Proceedings of the 52nd Hawaii International Conference on System Sciences | 2019

Autonomous Multi-Stage Flexible OPF for Active Distribution Systems with DERs

Sakis Meliopoulos[1]
sakis.m@gatech.edu

Chiyang Zhong[1]
c.zhong@gatech.edu

George Cokkinides[1]
cokkinides@comcast.net

Boqi Xie[1]
airxbq@gatech.edu

Catherine Dalton[2]
cdalton@epri.com

Paul Myrda[2]
pmyrda@epri.com

Evangelos Farantatos[2]
efarantatos@epri.com

[1]Georgia Institute of Technology

[2]Electric Power Research Institute

Abstract

The variability of renewable resources creates challenges in the operation and control of power systems. One way to cope with this issue is to use the flexibility of customer resources in addition to utility resources to mitigate this variability. We present an approach that autonomously optimizes the available distributed energy resources (DERs) of the system to optimally balance generation and load and/or levelize the voltage profile. The method uses a dynamic state estimator which is continuously running on the system providing the real-time dynamic model of the system and operating condition. At user selected time intervals, the real-time model and operating condition is used to autonomously assemble a multi-stage optimal power flow in which customer energy resources are represented with their controls, allowing the use of customer flexibility to be part of the solution. Customer DERs may include photovoltaic rooftops with controllable inverters, batteries, thermostatically controlled loads, smart appliances, etc. The paper describes the autonomous formation of the Multi-Stage Flexible Optimal Power Flow and the solution of the problem, and presents sample results.

1. Introduction

A variety of distributed energy resources (DERs), such as photovoltaic (PV) units, energy storage systems, thermostatically controlled loads, wind turbines, etc. are typically connected to modern-day electric power distribution systems. As DERs use advanced converters, the controllability of the distribution system has drastically increased. It is of great significance to optimally control the DERs to achieve reliable and secure operation as the variability of these resources can be substantial.

Different control schemes for distribution systems have been introduced in the literature. Demand response (DR) is a program utilized by utilities that motivates load changes on the customer side to improves system reliability. Both centralized and decentralized control strategies are studied in [1], reaching a conclusion stating that a hybrid approach achieves the best performance. A market-clearing scheme for DR is developed in [2], providing incentives to the customers enrolled in the DR program. Thermostatically controlled loads (TCLs) such as refrigerators and air conditioners can participate in direct load control. By utilizing the temperature dead-band of such loads, [3] proposes to control the ON/OFF operation of an aggregation of TCLs through state estimation using the Kalman filter. Similar to TCLs, plug-in electric vehicles (PEVs) can be controlled to improve the demand profile. By using concepts from non-cooperative games, a decentralized control strategy is introduced in [4] to achieve generation cost minimization and demand valley-filling. Another non-cooperative game based approach is proposed for the generation side by Chen and Zhu [5]. The distributed generation (DG) units can be used to mitigate voltage issues as stated in [6], which uses a voltage sensitivity approach with the help of surface fitting techniques. Due to the growing need for AC/DC conversions in distribution networks, AC/DC hybrid grids have gained much attention in the power system community. For such hybrid systems, a two-stage stochastic dispatch scheme is proposed in [7] and a flexible voltage control strategy using a linearized coupling model between DC voltage and AC frequency is described in [8].

Optimal control actions of a distribution system can be obtained by formulating and solving an optimal power flow (OPF) problem. Different optimization algorithms have been proposed. A gradient based centralized control strategy is discussed in [9] and a generic algorithm based method developed is compared

URI: https://hdl.handle.net/10125/59781
ISBN: 978-0-9981331-2-6

with a heuristic approach in [10]. Reference [11] proposes a distribution network restoration strategy using a multi-agent framework and it is formulated as a second-order cone programming problem. Quadratic programming is used to solve the AC OPF problem for distribution systems and compared against non-linear, quadratically constrained, and linearized approaches in [12]. Locational marginal pricing schemes at the distribution level can also be formulated as an OPF problem [13], [14].

The authors of this paper have previously proposed a distribution management system (DMS) that achieves real-time state estimation, protection, optimization and control of the system [15]. Here we propose an extension of this approach that uses the state estimation results to autonomously form and solve a multi-stage flexible OPF (F-OPF) problem for DER penetrated distribution systems. “Autonomous” means that the whole process is free of human/operator input. Once the user enters the models of the individual devices and the information of what is measured at the various meters and relays, the state estimation is performed autonomously, the multi-stage OPF is created autonomously from the output of the state estimator, the solution is also autonomously calculated, and the optimal controls are send to the DERs or other controllable devices. “Flexible” means that the proposed method includes customer controllable resources (DERs) for the operation of the system, thus customer flexibility is incorporated. In this paper, a stage is defined as a time period with a user-set interval. The approach is being implemented in an advanced distribution management system under a DoE/ENERGIZE project with the plan to demonstrate the method on several feeders of the National Grid and the New Mexico power company. To our knowledge this is the first time that dynamic state estimation is linked to autonomous optimization and control of distributed energy resources.

The paper is organized as follows. First we present the modeling approach that enables an object oriented approach to the state estimation and the subsequent autonomous formulation of the F-OPF in section 2. Modeling and state estimation are presented in section 3. The multi-stage F-OPF is presented in section 4. The solution method of the F-OPF is presented in section 5. Example test results are provided in section 6. Finally, concluding remarks are provided in section 7.

2. Description of Overall Approach

This section provides an overview of the autonomous approach, as illustrated in Figure 1. In a physical power system, relays and sensors are used to measure physical quantities, such as voltages, currents, temperatures, etc. These measurements and the information of the devices in the system are sent to a dynamic state estimator, which runs the quasi-dynamic state estimation and provides the validated model as well as the estimated operating condition of the system to setup the multi-stage F-OPF problem. Forecasts of load variations, generation from PV and wind, etc. are provided to set up the F-OPF for the future stages. Solving such OPF problem yields the optimal control actions to be implemented back into the physical plant so that the state of the system can be driven towards the optimal operating condition.

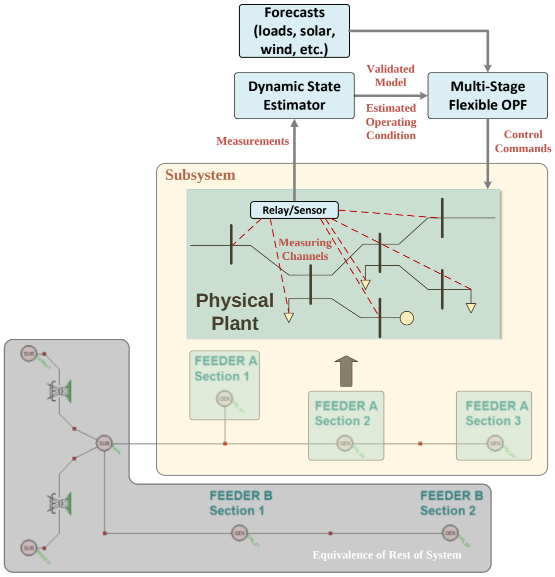


Figure 1: Implementation of the autonomous flexible OPF in a distribution system

3. Modeling and State Estimation

This section introduces the method of physically based modeling for devices, which are constructed as detailed mathematical objects derived from their physical circuit [16], and dynamic state estimation that uses these detailed device models to estimate the states of the system [17]. In this paper, detailed modeling is performed in the quasi-dynamic domain, where only slow dynamics like electromechanical transients in motors, generators, controllers, etc. are considered.

3.1. Quadratized Device Model

Each device is modeled as a quadratized device model (QDM). The QDM is derived by considering the mathematical model of a specific device and casting it into the syntax of QDM. In case there are nonlinearities of order higher than 2, additional state variables are introduced to reduce the nonlinearities to no higher than order 2. The general expression of a QDM is

$$\mathbf{i}(t) = Y_{x1}\mathbf{x} + Y_{u1}\mathbf{u} + D_{x1}\frac{d\mathbf{x}}{dt} + C_1 \tag{1a}$$

$$0 = Y_{x2}\mathbf{x} + Y_{u2}\mathbf{u} + D_{x2}\frac{d\mathbf{x}}{dt} + C_2 \tag{1b}$$

$$0 = Y_{x3}\mathbf{x} + Y_{u3}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{x3}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{u3}^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{ux3}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + C_3 \tag{1c}$$

$$\mathbf{h} = Y_{qx}\mathbf{x} + Y_{qu}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{qx}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{qu}^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{qux}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + C_q \tag{1d}$$

subject to $\mathbf{h} \le 0$, $\mathbf{u_{qmin}} \le \mathbf{u} \le \mathbf{u_{qmax}}$, and $|\mathbf{du}| \le \mathbf{u_{qlim}}$. Vectors $\mathbf{i}$, $\mathbf{x}$, and $\mathbf{u}$ are the terminal currents, states, and controls of the device at time t, respectively. The states include terminal voltages and device internal states. The functional constraints are expressed by $\mathbf{h}$, while the lower and upper control bounds correspond to vectors $\mathbf{u_{qmin}}$ and $\mathbf{u_{qmax}}$. $\mathbf{u_{hlim}}$ represents permissible step size for the controls of this device. The coefficients to the linear and differential terms are stored in matrices Y and D, respectively. The coefficients to the quadratic terms are defined in matrices F and the constant terms are given in vectors C. Note that only real-value matrices and vectors are used in the QDM. Each device model also includes the connectivity information (terminal node names).

3.2. Algebraic Quadratic Companion Form

The device QDM is integrated using quadratic integration to yield the algebraic quadratic companion form (AQCF). The integration time intervals and step are: $[t, t_m]$ and $[t, t+h]$, where $t_m = t + h/2$ and h is the time step [18]. The device AQCF syntax is:

$$\begin{Bmatrix} \mathbf{i}(t) \\ 0 \\ 0 \\ \mathbf{i}(t_m) \\ 0 \\ 0 \end{Bmatrix} = Y_x\mathbf{x} + Y_u\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_x^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_u^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{ux}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} - B \tag{2}$$

$$B = -N_x\mathbf{x}(t-h) - N_u\mathbf{u}(t-h) - M\mathbf{i}(t-h) - K$$

$$\mathbf{h} = Y_{fx}\mathbf{x} + Y_{fu}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{fx}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{fu}^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{fux}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + C_f$$

subject to $\mathbf{h} \le 0$, $\mathbf{u_{min}} \le \mathbf{u} \le \mathbf{u_{max}}$, and $|\mathbf{du}| \le \mathbf{u_{lim}}$. A past history vector B is introduced to store the model information from the previous simulation time step. The dimensionality of the AQCF is twice that of the QDM since it contains information from both time t and t_m. The matrices and vectors in (2) can be formed directly from those in (1) as follows.

$$Y_x = \begin{bmatrix} \frac{4}{h}D_{x1} + Y_{x1} & -\frac{8}{h}D_{x1} \\ \frac{4}{h}D_{x2} + Y_{x2} & -\frac{8}{h}D_{x2} \\ Y_{x3} & 0 \\ \frac{1}{2h}D_{x1} & \frac{2}{h}D_{x1} + Y_{x1} \\ \frac{1}{2h}D_{x2} & \frac{2}{h}D_{x2} + Y_{x2} \\ 0 & Y_{x3} \end{bmatrix} \quad Y_u = \begin{bmatrix} Y_{u1} & 0 \\ Y_{u2} & 0 \\ Y_{u3} & 0 \\ 0 & Y_{u1} \\ 0 & Y_{u2} \\ 0 & Y_{u3} \end{bmatrix} \quad K = \begin{bmatrix} 0 \\ 0 \\ C_3 \\ \frac{3}{2}C_1 \\ \frac{3}{2}C_2 \\ C_3 \end{bmatrix}$$

$$F_x = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ F_{x3} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & F_{x3} \end{bmatrix} \quad F_u = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ F_{u3} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & F_{u3} \end{bmatrix} \quad F_{ux} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ F_{ux3} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & F_{ux3} \end{bmatrix}$$

$$N_x = \begin{bmatrix} -Y_{x1} + \frac{4}{h}D_{x1} \\ -Y_{x2} + \frac{4}{h}D_{x2} \\ 0 \\ \frac{1}{2}Y_{x1} - \frac{5}{2h}D_{x1} \\ \frac{1}{2}Y_{x2} - \frac{5}{2h}D_{x2} \\ 0 \end{bmatrix} \quad N_u = \begin{bmatrix} -Y_{u1} \\ -Y_{u2} \\ 0 \\ \frac{1}{2}Y_{u1} \\ \frac{1}{2}Y_{u2} \\ 0 \end{bmatrix} \quad M = \begin{bmatrix} I_{size(i)} \\ 0 \\ 0 \\ -\frac{1}{2}I_{size(i)} \\ 0 \\ 0 \end{bmatrix}$$

$$Y_{fx} = \begin{bmatrix} Y_{qx} & 0 \\ 0 & Y_{qx} \end{bmatrix} \quad Y_{fu} = \begin{bmatrix} Y_{qu} & 0 \\ 0 & Y_{qu} \end{bmatrix} \quad C_f = \begin{bmatrix} C_q \\ C_q \end{bmatrix}$$

$$F_{fx} = \begin{bmatrix} F_{qx} & 0 \\ 0 & F_{qx} \end{bmatrix} \quad F_{fu} = \begin{bmatrix} F_{qu} & 0 \\ 0 & F_{qu} \end{bmatrix} \quad F_{fux} = \begin{bmatrix} F_{qux} & 0 \\ 0 & F_{qux} \end{bmatrix}$$

$$\mathbf{u_{min}} = \begin{bmatrix} \mathbf{u_{qmin}} \\ \mathbf{u_{qmin}} \end{bmatrix} \quad \mathbf{u_{max}} = \begin{bmatrix} \mathbf{u_{qmax}} \\ \mathbf{u_{qmax}} \end{bmatrix} \quad \mathbf{u_{lim}} = \begin{bmatrix} \mathbf{u_{qlim}} \\ \mathbf{u_{qlim}} \end{bmatrix}$$

AQCF is a standard syntax for modeling any device. The common syntax enables object-orientation of any application that uses the AQCF models. The states of the physically based device models satisfy every physical law and functional constraint, which makes the simulation results more realistic.

3.3. Device Modeling Example

We present the QDM model of a capacitor with discrete controls as an example that demonstrates that the discrete controls are transformed to continuous variables. Let u represent the discrete capacitor ON/OFF switching, $u = 0$ means capacitor is OFF, $u = 1$ means capacitor is ON. The QDM is:

$$\begin{aligned} I_{1r} &= -\omega C w_i & 0 &= w_r - (V_{1r} - V_{2r})u \\ I_{1i} &= \omega C w_r & 0 &= w_i - (V_{1i} - V_{2i})u \\ I_{2r} &= \omega C w_i & 0 &= p - (1-u)u \\ I_{2i} &= -\omega C w_r & 0 &= p - q^2 \end{aligned} \tag{3}$$

where $\omega = 2\pi f$ and f is the fundamental frequency. Subscripts r and i correspond respectively to the real and imaginary parts of the complex variables. Since in the QDM standard syntax, the quadratic terms are placed in equation set (1c), w_r and w_i are additional variables introduced to satisfy the standard. States p and q are used to convert the binary control into a continuous variable between 0 and 1. At a solution the value of u will be either 0 or 1. The through variables are I_{1r}, I_{1i}, I_{2r} and I_{2i}, while the state variables are V_{1r}, V_{1i}, V_{2r}, V_{2i}, w_r, w_i, p and q.

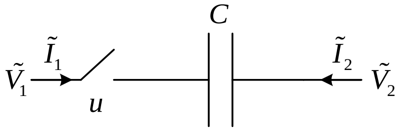


Figure 2: Capacitor circuit with switching capability

The next step is to obtain the corresponding AQCF of the capacitor. The total number of states in the AQCF is 16, including time t and t_m variables. Note that the resulting AQCF model does not have discrete control variables. However, when all equations of the model are satisfied the physical solution will correspond to the capacitor being ON or OFF.

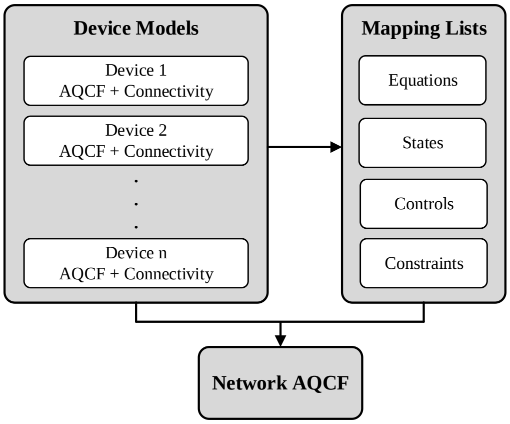


Figure 3: Network model construction process

3.4. Construction of the Network Model

Given the models of devices in the system in the AQCF syntax, the model of the entire system (network) is generated autonomously in the AQCF syntax as well. Mathematically, this is achieved as follows. At each node of the system, the physical law that relates the variables at this node are written. In case of an electrical node, this corresponds to Kirchhoff's current law (KCL). For nodes that are the interface of the system, the corresponding device equations remain the same. For common nodes, the device equations corresponding to one common node are combined by eliminating the through variables of the devices connected to that node and expressed in terms of the states.

The final expression of the network model consists of the equations at each node derived from KCL, as well as internal equations, functional constraints and control bounds of each device. During this process, mapping lists are first created based on the device connectivity to map the equations, states, controls and constraints from device level to network level. Then, with the device models, the network AQCF is automatically obtained. It has the same syntax as the device AQCF:

$$\begin{Bmatrix} \mathbf{i}(t) \\ 0 \\ 0 \\ \mathbf{i}(t_m) \\ 0 \\ 0 \end{Bmatrix} = Y_{nx}\mathbf{x} + Y_{nu}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{nx}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{nu}^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{nux}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} - B_n \tag{4}$$

$$B_n = -N_{nx}\mathbf{x}(t-h) - N_{nu}\mathbf{u}(t-h) - M_n\mathbf{i}(t-h) - K_n$$

$$\mathbf{h_n} = Y_{nfx}\mathbf{x} + Y_{nfu}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{nfx}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{nfu}^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{nfux}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + C_{nf}$$

subject to $\mathbf{h_n} \le 0$, $\mathbf{u_{nmin}} \le \mathbf{u} \le \mathbf{u_{nmax}}$, and $|\mathbf{du}| \le \mathbf{u_{nlim}}$, where subscript n refers to the network. The entire network formulation process is illustrated in Figure 3.

3.5. Dynamic State Estimator

Given the formulated network AQCF and available measurements in the network, the state estimator is able to operate automatically and output estimated states of the network for the OPF application.

With the available measurements in the network, the state estimator first creates the network measurement model consisting of four measurement types, i.e., a) actual measurement: measurements obtained from IEDs and created by the device AQCF and device-to-network mapping lists, b) derived measurement: created by derivations from actual measurements based on the network topology, c) virtual measurement: obtained from the network AQCF that provides the network KCL equations at the common nodes and the device internal equations, and d) pseudo measurement: created by knowing its approximate value (e.g, zero value for voltage at neutral phase during normal operation) with a relatively high measurement error. By combining all these measurement types together and substituting control variables with actual values from the control center, the network measurement model is formed in a similar syntax as the network AQCF:

$$\mathbf{z}(t, t_m) = y(\mathbf{x}) + \eta = Y_{zx}\mathbf{x} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{zx}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} - B_z + \eta \tag{5}$$

$$B_z = -N_{zx}\mathbf{x}(t-h) - M_z\mathbf{i}(t-h) - K_z$$

where subscript $\mathbf{z}$ refers to measurement model and η is the measurement error.

The state estimator works directly on the formulated network measurement model and provides the best estimate of the network states. The state estimator is based on the weighted least square method with the objective function being:

$$\text{Minimize } J = \left(\mathbf{z} - y(\mathbf{x})\right)^T W \left(\mathbf{z} - y(\mathbf{x})\right) \tag{6}$$

where W is a diagonal weight matrix with the weights defined as the inverse of the squared standard deviation of each measurement. The solution is computed by the following iterative equation:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(H^T W H\right)^{-1} H^T W \left(y\left(\mathbf{x}^k\right) - \mathbf{z}\right) \tag{7}$$

where H is the Jacobian matrix of $y(\mathbf{x})$.

Once the solution is obtained, chi-square test with computed confidence level is applied, which provides a mathematical method of evaluating the consistency between the measurements and the system model:

$$\begin{gathered} \xi = \sum_i \left(\frac{y_i(\mathbf{x}) - z_i}{\sigma_i} \right)^2 \\ P = 1 - \Pr(\xi, n) \end{gathered} \tag{8}$$

where σ_i is the standard deviation of each measurement, n is the degree of freedom (difference between number of measurements and states), ξ is the chi-square value, and P is the confidence level. A low confidence level indicates bad data or hidden failures, while a high value (e.g., 100%) implies that the network AQCF is consistent with the measurements and the estimated states are trustworthy.

4. Autonomous Multi-Stage Flexible OPF Formulation

A multi-stage OPF problem is autonomously formulated by stacking the network AQCF in (4) for several stages, as an extension of our previous work [19]. The state estimator also provides the best estimate of the states and controls. The main tasks are (a) objective function generation and (b) stacking network AQCF for multiple stages of the horizon.

4.1. Autonomous Objective Function Creation

The OPF objective function is user-selected from a list of objective functions. In this paper, we focus on an objective function that levelizes the voltage profile across the network. We define the objective function to be minimizing the sum of squared voltage magnitude mismatches (differences between voltage magnitude and desired voltage value) at selected buses. The mathematical expression of this objective function is:

$$\min \quad J = \sum_{i \in S_{bus}} \left(\frac{V_{i,mag} - V_{i,des}}{\alpha_i V_{i,des}} \right)^2 \tag{9}$$

where S_{bus} is the set of selected buses and S_{bc} is the set of binary control variables. $V_{i,mag}$ and $V_{i,des}$ are the voltage magnitude and desired voltage value at bus i, respectively, while α_i is a pre-defined tolerance value (e.g., 5%) at that bus.

The objective function in (9) is converted to a standard quandratized form:

$$J = Y_{ox}^T \mathbf{x} + Y_{ou}^T \mathbf{u} + \mathbf{x}^T F_{ox} \mathbf{x} + \mathbf{u}^T F_{ou} \mathbf{u} + \mathbf{u}^T F_{oux} \mathbf{x} + C_o \tag{10}$$

where subscript o is used to denote the coefficients corresponding to the objective function, whose terms are also at most second order.

Note also that any discrete control variables in the system are converted into continuous variables at the device level. We presented earlier the example of a switchable capacitor as an example with discrete control (ON/OFF). When the model is quadratized, the discrete control variable u is constrained by additional equations in terms of additional variables that make the system continuous, but the solution of these equations will yield the correct value for the variable u. Specifically, the additional variable p is introduced so that the equation $0 = p - u(1-u)$ is appended to the device model. The objective function in (9) becomes

$$\min \quad J = \sum_{i \in S_{bus}} \left(\frac{V_{i,mag} - V_{i,des}}{\alpha_i V_{i,des}} \right)^2 + \sum_{i \in S_{bc}} M_i p_i \tag{11}$$

where M_i corresponds to the weight of state p_i related to binary control u_i. The weights are large numbers to make sure that variables p are driven to 0, thereby making binary controls u equal to 0 or 1.

4.2. Quadratized OPF Model

Addition of an objective function to the network AQCF yields the quadratized OPF model (QOPFM):

$$\begin{array}{ll} \min & J(\mathbf{x},\mathbf{u}) = Y_{ox}^T \mathbf{x} + Y_{ou}^T \mathbf{u} + \mathbf{x}^T F_{ox} \mathbf{x} + \mathbf{u}^T F_{ou} \mathbf{u} + \mathbf{u}^T F_{oux} \mathbf{x} + C_o \\ \text{s.t.} & \mathbf{g}(\mathbf{x},\mathbf{u}) = 0 \\ & \mathbf{h}(\mathbf{x},\mathbf{u}) \le 0 \\ & \mathbf{u_{min}} \le \mathbf{u} \le \mathbf{u_{max}} \end{array} \tag{12}$$

The system operating point is referred to as $(\mathbf{x},\mathbf{u})$, consisting of all state and control variables introduced by the individual devices in the system. Equality vector $\mathbf{g}$ in (12) is derived from the equations in (4) as

$$\begin{gathered} \mathbf{g}(\mathbf{x},\mathbf{u}) = Y_{nx}\mathbf{x} + Y_{nu}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T \left\langle F_{nx}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{nu}^i \right\rangle \mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T \left\langle F_{nux}^i \right\rangle \mathbf{x} \\ \vdots \end{Bmatrix} \\ + N_{nx}\mathbf{x}(t-h) + N_{nu}\mathbf{u}(t-h) + M\mathbf{i}(t-h) + K - \mathbf{i}(t,t_m) \end{gathered} \tag{13}$$

where $\mathbf{i}(t,t_m)$ is $\left[\mathbf{i}(t) \quad 0 \quad 0 \quad \mathbf{i}(t_m) \quad 0 \quad 0\right]^T$. Note that (13) includes both KCL equations at common nodes and device internal equations. Inequality vector $\mathbf{h}$ as well as control bounds $\mathbf{u_{min}}$ and $\mathbf{u_{max}}$ in (12) are directly adopted from the network AQCF, containing device functional constraints and control bounds.

The QOPFM described is a single-stage quadratic OPF problem, including time t and t_m. A multi-stage problem is formed by combining multiple single-stage QOPFM into one. Three stages are considered in this paper, containing time periods t_2, t_1 and t_0, each having its own time t_m component. The time interval between two consecutive time periods is h. The multi-stage QOPFM can also be represented by the structure in (12), with its matrices and vectors formed as

$$Y_{ox} = \begin{bmatrix} Y_{ox}^s \\ Y_{ox}^s \\ Y_{ox}^s \end{bmatrix} \quad Y_{ou} = \begin{bmatrix} Y_{ou}^s \\ Y_{ou}^s \\ Y_{ou}^s \end{bmatrix} \quad C_o = \begin{bmatrix} C_o^s \\ C_o^s \\ C_o^s \end{bmatrix}$$

$$F_{ox} = \begin{bmatrix} F_{ox}^s & 0 & 0 \\ 0 & F_{ox}^s & 0 \\ 0 & 0 & F_{ox}^s \end{bmatrix} \quad F_{ou} = \begin{bmatrix} F_{ou}^s & 0 & 0 \\ 0 & F_{ou}^s & 0 \\ 0 & 0 & F_{ou}^s \end{bmatrix}$$

$$F_{oux} = \begin{bmatrix} F_{oux}^s & 0 & 0 \\ 0 & F_{oux}^s & 0 \\ 0 & 0 & F_{oux}^s \end{bmatrix} \quad Y_x = \begin{bmatrix} Y_x^s & 0 & 0 \\ 0 & Y_x^s & 0 \\ 0 & 0 & Y_x^s \end{bmatrix}$$

$$Y_u = \begin{bmatrix} Y_u^s & 0 & 0 \\ 0 & Y_u^s & 0 \\ 0 & 0 & Y_u^s \end{bmatrix} \quad K = \begin{bmatrix} K^s \\ K^s \\ -B_0 \end{bmatrix} \quad N_x = \begin{bmatrix} 0 & N_x^s & 0 \\ 0 & 0 & N_x^s \\ 0 & 0 & 0 \end{bmatrix}$$

$$N_u = \begin{bmatrix} 0 & N_u^s & 0 \\ 0 & 0 & N_u^s \\ 0 & 0 & 0 \end{bmatrix} \quad M = \begin{bmatrix} 0 & M^s & 0 \\ 0 & 0 & M^s \\ 0 & 0 & 0 \end{bmatrix}$$

$$Y_{fx} = \begin{bmatrix} Y_{fx}^s & 0 & 0 \\ 0 & Y_{fx}^s & 0 \\ 0 & 0 & Y_{fx}^s \end{bmatrix} \quad Y_{fu} = \begin{bmatrix} Y_{fu}^s & 0 & 0 \\ 0 & Y_{ou}^s & 0 \\ 0 & 0 & Y_{ou}^s \end{bmatrix}$$

$$F_{fx} = \begin{bmatrix} F_{fx}^s & 0 & 0 \\ 0 & F_{fx}^s & 0 \\ 0 & 0 & F_{fx}^s \end{bmatrix}, \quad F_{fu} = \begin{bmatrix} F_{fu}^s & 0 & 0 \\ 0 & F_{fu}^s & 0 \\ 0 & 0 & F_{fu}^s \end{bmatrix}$$

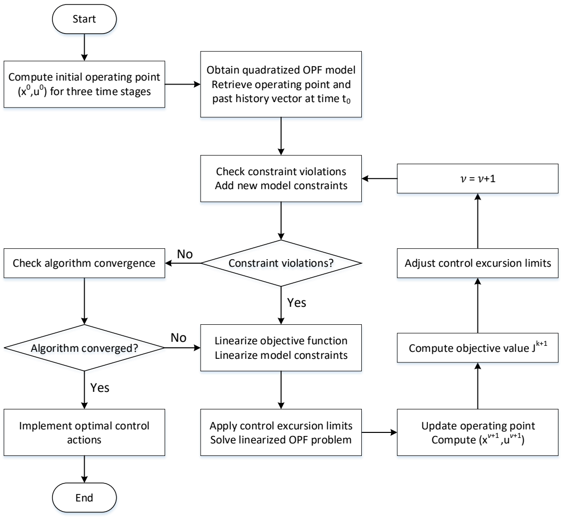
$$F_{fux} = \begin{bmatrix} F_{fux}^s & 0 & 0 \\ 0 & F_{fux}^s & 0 \\ 0 & 0 & F_{fux}^s \end{bmatrix}, \quad C_f = \begin{bmatrix} C_f^s \\ C_f^s \\ C_f^s \end{bmatrix}$$

$$\mathbf{u_{min}} = \begin{bmatrix} \mathbf{u}_{\mathbf{min}}^s \\ \mathbf{u}_{\mathbf{min}}^s \\ \mathbf{u}_{\mathbf{min}}^s \end{bmatrix}, \quad \mathbf{u_{max}} = \begin{bmatrix} \mathbf{u}_{\mathbf{max}}^s \\ \mathbf{u}_{\mathbf{max}}^s \\ \mathbf{u}_{\mathbf{max}}^s \end{bmatrix}, \quad \mathbf{u_{lim}} = \begin{bmatrix} \mathbf{u}_{\mathbf{lim}}^s \\ \mathbf{u}_{\mathbf{lim}}^s \\ \mathbf{u}_{\mathbf{lim}}^s \end{bmatrix}$$

where subscript s stands for single stage. B_0 is the past history vector for the first stage t_0 provided by the state estimator together with the initial operating condition at time t_0. The state, control and through vectors also include variables from the three stages. Thus, operating point $(\mathbf{x},\mathbf{u})$ contains information at time t_2, t_1 and t_0.

5. Solution Method by Sequential Linearization / Linear Programming

To solve the multi-stage F-OPF problem, we use a sequential linear programming (SLP) technique. The algorithm is illustrated in Figure 4. Given the multi-stage QOPFM, with the operating point and past history vector B_0 at time t_0 provided by the dynamic state estimator, the OPF problem is solved through a successive process of constraint violation check, linearization, linear program (LP) solving and operating point update.



Start
Compute initial operating point (x^0,u^0) for three time stages
Obtain quadratized OPF model
Retrieve operating point and past history vector at time t_0
Check constraint violations
Add new model constraints
$\nu = \nu+1$
No
Check algorithm convergence
Constraint violations?
Adjust control excursion limits
Yes
Algorithm converged?
No
Linearize objective function
Linearize model constraints
Compute objective value J^{k+1}
Yes
Implement optimal control actions
Apply control excursion limits
Solve linearized OPF problem
Update operating point
Compute $(x^{\nu+1},u^{\nu+1})$
End

Figure 4: Sequential linear programming algorithm

5.1. OPF Model Linearization

In the SLP algorithm, we define model constraints as the inequalities to be linearized and considered in the following LP. During each iteration ν, the operating point $(\mathbf{x}^\nu,\mathbf{u}^\nu)$ is substituted into the inequality constraints $\mathbf{h}(\mathbf{x},\mathbf{u})$. Constraint i is violated if $h_i(\mathbf{x}^\nu,\mathbf{u}^\nu) > \varepsilon_1$, where ε_1 is a preset small positive number. The newly violated constraints are added to the set of model constraints. Then, the multi-stage QOPFM is linearized with respect to the control variables, giving the linearized OPF model (LOPFM). These two procedures ensure that the size of the problem to be solved is the smallest possible in every iteration.

Linearization of the QOPFM in (12) around operating point $\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)$ yields

$$\begin{aligned} J\left(\mathbf{x},\mathbf{u}\right) &\approx J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)+\frac{dJ\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{d\mathbf{u}}\mathbf{d} \\ \mathbf{h_m}\left(\mathbf{x},\mathbf{u}\right) &\approx \mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)+\frac{d\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{d\mathbf{u}}\mathbf{d} \end{aligned} \tag{14}$$

where $\mathbf{d}=\mathbf{u}-\mathbf{u}^{\nu}$ and $\mathbf{h_m}\left(\mathbf{x},\mathbf{u}\right)$ is the vector of model constraints. The derivatives in (14) are computed by

$$\begin{aligned} \frac{dJ\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{d\mathbf{u}} &= \frac{\partial J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}+\frac{\partial J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\frac{d\mathbf{x}}{d\mathbf{u}} \\ \frac{d\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{d\mathbf{u}} &= \frac{\partial\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}+\frac{\partial\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\frac{d\mathbf{x}}{d\mathbf{u}} \end{aligned} \tag{15}$$

From equality constraints $\mathbf{g}\left(\mathbf{x},\mathbf{u}\right)=0$, we have

$$\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}+\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\frac{d\mathbf{x}}{d\mathbf{u}}=0 \tag{16}$$

where the derivative of $\mathbf{x}$ with respective to $\mathbf{u}$ is calculated as

$$\frac{d\mathbf{x}}{d\mathbf{u}}=-\left(\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\right)^{-1}\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}} \tag{17}$$

Since all the coefficient matrices have been defined and formed, the following partial derivatives can be computed directly.

$$\frac{\partial J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}=Y_{ou}^{T}+\left\{\mathbf{u}^{\nu T}F_{ou}^{i}\right\}+\left\{F_{ou}^{i}\mathbf{u}^{\nu}\right\}^{T}+\left\{F_{oux}^{i}\mathbf{x}^{\nu}\right\}^{T} \tag{18a}$$

$$\frac{\partial J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}=Y_{ox}^{T}+\left\{\mathbf{x}^{\nu T}F_{ox}^{i}\right\}+\left\{F_{ox}^{i}\mathbf{x}^{\nu}\right\}^{T}+\left\{\mathbf{u}^{\nu T}F_{oux}^{i}\right\} \tag{18b}$$

$$\frac{\partial\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}=Y_{fu}+\left\{\mathbf{u}^{\nu T}F_{fu}^{i}\right\}+\left\{F_{fu}^{i}\mathbf{u}^{\nu}\right\}^{T}+\left\{F_{fux}^{i}\mathbf{x}^{\nu}\right\}^{T} \tag{18c}$$

$$\frac{\partial\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}=Y_{fx}+\left\{\mathbf{x}^{\nu T}F_{fx}^{i}\right\}+\left\{F_{fx}^{i}\mathbf{x}^{\nu}\right\}^{T}+\left\{\mathbf{u}^{\nu T}F_{fux}^{i}\right\} \tag{18d}$$

$$\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}=Y_{u}+\left\{\mathbf{u}^{\nu T}F_{u}^{i}\right\}+\left\{F_{u}^{i}\mathbf{u}^{\nu}\right\}^{T}+\left\{F_{ux}^{i}\mathbf{x}^{\nu}\right\}^{T} \tag{18e}$$

$$\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}=Y_{x}+\left\{\mathbf{x}^{\nu T}F_{x}^{i}\right\}+\left\{F_{x}^{i}\mathbf{x}^{\nu}\right\}^{T}+\left\{\mathbf{u}^{\nu T}F_{ux}^{i}\right\} \tag{18f}$$

To reduce linearization errors, maximum permissible control excursions $\mathbf{u_{lim}}$ are utilized and $\mathbf{d_{min}}\le\mathbf{d}\le\mathbf{d_{max}}$ is imposed as bounds on the control variations between two consecutive iterations. Every entry i in $\mathbf{d_{min}}$ and $\mathbf{d_{max}}$ are given by

$$\begin{aligned} d_{\mathrm{min},i} &= \max\left(u_{\mathrm{min},i}-u_{i}^{\nu},-u_{\mathrm{lim},i}\right) \\ d_{\mathrm{max},i} &= \min\left(u_{\mathrm{max},i}-u_{i}^{\nu},u_{\mathrm{lim},i}\right) \end{aligned} \tag{19}$$

After completion of the linearization process, the LOPFM can be written in the general form:

$$\begin{aligned} \text{min} \quad & J\left(\mathbf{d}\right)=c^{T}\mathbf{d}+e \\ \text{s.t.} \quad & a\mathbf{d}+b\le 0 \\ & \mathbf{d_{min}}\le\mathbf{d}\le\mathbf{d_{max}} \end{aligned} \tag{20}$$

where from (14) – (17) the following holds.

$$\begin{aligned} c^{T} &= \frac{\partial J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}-\frac{\partial J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\left(\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\right)^{-1}\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}} \\ e &= J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right) \\ a &= \frac{\partial\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}}-\frac{\partial\mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\left(\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{x}}\right)^{-1}\frac{\partial\mathbf{g}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)}{\partial\mathbf{u}} \\ b &= \mathbf{h_m}\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right) \end{aligned} \tag{21}$$

5.2. Operating Point Update

Once the LP in (20) is solved in iteration ν and the solution is $\mathbf{d}^{\nu}$, control variables $\mathbf{u}^{\nu+1}=\mathbf{u}^{\nu}+\mathbf{d}^{\nu}$ are updated. As the state and control variables obey the equalities $\mathbf{g}\left(\mathbf{x},\mathbf{u}\right)=0$, these equations are used to solve for the updated states $\mathbf{x}^{\nu+1}$ through the Newton-Raphson method. The steps are listed as follows:

1) Let $\kappa=0$ and $\mathbf{x}^{\kappa}=\mathbf{x}^{\nu}$, where κ is the iteration number in the Newton-Raphson method used to obtain the state variables $\mathbf{x}^{\nu+1}$ in iteration ν.
2) Substitute $\mathbf{x}^{\kappa}$ and $\mathbf{u}^{\nu+1}$ into the QOPFM equations and compute $\mathbf{g}\left(\mathbf{x}^{\kappa},\mathbf{u}^{\nu+1}\right)$. If $\left\|\mathbf{g}\left(\mathbf{x}^{\kappa},\mathbf{u}^{\nu+1}\right)\right\|_{\infty}\le\varepsilon_{2}$, where ε_{2} is a pre-defined tolerance value, the procedure terminates and $\mathbf{x}^{\kappa}$ is the solution $\mathbf{x}^{\nu+1}$; otherwise, go the the next step.
3) Compute the Jacobian matrix $\partial\mathbf{g}\left(\mathbf{x}^{\kappa},\mathbf{u}^{\nu+1}\right)/\partial\mathbf{x}$, which can be easily achieved using equation (18f).
4) Calculate $\mathbf{x}^{\kappa+1}$ as
$$\mathbf{x}^{\kappa+1}=\mathbf{x}^{\kappa}-\left(\frac{\partial\mathbf{g}\left(\mathbf{x}^{\kappa},\mathbf{u}^{\nu+1}\right)}{\partial\mathbf{x}}\right)^{-1}\mathbf{g}\left(\mathbf{x}^{\kappa},\mathbf{u}^{\nu+1}\right) \tag{22}$$
5) Make $\kappa=\kappa+1$. If $\kappa\le\kappa_{\max}$, where $\kappa_{\max}$ is the maximum number of iterations allowed, go to step 2); otherwise, non-convergence is reported.

With the new operating point $\left(\mathbf{x}^{\nu+1},\mathbf{u}^{\nu+1}\right)$ found, the objective value $J\left(\mathbf{x}^{\nu+1},\mathbf{u}^{\nu+1}\right)$ is calculated and compared with the objective value from the previous iteration. If $J\left(\mathbf{x}^{\nu+1},\mathbf{u}^{\nu+1}\right)\ge J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)$, the control excursion limit for each non-binary control i is halved. The excursion limits for binary variables remain at 1. Then, the algorithm continues to iteration $\nu+1$ and the updated operating point is substituted back into the set of inequality constraints $\mathbf{h}\left(\mathbf{x},\mathbf{u}\right)$ to check if any of them

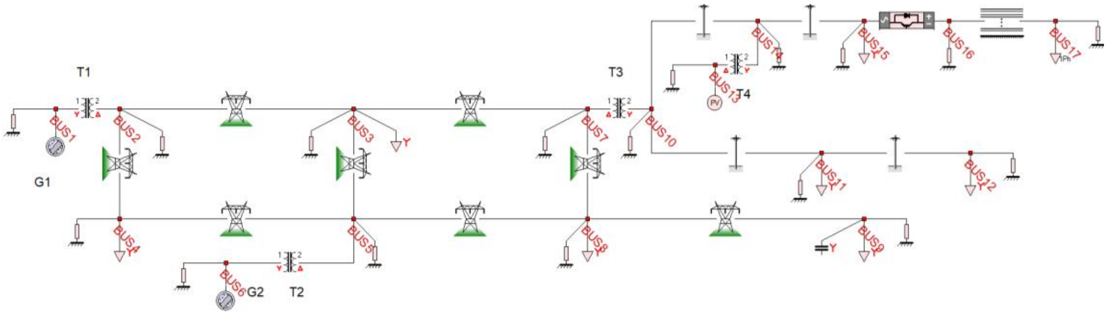


Figure 5: Example test system

are violated. Since the LOPFM only includes the previous model constraints, the updated operating point may not satisfy all inequality constraints in the QOPFM. The newly violated constraints become the new model constraints. The QOPFM is re-linearized with respect to the new operating point and same procedures follow.

Note that when the algorithm starts with the state and control variables, as well as the past history vector, at time t_0, the variables at time t_1 and t_2 are computed via the Newton-Raphson method, thus obtaining the initial operating point $\left(\mathbf{x}^0,\mathbf{u}^0\right)$.

5.3. Algorithm Convergence

If the updated operating point $\left(\mathbf{x}^{\nu+1},\mathbf{u}^{\nu+1}\right)$ satisfies all the inequality constraints during iteration $\nu+1$, the algorithm convergence is determined. There are two criteria to be checked:

$$\left|J\left(\mathbf{x}^{\nu+1},\mathbf{u}^{\nu+1}\right)-J\left(\mathbf{x}^{\nu},\mathbf{u}^{\nu}\right)\right|\le\varepsilon_3 \quad \text{and} \quad \left|\mathbf{du}_{\mathbf{n}}^{\nu}\right|\le\varepsilon_4 \qquad (23)$$

where ε_3 and ε_4 are small positive values. Since different controls have different units and are under different scales, the control excursions need to be normalized before they can be used as a convergence criterion. Hence, every element i in the vector of normalized control steps $\mathbf{du}_{\mathbf{n}}^{\nu}$ is computed as $du_{n,i}^{\nu}=d_i^{\nu}\big/u_{\lim,i}^{0}$, where $u_{\lim,i}^{0}$ is the original excursion limit before any adjustment is applied for control i.

The SLP algorithm converges when both criteria in (23) are satisfied. The optimal operating point is $\left(\mathbf{x}^{\nu+1},\mathbf{u}^{\nu+1}\right)$ with $\mathbf{u}^{\nu+1}$ being the optimal control actions to be implemented. If either criterion is not satisfied, a new iteration of QOPFM linearization and LOPFM solving is required considering the new operating point.

In general, convergence cannot be guaranteed for this optimization problem (multi-stage OPF) in the strict mathematical sense. The problem is a non-convex optimization problem and one of the most complex. From the practical point of view the solution does converge to a feasible and stable solution (typical a local optimum) as long as the system has enough controls.

6. Example Test Results

The proposed multi-stage F-OPF approach is tested on an example system with DER penetration at the distribution level. This section first describes the system and presents the test results afterwards. The model, as well as the initial operating point and past history vector at time t_0 are provided by the state estimator, which uses “measurements” created by a simulator.

6.1. System Description

The example test system is shown in Figure 5, including both transmission and distribution networks. The 115 kV transmission system has two generators, one (G1) at slack bus BUS1 and the other (G2) at BUS6. Transformers T1 and T2 are used to step up the voltages from 18 kV and 15 kV to 115 kV at BUS2 and BUS5, respectively. The 115 kV transmission lines connecting BUS2 to BUS3, BUS2 to BUS4, BUS3 to BUS5, BUS3 to BUS7, BUS4 to BUS5, BUS5 to BUS8, BUS7 to BUS8, and BUS8 to BUS9 are respectively 35 miles, 25 miles, 22 miles, 32 miles, 18 miles, 51 miles, 35.2 miles, and 19.2 miles long. A three-phase capacitor bank is located at BUS9. Note that the capacitor bank model is just like the capacitor model demonstrated in section 3.3, except that it has three phases instead of one. However, it is uncontrollable and set to be ON the whole time in this test system.

Transformer T3 steps down the transmission voltage at BUS7 to 13.2 kV distribution level at BUS10. A PV source with battery has a voltage setpoint of 480 V at BUS13, which is stepped up to 13.2 kV at BUS14 by transformer T4. An AC/DC converter converts 13.2 kV AC to 25 kV DC from BUS15 to BUS16, supplying a DC load at BUS17. The AC distribution lines connecting BUS10 to BUS11, BUS10 to BUS14, BUS11 to BUS12, BUS14 to BUS15 have lengths of 1 mile, 1 mile, 2 miles and 2 miles, respectively, while the DC line between BUS16 and BUS17 is 2 miles long.

Table 1: Loads at the various buses of the test system

Bus	*BUS3*	*BUS4*	*BUS8*	*BUS9*
Load	38 MW 10 MVar	100 MW 30 MVar	55 MW 15 MVar	31 MW 8 MVar
Bus	*BUS11*	*BUS12*	*BUS15*	*BUS17*
Load	2 MW 1 MVar	3.5 MW 1.2 MVar	3 MW 0.8 MVar	1 MW

Every bus in the test system has a ground impedance of 1 Ω. The rated power consumptions of the loads are provided in Table 1, while the controls with their excursion limits and initial values are listed in Table 2. The time interval h used in this example is 5 minutes.

Table 2: List of controls available in the test system

Device	**Control Variable**	**Excursion Limit**	**Initial Value**
Generator G1	Voltage Setpoint	0.01 pu	1.0 pu
Generator G2	Real Power Output	12.5 MW	180 MW
	Voltage Setpoint	0.01 pu	1.0 pu
Transformer T1	Tap Setting	0.02	1.0
Transformer T2	Tap Setting	0.02	1.0
Transformer T3	Tap Setting	0.02	1.0
PV Source with Battery	Voltage Setpoint	4.8 V	480 V
	Battery Real Power Output	0.06 MW	0.1 MW
AC/DC Converter	DC Voltage Setpoint	0.25 kV	25 kV
	AC Side Reactive Power Output	0.09 MVar	0.5 MVar

The voltages selected to be levelized are the three-phase voltages at BUS3, BUS4, BUS8, BUS9, BUS12 and BUS15, as well as the DC voltage at BUS17. The desired voltage V_{des} at transmission and distribution AC buses are 66.395 kV and 7.621 kV, respectively. The desired voltage at the DC bus is 25 kV. Tolerance α has the same value of 5% over all monitored buses.

6.2. F-OPF Results

Given the initial control values, the system simulation is run, providing the states and past history vector at time t_0, which are used to compute the initial operating point. Note that in practice, the information at time t_0 is provided by a dynamic state estimator. Pre-defined values ε_1 and ε_2 are 0.0001, while ε_3 and ε_4 are 0.005. The algorithm takes 40 iterations to converge. The objective value is plotted in Figure 6, while the three-phase voltages at BUS9 and BUS12 with the DC voltage at BUS17 at time t_2 are given in Figure 7.

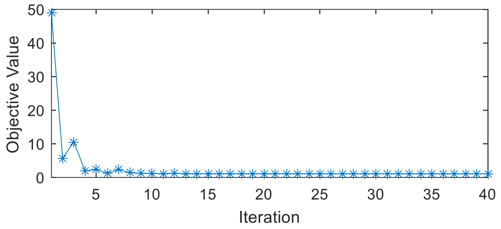


Figure 6: Objective function value vs iteration count

Although we show the evolution of the algorithm over 40 iterations, it can be noticed from Figure 6 and Figure 7 that the results do not change much after 10 iterations. Therefore, we can slightly increase tolerances ε_3 and ε_4 to speed up the process. The final objective value is about 1.1 and the voltages are close to their desired values. Other selected voltages are not shown here due to limited space, but they are also driven towards their desired values over the iterations.

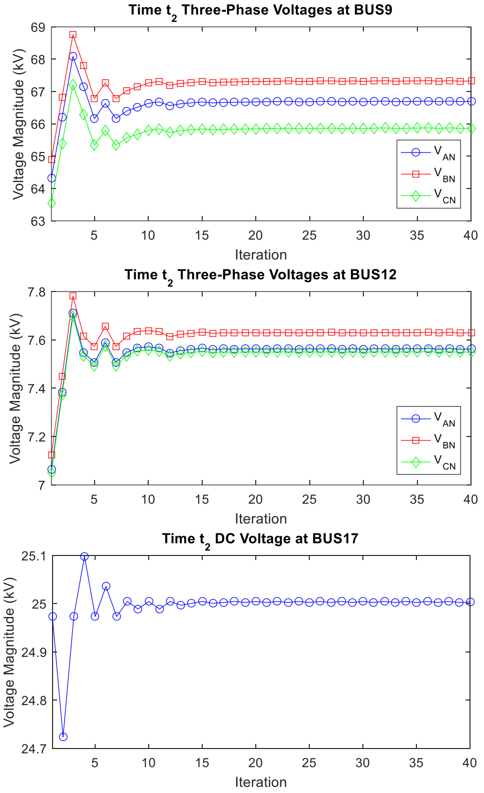


Figure 7: Voltages at BUS9, BUS12 and BUS17 vs iteration count

The optimal controls obtained upon convergence are the same at all three time stages. The voltage setpoints of G1 and G2 are 1.0106 pu and 1.0139 pu, respectively. The G2 real power output is 170.7715 MW. The tap settings of T1, T2 and T3 are respectively 1.0212, 1.0277 and 1.0412. For the PV source with battery, its voltage setpoint is 483.8438 V, while its battery real power output is 0.1443 MW. The controls of the AC/DC converter are 25.0303 kV for the DC voltage output and 0.5552 MVar for the AC side reactive power output.

7. Conclusions

This paper proposes a real-time autonomous approach of formulating and solving a multi-stage flexible OPF problem with the system operating data provided by a dynamic state estimator that estimates the system states with measurements collected by relays and sensors. Distribution systems penetrated with DERs are optimized against a list of user-selected objective functions. Optimizing the voltage profiles at selected buses over three consecutive time stages is illustrated.

The method is based on physically based modeling, which uses the AQCF standard syntax to represent the device and network models. The QOPFM is directly formed and linearized with respect to the system controls using the network model. The problem is solved using an SLP algorithm. Results from an example test system are provided. The test results demonstrate the autonomous formation and solution of the multi-stage OPF problem.

8. References

[1] S. Lu, N. Samaan, R. Diao, M. Elizondo, C. Jin, E. Mayhorn, Y. Zhang and H. Kirkham, “Centralized and decentralized control for demand response,” *2011 Innovative Smart Grid Technologies (ISGT)*, Hilton Anaheim, CA, 2011.

[2] M. Negnevitsky, T. D. Nguyen and M. de Groot, “An agent-based market-clearing scheme for the exchange of demand response,” *2011 IEEE Power and Energy Society General Meeting*, San Diego, CA, 2011.

[3] J. L. Mathieu, S. Koch and D. S. Callaway, “State estimation and control of electric loads to manage real-time energy imbalance,” *IEEE Transactions on Power Systems*, vol. 28, no. 1, pp. 430-440, Feb. 2013.

[4] Z. Ma, D. S. Callaway and I. A. Hiskens, “Decentralized charging control of large populations of plug-in electric vehicles,” *IEEE Transactions on Control Systems Technology*, vol. 21, no. 1, pp. 67-78, Jan. 2013.

[5] J. Chen and Q. Zhu, “A game-theoretic framework for resilient and distributed generation control of renewable energies in microgrids,” *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 285-295, Jan. 2017.

[6] Z. Zhang, L. F. Ochoa and G. Valverde, “A novel voltage sensitivity approach for the decentralized control of DG plants,” *IEEE Transactions on Power Systems*, vol. 33, no. 2, pp. 1566-1576, March 2018.

[7] A. A. Eajal, M. F. Shaaban, K. Ponnambalam and E. F. El-Saadany, “Stochastic centralized dispatch scheme for AC/DC hybrid smart distribution systems,” *IEEE Transactions on Sustainable Energy*, vol. 7, no. 3, pp. 1046-1059, July 2016.

[8] Y. Li, L. He, F. Liu, C. Li, Y. Cao and M. Shahidehpour, “Flexible voltage control strategy considering distributed energy storages for DC distribution network,” *IEEE Transactions on Smart Grid*. Online early access.

[9] Z. Shen and M. E. Baran, “Gradient based centralized optimal Volt/Var control strategy for smart distribution system,” *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT)*, Washington, DC, 2013.

[10] S. Paudyal, C. A. Cañizares and K. Bhattacharya, “Three-phase distribution OPF in smart grids: Optimality versus computational burden,” *2011 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies*, Manchester, 2011.

[11] H. Sekhavatmanesh and R. Cherkaoui, “Distribution network restoration in a multi-agent framework using a convex OPF model,” *IEEE Transactions on Smart Grid*. Online early access.

[12] J. Franco, L. Ochoa and R. Romero, “AC OPF for smart distribution networks: An efficient and robust quadratic approach,” *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 4613-4623, Sept. 2018.

[13] R. Yang and Y. Zhang, “Three-phase AC optimal power flow based distribution locational marginal price,” *2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Washington, DC, 2017.

[14] H. Yuan, F. Li, Y. Wei and J. Zhu, “Novel linearized power flow and linearized OPF models for active distribution networks with application in distribution LMP," *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 438-448, Jan. 2018.

[15] A. P. S. Meliopoulos, E. Polymeneas, Z. Tan, R. Huang and D. Zhao, “Advanced distribution management system,” *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 2109-2117, Dec. 2013.

[16] C. Zhong, A. P. S. Meliopoulos, J. Sun, M. Saeedifard and B. Xie, “Modeling of converter losses with high fidelity in a physically based object-oriented way,” *2018 IEEE Power and Energy Society General Meeting (PESGM)*, Portland, OR, 2018.

[17] B. Xie, A. P. S. Meliopoulos, Y. Liu and L. Sun, “Distributed quasi-dynamic state estimation with both GPS-synchronized and non-synchronized data,” *2017 North American Power Symposium (NAPS)*, Morgantown, WV, 2017.

[18] A. P. Meliopoulos, G. J. Cokkinides and G. K. Stefopoulos, “Quadratic integration method,” *Proc. of the 2005 International Conference on Power System Transients (IPST)*, Montreal, Canada, 2005.

[19] C. Zhong, A. P. Sakis Meliopoulos, G. J. Cokkinides and B. Xie, “Object-Oriented Voltage Control for AC-DC Hybrid Distribution Systems,” *2018 9th IEEE International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*, Charlotte, NC, USA, 2018.