Sample-Based Piecewise Linear Power Flow Approximations Using Second-Order Sensitivities

Paprapee Buason, Sidhant Misra
Los Alamos National Laboratory
Los Alamos, NM, USA
{buason, sidhant}@lanl.gov

Daniel K. Molzahn
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA, USA
molzahn@gatech.edu

***Abstract*—The inherent nonlinearity of the power flow equations poses significant challenges in accurately modeling power systems, particularly when employing linearized approximations. Although power flow linearizations provide computational efficiency, they can fail to fully capture nonlinear behavior across diverse operating conditions. To improve approximation accuracy, we propose conservative piecewise linear approximations (CPLA) of the power flow equations, which are designed to consistently over- or under-estimate the quantity of interest, ensuring conservative behavior in optimization. The flexibility provided by piecewise linear functions can yield improved accuracy relative to standard linear approximations. However, applying CPLA across all dimensions of the power flow equations could introduce significant computational complexity, especially for large-scale optimization problems. In this paper, we propose a strategy that selectively targets dimensions exhibiting significant nonlinearities. Using a second-order sensitivity analysis, we identify the directions where the power flow equations exhibit the most significant curvature and tailor the CPLAs to improve accuracy in these specific directions. This approach reduces the computational burden while maintaining high accuracy, making it particularly well-suited for mixed-integer programming problems involving the power flow equations.**

***Index Terms*—Conservative piecewise linear approximation; Second-order sensitivities; Power flow approximation**

I. Introduction

Describing the steady-state relationships between bus voltages and power injections, the power flow equations are fundamental to the operation, analysis, and planning of power systems. The inherent nonlinearity of these equations introduces non-convexities that lead to challenging optimization problems. This complexity is even more pronounced when dealing with mixed-integer problems or uncertainty quantification for systems with stochastic behavior.

To mitigate these challenges, various linearization techniques are used to simplify power flow representations. Methods such as DC power flow for transmission systems [1], LinDistFlow for distribution networks [2], and first-order Taylor expansions of the AC power flow equations are commonly employed. These approximations offer computational tractability but are often based on limiting assumptions such as near-nominal voltage magnitudes and small voltage angle differences. Although such assumptions may hold in certain operating conditions, they can lead to inaccuracies when applied to more complex or broader system conditions.

Recent research has increasingly focused on *adaptive* linear power flow approximations, which are tailored to provide greater accuracy for a specific system and operating range of interest. Approaches in this area include optimization-based methods [3] and data-driven techniques [4]–[8]; see the recent surveys in [9], [10]. These methods aim to strike a balance between computational efficiency and accuracy by customizing linearization coefficients to better capture power flow nonlinearities. Unlike traditional approaches such as DC power flow, LinDistFlow, or first-order Taylor expansions—which prioritize simplicity and computational speed but often sacrifice accuracy—adaptive methods allocate additional computational resources upfront to calculate coefficients that align with specific operating conditions. This upfront effort proves especially advantageous in applications with an offline/online split, where coefficients can be precomputed offline using forecasted data and later applied in real-time computations. By improving the precision of linearized models, adaptive approximations enable efficient solutions to complex optimization problems, such as infrastructure planning [11], [12], AC unit commitment [13], and sensor placement optimization [14], where traditional linearization methods may lead to significant inaccuracies.

Building on this foundation, sample-based conservative linear approximations (CLAs) were introduced to further enhance the usefulness of adaptive methods by incorporating conservativeness into the construction process [4]. These approximations are designed to systematically over- or under-estimate specific quantities, such as voltage magnitudes, to ensure robust constraint enforcement in optimization problems. By embedding conservativeness during their formulation, CLAs improve the reliability of inequality constraints, such as those imposed by voltage or power flow limits, making them especially effective in applications where strict adherence to constraints is critical; see [12], [14], [15] for relevant applications.

In this paper, we introduce the *conservative piecewise linear approximation* (CPLA) as an extension of the CLA approach to better capture the inherent nonlinearity of the power flow equations. While CLAs provide linear approximations that are conservative by design, they are limited in their ability to adapt to highly nonlinear regions. CPLA addresses this limitation by introducing piecewise linear approximations, which enable a more precise representation of nonlinear behavior.

To identify these nonlinear regions, we leverage a second-order sensitivity analysis to compute the directions in which the power flow equations exhibit the greatest curvature. The

This work is supported by the advanced grid modeling (AGM) program of the Department of Energy, Office of Electricity and NSF award #2145564.

results from [5] indicate that there are only a few dominant nonlinear directions. Thus, CPLA focuses on implementing piecewise linear approximations in a few highly nonlinear directions, while conservative linear approximations are applied in the remaining directions to manage computational complexity. This targeted approach balances improved accuracy in capturing nonlinearities with computational tractability.

In summary, the main contributions of this paper are:

(i) Introduction and formulation of a conservative piecewise linear approximation (CPLA) of the power flow equations based on a second-order sensitivity analysis.

(ii) Variable reduction using continuity of the piecewise linearization to improve tractability of the CPLA, reducing the number of decision variables from exponential to polynomial in the number of highly nonlinear directions.

(iii) Numerical studies on how the number of nonlinear directions and piecewise linearization breakpoints in the CPLAs impact approximation accuracy across test cases.

The remainder of this paper is organized as follows. Section II covers background material on the power flow equations as well as conservative linear approximations and second-order sensitivity analysis. Section III introduces the conservative piecewise linear approximation approach. Section IV provides the numerical results of our approach. Section V concludes the paper and discusses directions for future work.

II. Background

This section provides background on the AC power flow equations, recently developed conservative linear approximations of these equations, and the second-order sensitivity analysis used to capture the power flow nonlinearities.

A. The power flow equations

Let V and θ denote the vectors of voltage magnitudes and angles. Let P and Q denote the active and reactive power injections. At each bus i, the power flow equations are:

$$P_i = V_i^2 G_{ii} + \sum_{k\in\mathcal{B}_i} V_i V_k (G_{ik}\cos\theta_{ik} + B_{ik}\sin\theta_{ik}), \tag{1a}$$

$$Q_i = -V_i^2 B_{ii} + \sum_{k\in\mathcal{B}_i} V_i V_k (G_{ik}\sin\theta_{ik} - B_{ik}\cos\theta_{ik}), \tag{1b}$$

where $\theta_{ik} := \theta_i - \theta_k$. The subscript $(\cdot)_i$ denotes a quantity at bus i, the subscript $(\cdot)_{ik}$ denotes a quantity from or connecting bus i to k, and $\mathcal{B}_i$ denotes the set of all neighboring buses to bus i, inclusive of bus i itself. Here, G and B represent the real and imaginary parts, respectively, of the admittance matrix.

B. Conservative linear approximations

The nonlinear nature of power flow equations in (1) presents significant challenges for optimization problems where these equations act as constraints. Conservative linear approximations (CLAs) [4] simplify this complexity by providing linear models that over- or underestimate quantities like voltage magnitudes and current flows (Fig. 1), ensuring constraint satisfaction in optimization. By adopting a sample-based approach, CLAs can be tailored to specific power systems and operating conditions, allowing for efficient approximations within defined ranges of power injections and loads. For example, CLAs may sample loads within a predetermined range $\mathcal{S} = \{P_d^{\min} \leq P_d \leq P_d^{\max},\ Q_d^{\min} \leq Q_d \leq Q_d^{\max}$ for all $d \in \mathcal{N_D}\}$ using the uniform distribution, where $(\,\cdot\,)_d$ denotes the load demand, $\mathcal{N_D}$ denotes the set of all load demands, and the superscripts max (min) indicate upper (lower) limits.

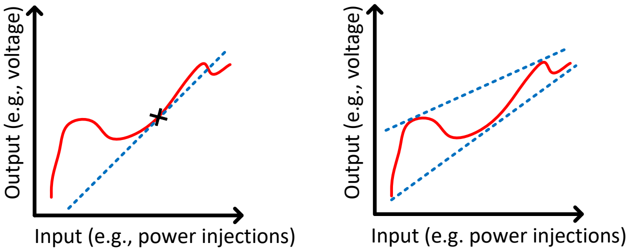


Fig. 1. The figure presents a visual comparison between a standard linear approximation (depicted on the left) and CLAs (depicted on the right). The solid line represents the nonlinear function being analyzed. In the left illustration, the dotted line portrays a conventional first-order Taylor approximation centered at point ×, whereas in the right illustration, the dotted line above (below) signifies an over- (under-)estimating approximation.

In addition to simplifying the representation of the power flow equations, CLAs can implicitly incorporate the characteristics of various devices, such as transformers and smart inverters [14]. This enhances CLAs accuracy in reflecting the behavior of complex systems.

Let bold quantities denote vectors and matrices. To facilitate the integration of CLAs into optimization frameworks, we denote the quantity of interest as γ. For instance, γ could be a particular voltage magnitude or line flow that one wishes to model as a linear function of the power injections. An overestimating CLA can be expressed as:

$$\gamma \leq a_0 + \boldsymbol{a}_1^T \begin{bmatrix} \boldsymbol{P} \\ \boldsymbol{Q} \end{bmatrix}, \tag{2}$$

where a_0 and $\boldsymbol{a}_1$ are coefficients of a CLA, and T is transpose. Assuming that (2) is indeed satisfied for all power injections $\mathbf{P}$ and $\mathbf{Q}$ in the operating range of interest, we can ensure that the constraint $\gamma \leq \gamma^{\max}$ is also satisfied via enforcing a linear constraint:

$$a_0 + \boldsymbol{a}_1^T \begin{bmatrix} \boldsymbol{P} \\ \boldsymbol{Q} \end{bmatrix} \leq \gamma^{\max}. \tag{3}$$

The CLA approach allows us to handle nonlinear AC power flow constraints while maintaining linear constraints within the optimization framework. To determine the coefficients a_0 and $\boldsymbol{a}_1$ in (2), we formulate the following regression problem that minimizes the error between the actual quantity of interest, denoted as γ, and its linear approximation:

$$\min_{a_0,\ \boldsymbol{a}_1} \quad \frac{1}{S}\sum_{s=1}^{S} \mathcal{L}\left(\gamma_s - \left(a_0 + \boldsymbol{a}_1^T \begin{bmatrix} \boldsymbol{P}_s \\ \boldsymbol{Q}_s \end{bmatrix}\right)\right) \tag{4a}$$

$$\text{s.t.} \quad \gamma_s - \left(a_0 + \boldsymbol{a}_1^T \begin{bmatrix} \boldsymbol{P}_s \\ \boldsymbol{Q}_s \end{bmatrix}\right) \leq 0, \quad \forall s = 1,\ldots,S. \tag{4b}$$

The subscript $(\,\cdot\,)_s$ represents the s^{th} sample and S represents the number of samples. The function $\mathcal{L}(\,\cdot\,)$ corresponds to a loss function, typically chosen to reflect the desired behavior of the approximation, such as the absolute value for ℓ_1 loss or the square for squared-ℓ_2 loss. The objective function in (4a) minimizes the mismatch between the approximated and actual quantities across all samples, while constraint (4b) ensures that the approximated quantity consistently overestimates the actual quantity for all samples. For underestimating CLAs, a similar process is followed, with the key distinction being the reversal of the inequality direction in (4b).

C. Second-order sensitivity of the power flow equations

Many traditional linearizations often neglect second-order and higher-order terms, assuming that the system behavior remains adequately approximated by linear models within the operating range of interest. While these linear approximations can provide reasonable estimates under certain conditions, their accuracy inherently depends on the curvature of the power flow equations within these operating ranges.

As power systems exhibit nonlinear behavior, particularly under highly varying operating conditions, overlooking second-order effects can lead to inaccuracies in the analysis results. To address this limitation, it becomes imperative to examine the second-order sensitivity of the power flow equations [5]. By considering the curvature and higher-order effects, we can gain deeper insights into the system behavior and refine their approximations accordingly.

Let g be a vector-valued function, denoted as $g(\boldsymbol{x})$, where $\boldsymbol{x}$ represents a vector input, yielding a vector output. To simplify the representation of variables in the power flow equations, let us define these as follows:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{P} \\ \boldsymbol{Q} \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{V} \end{bmatrix}. \tag{5}$$

We rewrite the power flow equations given in (1) as:

$$\boldsymbol{x} = g(\boldsymbol{y}). \tag{6}$$

Assume that a specific quantity of interest y_k can be written as $f(\boldsymbol{x})$. Let $\nabla_{\boldsymbol{x}} f(\boldsymbol{x_0})$ denote the gradient of f at $\boldsymbol{x} = \boldsymbol{x_0}$. The second-order Taylor approximation of y_k around $\boldsymbol{x_0}$ is:

$$y_k \approx f(\boldsymbol{x_0}) + \nabla f(\boldsymbol{x_0})^T(\boldsymbol{x}-\boldsymbol{x_0}) + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x_0})^T \boldsymbol{\Lambda}_{y_k}(\boldsymbol{x}-\boldsymbol{x_0}), \tag{7}$$

where $\boldsymbol{\Lambda}_{y_k}$ is the second-order sensitivity matrix for y_k around $\boldsymbol{x_0}$, which can be expressed as follows:

$$\boldsymbol{\Lambda}_{y_k} = \begin{bmatrix} \dfrac{\partial^2 y_k}{\partial x_1 \partial x_1} & \dfrac{\partial^2 y_k}{\partial x_1 \partial x_2} & \cdots & \dfrac{\partial^2 y_k}{\partial x_1 \partial x_{2N}} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2 y_k}{\partial x_{2N} \partial x_1} & \dfrac{\partial^2 y_k}{\partial x_{2N} \partial x_2} & \cdots & \dfrac{\partial^2 y_k}{\partial x_{2N} \partial x_{2N}} \end{bmatrix}. \tag{8}$$

The explicit form of the second-order sensitivity for voltages and its derivation can be found in [5]. We will use the second-order sensitivities to tailor the CLPA's piecewise linearizations based on the directions with the most significant nonlinearities.

III. Conservative Piecewise Linear Approximations

This section introduces conservative piecewise linear approximation (CPLAs) to improve the accuracy of power flow linearizations beyond what the CLA can offer. While CPLAs can provide more accurate representations by introducing multiple linear segments, applying CPLAs along all dimensions of the power flow equations can become computationally expensive, especially for large-scale optimization problems. In this section, we present the CPLA regression formulation and an efficient methodology that focuses on critical directions and variable reduction to lower the computational complexity.

A. Overall computational steps and parallel scalability

The CPLA computation consists of two main steps: identifying highly nonlinear directions via a second-order sensitivity analysis and solving the regression problem in (9) to construct piecewise linear approximations. These steps ensure that the approximations are tailored to accurately capture the nonlinear behavior of power flow equations. The flowchart in Fig. 2 illustrates the overall steps involved in computing the CPLA.

The approach is highly parallelizable, making it efficient for large-scale systems. The second-order sensitivity matrix computations for each bus and the power flow calculations for each sample are independent. Thus, they can be performed in parallel. Similarly, the regression problems for determining approximation coefficients are tractable linear problems that are independent across buses, enabling further parallelization. Furthermore, voltage solutions obtained from power flow computations can be reused across buses, reducing redundant calculations. This parallel structure ensures scalability and computational efficiency for large systems.

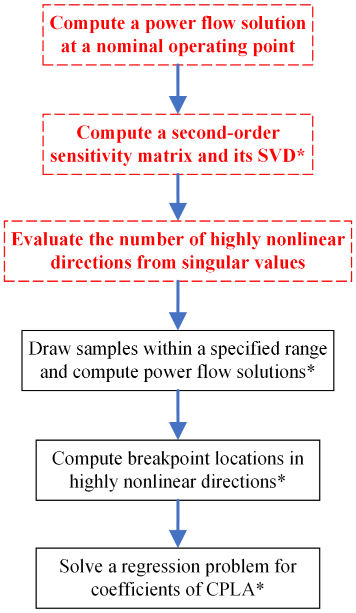

Compute a power flow solution at a nominal operating point
Compute a second-order sensitivity matrix and its SVD*
Evaluate the number of highly nonlinear directions from singular values
Draw samples within a specified range and compute power flow solutions*
Compute breakpoint locations in highly nonlinear directions*
Solve a regression problem for coefficients of CPLA*

Fig. 2. Flowchart illustrating the computation processes for the CPLA method. Red-dashed boxes highlight the steps involving the second-order sensitivity matrix. Steps marked with $*$ indicate parallelizable processes.

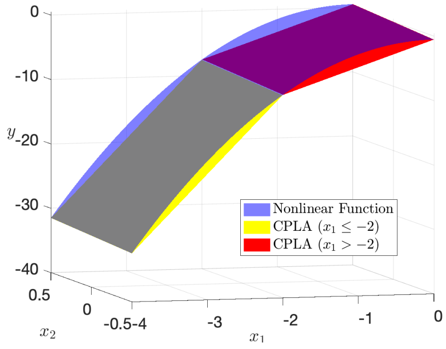


Fig. 3. An example of a CPLA is shown, where the yellow and red planes (each referred to as a "region" in this paper) underestimate the quadratic function $y = -4x_1^2 + x_2$ (blue manifold). The breakpoint is at $x_1 = -2$, with $x_1 \le -2$ and $x_1 > -2$ referred to as two segments. The yellow (red) plane corresponds to the function $y = 12x_1 + x_2 + 16$ ($y = 4x_1 + x_2$).

B. Problem formulation

CPLAs enhance the accuracy of power flow linearizations by addressing nonlinear behavior through multiple linear segments. However, not all dimensions of the power flow equations exhibit significant nonlinearity. Most directions display only mild curvature, while a select few are predominantly nonlinear and contribute substantially to the overall system behavior. To conceptually illustrate this concept, Fig. 3 presents an underestimating CPLA for a quadratic function with one highly nonlinear direction (x_1) and one linear direction (x_2). In this case, a piecewise linear function is only necessary along x_1, as x_2 exhibits linear curvature, highlighting the efficiency of focusing on critical dimensions.

Recognizing which dimensions require piecewise linearization is crucial for improving computational efficiency without sacrificing accuracy. This is where second-order sensitivity analysis becomes invaluable. By examining the second-order sensitivities of the power flow equations, we can identify and prioritize directions with pronounced nonlinear effects.

Let $[\boldsymbol{P}'_s; \boldsymbol{Q}'_s]$ represent the vector of rotated power injections (discussed in Section III-D) corresponding to highly nonlinear directions calculated from a second-order sensitivity analysis. Similar to the regression problem in (4), the formulation for computing a sample-based overestimating CPLA is as follows:

$$\min_{h(\,\cdot\,)} \quad \frac{1}{S}\sum_{s=1}^{S} \mathcal{L}\left(\gamma_s - h\left(\begin{bmatrix}\boldsymbol{P}_s\\ \boldsymbol{Q}_s\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}'_s\\ \boldsymbol{Q}'_s\end{bmatrix}, b_s\right)\right) \tag{9a}$$

$$\text{s.t. } \gamma_s - h\left(\begin{bmatrix}\boldsymbol{P}_s\\ \boldsymbol{Q}_s\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}'_s\\ \boldsymbol{Q}'_s\end{bmatrix}, b_s\right) \le 0, \tag{9b}$$

$$h\left(\begin{bmatrix}\boldsymbol{P}_s\\ \boldsymbol{Q}_s\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}'_s\\ \boldsymbol{Q}'_s\end{bmatrix}, b_s\right) = a_0 + \boldsymbol{a}_1^T \begin{bmatrix}\boldsymbol{P}_s\\ \boldsymbol{Q}_s\end{bmatrix} + g\left(\begin{bmatrix}\boldsymbol{P}'_s\\ \boldsymbol{Q}'_s\end{bmatrix}, b_s\right), \tag{9c}$$

$$\forall s = 1, \ldots, S,$$

where b_s denotes the segment which the s^{th} sample belongs to, $h(\,\cdot\,)$ represents the CPLA that consists of a CLA and a purely CPLA, denoted as $g(\,\cdot\,)$, which represents the piecewise linear function for which we optimize the coefficients across all segments (shown in Section III-C). Note that breakpoint locations in each direction are determined by evenly partitioning the sample space, allowing samples to share a segment in one direction but belong to different segments in another. By determining the breakpoint locations based on the drawn samples before solving for a CPLA in (9), we can effectively assign each sample to its corresponding segment.

We need to ensure that there are multiple samples within each region of the piecewise linear approximation. To avoid overfitting in the piecewise linear approximation, it is crucial to ensure that sufficient sample data exists within each region defined by the multidimensional piecewise linear functions. Overfitting is particularly likely when multiple highly nonlinear directions and several breakpoints in each direction are considered, as regions with sparse or no samples can lead to poorly generalized approximations. To address this, we propose the following two approaches:

1) Ensure an adequate number of drawn samples. Let C represent the number of regions and S represent the number of samples. Define ϵ as a predefined small positive confidence value. We ensure that the probability of the undesirable event where there are one or more regions with no samples is bounded by ϵ:
$$C\left(1 - \frac{1}{C}\right)^S \le \epsilon. \tag{10}$$
Assuming C is large, the bound on S is:
$$S \ge C \log \frac{C}{\epsilon}. \tag{11}$$
Note that this inequality approximates the probability that there is a region with no sample. This approximation is used to simplify the computation of the inequality in (10).
2) Use regularization by enforcing convexity or concavity in each direction, which can be determined by analyzing the singular values obtained from SVD. This approach reduces the required samples and, consequently, the computation time compared to the first approach.

C. Variable reduction from continuity

In this section, we consider a piecewise linear function in N dimensions with M breakpoints, resulting in $M+1$ segments within each dimension. Let $t_1, t_2, \ldots, t_N$ represent the N directions and $b_1, b_2, \ldots, b_N$ with $b_i \in \{0, 1, 2, \ldots, M\}$ represent the index of the segment in each direction. We define $\boldsymbol{t} = [t_1, t_2, \ldots, t_N]^T$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_N]$. The piecewise linear function $g(\boldsymbol{t}, \boldsymbol{b})$ is expressed as follows:

$$g(\boldsymbol{t}, \boldsymbol{b}) = a_0^{b_1 b_2 \ldots b_N} + a_1^{b_1 b_2 \ldots b_N} t_1 + \ldots + a_N^{b_1 b_2 \ldots b_N} t_N, \tag{12}$$

where the coefficients $a_0^{b_1 b_2 \ldots b_N}, a_1^{b_1 b_2 \ldots b_N}, \ldots, a_N^{b_1 b_2 \ldots b_N}$ correspond to $a_0, a_1, \ldots, a_N$ for the region defined by the segments $b_1, b_2, \ldots, b_N$. For instance, in Fig. 3, since there is only one nonlinear direction x_1, we have a single segment index $b_1 \in \{0, 1\}$, representing two segments. Specifically, $a_0^0 = 16$ and $a_1^0 = 12$ for the first segment ($b_1 = 0$), which corresponds to $x < -2$. The values of $t_1, t_2, \ldots, t_N$ fall within the specified range defined by the breakpoints.

We begin by examining the first breakpoint in the t_1 direction, denoted as t_1^0. At the point where $t_1 = t_1^0$, the continuity condition can be stated as follows:

$$a_0^{0b_2\ldots b_N} + a_1^{0b_2\ldots b_N} t_1^0 + \ldots + a_N^{0b_2\ldots b_N} t_N = \\ a_0^{1b_2\ldots b_N} + a_1^{1b_2\ldots b_N} t_1^0 + \ldots + a_N^{1b_2\ldots b_N} t_N. \tag{13}$$

Therefore, we can conclude that:

$$a_0^{0b_2\ldots b_N} + a_1^{0b_2\ldots b_N} t_1^0 = a_0^{1b_2\ldots b_N} + a_1^{1b_2\ldots b_N} t_1^0, \tag{14a}$$

$$a_2^{0b_2\ldots b_N} = a_2^{1b_2\ldots b_N}, \tag{14b}$$

$$\vdots$$

$$a_N^{0b_2\ldots b_N} = a_N^{1b_2\ldots b_N}. \tag{14c}$$

We apply a similar approach to all other breakpoints in the t_1 direction. For instance, considering the m^{th} breakpoint yields:

$$a_0^{(m-1)b_2\ldots b_N} + a_1^{(m-1)b_2\ldots b_N} t_1^{m-1} = a_0^{mb_2\ldots b_N} + a_1^{mb_2\ldots b_N} t_1^{m-1}, \tag{15a}$$

$$a_2^{(m-1)b_2\ldots b_N} = a_2^{mb_2\ldots b_N}, \tag{15b}$$

$$\vdots$$

$$a_N^{(m-1)b_2\ldots b_N} = a_N^{mb_2\ldots b_N}. \tag{15c}$$

By combining (14b)–(14c) and (15b)–(15c) for all $m = 0, 1, 2, \ldots, M-1$, we obtain:

$$a_0^{b_1' b_2\ldots b_N} = \sum_{i=0}^{b_1'-1} t_1^i \left(a_0^{ib_2\ldots b_N} - a_0^{(i+1)b_2\ldots b_N}\right) + a_0^{0b_2\ldots b_N}, \tag{16a}$$

$$a_2^{b_1 b_2\ldots b_N} = a_2^{b_1' b_2\ldots b_N}, \tag{16b}$$

$$\vdots$$

$$a_N^{b_1 b_2\ldots b_N} = a_N^{b_1' b_2\ldots b_N}, \tag{16c}$$

for any $b_1 \in \{0, 1, 2, \ldots, M\}$ and $b_1' \in \{1, 2, \ldots, M\}$.

We first focus specifically on the coefficient a_2 and make a claim regarding the other coefficients. By examining the breakpoints in the direction of t_3 and following an approach similar to the one outlined in equations (16b) to (16c), we derive an equality for the coefficient a_2 as follows:

$$a_2^{b_1 b_2 b_3\ldots b_N} = a_2^{b_1 b_2 b_3'\ldots b_N}, \tag{17}$$

for any $b_3, b_3' \in \{0, 1, 2, \ldots, M\}$.

Combining (16b) with (17), we obtain:

$$a_2^{b_1 b_2 b_3\ldots b_N} = a_2^{b_1' b_2 b_3\ldots b_N} = a_2^{b_1 b_2 b_3'\ldots b_N}. \tag{18}$$

Proceeding with the analysis, if we perform the same simplification across all directions except for direction t_2, we find that the coefficient a_2 is solely dependent on the segment b_2. Similarly, each coefficient a_n depends exclusively on its respective segment b_n for all $n = 1, 2, \ldots, N$.

Now, we analyze the coefficient a_0 and observe how it changes with directions. Utilizing the fact that a_n is dependent solely on direction t_n, coupled with information from (16a), a_0 is given by:

$$a_0^{b_1 b_2\ldots b_N} = \sum_{k=1}^{N} \left(\sum_{b_k'=0}^{b_k-1} t_k^{b_k'} \left(a_k^{b_k'} - a_k^{b_k'+1}\right) \right) + a_0^{00\ldots 0}. \tag{19}$$

Consequently, the piecewise linear function in (12) can be represented in a simplified form as:

$$g(\boldsymbol{t}, \boldsymbol{b}) = a_0^{b_1 b_2\ldots b_N} + a_1^{b_1} t_1 + a_2^{b_2} t_2 + \ldots + a_N^{b_N} t_N. \tag{20}$$

Leveraging continuity allows independent selection of breakpoints and simplifies the decision variables from $(N+1) \times (M+1)^N$ to $(N+1) \times (M+1)$.

D. Rotating coordinates from singular value decomposition

The directions of high curvature are identified using the significant singular values obtained from singular value decomposition (SVD) of the second-order sensitivity matrix $\boldsymbol{\Lambda}$ in (8). The singular vectors corresponding to these significant singular values are the directions in which we implement the CPLA. Let $\boldsymbol{\Lambda} = \boldsymbol{USU}^T$ be the singular value decomposition of $\boldsymbol{\Lambda}$ and $\widetilde{\boldsymbol{U}}$ be the sub-matrix of $\boldsymbol{U}$ obtained by selecting the columns corresponding to the significant singular values. Let $\boldsymbol{w}$ denote the complete vector of power injections. The vector of rotated power injections, $\boldsymbol{t}$ in (20), which represents the directions of high curvature is given by $\boldsymbol{t} = \widetilde{\boldsymbol{U}}^T \boldsymbol{w}$.

Since the segment indices b_i are automatically determined from the samples, the regression problem in (9a) using the representation of $g(\,\cdot\,)$ in (20) has a polyhedral constraint set and can be solved efficiently for several loss functions $\mathcal{L}(\,\cdot\,)$.

IV. Numerical results

In this section, we present numerical results to evaluate the effectiveness of the CPLA approach for approximating nonlinear power flow equations. The analysis is structured into three parts: examining the impact of singular vectors associated with dominant singular values, analyzing the effect of varying the number of segments, and exploring the combined influence of these factors to optimize performance.

Simulations are conducted on the *case30*, *case33bw*, *case141*, and *case2383wp* test cases from MATPOWER [16], with optimization problems solved using the YALMIP toolbox [17]. For each case, 10,000 samples are drawn by varying power injections within 50% to 150% of their nominal values. Reported voltages are in per unit (pu), and the ℓ_1 norm, denoted by $\mathcal{L}(\,\cdot\,)$, is used as the loss function. All computations are performed on a MacBook Pro with an Apple M1 Pro chip (10 cores, 16 GB RAM).

A. Effects of nonlinear directions and breakpoints

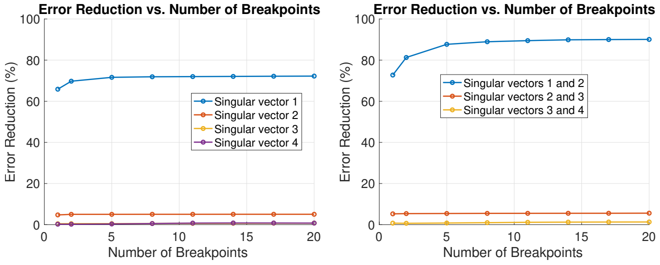
Piecewise linear approximation accuracy is strongly influenced by the specific nonlinear directions being considered. These directions are derived from the singular vectors corresponding to the dominant singular values of the system, with each singular vector representing a unique nonlinear direction. The amount of curvature in a given direction, as captured by its corresponding singular value, determines how well the approximation performs along that direction. Highly nonlinear directions are more challenging to approximate accurately. Analyzing one direction at a time provides insights into their influence on approximation quality.

In addition to the choice of nonlinear directions, the number of breakpoints used to represent a nonlinear function is another critical factor influencing the accuracy of a piecewise linear approximation. A finer segmentation can capture the underlying nonlinearity more effectively, resulting in significantly reduced approximation errors. However, this comes with the trade-off of increased computational demands. In this section, we analyze how the individual nonlinear directions and the number of breakpoints impact approximation accuracy.

Fig. 4 shows the error reduction compared to the CLA results as a function of the number of breakpoints for the 141-bus system at bus 80. Different lines represent the results obtained using various singular vectors. The singular vector associated with the most dominant singular value yields the greatest accuracy improvement. Increasing the number of breakpoints significantly reduces approximation errors for all singular vectors; however, this improvement diminishes beyond five breakpoints, where the error reduction stabilizes. For the most dominant singular vector, the error reduction reaches 72.22%, highlighting its substantial contribution to improving CPLA accuracy (refer to Fig. 4a). The improvement is particularly pronounced for the singular vector associated with the most dominant singular value, which achieves a significant error reduction. In contrast, other singular vectors provide smaller improvements, with the second singular vector achieving a maximum error reduction of about 5%, while subsequent singular vectors contribute even less.

Considering two singular vectors, Fig. 4b illustrates the error reduction for the CPLA of voltages at bus 80 of the 141-bus system. The results show that implementing CPLA in the directions of the first two singular vectors, corresponding to the two most dominant singular values, achieves a combined improvement of up to 90.07%. In particular, incorporating two singular vectors provides additional improvement over using only the first singular vector, demonstrating the advantage of capturing multiple high-curvature directions.

Table I presents the detailed approximation errors, error reductions, and computation times for computing the underestimating voltages by selecting the singular vector corresponding to the most dominant singular value in the *case30*, *case33bw*, *case141*, and *case2383wp* test cases at a specific bus. The results demonstrate that the proposed piecewise linear approach reduces errors by 41.30% to 71.93% while requiring comparable computation time to CLA, highlighting its effectiveness in balancing improved accuracy with computational efficiency. For better visualization, the plot comparing the results of CLA and CPLA with five breakpoints is shown in Fig. 5, demonstrating that CPLA provides a better approximation than CLA.



(a) One singular vector (b) Two singular vectors

Fig. 4. Percentage error reduction of voltage magnitudes vs. number of breakpoints in the 141-bus system at bus 80 when considered (a) one singular vector and (b) two singular vectors.

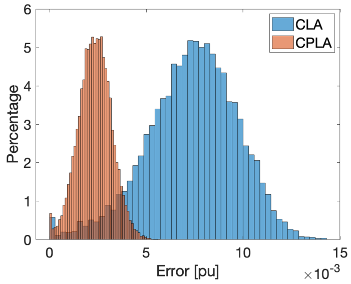
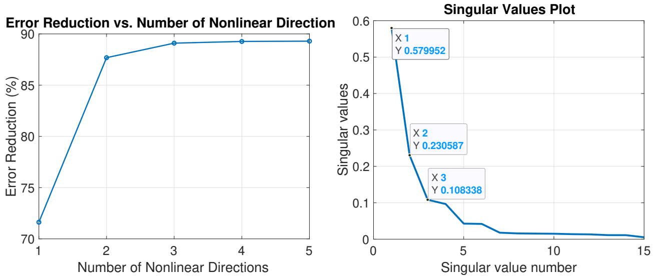


Fig. 5. Error histograms from the underestimating CLA (blue) and the underestimating CPLA (red) for voltage magnitudes at bus 20 in the IEEE 30-bus system.



(a) Error reduction (b) Singular values

Fig. 6. (a) Percentage error reduction of voltage magnitudes vs. the number of nonlinear directions and (b) plot of the first 15 most dominant singular values for the 141-bus system at bus 80.

B. Singular values and their effects to the nonlinear directions

The insights from analyzing nonlinear directions and segmentation granularity from the previous section highlight the trade-offs in achieving accurate piecewise linear approximations. Significant accuracy gains come from dominant nonlinear directions, with diminishing returns from subsequent singular vectors. Similarly, increasing breakpoints improves accuracy only up to a point, beyond which additional segments offer minimal benefit. These findings emphasize the need to balance dominant nonlinear directions with segmentation granularity. This section explores the impact of varying the number of nonlinear directions while keeping the number of breakpoints fixed and examines the trend of singular values.

Fig. 6 illustrates the reduction in error when varying the number of nonlinear directions with five breakpoints in each direction, alongside the first 15 most dominant singular values. The results in Fig. 6a indicate that incorporating up to five nonlinear directions achieves a significant error reduction of 89.29%. This performance is comparable to the results achieved by considering two nonlinear directions with ten breakpoints in each direction, as shown in Fig. 4b. The singular values plotted in Fig. 6b reveal a steep decline, dropping from

TABLE I
APPROXIMATION ERRORS FOR UNDERESTIMATING VOLTAGE MAGNITUDES AT A SPECIFIC BUS

Cases	Bus	t_{pf} [ms]	CLA		CPLA					
			Error/sample [pu]	time [s]	Error/sample for each #breakpoint(s)* [pu]			t_{CPLA} [s]		
					1	5	10	1	5	10
case30	20	2.26	7.36×10^{-3}	1.51	4.32×10^{-3} (41.30%)	4.24×10^{-3} (42.42%)	4.21×10^{-3} (42.84%)	1.67	1.72	1.73
case33bw	33	2.43	2.84×10^{-4}	2.27	1.37×10^{-4} (51.61%)	1.18×10^{-4} (58.51%)	1.16×10^{-4} (59.13%)	2.37	2.38	2.41
case141	80	5.13	6.75×10^{-5}	5.57	2.31×10^{-5} (65.86%)	1.92×10^{-5} (71.63%)	1.90×10^{-5} (71.93%)	5.95	7.98	8.16
case2383wp	466	169.52	7.45×10^{-5}	125.16	4.31×10^{-5} (41.83%)	3.64×10^{-5} (50.81%)	3.59×10^{-5} (51.43%)	148.24	175.86	182.47

t_{pf} = Average time required to solve 1 power flow solution in milliseconds.
t_{CPLA} = Time required to compute the second-order sensitivity matrix, perform the SVD, and compute the CPLA for a specified number of breakpoints.
*The percentage reduction in errors compared to the errors from conservative linear approximation (CLA).

0.58 for the most dominant direction to 0.11 for the third, and continuing to decrease for subsequent directions. This shows that only a few highly nonlinear directions contribute significantly to reducing errors, emphasizing the efficiency of focusing computations on these dominant directions.

V. CONCLUSION AND FUTURE WORK

This paper demonstrates the effectiveness of the CPLA approach for approximating nonlinear power flow equations, emphasizing a balance between accuracy and computational efficiency. The results highlight that only the first few singular vectors, corresponding to the most dominant singular values, contribute significantly to improving the accuracy of the approximation. This observation not only reinforces the importance of targeting key nonlinear directions but also reduces computational demands by limiting the focus to a small subset of directions. Furthermore, while increasing the number of segments improves precision, the diminishing returns observed beyond a certain point emphasize the need for an optimal segmentation strategy. This approach provides a practical and efficient solution for addressing the challenges of nonlinear power flow equations in optimization problems.

In future work, we aim to apply our CPLA method to power system planning and resilience tasks, with a focus on capacity expansion planning problems. Furthermore, we plan to explore advanced methods for determining breakpoint locations and develop CPLAs for other quantities, such as current flows, as well as for systems with varying topologies.

ACKNOWLEDGMENT

P. Buason thanks the Department of Mechatronics Engineering at Rajamangala University of Technology Phra Nakhon.

REFERENCES

[1] B. Stott, J. Jardim, and O. Alsaç, “DC power flow revisited,” *IEEE Trans. Power Syst.*, vol. 24, no. 3, pp. 1290–1300, Aug. 2009.

[2] M. E. Baran and F. F. Wu, “Optimal capacitor placement on radial distribution systems,” *IEEE Trans. Power Delivery*, vol. 4, no. 1, pp. 725–734, Jan. 1989.

[3] S. Misra, D. K. Molzahn, and K. Dvijotham, “Optimal adaptive linearizations of the AC power flow equations,” in *20th Power Systems Computation Conf. (PSCC)*, June 2018.

[4] P. Buason, S. Misra, and D. K. Molzahn, “A sample-based approach for computing conservative linear power flow approximations,” *Electric Power Systems Research*, vol. 212, p. 108579, 2022, presented at the *22nd Power Systems Computation Conf. (PSCC 2022)*.

[5] P. Buason, S. Misra, J.-P. Watson, and D. K. Molzahn, “Adaptive power flow approximations with second-order sensitivity insights,” to appear in *IEEE Trans. Power Syst.*

[6] P. Buason, S. Misra, and D. K. Molzahn, “Sample-based conservative bias linear power flow approximations,” in *IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia)*, 2024.

[7] J. Chen and L. A. Roald, “A data-driven linearization approach to analyze the three-phase unbalance in active distribution systems,” *Electric Power Systems Research*, vol. 211, p. 108573, 2022, presented at the *22nd Power Systems Computation Conf. (PSCC 2022)*.

[8] Y. Liu, N. Zhang, Y. Wang, J. Yang, and C. Kang, “Data-driven power flow linearization: A regression approach,” *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2569–2580, May 2019.

[9] M. Jia and G. Hug, “Overview of data-driven power flow linearization,” in *IEEE Belgrade PowerTech*, 2023.

[10] M. Jia, G. Hug, N. Zhang, Z. Wang, and Y. Wang, “Tutorial on data-driven power flow linearization–Part I: Challenges and training algorithms,” preprint available at https://doi.org/10.3929/ethz-b-000606654, 2023.

[11] B. Austgen, E. Kutanoglu, J. J. Hasenbein, and S. Santoso, “Comparisons of two-stage models for flood mitigation of electrical substations,” to appear in *INFORMS J. Comput.*

[12] A. D. Owen Aquino, S. Talkington, and D. K. Molzahn, “Managing vehicle charging during emergencies via conservative distribution system modeling,” in *8th Texas Power and Energy Conf. (TPEC)*, Feb. 2024.

[13] A. Castillo, C. Laird, C. A. Silva-Monroy, J.-P. Watson, and R. P. O’Neill, “The unit commitment problem with AC optimal power flow constraints,” *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 4853–4866, 2016.

[14] P. Buason, S. Misra, S. Talkington, and D. K. Molzahn, “A data-driven sensor placement approach for detecting voltage violations in distribution systems,” *Electric Power Systems Research*, vol. 232, p. 110387, 2024.

[15] R. Gupta, P. Buason, and D. K. Molzahn, “Fairness-aware photovoltaic generation limits for voltage regulation in power distribution networks using conservative linear approximations,” in *8th Texas Power and Energy Conf. (TPEC)*, Feb. 2024.

[16] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, “MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and education,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.

[17] J. Lofberg, “YALMIP: A toolbox for modeling and optimization in MATLAB,” in *IEEE International Conf. on Robotics and Automation*, 2004, pp. 284–289.