) + \left.\frac{\partial V}{\partial \tilde{\mathbf{u}}}\right|_{V=\overline{V}}\right\} \tag{2.20}$$

$$\mathbf{b} = \overline{\mathbf{v}} - \mathbf{B}\tilde{\mathbf{u}} \tag{2.21}$$

where $\overline{\mathbf{v}} = [\overline{V}_j]_{j=1,\dots,N}^\intercal$. In [40], an extension of this is presented for unbalanced distribution feeders, along with conditions of existence and uniqueness of the solution.

2.3 Modeling PV DGs

Recent literature highlights the possibility to manage and control DGs as one of the most promising and qualified alternatives for the efficient operation of ADNs [5]. Most of the works focus on solar and wind generators interfaced by a controlled voltage source converter (VSC). Here our focus is on PV generation, for which we give some modeling background.

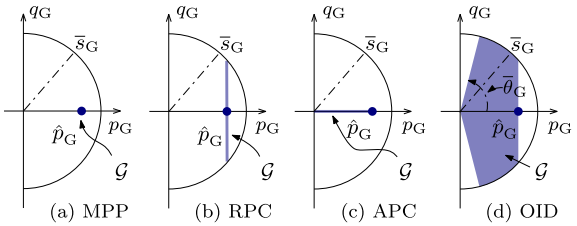

Figure 2.2: Operational chart of different operating modes for the DGs [1].

In steady–state operation, PV VSCs of size $\overline{s}_G$ may inject complex power (p_G, q_G) in four modes, as shown in Figure 2.2. We denominate them maximum power point (MPP), reactive power control (RPC), active power control (APC), and optimal inverter dispatch (OID). In all operating modes, the maximum active power $\hat{p}_G$ depends on solar irradiance and thus is uncertain. They are mathematically described as the set

$$\mathcal{G}(\hat{p}_\text{G}) := \left\{ (p_\text{G}, q_\text{G}) \in \mathbb{R}^2 \;\middle|\; \begin{array}{c} p_\text{G}^2 + q_\text{G}^2 \le \overline{s}_\text{G}^2 \\ 0 \le p_\text{G} \le \hat{p}_\text{G} \\ |q_G| \le \tan\overline{\theta}_\text{G}\left(\hat{p}_\text{G} - p_\text{G}\right) \end{array} \right\} \tag{2.22}$$

where $\overline{\theta}_{\mathrm{G}}$ represents the angle corresponding to the minimum power factor. We have made explicit the dependence of $\mathcal{G}$ with respect to $\hat{p}_{\mathrm{G}}$ to emphasize the impact of the uncertainty. The bound on both the active and reactive power output of DGs is subject to uncertainty. Note that (2.22) defines a tractable cone, i.e., a direct product of half–spaces and second–order cones, therefore is suitable for SOCP optimization models.

2.4 Operation of ADNs under uncertainty

The operation of ADNs with high penetration of DGs based on renewable energy sources (RES) has become one of the most studied topics by researchers and a big challenge for system operators. Traditional practices have been rendered insufficient to deal with the high variability of these sources, raising many concerns regarding efficient and secure operation. The variability of RES hinders the efficient and secure operation of controlled DGs and other active components. The reader may refer to Figure 2.3 for a glance on typical daily profiles of demand and solar available power. Furthermore, high RES penetration may cause reverse power flows, increasing the likelihood of voltages violating prescribed limits.

Consider an ADN under continuous operation conditions where the main sources of uncertainty are demand and RES power availability. The decision–maker has to design adequate policies capable to deal with the uncertainty. The main problem that those policies have to tackle is obtaining a good balance in the trade–off between economic and reliable solutions [57–59]. We classify the different control strategies that have been proposed into two main categories: *(i)* Traditional OPF–based scheduling and *(ii)* real–time adjustable policies.

Traditional OPF–based scheduling regards power output set–points as *here–and–now* decisions, i.e., decisions that have to be made with considerable anticipation, before the uncertainty reveals itself. Under this framework, a central controller periodically sends fixed set–points to the DGs, typically on a 10– to 15–minutes basis. The controller calculates the set–points based on historical observations of the uncertainty and global grid information by solving an optimization model that contains the OPF as a sub–problem. Under this framework, the endeavors have been placed in the uncertainty modeling.

On the other hand, approaches based on real–time adjustable policies regard the output of DGs as *wait–and–see* decisions, i.e., decisions that can be made at time instants when part (or all) of the uncertainty is revealed. A natural way to model real–time adjustable policies is as functions of the uncertainty. For example, a policy that continuously calculates the reactive power output of DGs as a function of maximum available solar power [12]. Moreover, note that the definition is broad enough to account for observed values of any uncertainty affected variable of the system. For example, observed values of voltage at some bus are also a function of the uncertainty. Therefore, traditional control applications such as voltage proportional control also fall into the category of real–time adjustable policies.

At the end, both approaches make *here–and–now* decisions. The former determines the output of DGs, meanwhile the latter determines control policy parameters. Also, uncertainty treatment in the tuning of adjustable policies is similar to the traditional OPF–based approach. Indeed, the main techniques are also stochastic optimization [24], chance constraint optimization [26] and robust optimization [12,20]. Our focus is on the robust case for which we present some background in the following section.

2.4.1 Robust adjustable optimization

In this section we develop relevant ideas regarding robust adjustable optimization based on [30].

General concept

Consider a general–type uncertain optimization problem, a collection

$$\mathcal{P} = \left\{ \min_{\mathbf{y}} \{ f(\mathbf{y}, \boldsymbol{\xi}) \mid F(\mathbf{y}, \boldsymbol{\xi}) \in \mathcal{C} \} \;\middle|\; \boldsymbol{\xi} \in \mathcal{U} \right\} \tag{2.23}$$

where $\mathbf{y} \in \mathbb{R}^n$ is the decision vector, $\boldsymbol{\xi} \in \mathbb{R}^L$ represents the uncertain data, the real–valued function $f(\mathbf{y}, \boldsymbol{\xi})$ is the objective, the vector–valued function $F(\mathbf{y}, \boldsymbol{\xi}) \in \mathbb{R}^m$ along with the set $\mathcal{C}$ specify the constraints, and $\mathcal{U}$ represents the uncertainty set.

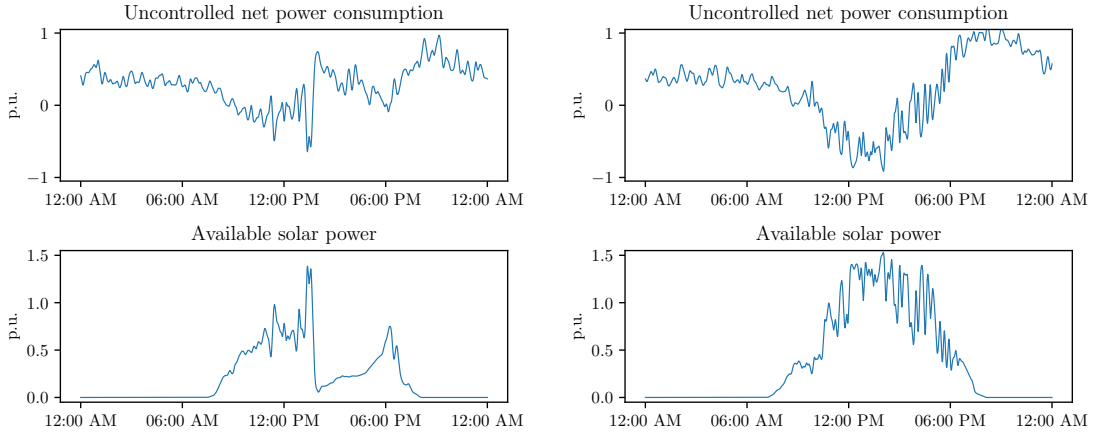

Figure 2.3: Aggregated daily profiles of two consecutive days reconstructed from real data of individual homes.

The construction of a model to obtain a robust optimal solution to (2.23) is not self–evident and depends upon a set of assumptions. For instance, what we traditionally consider as the robust counterpart (RC) of (2.23) depends upon the following design assumptions [30]:

1. All decision variables in (2.23) represent *here–and–now* decisions; they should be assigned specific numerical values as a result of solving the problem before the actual data "reveals itself".

2. The decision maker is fully responsible for consequences of the decisions to be made when, and only when, the uncertainty $\boldsymbol{\xi}$ is within the prespecified uncertainty set $\mathcal{U}$.

3. The constraints in (2.23) are "hard", we cannot tolerate violations of constraints, even small ones, when the uncertainty $\boldsymbol{\xi}$ is in $\mathcal{U}$.

Thereby, traditionally we leverage the RC of (2.23) as

$$\min_{\mathbf{y}} \max_{\boldsymbol{\xi}} \quad f(\mathbf{y}, \boldsymbol{\xi}) \tag{2.24a}$$

$$\text{s.t.} \quad F(\mathbf{y}, \boldsymbol{\xi}) \in \mathcal{C} \tag{2.24b}$$

$$\boldsymbol{\xi} \in \mathcal{U} \tag{2.24c}$$

However, in many cases, not all decisions in (2.23) represent *here–and–now* decisions. For instance, in this work, the dispatch of DGs is calculated in real–time at instants when the uncertainty is "revealing" itself. In other words, they correspond to *wait–and–see* decisions. Hence, we need a more general decision making environment where we drop the first assumption and introduce adjustable variables.

A natural way to model adjustability of operational variables is by letting $\mathbf{y}$ depend on the uncertainty

$$y_j = g_j(\mathbf{P}_j \boldsymbol{\xi}) \tag{2.25}$$

where $\mathbf{P}_1, ..., \mathbf{P}_n$ are given in advance matrices specifying the "information base" of the decisions $\mathbf{y}_j$, and $g(\cdot)$ are the decision rules to be chosen. For a given j, by setting $\mathbf{P}_j$ as the zero matrix, we force $\mathbf{y}_j$ to be completely independent of $\boldsymbol{\xi}$, that is, to be a *here–and–now* decision; by specifying $\mathbf{P}_j$ as the unit matrix, we allow for $\mathbf{y}_j$ to depend on the entire uncertain data; and choosing $1 \leq \text{Rank}(\mathbf{P_j}) < L$ enables the situation where $\mathbf{y}_j$ is only allowed to depend on a portion of the uncertain data.

The rules can, in principle, be arbitrary functions on the corresponding vector spaces. Now

replace the decision variables in (2.24) with the function (2.25), thereby obtaining the following

$$\min_{g(\cdot)} \max_{\boldsymbol{\xi}} \quad h(g(\boldsymbol{\xi})) \tag{2.26a}$$

$$\text{s.t.} \quad F(g(\boldsymbol{\xi}), \boldsymbol{\xi}) \in \mathcal{C} \tag{2.26b}$$

$$g(\boldsymbol{\xi}) = [g_1(\mathbf{P}_1\boldsymbol{\xi}), ..., g_n(\mathbf{P}_n\boldsymbol{\xi})]^\intercal \tag{2.26c}$$

$$\boldsymbol{\xi} \in \mathcal{U} \tag{2.26d}$$

Problem (2.26) is called the adjustable robust counterpart (ARC) of (2.23). Unfortunately, (2.26) is severely computational intractable in the general case. Indeed, it is an infinite–dimensional problem on a set of functions rather than vectors. It seems that the only option to tackle (2.26) is sticking to a chosen in advance parametric family of functions.

Affinely adjustable robust counterpart (AARC)

Here, instead of the general problem (2.23), our starting point is the following affinely–perturbed uncertain conic optimization problem

$$\mathcal{P} = \left\{ \min_{\mathbf{y}} \left\{ \mathbf{c}_\xi^\intercal \mathbf{y} + d_\xi \;\middle|\; \mathbf{A}_\xi \mathbf{y} + \mathbf{b}_\xi \in \mathcal{C} \right\} \;\middle|\; \boldsymbol{\xi} \in \mathcal{U} \right\} \tag{2.27}$$

where $\mathbf{A}_\xi, \mathbf{b}_\xi, \mathbf{y}_\xi, d_\xi$ are affine in $\boldsymbol{\xi}$, $\mathcal{C}$ is a tractable cone (direct product of nonnegative rays/Lorentz cones/Semidefinite cones), and $\mathcal{U}$ is convex compact and is given by a strictly feasible semidefinite representation[2] [60].

As said before, the only option is to restrict (2.25) to a parametric family of functions. In this case, the adjustability is given by the following affine decision rules

$$y_j = g_j(\mathbf{P}_j) = p_j + \mathbf{q}_j^\intercal \mathbf{P}_j \boldsymbol{\xi}, \;\; j = 1, ..., n \tag{2.28}$$

Then, the following is the AARC of (2.27)

$$\min_{\mathbf{P},\mathbf{q}} \max_{\boldsymbol{\xi}} \quad \sum_{j=1}^{n} c_{\xi,j} \left(p_j + \mathbf{q}_j^\intercal \mathbf{P}_j \boldsymbol{\xi} \right) + d_\xi \tag{2.29a}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} \mathbf{A}_{\xi,j} \left(p_j + \mathbf{q}_j^\intercal \mathbf{P}_j \boldsymbol{\xi} \right) + \mathbf{b}_\xi \in \mathcal{C} \tag{2.29b}$$

$$\boldsymbol{\xi} \in \mathcal{U} \tag{2.29c}$$

where $\mathbf{P}$ and $\mathbf{q}$ are the coefficients that define affine decision rules. The reason behind focusing on the affine case rather than other parametric families is that there exists at least one important case when (2.29) is, essentially, as tractable as (2.27). The case in question is the AARC with fixed recourse.

Definition 1. *Consider an uncertain conic problem (2.27) augmented with an information base $\{\mathbf{P}_j\}_{j=1}^n$. We say that this pair has fixed recourse if the coefficients of every adjustable variable y_j are certain:*

$$\forall (j : \mathbf{P_j} \neq \mathbf{0}): \;\; \textit{both } \mathbf{A}_{\xi,j} \textit{ and } c_{\xi,j} \textit{ are independent of } \boldsymbol{\xi}$$

Proposition 1. *Consider (2.27), and $\{\mathbf{P}_j\}$ with fixed recourse, then (2.29) can be formulated as an instance of (2.27).*

Proof. Comprise the affine decision rule parameters into the column vector $\tilde{\mathbf{y}} = (\mathbf{p}, \{\mathbf{q}_j\}_{j=1}^n)$ and define $\tilde{\mathbf{c}}$ and $\tilde{\mathbf{A}}_\xi$ such that $\tilde{\mathbf{c}}_\xi^\intercal \tilde{\mathbf{y}} = \sum_j c_{\xi,j} \left(p_j + \mathbf{q}_j^\intercal \mathbf{P}_j \boldsymbol{\xi} \right)$ and $\tilde{\mathbf{A}}_\xi \tilde{\mathbf{y}} = \sum_j \mathbf{A}_{\xi,j} \left(p_j + \mathbf{q}_j^\intercal \mathbf{P}_j \boldsymbol{\xi} \right)$. Now we can write (2.29) as

$$\min_{\tilde{\mathbf{y}}} \max_{\boldsymbol{\xi}} \quad \tilde{\mathbf{c}}_\xi^\intercal \tilde{\mathbf{y}} + d_\xi \tag{2.30a}$$

$$\text{s.t.} \quad \tilde{\mathbf{A}}_\xi \tilde{\mathbf{y}} + \mathbf{b}_\xi \in \mathcal{C} \tag{2.30b}$$

$$\boldsymbol{\xi} \in \mathcal{U} \tag{2.30c}$$

where $\tilde{\mathbf{A}}_\xi$ and $\tilde{\mathbf{c}}_\xi$ are affine in ξ, therefore, (2.30) is equivalent to (2.27) □

[2] Polyhedra, balls and many other categories of convex sets are semidefinite–representable.

Note that is the fixed recourse assumption what makes the formulation (2.30) possible; without this assumption $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{c}}$ would be quadratic on $\boldsymbol{\xi}$. Proposition 1 constitutes the main reason in favor of affine decision rules, assuming fixed recourse. This is because tractable solution approaches for semi–infinite conic problems such as (2.27) are applicable to (2.29), provided the fixed recourse property.

Although affinely adjustable policies exhibit desirable computational tractability, we may wonder if restricting ourselves to the affine case, instead of arbitrary decision rules, is a dramatic simplification. However, the simplification is not as dramatic as it looks at first glance. For example, consider a decision rule that conveys quadratic terms on the uncertainty. To utilize the same framework, all we need to do is to augment the uncertainty vector $\boldsymbol{\xi}$ by extra entries—the pairwise products $\xi_i\xi_j$—and to treat the extended uncertain vector $\tilde{\boldsymbol{\xi}} = \tilde{\boldsymbol{\xi}}(\boldsymbol{\xi})$ as our new data. By doing this, the decisions that were quadratic in $\boldsymbol{\xi}$ become affine in $\tilde{\boldsymbol{\xi}}$. This redefinition of the uncertainty allows us to incorporate more general decision rules, at the cost of increasing the size of the uncertainty vector.

2.4.2 Two stage robust optimization under scenario–based uncertainty

Observe that to construct the AARC of (2.27) we replaced all of the original decision variables by their corresponding decision rules. This was done under the assumption that all of the original decision variables were control actions. However, in the context of this work not all variables correspond to control actions. For instance, some of them represent the state of the system and others are just auxiliary variables, which by default correspond to freely adjustable decisions. We want a general scheme to represent control parameters as *here–and–now* decisions and the rest of the variables as *wait–and–see* decisions.

Consider a fixed recourse pair consisting of an instance of (2.27) together with an information base $\{\mathbf{P}_j\}_{j=1}^n$. Let the entries in $\mathbf{y}$ be either freely adjustable (with $\mathbf{P}_j \neq 0$) or non–adjustable variables (with $\mathbf{P}_j = 0$). Without loss of generality we can assume that $\mathbf{y} = [\mathbf{u}^\intercal, \mathbf{v}^\intercal]^\intercal$ where $\mathbf{u}$ comprises the non–adjustable variables and $\mathbf{v}$ the freely adjustable variables. The problem can be written as the collection

$$\left\{ \min_{\mathbf{u},\mathbf{v}} \left\{ \mathbf{p}_\xi^\intercal \mathbf{u} + \mathbf{q}^\intercal \mathbf{v} + d_\xi \;\middle|\; \mathbf{P}_\xi \mathbf{u} + \mathbf{Q}\mathbf{v} + \mathbf{r}_\xi \in \mathcal{C} \right\} \;\middle|\; \boldsymbol{\xi} \in \mathcal{U} \right\} \tag{2.31}$$

where[3] $\mathbf{p}_\xi, d_\xi, \mathbf{P}_\xi, \mathbf{r}_\xi$ are affine in ξ. Then, the ARC of (2.31) can be written as the trilevel optimization problem

$$\min_{\mathbf{u}\in\mathcal{W}} \max_{\boldsymbol{\xi}\in\mathcal{U}} \min_{\mathbf{v}\in\mathcal{V}(\mathbf{u},\boldsymbol{\xi})} \mathbf{p}_\xi^\intercal \mathbf{u} + \mathbf{q}^\intercal \mathbf{v} + d_\xi \tag{2.32a}$$

where $\mathcal{V}(\mathbf{u},\boldsymbol{\xi}) := \{\mathbf{v} \mid \mathbf{P}_\xi \mathbf{u} + \mathbf{Q}\mathbf{v} + \mathbf{r}_\xi \in \mathcal{C}\}$ and $\mathcal{W} \subseteq \mathbb{R}^w$ may comprise any constraint that is independent of $\mathbf{v}$ and $\boldsymbol{\xi}$. Although, any constraint can be encoded as a proposition in the definition of $\mathcal{V}(\mathbf{u},\boldsymbol{\xi})$. Problem (2.32) represents a two–stage optimization problem [28], where the first (outer) level decides the non–adjustable variables, the second level finds the worst–case scenario, and the third level determines the adjustable variables.

Now that we have presented (2.32), we focus on a particular case. Simplify the representation of the uncertainty set restricting to the polyhedral case. In particular, consider an uncertainty set described by the convex hull of a set of scenarios

$$\mathcal{U} = \mathrm{Conv}\left(\{\boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_S\}\right) \tag{2.33}$$

Proposition 2. *Solving* (2.32) *with scenario–based uncertainty given by (2.33) is equivalent to solve the following*

$$\min_{t,\mathbf{u},\{\mathbf{v}\}_{s=1}^S} \quad t \tag{2.34a}$$

$$\text{s.t.} \quad \mathbf{p}_{\xi_s}^\intercal \mathbf{u} + \mathbf{q}^\intercal \mathbf{v}_s + d_{\xi_s} \leq t, \quad \forall s \in \{1, ..., S\} \tag{2.34b}$$

$$\mathbf{P}_{\xi_s} \mathbf{u} + \mathbf{Q}\mathbf{v}_s + \mathbf{r}_{\xi_s} \in \mathcal{C}, \quad \forall s \in \{1, ..., S\} \tag{2.34c}$$

[3] Do not confuse the matrix of coefficients $\mathbf{P}_\xi$ with the information base matrices $\mathbf{P}_j,\ j = 1, ..., n$.

Specifically, the optimal values of (2.34) and (2.31) are equal. Moreover, if $\left(\bar{t}, \overline{\mathbf{u}}, \{\overline{\mathbf{v}}\}_{s=1}^{S}\right)$ *is a feasible solution to (2.34) then* $(\bar{t}, \overline{\mathbf{y}})$ *is a feasible solution to (2.29) considering*

$$\overline{\mathbf{v}} = \sum_{s=1}^{S} \boldsymbol{\omega}_s(\boldsymbol{\xi}) \overline{\mathbf{v}}_s \tag{2.35}$$

where $\boldsymbol{\omega}(\boldsymbol{\xi})$ *is an arbitrary vector with non–negative and unit–sum entries such that*

$$\boldsymbol{\xi} = \sum_{s=1}^{S} \boldsymbol{\omega}_s(\boldsymbol{\xi}) \boldsymbol{\xi}_s \tag{2.36}$$

Proof. See Theorem 14.2.3 in [30, Chapter 14] □

Proposition 2 is at the core of this work. Indeed, the solution approach in Section 3.2 is based on approximating the control tuning model (3.18) to an instance of (2.31).

2.4.3 Polyhedral uncertainty set design

The construction of the uncertainty set $\mathcal{U}$ is a critical design problem. In fact, different approaches for the construction of uncertainty sets will also impact the risk adversity of the problem. A straight forward approach, utilized in [31], constructs a polyhedral uncertainty set as a box around the expected value of $\boldsymbol{\xi}$

$$\mathcal{U} = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{N_\xi} \;\middle|\; \sum_{i=1}^{N_\xi} \frac{|\xi_i - \overline{\xi}_i|}{\tilde{\xi}_i} \leq \delta \;,\; \xi_i \in \left[\overline{\xi}_i - \tilde{\xi}_i, \overline{\xi}_i + \tilde{\xi}_i\right] \;\; \forall i \in 1...N_\xi \right\} \tag{2.37}$$

where $\boldsymbol{\xi} = [\xi_i]_{i=1...N_\xi}^{\intercal}$. The variables $\overline{\xi}_i$ and $\tilde{\xi}_i$ represent the expected value and the standard deviation of ξ_i respectively, based on observed values. The hyperparameter δ is interpreted as an uncertainty budget and determines the size of the box, thus controlling the degree of conservativeness of the solution. Nevertheless, this approach does not capture the complex correlations between the elements of $\boldsymbol{\xi}$ as the degree of freedom of each element ξ_i is independent of each other. Another drawback is that the dimension of $\mathcal{U}$ scales with the dimension of $\boldsymbol{\xi}$.

The authors in [35] addressed the aforementioned issues by building a scenario–based polyhedral uncertainty set where each scenario represent an extreme point of $\mathcal{U}$

$$\mathcal{U} = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{N_\xi} \;\middle|\; \boldsymbol{\xi} = \sum_{k \in \mathcal{K}} \alpha_k \boldsymbol{\xi}_k, \sum_{k \in \mathcal{K}} \alpha_k = 1, \alpha_k \geq 0 \right\} \tag{2.38}$$

where $\boldsymbol{\xi}_k, k \in \mathcal{K}$ is an observed realization of $\boldsymbol{\xi}$, i.e. a scenario, and $\mathcal{K}$ is the set of scenarios considered. Note that $\mathcal{U}$ is the convex hull of the selected scenarios capturing observed correlations. In particular, the authors consider the $|\mathcal{K}|$ most recent observations. Then, $|\mathcal{K}|$ is interpreted as a one–dimensional hyperparameter that controls the robustness of the solution, introducing adaptivity through the modification of $|\mathcal{K}|$, as more scenarios considered imply a more conservative solution.

In this work, the adaptivity of the uncertainty set is based on one of the simplest non–parametric machine learning algorithms for classification: the KMeans [3]. Although, more sophisticated adaptivity policies can be implemented. An extensive overview of adaptive data management policies is made in [59] identifying promising non–covered fields of research in this area. The study claims that through machine learning techniques, any strategy based on SO, ChCO, or RO, could be significantly enhanced by the dynamic adaptation of the uncertainty management policies. In fact, in [61] after an extensive comparison between stochastic, robust and CVaR–based models, the results showed optimal values both for budget levels on the RO model, and risk–averse level for the CVaR–based model, where the out of sample results reached a minimum value. Concurrently, in [35], several experiments with uncertainty sets containing different number of scenarios showed significant differences in the trade–off between robustness and cost.

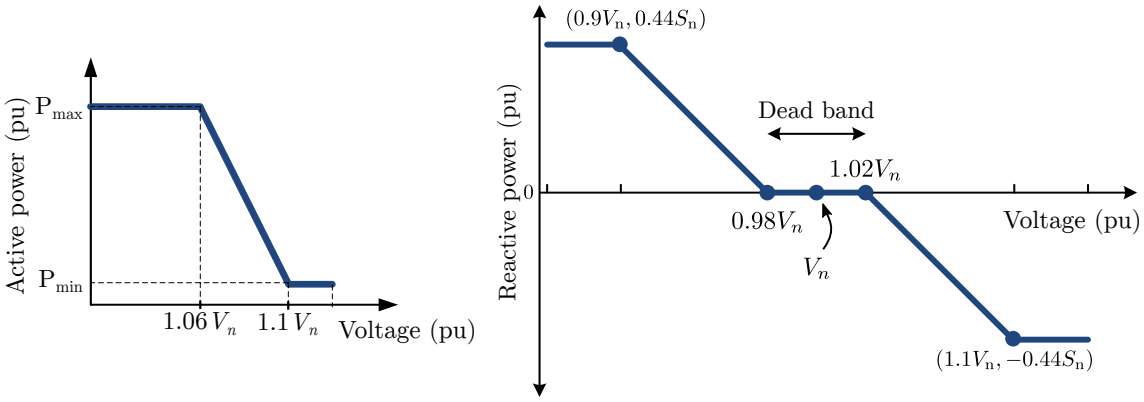

Figure 2.4: IEEE–1547 control rule [2].

2.4.4 Droop control

The concept of droop control ambiguously applies to a variety of controllers. As the name suggests, its fundamental feature is that the controlled variable is a nonincreasing continuous function of some input. The system designer is in charge of determining the shape of the droop policy. Thus, a variety of droop controllers have been proposed in the form of linear [20], piecewise linear [2], and quadratic functions [11]. The classic example corresponds to frequency control utilizing a speed–droop characteristic where variations in frequency are proportional to variations in power output of generators [62, Chapter 10]. On the same line, different droop control strategies have been applied to achieve primary control goals such as stability and load sharing in the context of microgrids [63]. Additionally, many droop control strategies have been recently proposed to achieve voltage regulation, both in microgrids and ADNs.

For instance, the IEEE 1547.8 Standard [2] recommends the application of droop control for the dispatch of DGs using the policies shown in Figure 2.4. Where S_n correspond to the nominal rating of the DG, P_max is the maximum available power, P_min is the minimum active power, and V_n is the nominal voltage typically set to 1 (pu). Its goal is to support voltage regulation in distribution networks. Note that the settings of this droop controller depend only on the attributes of each generator, being network agnostic. On one side, this fact is convenient because there is no need for global information. However, ignoring system wide information may lead to heavily suboptimal operational performance. Therefore, in this work we propose a systematic approach to design droop characteristics based on global network information. In this work, the focus is on improving operational performance, which takes us away from the task of explicitly controlling voltage and raises questions regarding optimal control settings to minimize total energy consumption while operating within a prescribed voltage band. This naturally drives us to leverage an optimization model to determine the control parameters.

Two main problems arise in tuning voltage droop controllers: system stability and characterization of quasistationary equilibrium. Stability has been handled by imposing bounds on the magnitude of the proportional gains, explicitly as in [18], or, more precisely, by constraining the maximum singular value of the linearized closed–loop system matrix as in [7]. However, the latter condition cannot be handled by off–the–shelf solvers. In [20], the authors proposed a second–order cone inner approximation of the condition in [7], improving computational tractability at the cost of constraining the feasible set.

In addition, extra difficulties come from the characterization of quasistationary equilibrium, which naturally introduces products of continuous variables in the control tuning model. Some approaches circumvent the issue by selecting arbitrary stable fixed slopes a priori such as [18] and [6]. Others apply ex–post linear regressions on OPF solutions [15]. In [14], the authors approximate the bilinear terms applying a binary expansion of the feasible set of droop slopes and the Big–M method for the product of binary and continuous variables. In [20], the Neuman series approximation models the controlled system equilibrium without bilinear terms but is not flexible enough to incorporate affine policies. More general approaches to deal with bilinear terms, such as McCormick envelopes [23], provide enough versatility to co–optimize both real–time control rules.

2.5 Model selection: AdaLASSO

In this work, the controller is a parametric function of both endogenous variables representing the state of the system, and exogenous variables, i.e., the uncertainty. The out–of–sample performance of the proposed controller relates to its prediction capability on unknown data that is revealed a posteriori. Assessment of this performance is extremely important in practice, since it guides the controller design, and gives us a measure of the quality of the ultimately chosen model. Focusing on out–of–sample performance rather than in–sample brings some issues to the table, namely, the so called bias–variance trade–off. Here we give insight into this issue and present some tools to deal with it.

2.5.1 The LASSO

Consider a linear regression model of the form

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j \tag{2.39}$$

where $\beta_0, ..., \beta_p$ are the unknown parameters, and $X_1, ..., X_p$ are the inputs modeled as random variables. Note the similarities between the affine policies (2.28) and (2.39).

Typically we have a set of in–sample data $(\mathbf{X}, \mathbf{y})$ from which to estimate the parameters, where $\mathbf{X} \in \mathbb{R}^{N \times (p+1)}$ and $\mathbf{y} \in \mathbb{R}^N$. The most popular estimation method is the well known least squares estimator defined as

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\intercal (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \tag{2.40}$$

where $\boldsymbol{\beta} = [\beta_0, ..., \beta_p]^\intercal$. Of course, $\mathbf{X}$ has a $\mathbf{1}$ as the first column.

We are often not satisfied with the least squares estimation because of its out–of–sample prediction accuracy. It often has low bias but large variance. Prediction accuracy can sometimes be improved by shrinking or setting some coefficients to zero. By doing so, we sacrifice a little bit of bias to reduce the variance of the predicted values and hence may improve overall out–of–sample prediction accuracy.

There exist many approaches for subset selection such as forward– and backward–stepwise selection, forward–stagewise selection, and best–subset selection by incorporating zero–norm constraints in the optimization problem embedded in (2.40); refer to [3, Chapter 3] for insight into these techniques. All of them act by retaining a subset of the predictors and discarding the rest. Subset selection methods produce a model that is interpretable and has possibly lower prediction error than the full model. However, they are not computationally scalable to the extent that we need in this work.

On the other hand, shrinking methods act by imposing a penalty on the size of the model. In general, they are more computationally tractable. The most common methods are the Ridge and the LASSO. They differ in the penalty function that is introduced. For instance, the following is the LASSO estimator

$$\hat{\boldsymbol{\beta}}_{\text{lasso}} = \arg\min_{\boldsymbol{\beta}} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\intercal (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda \sum_{j=0}^{p} |\beta_j| \tag{2.41}$$

where λ is a weighting factor. The penalty function $\lambda \sum_{j=0}^{n} |\beta_j|$ continuously shrinks the model toward zero. Indeed, making λ sufficiently large will cause some of the coefficients to be exactly zero. Hence, the LASSO does a kind of continuous subset selection. In the signal processing literature, the LASSO is known as *basis pursuit*, which gives more insight into its purpose.

Effective degrees of freedom

Consider a linear regression using a subset of k input variables. If this subset is prespecified a priori without reference to the training data, then the number of degrees of freedom used in the fitted model is defined as k. Alternatively, suppose that we carry out a best subset selection to determine the "optimal" subset of k predictors. Then, the resulting model has k parameters, but, in some sense, we have used up more than k degrees of freedom. Similarly, if we obtain the LASSO

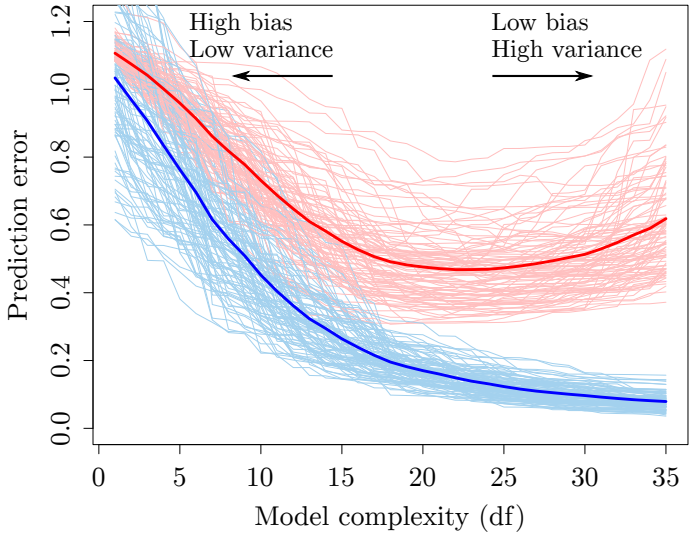

Figure 2.5: Behavior of in–sample and out–of–sample error as the model complexity is varied. Light blue curves show the in–sample error, while the red curves show the out–of–sample error [3].

estimator utilizing some $\lambda > 0$ and all the resulting predictors are non–zero values, how many degrees of freedom have we used?

There is a need for a more general definition of degrees of freedom. Suppose we stack the realizations of Y into the vector $\mathbf{y}$ and the corresponding predictions into the vector $\hat{\mathbf{y}}$. Then we can write

$$\hat{\mathbf{y}} = \mathbf{S}\mathbf{y} \tag{2.42}$$

where $\mathbf{S}$ is a matrix depending on the input data $\mathbf{X}$ but not on $\mathbf{y}$. Then, the effective degrees of freedom is defined as

$$\mathrm{df}(\mathbf{S}) = \mathrm{Tr}(\mathbf{S}) \tag{2.43}$$

2.5.2 Bias–variance trade–off and model complexity

Consider the case of a quantitative prediction model. We have the following random variables: a target variable Y, a vector of inputs X, and a prediction model $\hat{f}(X)$. The loss function for measuring errors between Y and $\hat{f}(X)$ is denoted by $L(Y, \hat{f}(x))$. Typical choices are

$$L(Y, \hat{f}(X)) = \begin{cases} (Y - \hat{f}(X))^2 & \text{squared error} \\ |Y - \hat{f}(X)| & \text{absolute error} \end{cases} \tag{2.44}$$

The out–of–sample error, also known as the *test error* or the *generalization error*, is the prediction error over a data sample $\mathcal{T}$ that was not utilized in the fitting of $\hat{f}(X)$, given by

$$\mathrm{Err}_{\mathcal{T}} = \mathbb{E}\left[L(Y, \hat{f}(X)) \mid \mathcal{T}\right] \tag{2.45}$$

where both X and Y are drawn randomly from their joint distribution (population). Here the out–of–sample data set $\mathcal{T}$ is fixed, and the out–of–sample error refers to the error of this specific out–of–sample data set. The quantity of interest is

$$\mathrm{Err} = \mathbb{E}\left[L(Y, \hat{f}(X))\right] = \mathbb{E}\left[\mathrm{Err}_{\mathcal{T}}\right] \tag{2.46}$$

Note that this expectation averages over everything that is random, including the randomness in the out–of–sample data that produces $\hat{f}(\cdot)$. On the other hand, the average in–sample error is defined by

$$\mathrm{err} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(\mathbf{x}_i)) \tag{2.47}$$

where N is the number of in–sample data points $(y_i, \mathbf{x}_i)$ and the variables $\mathbf{x}_1, ..., \mathbf{x}_N$ represent the rows of the in–sample data matrix $\mathbf{X}$.

Figure 2.5 shows the out–of–sample error (light red curves) $\text{Err}_{\mathcal{T}}$ for 100 simulated data sets, each of size 50. The LASSO was used to produce the sequence of fits. The light red curves show the out–of–sample error $\text{Err}_{\mathcal{T}}$ given a particular data set. The solid red curve is the average, and hence an estimate of Err. As the model becomes more complex, i.e., with more effective degrees of freedom (df), it can adapt to more complicated underlying structures. Hence, there is a decrease in bias but an increase in variance. Note that there is a desired level of complexity somewhere in the middle that maximizes out–of–sample prediction accuracy. Also, Figure 2.5 shows the average in–sample error over a particular data set (light blue curves), and the expected error $\mathbb{E}\left[\overline{\text{err}}\right]$ (solid blue). Note that the in–sample error consistently decreases with model complexity. Indeed, if the model complexity is high enough, typically, it can achieve zero in–sample error. However, a model with zero in–sample error usually generalizes bad to out–of–sample data, and we say that it suffers from in–sample overfitting.

2.5.3 The AdaLASSO

Assume that the true estimator can be represented as a linear function, so that

$$\mathbb{E}(Y|X) = \sum_{j=1}^{p} X_j \beta_j^* \tag{2.48}$$

Without loss of generality, we assume that the data is centered, i.e., $\beta_0 = 0$. Let $\mathcal{A} = \left\{j : \beta_j^* \neq 0\right\}$ and $|\mathcal{A}| = p_0 < p$, meaning that (2.48) depend only on a subset of the predictors.

Consider the LASSO estimate

$$\hat{\boldsymbol{\beta}}_{\text{lasso}}^{(N)} = \arg\min_{\boldsymbol{\beta}} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\intercal}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda_N \sum_{j=0}^{p} |\beta_j| \tag{2.49}$$

where λ_N varies with N. Let $\mathcal{A}_N = \left\{j : \hat{\beta}_{\text{lasso},j}^{(N)} \neq 0\right\}$. The lasso estimation procedure is consistent if and only if it can asymptotically identify the true underlying model, i.e.,

$$\lim_{N\to\infty} \mathbb{P}(\mathcal{A}_N = \mathcal{A}) = 1 \tag{2.50}$$

In addition to consistency, we would want an estimation procedure that rapidly converges to the true model. For that, the asymptotic normality condition

$$\sqrt{N}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{(N)} - \hat{\boldsymbol{\beta}}_{\mathcal{A}}^{*}) \to_d \mathcal{N}(0, \boldsymbol{\Sigma}^*) \tag{2.51}$$

is required. Meaning that $\sqrt{N}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{(N)} - \hat{\boldsymbol{\beta}}_{\mathcal{A}}^{*})$ distributes normally, with covariance matrix $\boldsymbol{\Sigma}^*$, as N goes to infinity. Details about (2.51) can be found in [27] together with the definition of $\boldsymbol{\Sigma}^*$. Note that the required rate of convergence is $\sqrt{N}$. The aforementioned conditions, regarding consistency and convergence rate, are known as the oracle properties [64]. The authors in [27] show that the LASSO does not hold the oracle properties, but they "fix" the LASSO by modifying the penalty function leveraging a fitting procedure that does hold the oracle properties: the AdaLASSO.

The AdaLASSO estimate is given by

$$\hat{\boldsymbol{\beta}}_{\text{ada}}^{(N)} = \arg\min_{\boldsymbol{\beta}} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\intercal}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda_N \sum_{j=0}^{p} \hat{\omega}_j |\beta_j| \tag{2.52}$$

where $\hat{\omega}_j = 1/|\hat{\beta}_{\text{ols},j}|^{\gamma}$, $\hat{\boldsymbol{\beta}}_{\text{ols}}$ is the ordinary least squares estimator, and $\gamma > 0$. Note that solving the underlying optimization problem in (2.52) is as efficient as solving (2.41). However, we need to calculate $\hat{\boldsymbol{\beta}}_{\text{ols}}$ before obtaining the AdaLASSO estimator. Note that, similarly to the LASSO, λ_N is a hyper–parameter that must be defined a priori.

2.6 McCormick envelopes

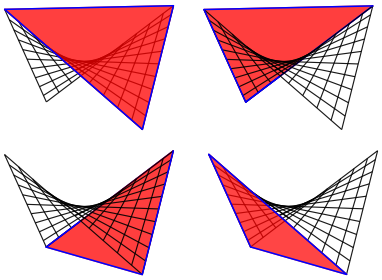

Figure 2.6: McCormick envelopes facets.

A common technique applied in global optimization is the construction of convex envelopes to non–convex functions. There is a general approach proposed by McCormick [65] that is based on a polyhedral relaxation for bilinear terms with bounded variables. Consider the following set

$$\mathcal{B} = \left\{ (x, y, z) \in \mathbb{R}^3 \;\middle|\; \begin{array}{c} z = xy,\ l_x \leq x \leq u_x, \\ l_y \leq y \leq u_y \end{array} \right\} \tag{2.53}$$

where l_x, u_x, l_y and u_y are known constants. The McCormick envelopes of $\mathcal{B}$ describe a polyhedron $\mathcal{C}$ given by

$$\mathcal{C} = \left\{ (x, y, z) \in \mathbb{R}^3 \;\middle|\; \begin{array}{c} -z + l_x y + l_y x \leq l_x l_y, \\ -z + u_x y + u_y x \leq u_x u_y, \\ -z + u_x y + l_y x \geq u_x l_y, \\ -z + l_x y + u_y x \geq l_x u_y \end{array} \right\} \tag{2.54}$$

Figure 2.6 shows the surface $\mathcal{B}$ in black, and the four facets of $\mathcal{C}$ in red, one at a time. In [66], it was proven that

$$\mathcal{C} = \text{Conv}\,(\mathcal{B}) \tag{2.55}$$

In [67], the authors investigate the gap between bilinear functions with multiple terms and the corresponding McCormick relaxation. In [23], the authors study the convex hull of $\mathcal{B}$ when (x, y) lie in a box and there are explicit upper and/or lower bounds on the product xy.

Further, as we will present later, in this work we are not only interested in the results of [23], but also in the case when there are products of variables $z_1 = x_1 y_1$ and $z_2 = x_2 y_2$ such that they belong to a translated second order cone $(a + z_1)^2 + (b + z_2)^2 < c$, where a, b and c are constant scalars.

Chapter 3

Methodology

This chapter characterizes the main problem of this work in terms of a mathematical optimization model and then presents the proposed solution approach in detail. First, a general discrete–time model of the closed-loop controlled ADN is presented as the starting point. Second, the controllers and the system equilibrium are represented in terms of a tractable model, together with stability conditions. Consequently, the pursuit of high–performance control settings is translated into a control tuning optimization model, which constitutes the main problem to tackle. The difficulties regarding the control tuning model raise the need for a convenient reformulation of the original problem. Therefore, the final part of this chapter deals with leveraging a tractable SOCP model approximation.

3.1 Problem formulation

As mentioned before, the main objective is to leverage a control strategy for DGs to improve system–wide operational performance. We assume an ADN with poor communication infrastructure, i.e., typical delays and transient communication failures. Therefore, the design principle is to consider real–time operation as a critical task that cannot directly depends on communication between buses. This local approach implies a fundamental limit in information availability. Although, to achieve near–optimal operational performance, there is a need for network-wide information in some way. Therefore, we also consider another control layer, which updates the local controllers once a day. The idea is to for the updating time–intervals to be so long that they are immune to typical delays and transient communication failures in non–dedicated cellular o satellite based technologies [22].

Figure 3.1 presents a simple instance of the cyber-physical system that supports the proposed strategy. It contains all the fundamental elements: a distribution network, controlled DGs, and a WAN. Each controlled DG operates based on local observed values of voltage, demand, and active power availability. Note that, unlike classical voltage droop control, we additionally incorporate real-time local measurements of the uncertainty. They give extra information for the controllers to compensate for their effect and achieve efficient operation.

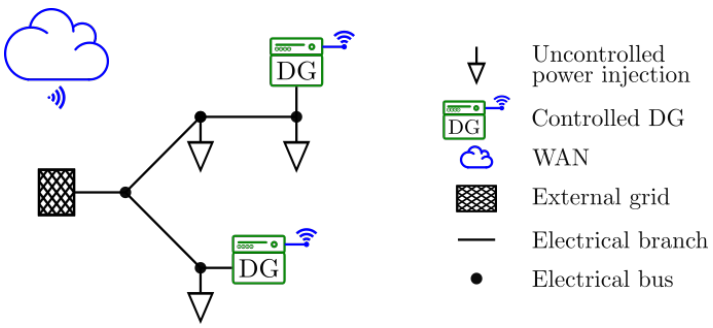

Figure 3.1: General overview of the system elements.

Controlled DGs actuate by injecting active and reactive power at their corresponding buses. The control policy is the function that maps local measurements to actuation. The fundamental problem is to determine those control policies optimally. To do so, a central entity that may be an arbitrarily chosen local controller calculates those policies and periodically updates all the local controllers through a WAN. Therefore, the real-time operation and the adaptivity are separate tasks.

Consider a radial active distribution network provided with controlled DGs, let $\mathcal{N}^+ := \mathcal{N} \cup \{0\}$ be the set of buses, and $\mathcal{N} := \{1, ..., N\}$ the set of buses without the substation node 0. Let k be the discrete–time index, and $\mathbf{x}_k \in \mathbb{R}^N$ be the vector of states at time k, representing voltage variations such that $\mathbf{x}_k := \mathbf{v}_k - \overline{\mathbf{v}}$, where $\mathbf{v}_k$ is the vector of bus voltage magnitudes and $\overline{\mathbf{v}}$ is the nominal set–point. Let $\mathbf{p}_{\mathrm{G},k} \in \mathbb{R}^N$ and $\mathbf{q}_{\mathrm{G},k} \in \mathbb{R}^N$ be the vectors of controlled active and reactive bus power injections respectively, and define their variations $\boldsymbol{\Delta}\mathbf{p}_{\mathrm{G},k} := \mathbf{p}_{\mathrm{G},k} - \overline{\mathbf{p}}_{\mathrm{G}}$ and $\boldsymbol{\Delta}\mathbf{q}_{\mathrm{G},k} := \mathbf{q}_{\mathrm{G},k} - \overline{\mathbf{q}}_{\mathrm{G}}$ with respect to the nominal set–points $\overline{\mathbf{p}}_{\mathrm{G}}$ and $\overline{\mathbf{q}}_{\mathrm{G}}$. Comprise the variations into the vector of control actions $\mathbf{u}_k := [\boldsymbol{\Delta}\mathbf{p}_{\mathrm{G},k}^\intercal, \boldsymbol{\Delta}\mathbf{q}_{\mathrm{G},k}^\intercal]^\intercal$. Similarly, stack the nominal set–point vectors into $\overline{\mathbf{u}} := [\overline{\mathbf{p}}_{\mathrm{G}}^\intercal, \overline{\mathbf{q}}_{\mathrm{G}}^\intercal]^\intercal$. Additionally, let $\boldsymbol{\xi}_{\mathrm{L},k} := [\mathbf{p}_{\mathrm{L}}^\intercal, \mathbf{q}_{\mathrm{L}}^\intercal]^\intercal$ be the column vector of uncontrolled power injection, where $\mathbf{p}_{\mathrm{L},k} \in \mathbb{R}^N$ and $\mathbf{q}_{\mathrm{L},k} \in \mathbb{R}^N$ are the active and reactive components. Moreover, define the vector of active power availability as $\hat{\mathbf{p}}_{\mathrm{G},k}$ and comprise all the uncertain realizations in the column vector $\boldsymbol{\xi}_k := [\boldsymbol{\xi}_{\mathrm{L},k}^\intercal, \hat{\mathbf{p}}_{\mathrm{G},k}^\intercal]^\intercal$.

The following equations represent the discrete–time quasistationary model of the closed–loop controlled system

$$\mathbf{x}_k = f\left(\mathbf{u}_k, \boldsymbol{\xi}_{\mathrm{L},k}\right) \tag{3.1a}$$

$$\mathbf{u}_k = g\left(\mathbf{x}_{k-1}, \boldsymbol{\xi}_{k-1}\right) \tag{3.1b}$$

The plant is modeled by (3.1a) and it represents the non–linear AC power flow equations for balanced systems. Under mild conditions as described in [40], we assume that for each pair $(\mathbf{u}, \boldsymbol{\xi})$ there exists a unique value of $\mathbf{x}$ so we can write (3.1a). Meanwhile, (3.1b) models the application of the proposed control policy $g(\cdot)$. The notation used in (3.1) allows to clearly identify the relations between the state of the system $\mathbf{x}$, the control actions $\mathbf{u}$, and the uncertainty $\boldsymbol{\xi}$. We consider all buses in $\mathcal{N}$ as P–Q buses, therefore, bus voltages are controlled by the active and reactive power injections comprised in $\mathbf{u}$.

The overall objective is to design $g(\cdot)$ in (3.1b) to efficiently and securely operate the distribution network. More precisely, to minimize system–wide energy consumption while maintaining acceptable voltage levels under "continuous operation" conditions as defined by [2]. In the rest of this section, we define the elements of our proposed strategy.

3.1.1 Controllers

To leverage a systematic methodology for the design of $g(\cdot)$, is necessary to arbitrarily restrict $g(\cdot)$ to a particular parametric family of functions. The family of control policies should be flexible enough to approximate the optimal behavior of DGs. However, the flexibility comes with the price of needing many parameters in the representation of the controllers, thereby introducing tractability concerns. Considering the trade–off between flexibility and tractability we propose the following control policy

$$g(\mathbf{x}, \boldsymbol{\xi}) := \underset{\mathcal{G}(\boldsymbol{\xi})}{\mathrm{proj}} \left(\mathbf{G}\mathbf{x} + \sum_{j=0}^{d} \mathbf{D}_j \psi^j\left(\boldsymbol{\xi}\right) \right) \tag{3.2}$$

where the matrices $\mathbf{D}_0, ..., \mathbf{D}_d$ and $\mathbf{G}$ comprise the control parameters, the scaling function $\psi(\cdot)$ normalizes the values of $\boldsymbol{\xi}$, and the operator $\mathrm{proj}_{\mathcal{G}(\boldsymbol{\xi})}(\cdot)$

$$\underset{\mathcal{G}}{\mathrm{proj}}\left(\mathbf{s}\right) := \underset{\mathbf{r} \in \mathcal{G}}{\arg\min}(||\mathbf{r} - \mathbf{s}||_2) \tag{3.3}$$

which projects its argument to the set of feasible actuation $\mathcal{G}(\boldsymbol{\xi})$. The matrices of control parameters have a block–partitioned structure such that $\mathbf{G} := [\mathbf{G_p}^\intercal, \mathbf{G_q}^\intercal]^\intercal$ is composed by two $N \times N$ diagonal matrices $\mathbf{G_p}$ and $\mathbf{G_q}$, and

$$\mathbf{D}_j := \begin{bmatrix} \mathbf{D}_{\mathbf{p}_\mathrm{G}\mathbf{p}_\mathrm{L},j} & \mathbf{D}_{\mathbf{p}_\mathrm{G}\mathbf{q}_\mathrm{L},j} & \mathbf{D}_{\mathbf{q}_\mathrm{G}\hat{\mathbf{p}}_\mathrm{G},j} \\ \mathbf{D}_{\mathbf{q}_\mathrm{G}\mathbf{p}_\mathrm{L},j} & \mathbf{D}_{\mathbf{q}_\mathrm{G}\mathbf{q}_\mathrm{L},j} & \mathbf{D}_{\mathbf{p}_\mathrm{G}\hat{\mathbf{p}}_\mathrm{G},j} \end{bmatrix} \tag{3.4}$$

for $j = 1, ...d$, where each entry in (3.4) is an $N \times N$ diagonal matrix whose naming subscript indicates the corresponding component–wise interactions between $\mathbf{u}$ and $\boldsymbol{\xi}$. Compactly, we write $\mathbf{D} := [\mathbf{D}_0, ..., \mathbf{D}_d]$, and we refer to $\mathbf{D}$ as the matrix of polynomial policy parameters. Recall that we apply the polynomial function on normalized values of the observed uncertainty. The normalizing function $\psi(\cdot)$ scales the values of $\boldsymbol{\xi}$ and is defined as

$$\psi(\boldsymbol{\xi}) := \frac{\boldsymbol{\xi} - \overline{\boldsymbol{\xi}}}{\sigma(\boldsymbol{\xi})} \tag{3.5}$$

where $\overline{\boldsymbol{\xi}}$ and $\sigma(\boldsymbol{\xi})$ are the expected value and the standard deviation of $\boldsymbol{\xi}$ respectively.

The controller in (3.2) is flexible enough to capture traditional droop control and affine policies. Moreover, the affine policy terms conveys many different relations between controlled variables and uncertainty sources. The idea is for the control tuning model to decide which instance of (3.2) achieves the best operational performance.

3.1.2 DGs feasible region

For simplicity in the mathematical notation, we assume at most one controlled DG per bus. Nevertheless, the presented approach is generalizable without much effort to cases with more than one DG per bus. For each DG, the following represents its feasible dispatch set

$$\mathcal{G}_i(\hat{p}_{\mathrm{G},i}) := \left\{ (p_{\mathrm{G},i}, q_{\mathrm{G},i}) \;\middle|\; \begin{array}{c} p_{\mathrm{G},i}^2 + q_{\mathrm{G},i}^2 \leq S_{\mathrm{nom},i}^2 \\ 0 \leq p_{\mathrm{G},i} \leq \hat{p}_{\mathrm{G},i} \end{array} \right\}, \; \forall i \in \mathcal{N}_{\mathrm{G}} \tag{3.6}$$

where $\mathcal{N}_{\mathrm{G}} \subseteq \mathcal{N}$ is the set of buses with a controlled DG, $S_{\mathrm{nom},i}$ is the nominal rating of the DG connected to bus $i \in \mathcal{N}_G$, and $\mathcal{G}_i(\hat{p}_{\mathrm{G},i}) = \{(0,0)\}, \; \forall i \in \mathcal{N} \setminus \mathcal{N}_G$. Recall that $p_{G,i} = u_i + \overline{p}_{G,i}$, $q_{G,i} = u_{i+N} + \overline{q}_{G,i}$, and $\xi_{2N+i} = \hat{p}_{G,i}$ for $i \in \mathcal{N}$. Then, as a consequence of the component–wise definition (3.6), it is easy to construct the set $\mathcal{G}(\boldsymbol{\xi}) \subset \mathbb{R}^{2N}$ as

$$\mathcal{G}(\boldsymbol{\xi}) := \mathcal{G}_1(\hat{p}_{\mathrm{G},i}) \times ... \times \mathcal{G}_1(\hat{p}_{\mathrm{G},N}) \tag{3.7}$$

such that

$$\mathbf{u} \in \mathcal{G}(\boldsymbol{\xi}) \tag{3.8}$$

The feasible dispatch region $\mathcal{G}(\boldsymbol{\xi})$ is the intersection of halfspaces and second–order cones, thereby convex and easily manageable by off-the–self solvers. Note that, by definition the control actions always satisfy (3.8) because of the projection operator (3.3).

3.1.3 Power flow approximation

We consider linear representations such as the *DistFlow* equations [39] or a first order Taylor (FOT) approximation [40] to include the power flow equations in our model. In general, it can be represented by

$$\mathbf{x} = \mathbf{B}(\mathbf{u} + \boldsymbol{\xi}_{\mathrm{L}}) + \mathbf{m} \tag{3.9}$$

where $\mathbf{B} \in \mathbb{R}^{N \times 2N}$ and $\mathbf{m} \in \mathbb{R}^{2N}$ are known parameters.

Remark 1. *The structure of the control tuning problem (3.26) is flexible enough to incorporate more precise OPF models such as branch flow model SOCP relaxations. The reason why we chose a linear model is that SOCP relaxations do not offer exactness guarantees when explicit voltage upper bounds bind at optimality [54]. This issue is problematic under high solar penetration scenarios as the voltage rises with the active power injections because of distribution networks' significant resistance to reactance ratio. However, voltage upper bounds in branch flow SOCP–OPF formulations can be enforced by suitably shaping the power injection feasible set, potentially offering more precise solutions than (3.9), although it is not in the scope of this work.*

3.1.4 System stability linear approximation

Within the feasible dispatch set, the controlled dynamical system is

$$\mathbf{x}_{k+1} = \mathbf{B}\mathbf{G}\mathbf{x}_k + \mathbf{B}\sum_{j=1}^{d} \mathbf{D}_j \psi^j(\boldsymbol{\xi}_k) + \mathbf{B}\boldsymbol{\xi}_L + \mathbf{m} \tag{3.10}$$

From elementary control theory of discrete–time linear systems [68], exponential stability of (3.10) can be achieved by constraining the spectral radius of $\mathbf{BG}$, such that

$$\rho(\mathbf{BG}) < 1 \tag{3.11}$$

The set defined by (3.11) is convex but cannot be handled by current off–the shelf–solvers. Nevertheless, it can be constricted to

$$||\mathbf{BG}||_F \leq 1 \tag{3.12}$$

relying on the fact that any vector–induced matrix norm is an upper bound on the spectral radius. Therefore, we obtain a computationally tractable reformulation while preserving stability. The authors in [20] proved that, under mild assumptions, (3.12) is a sufficient condition for bounded-input, bounded-output (BIBO) stability of the non–linear system (3.1) when $\mathbf{D} = \mathbf{0}$, i.e. for a pure voltage droop controller. In this work, we assume that the polynomial policy terms in (3.10) remain bounded, i.e. $||\mathbf{B}\sum_{j=1}^{d}\mathbf{D}_j\psi^j(\boldsymbol{\xi}_k) + \mathbf{B}\boldsymbol{\xi}_L + \mathbf{m}||_1 < M$, for $M \in \mathbb{R}$. Therefore, under this assumption, it is not difficult to extend the proof in [20] to show that (3.12) guarantees BIBO stability of system (3.1) for $\mathbf{D} \neq \mathbf{0}$.

3.1.5 Operational cost

The operational cost accounts for the distribution network's total net power consumption and voltage violations. We define it as

$$h(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}) := \zeta(\mathbf{u}, \boldsymbol{\xi}) + \alpha e(\mathbf{x}) \tag{3.13}$$

where $\zeta(\mathbf{u}, \boldsymbol{\xi})$ represents the net power injected by the utility at the slack bus, $e(\mathbf{x})$ accounts for voltage violations, and α is a positive scalar constant. We do not directly consider the utility power injection as a variable, instead, we write $\zeta(\mathbf{u}, \boldsymbol{\xi})$ as the difference between the losses and the net power injected by loads and DGs. Therefore,

$$\zeta(\mathbf{u}, \boldsymbol{\xi}) = \ell(\mathbf{u}, \boldsymbol{\xi}_{\mathrm{L}}) - \mathbf{1}^{\intercal}(\overline{\mathbf{u}} + \mathbf{u} + \boldsymbol{\xi}_{\mathrm{L}}) \tag{3.14}$$

where $\ell(\mathbf{u}, \boldsymbol{\xi}_{\mathrm{L}})$ represents the system losses. For radial networks, the system losses can be approximated by a convex quadratic function

$$\ell(\mathbf{u}, \boldsymbol{\xi}_{\mathrm{L}}) \approx (\overline{\mathbf{u}} + \mathbf{u} + \boldsymbol{\xi}_{\mathrm{L}})^{\intercal}\tilde{\mathbf{R}}(\overline{\mathbf{u}} + \mathbf{u} + \boldsymbol{\xi}_{\mathrm{L}}) \tag{3.15}$$

where $\tilde{\mathbf{R}} := \mathrm{diag}(\mathbf{R}, \mathbf{R})$ and $\mathbf{R} := \left(\mathbf{A}^{-1}\mathrm{diag}(\mathbf{r})\left(\mathbf{A}^{-1}\right)^{\intercal}\right)$, $\mathbf{r}$ is the column vector of branch resistances and $\mathbf{A}$ is the bus incidence matrix of the system [69]. Note that the matrix $\tilde{\mathbf{R}}$ is symmetric positive definite and have non–negative entries. Finally, we define the function of voltage violation as

$$e(\mathbf{x}) := ||(\check{\mathbf{x}} - \mathbf{x})^{+}||_1 + ||(\mathbf{x} - \hat{\mathbf{x}})^{+}||_1 \tag{3.16}$$

where $\check{\mathbf{x}}$ and $\hat{\mathbf{x}}$ are the lower and upper bounds on $\mathbf{x}$, respectively. Note that we only penalize values of voltage that lie outside the normal operation range, e.g. $\pm 5\%$. Therefore, in normal operating conditions the focus is on minimizing the overall power consumption of the distribution network.

3.1.6 Two–stage robust tuning model

So far, we have described the control scheme and the shape of the real–time controllers. Nevertheless, nothing has been said about the methodology for determining optimal control parameters $\mathbf{G}$ and $\mathbf{D}$. We propose a scenario–based two–stage robust optimization problem, comprising three optimization levels

$$\min_{\mathbf{G},\mathbf{D}\in\mathcal{W}}\left(\max_{\boldsymbol{\xi}\in\mathcal{S}}\left(\min_{\mathbf{y}\in\mathcal{Y}(\mathbf{G},\mathbf{D})} h(\mathbf{y}, \boldsymbol{\xi})\right)\right) \tag{3.17}$$

In the first level, the first–stage decision $(\mathbf{G}, \mathbf{D})$ determines the controller settings. In the second–level problem, the worst–case scenario of $\boldsymbol{\xi}$ is selected, where $\boldsymbol{\xi}$ is modeled as a discrete random vector whose probability distribution is supported on a sample set $\mathcal{S} := \{\boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_{N_s}\}$. The metric used to find the worst–case scenario is defined by the optimal objective function of the third–level problem as a function of first–stage and second–stage variables, hereinafter referred to as $z(\mathbf{G}, \mathbf{D}, \boldsymbol{\xi})$. Thus, the objective of the third level, inner minimization in (3.17), is to determine the second–stage recourse vector $\mathbf{y} := (\mathbf{x}, \mathbf{u})$ which comprises voltage variations and controlled power injections.

3.1.7 Adaptive data–driven uncertainty sets (DDUS)

As mentioned before, we assume that the uncertainty of the model originates only from solar power availability $\hat{\mathbf{p}}_{\mathrm{G}}$ and the uncontrolled net power generation $\boldsymbol{\xi}_{\mathrm{L}} \in \mathbb{R}^{2N}$, comprising into $\boldsymbol{\xi} := [\boldsymbol{\xi}_{\mathrm{L}}^{\intercal}, \hat{\mathbf{p}}_{\mathrm{G}}^{\intercal}]^{\intercal}$. Based on [1,35] we propose building the set of scenarios $\mathcal{S}$ in an adaptive and data-driven fashion. The term data-driven is associated with the fact that the scenarios used to build up the uncertainty set directly rely on actual data. Therefore, this approach recognizes the complexity of the proper data generation process and utilizes historical observations to shape the uncertainty set with the dependencies found on data.

The synthesis of the DDUS may include preprocessing schemes such as filtering and trend shifts based on adaptive forecasts of the planning horizon. Furthermore, sophisticated machine learning techniques [59] could be applied to design the uncertain data management policy. Despite that, our approach is simple, we take into account the daily seasonality of the data and directly incorporate the historical profiles of uncertainty realizations from the last days. Specifically, we utilize observed values of $\boldsymbol{\xi}$ from time intervals that result from shifting the control horizon backwards. For instance, if the current time is 9:00 am and we are solving the control tuning problem for the following 2 hours, we consider the historical measurements of $\boldsymbol{\xi}$ from 9:00 am to 11:00 am from the last N_D consecutive days. Then we filter them by selecting N_s representative samples to construct $\mathcal{S}$.

The filter is based on the KMeans algorithm [3], which clusters data by trying to separate samples of $\boldsymbol{\xi}$ in groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squares. This algorithm allows for the specification of a predefined number of clusters N_s. It scales well to large number of samples and has been used across a large range of application areas in many different fields. From each cluster, we select the sample that is closer to the centroid. Therefore, we obtain N_s samples as the output. Note that the KMeans algorithm tries to obtain clusters with equal variance. Hence, we may interpret the filtering as a method for selecting a set of scenarios with similar amount of information about $\boldsymbol{\xi}$.

Note that by applying the process mentioned above we avoid the need for sophisticated forecast techniques and the number of scenarios remains as a user–defined parameter.

3.1.8 Optimal operation model

The third–level optimization problem in (3.17) represents the optimal operation of the system as a function of $(\mathbf{G}, \mathbf{D}, \boldsymbol{\xi})$ and is given by

$$
\begin{aligned}
z(\mathbf{G}, \mathbf{D}, \boldsymbol{\xi}) := & \\
\min_{\mathbf{y}} \quad & h(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}_{\mathrm{L}}) && \text{(3.18a)} \\
\text{s.t.} \quad & \mathbf{u} = \mathbf{G}\mathbf{x} + \sum_{j}^{d} \mathbf{D}_j \psi^j(\boldsymbol{\xi}) && \text{(3.18b)} \\
& \mathbf{x} = \mathbf{B}(\mathbf{u} + \boldsymbol{\xi}_{\mathrm{L}}) + \mathbf{m} && \text{(3.18c)} \\
& \mathbf{u} \in \mathcal{G}(\boldsymbol{\xi}_j) && \text{(3.18d)} \\
& ||\mathbf{G}\mathbf{B}||_F \leq 1 && \text{(3.18e)}
\end{aligned}
$$

where the objective function (3.18a) minimizes the voltage violations and the energy drawn by the distribution network. Equations (3.18b) and (3.18c) characterize the equilibrium of the closed–loop controlled system. Note that we dropped the projection operator in (3.2) because we have already enforced the control actions to belong to feasible actuation set in (3.18d). Finally, (3.18e) constraints the domain of $z(\mathbf{G}, \mathbf{D}, \boldsymbol{\xi})$ to the set of stable controllers.

Observe that problem (3.18) presents some difficulties. First, note that (3.18a) has non–linearities from the system losses approximation (3.15) and piecewise linear function (3.16). Nevertheless, it is easy to reformulate (3.18a) to a second–order cone by incorporating auxiliary variables. Second and more important, the bilinear terms $\mathbf{G}\mathbf{x}$ introduced by (3.18b) make the single–level reformulation of (3.17) a quadratically constrainted quadratic problem (QCQP) which cannot be solved efficiently by off–the–shelf solvers. Bilinear terms composed of two continuous variables do not have an exact convex reformulation. In the next section we will tackle these difficulties and obtain a single–level second–order cone approximation of (3.17).

Before we go further, we summarize the proposed control strategy in Fig. 3.2. The centralized control collects the system's data, constructs the scenarios, and periodically calculates the nominal

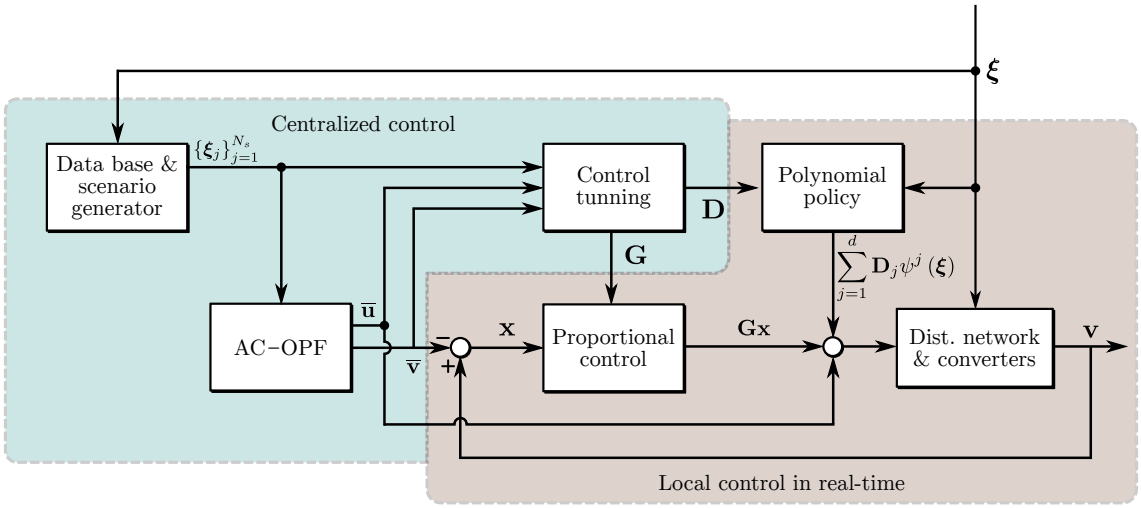

Figure 3.2: Proposed control block diagram.

set–point and control parameters. Meanwhile, the local control calculates and superposes, in real–time, the voltage droop term and the polynomial terms to define the DGs power injection references. Note that the droop control and the polynomial policy act as feedback and feedforward controllers, respectively.

3.2 Solution approach

3.2.1 Control policy linear approximation

Note that the equality constraint (3.18b) is non–convex due to the bilinear term $\mathbf{Gx}$. In this section, we relax (3.18b) applying classical McCormick envelopes to form a bounded polyhedron arround $\mathbf{Gx}$ [65]. The McCormick relaxation requires $\mathbf{x}$ and $\mathbf{G}$ to have fixed upper and lower bounds on each entry. We capitalize on the fact that the entries of $\mathbf{G}$ are implicitly bounded by (3.18e), and $\mathbf{x}$, in practice, belongs to a normal operating range near zero. Hence, we define fixed bounds on variables $\mathbf{G}$ and $\mathbf{x}$ and substitute $\mathbf{Gx}$ in (3.18b) by the auxiliary variable $\boldsymbol{\omega}$. We reformulate our problem, replacing (3.18b) by (3.19)

$$\mathbf{u} = \boldsymbol{\omega} + \sum_{j=1}^{d} \mathbf{D}_j \psi^j(\boldsymbol{\xi}) \tag{3.19}$$

and incorporating the constraint

$$(\boldsymbol{\omega}, \mathbf{x}, \mathbf{G}) \in \mathcal{C} \tag{3.20}$$

where $\mathcal{C}$ is the bounded polyhedron defined by the McCormick inequalities. Equations (3.19) and (3.20) represent the McCormick relaxation of (3.18b). Refer to [23] for constructive aspects of (3.20) and further insight.

3.2.2 AdaLASSO regularization

Ideally, we would expect to choose a controller that accurately captures the regularities in the in–sample (training) data and generalizes well to out–of–sample (validation) data. In this context, complex control rules, i.e., $(\mathbf{G}, \mathbf{D})$ with many non–zero parameters, may be able to represent the in–sample data well but are at risk of overfitting to low–dimensional and unrepresentative in–sample data [24]. In contrast, more sparse control rules may fail to capture the complexity of the underlying process, i.e., underfit.

AdaLASSO is a regularization technique that systematically shrinks a given linear regression model. It has the oracle properties, roughly meaning that it can identify the relevant subset of control parameters [27]. It works by introducing the following penalty in the objective function of

the control tuning problem

$$c_\lambda(\mathbf{D}) := \lambda \sum_{i=1}^{2N} \sum_{j=1}^{3N} \left| \frac{D_{ij}}{D_{ij}^{(0)}} \right| \tag{3.21}$$

where λ is a non–negative scalar reflecting the penalization level. Note that it continuously shrinks the control coefficients toward 0 as λ increases. This continuous shrinkage often improves out–of–sample performance up to a certain value of λ. The weighting matrix $\mathbf{D}^{(0)}$ is the matrix of polynomial control parameters obtained by solving the control tuning model (3.26) without AdaLASSO regularization, i.e., $\lambda = 0$. So in order to solve the regularized model, the non–regularized solution is required. For computational implementation purposes we do not consider the control coefficients in $\mathbf{D}$ with corresponding zero–valued entries in $\mathbf{D}^{(0)}$.

The function (3.21) is non–linear because of the absolute value terms. However, it can be easily reformulated by defining the following

$$\tilde{c}_\lambda(\mathbf{\Lambda}) := \lambda \sum_{i=1}^{2N} \sum_{j=1}^{3N} \frac{\Lambda_{ij}}{\left| D_{ij}^{(0)} \right|} \tag{3.22a}$$

$$-\mathbf{\Lambda} \leq \mathbf{D} \leq \mathbf{\Lambda} \tag{3.22b}$$

where $\mathbf{\Lambda}$ is an auxiliary variable of the same dimension as $\mathbf{D}$.

3.2.3 Objective function

Equation (3.13) is non–linear due to (3.15) and (3.16), nevertheless we can easily reformulate (3.16) as a linear function by introducing auxiliary variables $\boldsymbol{\epsilon}^+, \boldsymbol{\epsilon}^- \geq \mathbf{0}$. Define

$$\tilde{e}(\boldsymbol{\epsilon}^+, \boldsymbol{\epsilon}^-) := \mathbf{1}^\intercal \left(\boldsymbol{\epsilon}^+ + \boldsymbol{\epsilon}^- \right) \tag{3.23}$$

and incorporate the constraints

$$\check{\mathbf{x}} - \boldsymbol{\epsilon}^- \leq \mathbf{x} \leq \hat{\mathbf{x}} + \boldsymbol{\epsilon}^+ \tag{3.24}$$

Therefore, we represent the operational costs as a convex quadratic function

$$\tilde{h}(\mathbf{x}, \mathbf{u}, \boldsymbol{\epsilon}^+, \boldsymbol{\epsilon}^-, \boldsymbol{\xi}) = \ell\left(\mathbf{u}, \boldsymbol{\xi}_{\mathrm{L}}\right) - \mathbf{1}^\intercal \left(\overline{\mathbf{u}} + \mathbf{u} + \boldsymbol{\xi}_{\mathrm{L}} \right) + \tilde{e}(\boldsymbol{\epsilon}^+, \boldsymbol{\epsilon}^-) \tag{3.25}$$

3.2.4 Control tuning problem SOCP reformulation

Here we present a single–level reformulation of (3.17) where (3.18) is approximated by a SOCP problem. The model is formulated as follows:

$$\min_{\mathbf{w}, \tilde{\mathbf{y}}} \quad \beta - \gamma\varepsilon + \tilde{c}_\lambda(\mathbf{\Lambda}) \tag{3.26a}$$

$$\text{s.t.} \quad \tilde{h}(\mathbf{x}_s, \mathbf{u}_s, \boldsymbol{\epsilon}_s^+, \boldsymbol{\epsilon}_s^-, \boldsymbol{\xi}_s) \leq \beta, \qquad \forall s \in \mathcal{S} \tag{3.26b}$$

$$\mathbf{x}_s = \mathbf{B}\mathbf{u}_s + \mathbf{m}, \qquad \forall s \in \mathcal{S} \tag{3.26c}$$

$$\mathbf{u}_s = \boldsymbol{\omega}_s + \sum_{j}^{d} \mathbf{D}_j \psi^j\left(\boldsymbol{\xi}_s\right), \qquad \forall s \in \mathcal{S} \tag{3.26d}$$

$$(\mathbf{G}, \mathbf{x_s}, \boldsymbol{\omega}_s) \in \mathcal{C}, \qquad \forall s \in \mathcal{S} \tag{3.26e}$$

$$\mathbf{u}_s \in \mathcal{G}\left(\boldsymbol{\xi}_s\right), \qquad \forall s \in \mathcal{S} \tag{3.26f}$$

$$\check{\mathbf{x}} - \boldsymbol{\epsilon}_s^- \leq \mathbf{x}_s \leq \hat{\mathbf{x}} + \boldsymbol{\epsilon}_s^+, \qquad \forall s \in \mathcal{S} \tag{3.26g}$$

$$||\mathbf{BG}||_F \leq 1 - \varepsilon \tag{3.26h}$$

$$-\mathbf{\Lambda} \leq \mathbf{D} \leq \mathbf{\Lambda} \tag{3.26i}$$

$$\boldsymbol{\epsilon}_s^+ \geq \mathbf{0}, \boldsymbol{\epsilon}_s^- \geq \mathbf{0}, \qquad \forall s \in \mathcal{S} \tag{3.26j}$$

$$\varepsilon \geq 0 \tag{3.26k}$$

Where $\mathbf{w} := (\mathbf{G}, \mathbf{D}, \beta, \varepsilon, \mathbf{\Lambda})$ and $\tilde{\mathbf{y}} := (\mathbf{x}, \mathbf{u}, \boldsymbol{\omega}, \boldsymbol{\epsilon}^+, \boldsymbol{\epsilon}^-)$ correspond to the first and second stage variables respectively. The objective function (3.26a) minimizes the upper bound on operational costs β, the control instability penalty $-\gamma\varepsilon$, and the AdaLASSO penality (3.22a). The control

instability penalty is introduced to promote controllers with higher stability margin ε [20]. Equation (3.26b) is necessary to define β as the maximum operational cost (3.25), over all scenarios. Equation (3.26c) is the linear power flow model enforced for each scenario. Equation (3.26d) and (3.26e) account for the McCormick reformulation of (3.18b) explained in Section 3.2.1. We enforce the feasibility of the actuation for each scenario in (3.26f). Equation (3.26g) impose soft bounds on voltage with the help of auxiliary variables forced to be non–negative by (3.26j). Equation (3.26h) is a constriction of (3.18e) by a variable gap ε in order to apply the control instability penalty mentioned above. The bounds (3.26i) correspond to the absolute value reformulation (3.22b) necessary for the formulation of the AdaLASSO penalty in the objective function. Finally, (3.26k) constrains the stability margin ε to be non–negative.

Chapter 4

Computational Experiments

This chapter characterizes the experimental design and shows the obtained results. First, the simulation methodology is described in detail together with the data that was utilized in the experiments. Next, alternative control strategies are presented highlighting their main properties. Then, the AdaLASSO coefficient λ is determined by enumeration. After that, the performance of the proposed and the alternative approaches is evaluated considering three instances of IEEE tests feeders. Additionally, the proposed real–time controllers are simulated disabling the updating layer to evaluate their behavior under strictly local operation. Finally, computational tractability experiments are shown.

4.1 Simulation methodology

We have performed extensive computational experiments utilizing modified versions of three IEEE test feeders, namely the 4–bus, 34–bus, and 123–bus feeders [70]. Line segments have been balanced using Carson's equation and Kron reduction. Distributed loads have been converted into balanced three–phase loads. One–minute time series data is obtained from a set of individual homes from the U.S. Austin region [71]. We have downsampled the time series data to a 6–second sampling frequency. For each home, P–Q load measurements and solar power injection are available. The aggregation of individual loads into equivalent bus–injection profiles have been done by applying an affine combination of the individual load time series and applying a moving average low–pass filter to emulate superposition. To model the penetration of EVs, we superposed synthetic EV demand components to the distributed loads, following the method described in [72]. Finally, we scaled the time series to obtain study cases with 100% PV penetration and 10% EV penetration, relative to the total demand of the system. Only 50% of the PV penetration is available for controlled DGs. The rest represents uncontrolled DGs functioning at maximum power point (MPP). We obtain daily profiles similar to the one in Figure 2.3, at each bus. Note that the uncontrolled PV penetrations are considered in the profile of uncontrolled net power consumption of Figure 2.3 where negative values are observed near midday.

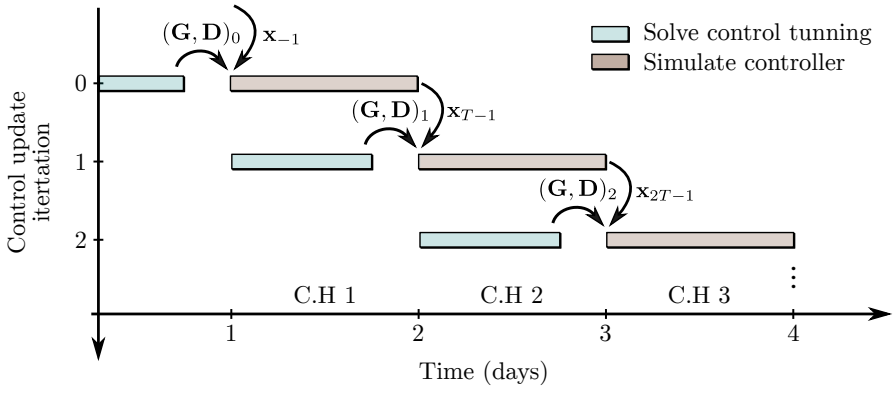

Figure 4.1: Rolling horizon concept (1 day control horizon).

At a given bus, the time series corresponding to the uncontrolled PV power output and the maximum power availability have the same shape but were scaled differently. However, they have different shapes between different buses; although, they are highly correlated as the houses where the measurements were performed belong to the same region. Note that by no means we are capturing spatial correlations in solar irradiance. Additionally, the time–series of load and PV output were randomly assigned to each bus. Similarly, the three feeders were randomly populated with controlled DGs at different locations. For the 4–bus, controlled DGs are installed in buses 1,2, and 3. For 34–bus, controlled DGs are installed in buses 8, 9, 10, 11, 13, 15, 17, 18, 19, 21, 22, 24, 26, 28, 30, 31, 32 and 33; and for the IEEE–123bus, DGs are installed in buses 9, 10, 15, 16, 18, 19, 21, 23, 27, 29, 30, 31, 32, 33, 34, 36, 37, 38, 40, 41, 44, 45, 46, 47, 48, 49, 50, 51, 52, 54, 55, 57, 58, 59, 62, 63, 64, 65, 67, 68, 69, 72, 73, 75, 76, 78, 79, 81, 82, 83, 84, 85, 86, 87, 89, 91, 93, 94, 95, 98, 99, 102, 103, 105, 106, 108, 110, 112, 113 and 119. A systematic criterion for DG allocation is out of the scope of this work.

To assess the performance of the proposed model, we emulate a real–time dispatch process using a rolling–window out–of–sample evaluation methodology. First, we divide our time series data into in–sample (training) and out–of–sample (evaluation) time intervals. The in–sample represents historical data available to make *here–and–now* decisions, e.g., calculating $(\mathbf{G}, \mathbf{D})$; meanwhile the out–of–sample emulates the real–time realization of the uncertainty, available only to make *wait–and–see* decisions, e.g. calculating $\mathbf{u}$ applying (3.2). Let the out–of–sample time series be partitioned into consecutive control horizons. Let i denote a control horizon index such that $i \in \{1, 2, ..., H\}$, then the matrices $(\mathbf{G}, \mathbf{D})_i$ calculated at the control horizon i are simulated in the control horizon $i + 1$, see Figure 4.1. In addition to the control matrices, previous *here–and–now* decisions have to be made. Namely, the construction of the set of scenarios and the calculation of the nominal setpoint, see Figure 4.2 for a summary on the flow of information in the simulation process.

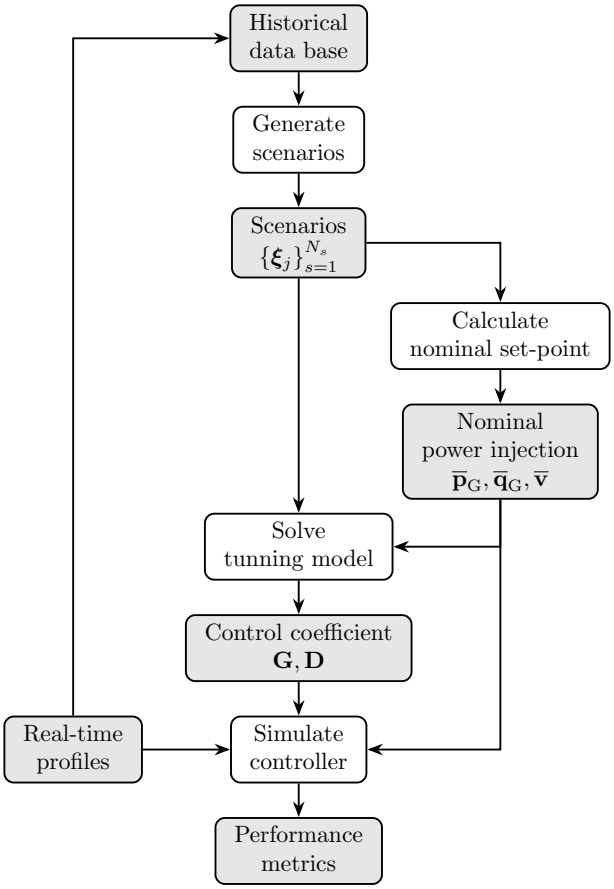

Figure 4.2: Simulation flowchart.

4.2 Computational implementation

In this section, we give details regarding the actual computational implementation of (3.26).

4.2.1 Model selection: Hyperparameters

Multi–objective relative weight α

First, recall that the operational cost function conveys terms associated with utility power injection and voltage limit violations. The relative weight between them is given by a positive scalar α, which must be defined in advance. The selection criterion for this number depends on the relevance of the voltage violations. Many factors influence this decision, such as actual cost penalties for voltage violations for system operators. Here, our goals are, in the first place, to operate the system within the $\pm 5\%$ voltage range, and second, to minimize total power consumption. Therefore, we have selected $\alpha = 10^5$, a number sufficiently large to represent our design goals but low enough to avoid solver related numerical issues.

By the way, we could avoid this arbitrary hyperparameter selection process by eliminating the voltage violation terms from the operational cost function and forcing the voltages to lie within a prespecified range. Nevertheless, fixing voltage limits does not follow our design criterion because the space of feasible controllers would be overly constrained compared to the set of all controllers that ensure continuous operation conditions. This is because the desired voltage range can be a subset of the range of voltages that ensure continuous operation. Therefore, we assume that the system operator may prefer to operate within that tighter voltage range if possible but could operate outside that range if necessary. In this context, the idea is for the controllers to achieve high-performance operation even outside the desired voltage operation range.

McCormick bounds

Recall that the McCormick relaxation requires the variables involved in the bilinear terms to have fixed bounds. In this work, we set

$$\begin{aligned} x_j &\in [-0.06, 0.06] \\ G_{p,j,j} &\in [-0.2, 0] \\ G_{q,j,j} &\in [-0.2, 0] \end{aligned}$$

for all $j \in \mathcal{N}_G$. The bounds were determined by enumeration, i.e., running the proposed control tuning model (3.26) for different bound values of $\mathbf{G}$ and $\mathbf{x}$ separately. In particular we utilized an instance of the 34–bus feeder following the simulation procedure described in Section 4. The selection criterion was to avoid binding bounds at optimality. The enumeration procedure considered the same value for all DGs; nevertheless, as shown in [20], we should expect for the entries of $\mathbf{G}$ to heavily depend on the location of each DG. Therefore, more sophisticated procedures for McCormick bound selection can be incorporated to consider spatial dependency.

The control horizon

Later in this section, the reader will notice that we have selected a control horizon of 1 day. This selection may seem strange considering that alternative approaches typically utilize 10– to 15–minutes control horizons, such as in [20]. Note that generating scenarios from whole days of historical data produces an uncertainty set with higher variance than generating scenarios from a 15–minute range from the same days. The intuition says that shorter control horizons imply less uncertainty and better out–of–sample performance. So why select a 1–day control horizon?. The reason is simple, we have noticed that our controller behaves better with a 1–day control horizon than with a 15–minutes one by comparing simulation results. We have no analytical justification for this behavior. However, we believe that more information about different operating points is incorporated into the controller by considering a set of scenarios with higher variance. Additionally, in-sample overfitting is more likely to happen with low–variance in–sample data sets. A systematic approach for selecting the control horizon is regarded as future work.

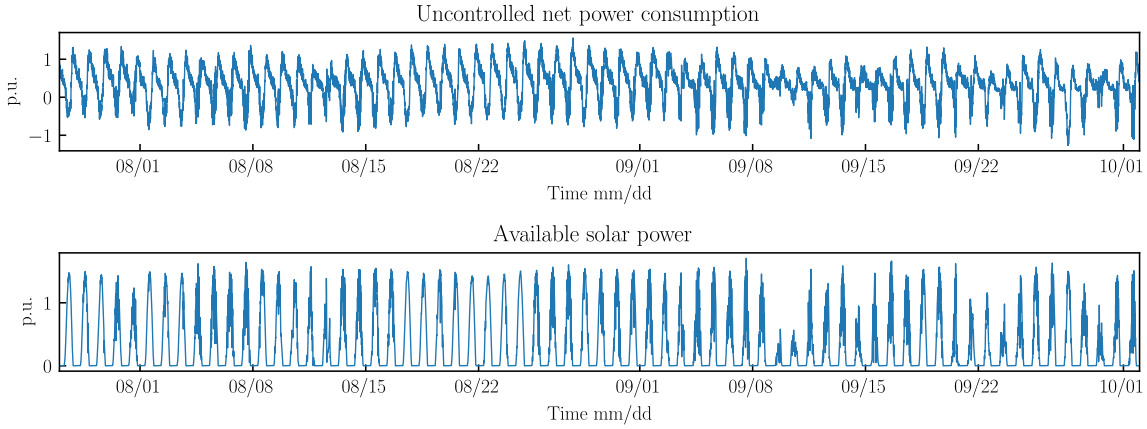

Figure 4.3: Available data with uncertainty on a random bus of the IEEE 4-bus test feeder.

Number of scenarios and historical data sample subset

Although we have defined the scenario generation procedure in section 3.1.7, note that there are still two hyperparameters to define, namely the number of scenarios N_s and the number of previous days to consider N_D. By exploring different alternatives we chose $N_s = 500$ and $N_D = 10$ based on out–of–sample performance. Figure 4.3 depicts a sample of the available data utilized for the experiments. Note that there is a daily seasonality and $N_D = 10$ days seems a reasonable choice. A systematic approach for selecting N_D and N_s is regarded as future work.

4.2.2 The AMPL model

There is a gap between (3.26) and the computationally implemented model. Here we analyze the differences between (3.26) and the actual implementation characterized by the AMPL model declaration file in Appendix A, which we refer to as the *AMPL model*.

Orthant–based partition of the bilinear surfaces

First, define $\mathbf{x} := \mathbf{x}^+ - \mathbf{x}^-$, where $\mathbf{x}^+$ and $\mathbf{x}^-$ are column vectors with non–negative entries. Now (3.18b) can be written as

$$\mathbf{u} = \mathbf{G}\mathbf{x}^+ - \mathbf{G}\mathbf{x}^- + \sum_j^d \mathbf{D}_j \psi^j(\boldsymbol{\xi}) \tag{4.1}$$

Note that by doing this, we partition the original bilinear manifold into two sets, each of them belonging to a single orthant. Hence, the McCormick relaxation of the resulting manifold partitions is tighter that the original. Following this, in the AMPL model, the voltage variation $\mathbf{x}$ is represented as the difference between two positive variables `v_pos` and `v_neg`. Additionally $\mathbf{G}$ is represented by the vectors of diagonal elements $\mathbf{g}_\mathrm{p} := \mathrm{diag}(\mathbf{G}_\mathrm{p})$ and $\mathbf{g}_\mathrm{q} := \mathrm{diag}(\mathbf{G}_\mathrm{q})$, where `g_p` and `g_q` are the corresponding variables in AMPL. Finally, note that there are 4 types of auxiliary variables summarized in Table 4.1

Table 4.1: McCormick auxiliary variables in the AMPL model.

Bilinear term	Auxiliary variable
`dv_pos*g_p`	`k_p_pos`
`dv_pos*g_q`	`k_q_pos`
`dv_neg*g_q`	`k_p_pos`
`dv_neg*g_q`	`k_q_pos`

Shrinkage of G

Note that if we utilize a shorter control horizon period, e.g., 15 minutes, there are time intervals of the day where $\mathbf{x} \approx \mathbf{0}$. If the values of $\mathbf{x}$ are sufficiently small, then the solver algorithm may assign

to $\mathbf{G}$ any value within the corresponding bounds. To avoid non–zero values of $\mathbf{G}$ when $\mathbf{x} \approx \mathbf{0}$ we incorporate a small weighting factor that shrinks $\mathbf{G}$ to avoid unmeaningful values that may impact out–of–sample performance. This is implemented by the terms

```
cost_lasso_v * sum {i in BUSES} (gp[i] + gq[i])
```

in the objective function. Where `cost_lasso_v` is a small positive constant. we set `cost_lasso_v` = 10^{-5} for shorter control horizons, and `cost_lasso_v` = 0 far daily control horizons.

4.2.3 Solver configuration: dealing with sub–optimal termination

All instances of the AMPL model in Appendix A were solved utilizing Gurobi 9.0.1. For some rare instances the solver may terminate with a sub–optimal termination status, meaning that optimality has not been achieved. We configured the solver to achieve optimality in such instances according to the following.

Improving barrier convergence rate

As mentioned before, the solver terminate with a sub–optimal termination status for some rare instances. Typically, those instances present a low convergence rate in the late iterations. Large-scale quadratic programming problems are typically solved by utilizing the barrier algorithm. The barrier solver terminates when the relative difference between the primal and dual objective values is less than the specified tolerance. Tightening this tolerance often produces a more accurate solution, which can sometimes improve the convergence rate in the late iterations [73]. Therefore, if we end up with a sub–optimal termination condition, we follow the recommendations above and rerun the instance with a tighter barrier convergence tolerance (barconvtol=1e-10) and further configuration to avoid numerical issues. In all the instances, we achieve optimality after these extra settings.

Avoiding numerical issues

To deal with the sub–optimal termination described above, we also apply some easy–to–implement strategies to avoid numerical issues. Although, they come with the price of increasing the computational burden. First, we shut down presolve aggregation (aggregate=0), because aggregation can lead to an accumulation of numerical errors. In addition, Gurobi has a high level configuration parameter called NumericFocus to control the degree to which the code attempts to detect and manage numerical issues. The default setting (0) makes an automatic choice, with a slight preference for speed. Settings 1-3 increasingly shift the focus towards being more careful in numerical computations. With higher values, the code will spend more time checking the numerical accuracy of intermediate results, and it will employ more expensive techniques in order to avoid potential numerical issues. We set the NumericFocus parameter to 3 when rerunning the instances that initially terminated with sub optimal termination status.

Other strategies to deal with numerical issues such as proper model scaling can be implemented without incurring in extra computational burden. They are regarded as future work.

4.3 Alternative control strategies

We test our strategy against similar local approaches in current literature, namely, the Standard IEEE–1547 [2] (IEEE1547), and the voltage droop control approach presented in [20] (NCognizant). The former defines a priori values of voltage droop setting for each DG, without grid information. In contrast, the latter periodically solves a centralized robust optimization problem using network–wide data. NCognizant applies voltage droop control based only on local measurements, and the authors tackle the bilinear terms that naturally arise from proportional control tuning using the Neuman approximation. Additionally, we consider a fully centralized scenario–based stochastic OPF strategy (SOPF), which, at each control iteration, defines fixed nominal set–points for the DGs. Table 4.2 summarizes the control strategies mentioned above.

Table 4.2: Comparative table of the simulated strategies.

Control strategy	Control horizon	Strictly local
IEEE1547	—	Yes
NCognizant	15 minutes	No
SOPF	15 minutes	No
Proposed	1 day	No

4.4 Model selection: AdaLASSO

AdaLASSO regularization can be done with different levels depending on the chosen value of λ. Here we carry out a simple parametric model selection process: given a finite number of alternative models, i.e., instances of (3.26) with different λ values, we choose the best according to out–of–sample average operational cost. We simulated three days of real–time operation on the IEEE 34–bus case for each λ. At each control iteration (daily), 500 scenarios were constructed from 15 previous days of in–sample data according to Section 3.1.7. We consider polynomial policies of degree 3 and 18 controlled DGs. Therefore the number of control parameters scales to 468, which is close to the number of scenarios.

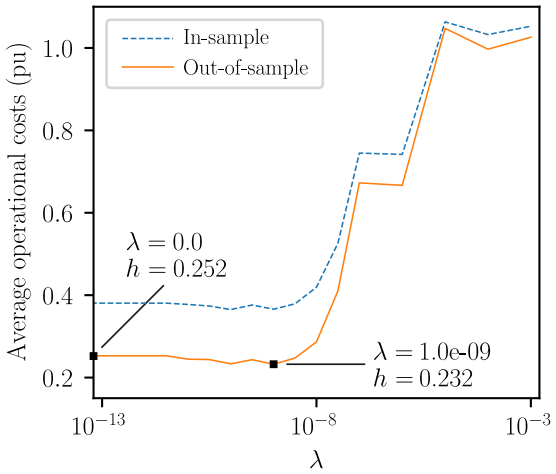

Figure 4.4: Average operational costs as a function of the AdaLASSO penalty coefficient λ.

Figure 4.4 shows the out–of–sample average operational costs, calculated using (3.13), as a function of λ. The value $\lambda = 10^{-9}$ is selected for the regularized model as it minimizes the operational costs. We use an appropriate plot scale to draw the non–regularized solution, i.e., with $\lambda = 0$, as a point in the y-axis. The AdaLASSO regularization improved the operational costs by 8%. To evaluate the significance of the improvement, we take both solutions and construct corresponding sets of the minute–based average operational cost samples. We evaluate the plausibility of both sets being collections of realizations of the same random variable (null–hypothesis). We applied a test of significance (t–test), and the results indicate that the hypothesis mentioned above is rejected with a p–value of 0.26.

Recall that the AdaLASSO penalty (3.21) has a shrinking effect on $\mathbf{D}$. At the same time, $\mathbf{D}$ has a block–partitioned structure conveying different relations between the elements of the uncertainty and the control actions as shown in (3.4). Figure 4.5 shows the percentage of control parameters that have non–zero values for each submatrix. This pattern gives a hint on what types of parameters in $\mathbf{D}$ are the most relevant. For example terms with degree 2 and 3 that calculate $\mathbf{p}_{\mathrm{G}}$ as a function of $\hat{\mathbf{p}}_{\mathrm{G}}$ seem to be the least relevant. On the other hand, policies that calculate $\mathbf{q}_{\mathrm{G}}$ as a function of the maximum power availability $\hat{\mathbf{p}}_{\mathrm{G}}$ seem promising. This observation is coherent with the design of reactive power control in [12].

Remark 2. *The out–of–sample curve shown in Figure 4.4 is calculated utilizing the "true" value of $\ell(\mathbf{u}, \boldsymbol{\xi}_L)$ in (3.13) given by the non–linear AC power flow equations applied to the out–of–sample*

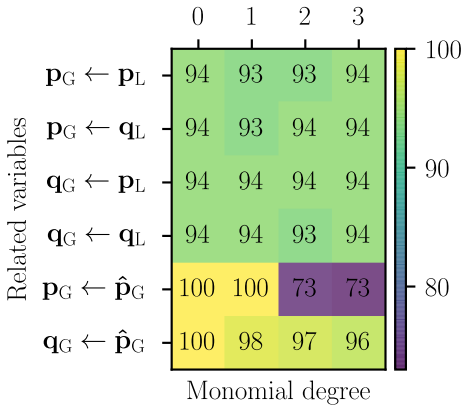

Figure 4.5: Percentage of non-zero polynomial control parameters.

data, meanwhile, the in–sample curve is calculated utilizing the second–order–cone approximation (3.15) over the in–sample scenarios. Therefore, the results suggest that the proposed approach is robust enough to cope with the effect of both modeling errors and uncertainty. This feature prevailed for all values of λ that were explored.

4.5 Results

In this section, we compare the operational performance of the proposed strategy with other existing approaches described in Section 4.3. The metrics selected for performance assessment are: average voltage violation per bus (AVV), hourly average utility power injection[1] (HUPI), hourly curtailed PV power (HCPVP), and hourly network losses (HNL). We evaluate the real–time operation for the three test feeders along a simulation horizon of one week. Except for IEEE1547, the nominal set–points are determined using the same stochastic OPF model and scenario generation procedure for all control strategies. The number of scenarios generated from in–sample data is 100 for the IEEE 4–bus feeder and 500 for the IEEE 34–bus and the IEEE 123–bus feeders.

Case of study	Strategy	AVV $\times 10^{-7}$	HUPI (kW)	HCPVP (kW)	HNL(kW)
IEEE 4-bus test feeder	SOPF	1.5	479.6	433.8	32.1
	NCognizant	0	482.5	435.3	33.6
	Proposed	0	476.6	440.6	22.3
	IEEE1547	1135.5	192.3	147.7	30.9
IEEE 34-bus test feeder	SOPF	1.4	86.4	98.3	12.6
	NCognizant	0	90.2	98.8	15.9
	Proposed	0	43.5	51.3	16.7
	IEEE1547	779.2	-14.9	0	9.6
IEEE 123-bus test feeder	SOPF	401.2	1319	1431.1	57.9
	NCognizant	310.2	1353.6	1467.8	56
	Proposed	0	185.5	497.4	117
	IEEE1547	13438.5	-283.7	0	145.2

Table 4.3: Mean value of selected metrics

Table 4.3 summarizes the simulation results. We observe that only NCognizant, SOPF, and Proposed achieve voltage regulation satisfactorily. Between them, note that the proposed approach produces the most economically optimal solution by minimizing the other metrics, i.e., HUPI, HCPVP, and HNL. To make this more explicit, Figure 4.7a shows the probability distribution function of daily average operational costs, which considers a weighted sum of AVV and HUPI (see (3.13)). The experiments suggest that the proposed strategy captures a multi–objective approach achieving more economically optimal decisions while regulating voltage effectively.

[1]The hourly average of energy drawn from the substation by the ADN.

An important fact that is not evident in Table 4.3 is that the out–of–sample results show important reverse power flows near midday. To make this explicit, Figure 4.6 shows an hourly boxplot corresponding to the utility power injection obtained from the out–of–sample simulations of the proposed strategy for the 34–bus feeder.

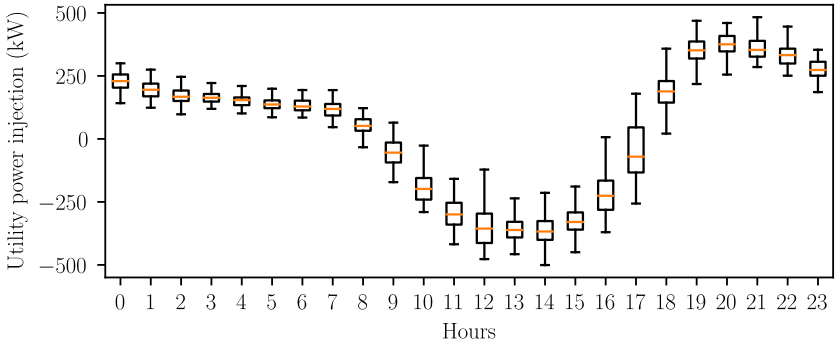

Figure 4.6: Hourly boxplot of the utility power injection for the proposed approach 34–bus feeder.

For the IEEE 123–bus instance, which contains 70 DGs, the number of control parameters scales to 1820. The number of control parameters is more than three times the number of scenarios utilized in the control tuning model. Nevertheless, Table 4.3 shows that the proposed strategy still presents high performance for the IEEE 123–bus. It is important to note that, in this case, the regularization procedure only improved the solution by 2.3%. Therefore, the experiments suggest that even for cases where the number of tuning parameters is considerably higher than the number of scenarios, the structure of the control design is immune to in–sample overfitting. This observation enables the scalability of the proposed control scheme for ADNs with many DGs without incorporating an untractable number of scenarios into the model.

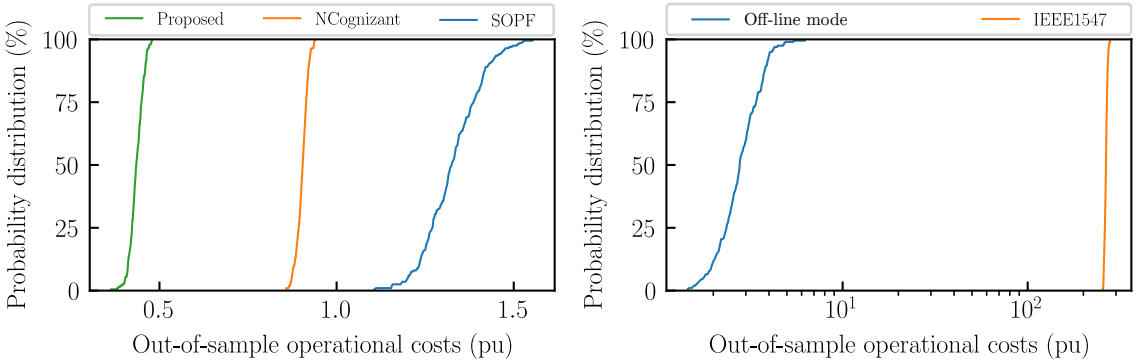

Figure 4.7: Out-of-sample performance evaluation of daily average operational costs, IEEE 34-bus. (a) Proposed, NCorgnizant, and SOPF; (b) Off–line mode and IEEE1547.

4.5.1 Proposed strategy in off–line mode

In this section, we drop the centralized updating process (see Figure 3.2) and evaluate the proposed strategy under a strictly local operation mode. For that, we perform the same simulations described in Section 4.5, but without a daily updating process, i.e., fixed control parameters $(\mathbf{G}, \mathbf{D})$ and fixed nominal set–point $(\overline{\mathbf{p}}_{\mathrm{G}}, \overline{\mathbf{q}}_{\mathrm{G}}, \overline{\mathbf{v}})$ throughout the whole simulation horizon. We obtain a purely local strategy such as IEEE1547, suitable for systems that lack sophisticated communication infrastructure.

Table 4.4: Mean value of selected metrics for the proposed strategy in off–line mode.

Case	AVV $\times 10^{-7}$	HUPI (kW)	HCPVP (kW)	HNL (kW)
IEEE 4–bus	31.6	479.6	443.1	22.9
IEEE 34–bus	7.3	42.6	54.5	12.5
IEEE 123–bus	54.0	416.5	551.8	99.7

Table 4.4 summarizes the simulation results obtained for the offline operation mode. Observe that the offline operation mode achieves better voltage regulation than the IEEE1547, with a difference in AVV of 2 orders of magnitude. The experiments suggest that polynomial policies combined with voltage droop controllers can be used as a strictly local control strategy for DGs.

4.5.2 Computational tractability

For the three test feeders, all instances of (3.26) were solved with Gurobi 9.0.1, setting a time limit termination condition of 15 minutes. The experiments were carried out on a laptop with a 7th Gen Intel(R) Core(TM) i7–7700HQ CPU @ 2.80GHz processor and 12 Gb of RAM. Figure 4.8 shows how the computational time increases with the number of controlled DGs. Recall that the proposed approach only requires (3.26) to be solved only once a day. Therefore, the results concerning computational tractability indicate the potential as an actual industrial application.

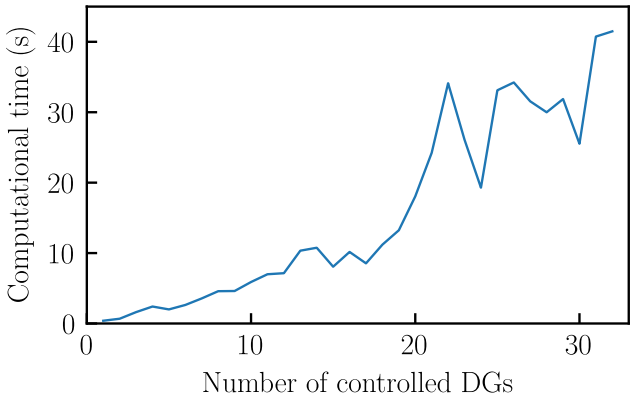

Figure 4.8: Computational time of solving the control tuning model (3.26) with 500 scenarios, IEEE 34–bus.

Chapter 5

Conclusions and future work

This work proposes a distributed control strategy for the operation of DGs in distribution networks. The control scheme is based on real-time local control rules that combine voltage proportional control and polynomial policies. We designed the control rules as parametric functions with suitable linear approximations, which enabled leveraging a tractable control tuning model that considers system-wide information. In this way, the tuning process embeds optimal global decisions within the local control rules.

Unlike similar approaches, we have investigated more flexible policies that can take different forms depending on the solution to the control tuning problem. In particular, besides proportional control, we have explored affine policies that are linear combinations of monomials of degrees 0, 1, 2, and 3 applied to normalized uncertainty values. A finite set of control parameters fully determines the proposed local control rules. The number of parameters is proportional to the number of DGs. To calculate them, we leveraged a control tuning model in the form of a two-stage robust SOCP optimization problem with a scenario-based uncertainty set. The objective function minimizes the total system energy consumption and voltage violations.

The overall control strategy is organized under a two-layer control hierarchy: a real-time control layer in charge of evaluating the local control policies in the order of $10 - 100$ms; and a centralized adaptivity layer in charge of periodically solving the control tuning model to update the local controllers on a daily basis. This scheme avoids communication overhead while still capitalizing on global available information.

Extensive computational experiments show that the proposed approach considerably outperforms similar state-of-the-art droop-based schemes. The algorithm is tested in three modified versions of the IEEE feeders under high solar penetration scenarios with nearly 100% solar penetration. Results reveal several features:

1. We have investigated operational conditions with high solar penetration. Indeed, for every day in the simulation horizon there is an interval where each of the studied control methods presented reverse power flow. Moreover, the 50% of the solar power injection is uncontrolled. Under these conditions, the proposed approach achieved the best performance among the alternatives, regulating voltage effectively meanwhile achieving high economical efficiency compared to the compared schemes. By means of this observation we conclude that the proposed approach has potential as a viable solution for the integration of high amounts of solar penetration.

2. One of the main concerns regarding the applicability of affine policies is the dependence on a large number of parameters to define them. Experimental results corresponding to the IEEE 123-bus feeder suggest that the control tuning model obtains a high-quality solution even though the number of scenarios is considerably less than the number of control parameters to be determined. Indeed, in that case, the number of scenarios was 500; meanwhile, the number of unknown control parameters to be determined was 1820. Furthermore, recall that the objective of the AdaLASSO regularization is to improve the out-of-sample performance by reducing the effect of in-sample overfitting. When analyzing the impact of the regularization, it is noticeable that even though it did improve the solution for the IEEE 34-bus case by 8%, the non-regularized solution still outperformed the alternative approaches. Moreover, in the IEEE 123-bus case, the improvement of the regularization was not significant, and again the proposed approach outperformed the others.

3. Experiments were carried out considering updating intervals of one day. In other words, the control tuning model is solved only once a day, and the parameters of the real-time control policies remain fixed during that period. The simulation results showed that the proposed approach outperforms similar proportional control and OPF based approaches, even though alternative approaches utilize updating cycles of 15 minutes. Therefore, the proposed approach reduces the communication overhead with respect to similar approaches that tend to be based on a shorter control horizon.

4. From the previous observation, natural questions arise concerning the applicability of the proposed real-time control rules to a strictly local design, i.e., calculating the control parameters only once and then fix them indefinitely (off-line mode). Off-line simulation suggests that the detriment in performance is not dramatic. In fact, in terms of voltage regulation the off-line mode outperformed the recomended standard IEEE-1547 [2]. However, the simulation was carried out for only one week therefore no definitive conclusions can be made because we are not considering slower seasonal components of the uncertainty.

5. We have proposed a SOCP tuning model suitable for off-the-shelf solvers. All instances of the tuning model were solved with Gurobi with a time limit of 15 minutes. Recall that the solution is actually required once a day. Therefore experiments suggest that the proposed approach satisfy actual computational tractability requirements, at least for systems of similar size.

As a consequence of the experimental results and the observations mentioned above, future research will consider the following problems:

1. The implementation of the proposed approach under strictly local operation conditions.

2. Controlling the risk aversion of the control tuning model. Note that the simulations corresponding to the AdaLASSO coefficient determination showed a noticeable gap between in-sample expected operational costs and out-of-sample operational costs. This difference is observed in Figure 4.4 and suggests that there is a need for controlling or at least evaluate the risk aversion of the control tuning problem because the obtained solution seem too robust for that particular simulation horizon.

3. The implementation of a more sophisticated adaptivity policy to construct the uncertainty set. For instance, the construction of the uncertainty set may consider as input the historical values of the state of the system and the operational performance. This issues is highly related to the previous item.

4. A more precise solution approach to deal with the bilinear terms $\mathbf{Gx}$ in (3.18) that naturally arise in the tuning of proportional controllers. This is important as many alternative local control strategies rely on proportional control in some way. In general, we are interested in solving problems where there are products of variables $z_1 = x_1 y_1$ and $z_2 = x_2 y_2$ are constrained by $(a + z_1)^2 + (b + z_2)^2 < c$, where a, b and c are constant scalars.

5. Explore similar families of functions for the real-time control rules:

   (a) Consider alternative approaches for the basis of polynomial functions (see [26] and [12]).

   (b) Consider dead-band incorporation for the proportional control policies (see [7]).

6. A solution approach that avoids numerical issues. Currently the barrier method is utilized to solve the control tuning model. For some instances the problem may present slow rate of convergence in the last iterations. In the present work a time limit termination condition is imposed forcing the solver to return sub-optimal solutions if optimality has not been achieved within a 15 minutes time limit. The numerical issues are present even though recommended settings have been applied regarding tightening the barrier converse tolerance and suppressing presolve reductions, among others. This raises the need for a more numerically reliable solution algorithm.

7. The evaluation of the risk introduced by the communication network delays and failures.

8. The incorporation of a more precise OPF sub-problem to reduce modeling error. In particular SOCP-OPF relaxations have the potential to improve the solution without dramatic impact in computational tractability. However SOCP relaxations do not hold exactness guarantees when explicit voltage upper limits bind at optimality. In the context of high solar penetration, BFM SOCP relaxations require the representation of voltage upperbounds without explicit constraints, for example, by properly constraining the net active and reactive power injection.

9. The extension of the proposed approach to a three-phase formulation.

10. The design of tighter convex reformulations for the stability constraint. Currently the stability constraint is enforced by the Frobenius norm constriction (3.12) of of the bound on the maximum singular value (3.11). This may introduce a significant error.

Appendix A

AMPL implementation of the control tuning model

```
# ************************************************************* #
# ############################ SETS ############################# #
# ************************************************************* #
set BUSES;
set LINKS within (BUSES cross BUSES);
set SCENARIOS;
param n_deg >= 0, integer;

# Polynomial policy degrees
set DEGS := 0..n_deg;
# ************************************************************* #
# ######################### PARAMETERS ########################## #
# ************************************************************* #
param cost_stability, >= 0;
param cost_lasso_v >= 0;
param cost_lasso_x >= 0;
param cost_putility >= 0;
param cost_vlim >= 0;

# Linear power flow model matrices
param R {LINKS}; # Active power to voltage sensitivity
param X {LINKS}; # Reactive power to voltage sensitivity

# Voltage soft bounds parameters
param v_lb {BUSES};  # Voltage lower bound
param v_ub {BUSES};  # Voltage upper bound
param v_set {BUSES}; # Voltage nominal set-point

# McCormick bounds on voltage variations and droop parameters
param dv_ub {BUSES}; # dv upper bound
param dv_lb {BUSES}; # dv lower bound
param gp_lb {BUSES}; # gp lower bound (upper bound is zero)
param gq_lb {BUSES}; # gq lower bound (upper bound is zero)

# DG size
param smax {BUSES};
# Maximum available controlled active power
param up_max {BUSES, SCENARIOS};
# Non-controlled active power injection variation
param dp_nc {BUSES, SCENARIOS};
# Non-controlled reactive power injection variation
```

```
param dq_nc {BUSES, SCENARIOS};
# Scaled maximum available controlled active power
param up_max_scaled {BUSES, SCENARIOS}, default 0;
# Scaled non-controlled active power injection variation
param dp_nc_scaled {BUSES, SCENARIOS}, default 0;
# Scaled non-controlled reactive power injection variation
param dq_nc_scaled {BUSES, SCENARIOS}, default 0;

param up_mean{BUSES}; # Nominal controlled active power output
param uq_mean{BUSES}; # Nominal controlled reactive power output
param p_mean{BUSES};  # Bus net active power injection
param q_mean{BUSES};  # Bus net reactive power injection

# AdaLASSO weighting factors
param w_DL_pp {BUSES, DEGS}, default 1;
param w_DL_pq {BUSES, DEGS}, default 1;
param w_DL_qp {BUSES, DEGS}, default 1;
param w_DL_qq {BUSES, DEGS}, default 1;
param w_DP_p {BUSES, DEGS}, default 1;
param w_DP_q {BUSES, DEGS}, default 1;

# ************************************************************* #
# ######################### VARIABLES ############################ #
# ************************************************************* #
# Operational costs upper bound
var epi_ub;

# Voltage limits auxiliary variables
var e_pos {BUSES, SCENARIOS}, >= 0;
var e_neg {BUSES, SCENARIOS}, >= 0;

var eps, >= 0; # Stability margin

# Voltage variation positive term
var dv_pos {BUSES, SCENARIOS}, >=0;
# Voltage variation negative term
var dv_neg {BUSES, SCENARIOS}, >=0;

# McCormick auxiliary variables
var k_p_pos {BUSES, SCENARIOS};
var k_p_neg {BUSES, SCENARIOS};
var k_q_pos {BUSES, SCENARIOS};
var k_q_neg {BUSES, SCENARIOS};

# Actuation
var dup {BUSES, SCENARIOS};
var duq {BUSES, SCENARIOS};

# Droop control coefficients
var gp {BUSES}, <= 0;
var gq {BUSES}, <= 0;

# Polynomial policy control coefficients
var DL_pp_pos {BUSES, DEGS}, >= 0;
var DL_pq_pos {BUSES, DEGS}, >= 0;
var DL_qp_pos {BUSES, DEGS}, >= 0;
var DL_qq_pos {BUSES, DEGS}, >= 0;
var DL_pp_neg {BUSES, DEGS}, >= 0;
```

```
var DL_pq_neg {BUSES, DEGS}, >= 0;
var DL_qp_neg {BUSES, DEGS}, >= 0;
var DL_qq_neg {BUSES, DEGS}, >= 0;

var DP_p_pos {BUSES, DEGS}, >= 0;
var DP_q_pos {BUSES, DEGS}, >= 0;
var DP_p_neg {BUSES, DEGS}, >= 0;
var DP_q_neg {BUSES, DEGS}, >= 0;

# ************************************************************* #
# ###################### MODEL DEFINITION ##################### #
# ************************************************************* #
# Objective function
minimize FOBJ_ROBUST:
    # Operational costs upper bound
    + epi_ub
    # Stability cost
    - cost_stability * eps
    # Droop coefficients shrinkage
    - cost_lasso_v * sum {i in BUSES} (gp[i] + gq[i])
    # AdaLASSO penalty
    + cost_lasso_x * sum {i in BUSES, k in DEGS} (
        + w_DL_pp[i, k] * (DL_pp_pos[i,k] + DL_pp_neg[i,k])
        + w_DL_pq[i, k] * (DL_pq_pos[i,k] + DL_pq_neg[i,k])
        + w_DL_qp[i, k] * (DL_qp_pos[i,k] + DL_qp_neg[i,k])
        + w_DL_qq[i, k] * (DL_qq_pos[i,k] + DL_qq_neg[i,k])
        + w_DP_p[i, k]  * (DP_p_pos[i,k]  + DP_p_neg[i,k])
        + w_DP_q[i, k]  * (DP_q_pos[i,k]  + DP_q_neg[i,k])
      )
;

# ************************************************************* #
# epi_ub belongs to the operational costs epigraph for all      #
# scenarios                                                     #
# ************************************************************* #
s.t. EPIGRAPH{t in SCENARIOS}:
    epi_ub
    >=
    # Total utility power injection
    cost_putility * (
        # Losses
        sum{i in BUSES} (
            (p_mean[i] + dp_nc[i, t] + dup[i, t]) *
            sum{j in BUSES} (
                R[i, j] * (p_mean[j] + dp_nc[j, t] + dup[j, t])
            )
            +
            (q_mean[i] + dq_nc[i, t] + duq[i, t]) *
            sum{j in BUSES} (
                R[i, j] * (q_mean[j] + dq_nc[j, t] + duq[j, t])
            )
        )
        # Negative net bus power injection
        - sum{i in BUSES} (p_mean[i] + dp_nc[i, t] + dup[i, t])
    )
    # Cost voltage limits
    + cost_vlim * sum{i in BUSES} (e_neg[i, t] + e_pos[i, t])
;
```

```
# Voltage bounds
s.t. VOLTAGE_LB {i in BUSES, t in SCENARIOS}: dv_lb[i] <= -
    dv_neg[i, t];
s.t. VOLTAGE_UB {i in BUSES, t in SCENARIOS}: dv_ub[i] >=
    dv_pos[i, t];

# Voltage soft bounds
s.t. VOLTAGE_LB_SOFT {i in BUSES, t in SCENARIOS}:
    v_lb[i] - e_neg[i, t] <= v_set[i] + (dv_pos[i,t]  -
        dv_neg[i,t])
;
s.t. VOLTAGE_UB_SOFT {i in BUSES, t in SCENARIOS}:
    v_ub[i] + e_pos[i, t] >= v_set[i] + (dv_pos[i,t]  -
        dv_neg[i,t])
;

# Linear power flow model
s.t. LINEAR_PFLOW {i in BUSES, t in SCENARIOS}:
    (dv_pos[i,t]  -  dv_neg[i,t])
    ==
    sum{j in BUSES} (
        # Controlled part
        + R[i, j] * dup[j, t]
        + X[i, j] * duq[j, t]
        # Uncontrolled part
        + R[i, j] * dp_nc[j, t] + X[i, j] * dq_nc[j, t]
    )
;

# Stability
s.t. STABILITY:
    sum {(i, j) in LINKS} ((R[i, j] * gp[j] + X[i, j] * gq[j])^2)
        <= 1 - eps
;

# Control policy dup
s.t. PROPOSED_CONTROL_P_MCCORMICK{j in BUSES, t in SCENARIOS}:
    dup[j, t]
    ==
    + k_p_pos[j, t] - k_p_neg[j, t]
    + sum{k in DEGS} (
        # \xi_L laws
        + (DL_pp_pos[j, k] - DL_pp_neg[j, k]) * (dp_nc_scaled[j,
            t])^k
        + (DL_pq_pos[j, k] - DL_pq_neg[j, k]) * (dq_nc_scaled[j,
            t])^k
        # \xi_P laws
        + (DP_p_pos[j, k] - DP_p_neg[j, k]) * (up_max_scaled[j,
            t])^k
    )
;

# Control policy duq
s.t. PROPOSED_CONTROL_Q_MCCORMICK{j in BUSES, t in SCENARIOS}:
    duq[j, t]
    ==
    + k_q_pos[j, t] - k_q_neg[j, t]
```

```
    + sum{k in DEGS} (
        # \xi_L laws
        + (DL_qp_pos[j, k] - DL_qp_neg[j, k]) * (dp_nc_scaled[j,
           t])^k
        + (DL_qq_pos[j, k] - DL_qq_neg[j, k]) * (dq_nc_scaled[j,
           t])^k
        # \xi_D laws
        + (DP_q_pos[j, k] - DP_q_neg[j, k]) * (up_max_scaled[j,
           t])^k
    )
;

# DG size constraint
s.t. PROPOSED_SMAX{i in BUSES, t in SCENARIOS}:
    (up_mean[i] + dup[i, t])^2 + (uq_mean[i] + duq[i, t])^2 <=
       smax[i]^2
;

# DG maximum available power constraint
s.t. PROPOSED_PMAX{i in BUSES, t in SCENARIOS}:
    up_mean[i] + dup[i, t] <= up_max[i, t]
;

# DG minimum controlled active power
s.t. PROPOSED_PMIN{i in BUSES, t in SCENARIOS}:
    up_mean[i] + dup[i, t] >= 0
;

# McCormick envelopes
s.t. MCCORMICK_P_POS0{i in BUSES, t in SCENARIOS}:
-k_p_pos[i, t] + gp_lb[i]*dv_pos[i, t] + 0*gp[i] <= gp_lb[i]*0
;
s.t. MCCORMICK_P_POS1{i in BUSES, t in SCENARIOS}:
-k_p_pos[i, t] + 0*dv_pos[i, t] + dv_ub[i]*gp[i] <= 0*dv_ub[i]
;
s.t. MCCORMICK_P_POS2{i in BUSES, t in SCENARIOS}:
-k_p_pos[i, t] + 0*dv_pos[i, t] + 0*gp[i] >= 0
;
s.t. MCCORMICK_P_POS3{i in BUSES, t in SCENARIOS}:
-k_p_pos[i, t] + gp_lb[i]*dv_pos[i, t] + dv_ub[i]*gp[i] >=
   gp_lb[i]*dv_ub[i]
;

s.t. MCCORMICK_P_NEG0{i in BUSES, t in SCENARIOS}:
-k_p_neg[i, t] + gp_lb[i]*dv_neg[i, t] + 0*gp[i] <= gp_lb[i]*0
;
s.t. MCCORMICK_P_NEG1{i in BUSES, t in SCENARIOS}:
-k_p_neg[i, t] + 0*dv_neg[i, t] + -dv_lb[i]*gp[i] <= 0*-dv_lb[i]
;
s.t. MCCORMICK_P_NEG2{i in BUSES, t in SCENARIOS}:
-k_p_neg[i, t] + 0*dv_neg[i, t] + 0*gp[i] >= 0
;
s.t. MCCORMICK_P_NEG3{i in BUSES, t in SCENARIOS}:
-k_p_neg[i, t] + gp_lb[i]*dv_neg[i, t] + -dv_lb[i]*gp[i] >=
   gp_lb[i]*-dv_lb[i]
;

s.t. MCCORMICK_Q_POS0{i in BUSES, t in SCENARIOS}:
```

```
-k_q_pos[i, t] + gq_lb[i]*dv_pos[i, t] + 0*gq[i] <= gq_lb[i]*0
;
s.t. MCCORMICK_Q_POS1{i in BUSES, t in SCENARIOS}:
-k_q_pos[i, t] + 0*dv_pos[i, t] + dv_ub[i]*gq[i] <= 0*dv_ub[i]
;
s.t. MCCORMICK_Q_POS2{i in BUSES, t in SCENARIOS}:
-k_q_pos[i, t] + 0*dv_pos[i, t] + 0*gq[i] >= 0
;
s.t. MCCORMICK_Q_POS3{i in BUSES, t in SCENARIOS}:
-k_q_pos[i, t] + gq_lb[i]*dv_pos[i, t] + dv_ub[i]*gq[i] >=
   gq_lb[i]*dv_ub[i]
;

s.t. MCCORMICK_Q_NEG0{i in BUSES, t in SCENARIOS}:
-k_q_neg[i, t] + gq_lb[i]*dv_neg[i, t] + 0*gq[i] <= gq_lb[i]*0
;
s.t. MCCORMICK_Q_NEG1{i in BUSES, t in SCENARIOS}:
-k_q_neg[i, t] + 0*dv_neg[i, t] + -dv_lb[i]*gq[i] <= 0*-dv_lb[i]
;
s.t. MCCORMICK_Q_NEG2{i in BUSES, t in SCENARIOS}:
-k_q_neg[i, t] + 0*dv_neg[i, t] + 0*gq[i] >= 0
;
s.t. MCCORMICK_Q_NEG3{i in BUSES, t in SCENARIOS}:
-k_q_neg[i, t] + gq_lb[i]*dv_neg[i, t] + -dv_lb[i]*gq[i] >=
   gq_lb[i]*-dv_lb[i]
;
```

Bibliography

[1] F Mancilla-David, A Angulo, and A Street. Power Management in Active Distribution Systems Penetrated by Photovoltaic Inverters: A Data-Driven Robust Approach. *IEEE Transactions on Smart Grid*, page 1, 2019.

[2] IEEE Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interfaces. *IEEE Std 1547-2018 (Revision of IEEE Std 1547-2003)*, pages 1–138, apr 2018.

[3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning, second edition.* Springer, 2 edition, 2009.

[4] U.S. Renewable Portafolio Standards 3. `https://emp.lbl.gov/publications/us-renewables-portfolio-standards-3` (Accessed: 2021-09-14).

[5] K Turitsyn, P Sulc, S Backhaus, and M Chertkov. Options for Control of Reactive Power by Distributed Photovoltaic Generators. *Proceedings of the IEEE*, 99(6):1063–1073, jun 2011.

[6] Cuo Zhang and Yan Xu. Hierarchically-Coordinated Voltage/VAR Control of Distribution Networks Using PV Inverters. *IEEE Transactions on Smart Grid*, 11(4):2942–2953, jul 2020.

[7] Xinyang Zhou, Lijun Chen, Masoud Farivar, Zhiyuan Liu, and Steven Low. Reverse and Forward Engineering of Local Voltage Control in Distribution Networks. *IEEE Transactions on Automatic Control*, 2020.

[8] Yixin Liu, Li Guo, and Chengshan Wang. A robust operation-based scheduling optimization for smart distribution networks with multi-microgrids. *Applied Energy*, 228:130–140, 2018.

[9] T Ding, S Liu, W Yuan, Z Bie, and B Zeng. A Two-Stage Robust Reactive Power Optimization Considering Uncertain Wind Power Integration in Active Distribution Networks. *IEEE Transactions on Sustainable Energy*, 7(1):301–311, 2016.

[10] Yongjie Wang, Wenchuan Wu, Boming Zhang, Zhengshuo Li, and Weiye Zheng. Robust voltage control model for active distribution network considering PVs and loads uncertainties. In *2015 IEEE Power Energy Society General Meeting*, pages 1–5, 2015.

[11] J W Simpson-Porco, F Dörfler, and F Bullo. Voltage Stabilization in Microgrids via Quadratic Droop Control. *IEEE Transactions on Automatic Control*, 62(3):1239–1253, mar 2017.

[12] R A Jabr. Linear Decision Rules for Control of Reactive Power by Distributed Photovoltaic Generators. *IEEE Transactions on Power Systems*, 33(2):2165–2174, mar 2018.

[13] R A Jabr. Robust Volt/VAr Control With Photovoltaics. *IEEE Transactions on Power Systems*, 34(3):2401–2408, may 2019.

[14] Yusuf Gupta, Suryanarayana Doolla, Kishore Chatterjee, and Bikash Chandra Pal. Optimal DG Allocation and Volt–Var Dispatch for a Droop-Based Microgrid. *IEEE Transactions on Smart Grid*, 12(1):169–181, jan 2021.

[15] Xianzhuo Sun, Wen Zhang, and Jian Chen. Optimize globally, control locally: Coordinated optimal local voltage control in hybrid AC/DC microgrid. *International Journal of Electrical Power & Energy Systems*, 118:105734, 2020.

[16] A Bernstein and E Dall'Anese. Real-Time Feedback-Based Optimization of Distribution Grids: A Unified Approach. *IEEE Transactions on Control of Network Systems*, 6(3):1197–1209, 2019.

[17] A Hauswirth, A Zanardi, S Bolognani, F Dörfler, and G Hug. Online optimization in closed loop on the power flow manifold. In *2017 IEEE Manchester PowerTech*, pages 1–6, 2017.

[18] Brett A Robbins, Christoforos N Hadjicostis, and Alejandro D Domínguez-García. A Two-Stage Distributed Architecture for Voltage Control in Power Distribution Systems. *IEEE Transactions on Power Systems*, 28(2):1470–1482, may 2013.

[19] V Kekatos, L Zhang, G B Giannakis, and R Baldick. Voltage Regulation Algorithms for Multiphase Power Distribution Grids. *IEEE Transactions on Power Systems*, 31(5):3913–3923, 2016.

[20] K Baker, A Bernstein, E Dall'Anese, and C Zhao. Network-Cognizant Voltage Droop Control for Distribution Grids. *IEEE Transactions on Power Systems*, 33(2):2098–2108, mar 2018.

[21] Shichao Liu, Xiaoyu Wang, and Peter Xiaoping Liu. Impact of Communication Delays on Secondary Frequency Control in an Islanded Microgrid. *IEEE Transactions on Industrial Electronics*, 62(4):2021–2031, apr 2015.

[22] Murat Kuzlu and Manisa Pipattanasomporn. Assessment of communication technologies and network requirements for different smart grid applications. In *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–6, feb 2013.

[23] Kurt M Anstreicher, Samuel Burer, and Kyungchan Park. Convex Hull Representations for Bounded Products of Variables, 2020.

[24] N Nazare, A Street, A Lorca, and E Sauma. Long-term multistage hydrothermal dispatch via regularized affine policies. In *21st Power System Computation Conference (PSCC)*, 2020.

[25] Dimitris Bertsimas and Vineet Goyal. On the power and limitations of affine policies in two-stage adaptive optimization. *Mathematical Programming*, 134(2):491–531, 2012.

[26] Tillmann Mühlpfordt, Timm Faulwasser, Veit Hagenmeyer, Line Roald, and Sidhant Misra. On polynomial real-time control policies in stochastic AC optimal power flow. *Electric Power Systems Research*, 189:106792, 2020.

[27] Hui Zou. The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

[28] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 2014.

[29] Arkadi Nemirovski and Alexander Shapiro. Convex Approximations of Chance Constrained Programs. *SIAM Journal on Optimization*, 17:969–996, 2006.

[30] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust Optimization:*. Princeton University Press, 2009.

[31] D Bertsimas, E Litvinov, X A Sun, J Zhao, and T Zheng. Adaptive Robust Optimization for the Security Constrained Unit Commitment Problem. *IEEE Transactions on Power Systems*, 28(1):52–63, feb 2013.

[32] Tao Ding, Cheng Li, Yongheng Yang, Jiangfeng Jiang, Zhaohong Bie, and Frede Blaabjerg. A Two-Stage Robust Optimization for Centralized-Optimal Dispatch of Photovoltaic Inverters in Active Distribution Networks. *IEEE Transactions on Sustainable Energy*, 8(2):744–754, apr 2017.

[33] Noemi G Cobos, José M Arroyo, and Alexandre Street. Least-Cost Reserve Offer Deliverability in Day-Ahead Generation Scheduling Under Wind Uncertainty and Generation and Network Outages. *IEEE Transactions on Smart Grid*, 9(4):3430–3442, jul 2018.

[34] Dimitris Bertsimas and Melvyn Sim. The Price of Robustness. *Operations Research*, 52(1):35–53, 2004.

[35] Alexandre Velloso, Alexandre Street, David Pozo, Jose M Arroyo, and Noemi G Cobos. Two-Stage Robust Unit Commitment for Co-Optimized Electricity Markets: An Adaptive Data-Driven Approach for Scenario-Based Uncertainty Sets. *IEEE Transactions on Sustainable Energy*, 11(2):958–969, apr 2020.

[36] S H Low. Convex Relaxation of Optimal Power Flow—Part I: Formulations and Equivalence. *IEEE Transactions on Control of Network Systems*, 1(1):15–27, mar 2014.

[37] J Lavaei and S H Low. Zero Duality Gap in Optimal Power Flow Problem. *IEEE Transactions on Power Systems*, 27(1):92–107, feb 2012.

[38] B. Stott and O. Alsac. Fast decoupled load flow. *IEEE Transactions on Power Apparatus and Systems*, PAS-93(3):859–869, May 1974.

[39] M E Baran and F F Wu. Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Transactions on Power Delivery*, 4(2):1401–1407, 1989.

[40] A Bernstein, C Wang, E Dall'Anese, J Le Boudec, and C Zhao. Load Flow in Multiphase Distribution Networks: Existence, Uniqueness, Non-Singularity and Linear Models. *IEEE Transactions on Power Systems*, 33(6):5832–5843, nov 2018.

[41] R A Jabr. Radial distribution load flow using conic programming. *IEEE Transactions on Power Systems*, 21(3):1458–1459, aug 2006.

[42] Xiaoqing Bai, Hua Wei, Katsuki Fujisawa, and Yong Wang. Semidefinite programming for optimal power flow problems. *International Journal of Electrical Power & Energy Systems*, 30(6):383–392, 2008.

[43] B Subhonmesh, S H Low, and K M Chandy. Equivalence of branch flow and bus injection models. In *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1893–1899, oct 2012.

[44] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, USA, 2004.

[45] S H Low. Convex Relaxation of Optimal Power Flow—Part II: Exactness. *IEEE Transactions on Control of Network Systems*, 1(2):177–189, jun 2014.

[46] S Bose, D F Gayme, K M Chandy, and S H Low. Quadratically Constrained Quadratic Programs on Acyclic Graphs With Application to Power Flow. *IEEE Transactions on Control of Network Systems*, 2(3):278–287, 2015.

[47] Z Luo, W Ma, A M So, Y Ye, and S Zhang. Semidefinite Relaxation of Quadratic Optimization Problems. *IEEE Signal Processing Magazine*, 27(3):20–34, may 2010.

[48] B Kocuk, S S Dey, and X A Sun. Inexactness of SDP Relaxation and Valid Inequalities for Optimal Power Flow. *IEEE Transactions on Power Systems*, 31(1):642–651, jan 2016.

[49] B C Lesieutre, D K Molzahn, A R Borden, and C L DeMarco. Examining the limits of the application of semidefinite programming to power flow problems. In *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1492–1499, 2011.

[50] J Lavaei, D Tse, and B Zhang. Geometry of Power Flows and Optimization in Distribution Networks. *IEEE Transactions on Power Systems*, 29(2):572–583, mar 2014.

[51] L Gan, N Li, U Topcu, and S H Low. Exact Convex Relaxation of Optimal Power Flow in Radial Networks. *IEEE Transactions on Automatic Control*, 60(1):72–87, jan 2015.

[52] Burak Kocuk, Santanu S Dey, and X Andy Sun. Strong SOCP Relaxations for the Optimal Power Flow Problem. *Operations Research*, 64(6):1177–1196, 2016.

[53] Carleton Coffrin, Hassan L. Hijazi, and Pascal Van Hentenryck. The QC Relaxation: A Theoretical and Computational Study on Optimal Power Flow. *IEEE Transactions on Power Systems*, 31(4):3008–3018, jul 2016.

[54] L Gan, N Li, U Topcu, and S H Low. Exact Convex Relaxation of Optimal Power Flow in Radial Networks. *IEEE Transactions on Automatic Control*, 60(1):72–87, jan 2015.

[55] C Wang, A Bernstein, J Le Boudec, and M Paolone. Explicit Conditions on Existence and Uniqueness of Load-Flow Solutions in Distribution Networks. *IEEE Transactions on Smart Grid*, 9(2):953–962, mar 2018.

[56] M E Baran and F F Wu. Optimal capacitor placement on radial distribution systems. *IEEE Transactions on Power Delivery*, 4(1):725–734, jan 1989.

[57] Morteza Aien, Ali Hajebrahimi, and Mahmud Fotuhi-Firuzabad. A comprehensive review on uncertainty modeling techniques in power system studies. *Renewable and Sustainable Energy Reviews*, 57:1077–1089, 2016.

[58] Mohammad Hemmati, Behnam Mohammadi-Ivatloo, and Alireza Soroudi. Chapter 2 - Uncertainty management in decision-making in power system operation. In Shady H E [Abdel Aleem], Almoataz Youssef Abdelaziz, Ahmed F Zobaa, and Ramesh Bansal, editors, *Decision Making Applications in Modern Power Systems*, pages 41–62. Academic Press, 2020.

[59] Chao Ning and Fengqi You. Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming. *Computers & Chemical Engineering*, 125:434–448, 2019.

[60] J William Helton and Jiawang Nie. Semidefinite representation of convex sets. *Mathematical Programming*, 122(1):21–64, 2010.

[61] Narges Kazemzadeh, Sarah M Ryan, and Mahdi Hamzeei. Robust optimization vs. stochastic programming incorporating risk measures for unit commitment with uncertain variable renewable generation. *Energy Systems*, dec 2017.

[62] Allen J Wood, Bruce F Wollenberg, and Gerald B Sheble. *Power generation, operation and control*. John Wiley & Sons, 3 edition, 2014.

[63] Josep M. Guerrero, Juan C. Vasquez, José Matas, Luis García de Vicuna, and Miguel Castilla. Hierarchical control of droop-controlled ac and dc microgrids—a general approach toward standardization. *IEEE Transactions on Industrial Electronics*, 58(1):158–172, Jan 2011.

[64] Jianqing Fan and Runze Li. Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[65] Garth P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I - convex underestimating problems. *Math. Program.*, 10(1):147–175, 1976.

[66] Faiz A Al-Khayyal and James E Falk. Jointly Constrained Biconvex Programming. *Mathematics of Operations Research*, 8(2):273–286, 1983.

[67] Natashia Boland, Santanu S. Dey, Thomas Kalinowski, Marco Molinaro, and Fabian Rigterink. Bounding the gap between the mccormick relaxation and the convex hull for bilinear functions. *Math. Program.*, 162(1-2):523–535, 2017.

[68] Goodwin GC, Graebe SF, and Maria Elodia Salgado. *Control System Design*. Prentice Hall, 2001.

[69] Zhaojian Wang, Feng Liu, Yifan Su, Peng Yang, and Boyu Qin. Asynchronous distributed voltage control in active distribution networks. *Automatica*, 122:109269, 2020.

[70] IEEE tests feeders, 2020. Accessed: 2021-03-30.

[71] Pecan Street Dataport. `https://www.pecanstreet.org/dataport/` (Accessed: 2020-12-30).

[72] K Qian, C Zhou, M Allan, and Y Yuan. Modeling of Load Demand Due to EV Battery Charging in Distribution Systems. *IEEE Transactions on Power Systems*, 26(2):802–810, may 2011.

[73] Gurobi reference: Barconvtol parameter. `https://www.gurobi.com/documentation/9.0/refman/barconvtol.html` (Accessed: 2021-10-02).