Ensemble Learning Based Convex Approximation of Three-Phase Power Flow

Ren Hu, *Student Member, IEEE*, Qifeng Li, *Senior Member, IEEE*, and Feng Qiu, *Senior Member, IEEE*

***Abstract*—Though the convex optimization has been widely used in power systems, it still cannot guarantee to yield a tight (accurate) solution to some problems. To mitigate this issue, this paper proposes an ensemble learning based convex approximation for alternating current (AC) power flow equations that differs from the existing convex relaxations. The proposed approach is based on three-phase quadratic power flow equations in rectangular coordinates. To develop this data-driven convex approximation of power flows, the polynomial regression (PR) is first deployed as a basic learner to fit convex relationships between the independent and dependent variables. Then, ensemble learning algorithms such as gradient boosting (GB) and bagging are introduced to combine learners to boost model performance. Based on the learned convex approximation of power flow, optimal power flow (OPF) is formulated as a convex quadratic programming problem. The simulation results on IEEE standard cases of both balanced and unbalanced systems show that, in the context of solving OPF, the proposed data-driven convex approximation outperforms the conventional semi-definite programming (SDP) relaxation in both accuracy and computational efficiency, especially in the cases that the conventional SDP relaxation fails.**

***Index Terms*—Convex approximation, data-driven, ensemble learning, power flow.**

I. Introduction

POWER flow analysis plays a significant role in power system planning and operation. Many decision-making processes in power systems rely heavily on accurate and effective power flow calculations [1]–[3]. Power flow models are also considered as inevitable system constraints in optimization problems like transmission or generation expansion planning and optimal power flow (OPF) [3]–[5]. However, the nonlinearity and nonconvexity of AC power flow (ACPF) make the OPF problem computationally expensive due to the nondeterministic polynomial time (NP) problem.

One approach to handling these challenges is to linearize the power flow equations, which has been widely adopted in power generation dispatching [6], [7] and power market trading [8], [9]. One of the most well-known linear PF models, i.e., the direct current (DC) power flow, captures the linear relationship between the active power flow injection and the bus voltage phase angle. Other extended versions of linear power flow, that involves reactive power, have also attracted substantial attention [10]–[14]. For the sake of a better predictive accuracy, the authors in [14] develops a linear regression model for power flow through Principal component analysis (PCA) based and Bayesian inference with regularization. Based on [14], the linear model is further improved considering measurement noise in data to simulate the practical data in [15]. Reference [16] explores the generalization errors of linear regression methods discussed in [14], [15] in detail through bounding the training and test error, which testifies the robustness of these methods and enriches the work of regression-based linear approximation of power flow. Although linear power flow models are computationally tractable, they are generally based on some critical assumptions, such as ignoring the inherent quadratic terms between bus voltages. Additionally, there may be some concerns on their applicability to some problems, such as how to explain and apply the new explanatory variables created by the PCA-based method.

With the deluge of data generated by sensors like phasor measurement units today, data-driven methods have attracted massive research efforts in power system analysis, such as in estimating distribution factors [17] and the Jacobian matrix [18] and identifying the admittance matrix [19]. To overcome the issues above, the convex quadratic approximation of the originally nonconvex quadratic power flow is learnt in this paper based on machine learning techniques, considering the inherent quadratic terms between bus voltages. More precisely, the polynomial regression [20], [21] is employed as a basic learner to infer convex relationships between the active or reactive power and the bus voltage. Since the original power flow is nonlinear quadratic, and the polynomial regression can learn the nonlinear quadratic relationship, the polynomial regression is used to fit the convex nonlinear quadratic relationship in power flow. Ensemble learning algorithms, i.e., gradient boosting [22], [23] and bagging [24]–[26], are then introduced to assemble every basic learner and tune the learning parameters to avoid overfitting, and eventually yield a stronger learner. Boosting and bagging tend to use homogeneous basic learners with the good interpretability and applicability, while others may involve heterogeneous basic learners with different assumptions. Hence, any other ensemble learning methods, such as stacking,

Manuscript received May 14, 2020; revised August 31, 2020, October 21, 2020, and December 16, 2020; accepted January 16, 2021. Date of publication January 28, 2021; date of current version August 19, 2021. The work of Ren Hu and Qifeng Li was supported by the U.S. National Science Foundation under Grant #1808988. The work of Feng Qiu was supported by the U.S. Department of Energy Advanced Grid Modeling Program under Grant DE-OE0000875. Paper no. TPWRS-00793-2020. *(Corresponding author: Qifeng Li.)*

Ren Hu and Qifeng Li are with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816 USA (e-mail: hurenlaker@knights.ucf.edu; qifeng.li@ucf, edu).

Feng Qiu is with the Energy Systems Division, Argonne National Laboratory, Lemont, IL 60439 USA (e-mail: fqiu@anl.gov).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPWRS.2021.3055481.

Digital Object Identifier 10.1109/TPWRS.2021.3055481

error-correcting output codes (ECOC), mixture of experts, Bayesian model averaging and combination, etc., will not be discussed here.

Although nonlinear programming problems are generally NP-hard to solve, many convex nonlinear optimization problems admit polynomial-time algorithms [27]. In recent years, high-performance solvers such as MOSEK, CPLEX, and GUROBI have been developed to effectively solve major types of convex problems. Various convex relaxations, such as second-order cone (SOC) [28], semi-definite programming (SDP) [29], enhanced SDP [30] convex DistFlow (CDF) [31], [32], quadratic convex (QC) [33], moment-based [34], and convex hull relaxation [35], have been introduced to convexify a fundamental power system optimization problem. Of these, SDP-based relaxations have attracted the most attention due to its general applicability to nonconvex quadratic problems [32]–[34] including the three-phase ACPF investigated in this paper. However, they are generally computationally hard and not tight enough to guarantee a satisfactory solution for many OPF cases and result in failures [38], [46], [47]. The SOC relaxation is computationally easier and, however, not as tight as SDP [35], while CDF relaxation is only suitable for balanced tree networks. To avoid the limitations of the existing convex relaxations, this paper develops a novel convex approximation of power flow models through using data-driven methods. The main contributions of this paper can be summarized as follows:

- A data-driven convex quadratic approximation (DDCQA) of power flow is developed based on the polynomial regression and ensemble learning. In the parameter-fitting process, ensemble learning algorithms are applied to incorporate all basic learners, i.e., the polynomial regression, into a stronger learner and to enhance the accuracy of DDCQA.
- The proposed DDCQA is applied to convexify the OPF and a DDCQA-based OPF is proposed. The resulting OPF is a convex quadratic programming problem that outperforms the existing SDP relaxation in computational efficiency and accuracy. More importantly, the accuracy of the proposed DDCQA can be improved by learning from operation experience.

The rest of this paper is organized as follows: Section II depicts the existing problems and proposed solutions for computing and fitting power flow. In Section III, data-driven convexification of power flow is formulated through ensemble learning. The empirical IEEE case analyses and conclusions are displayed in Section IV and V, respectively.

II. Problem Formulation and Statement

This section first presents a three-phase ACPF models in rectangular coordinates. Then, the existing problems are discussed, and an overview of the proposed method is provided.

A. Three-Phase AC Power Flows in Rectangular Coordinates

In an *n*-bus three-phase power networks, the equations of ACPF in rectangular coordinates for each phase involving the coupling terms from another two phases can be generally illustrated as below.

$$\begin{aligned}
P_i^\phi &= e_i^\phi \sum_{j\in N}\sum_{\gamma\in\Phi}\left(G_{ij}^{\phi\gamma}e_j^\gamma - B_{ij}^{\phi\gamma}f_j^\gamma\right) \\
&\quad + f_i^\phi \sum_{j\in N}\sum_{\gamma\in\Phi}\left(G_{ij}^{\phi\gamma}f_j^\gamma + B_{ij}^{\phi\gamma}e_j^\gamma\right) \\
Q_i^\phi &= f_i^\phi \sum_{j\in N}\sum_{\gamma\in\Phi}\left(G_{ij}^{\phi\gamma}e_j^\gamma - B_{ij}^{\phi\gamma}f_j^\gamma\right) \\
&\quad - e_i^\phi \sum_{j\in N}\sum_{\gamma\in\Phi}\left(G_{ij}^{\phi\gamma}f_j^\gamma + B_{ij}^{\phi\gamma}e_j^\gamma\right) \\
P_{ij}^\phi &= e_i^\phi \sum_{\gamma\in\Phi}\left[G_{ij}^{\phi\gamma}(e_i^\gamma - e_j^\gamma\right) + B_{ij}^{\phi\gamma}\left(f_j^\gamma - f_i^\gamma\right)] \\
&\quad + f_i^\phi \sum_{\gamma\in\Phi}\left[G_{ij}^{\phi\gamma}(f_i^\gamma - f_j^\gamma\right) + B_{ij}^{\phi\gamma}\left(e_i^\gamma - e_j^\gamma\right)] \\
Q_{ij}^\phi &= e_i^\phi \sum_{\gamma\in\Phi}[G_{ij}^{\phi\gamma}\left(f_j^\gamma - f_i^\gamma\right) - B_{ij}^{\phi\gamma}\left(e_i^\gamma - e_j^\gamma\right)] \\
&\quad + f_i^\phi \sum_{\gamma\in\Phi}[G_{ij}^{\phi\gamma}\left(e_i^\gamma - e_j^\gamma\right) - B_{ij}^{\phi\gamma}\left(f_i^\gamma - f_j^\gamma\right)] \qquad (1)
\end{aligned}$$

where $\Phi = \{a, b, c\}$ denotes the phase set and $\gamma, \phi \in \Phi$; $N = \{01, 2, .., n\}$ is the bus set, $i, j \in N$; P_i^ϕ and Q_i^ϕ represent the ϕ-phase active and reactive power injections at bus i; e_i^ϕ and f_i^ϕ are the real and imaginary parts of the ϕ-phase voltage at bus i; P_{ij}^ϕ and Q_{ij}^ϕ are the ϕ-phase active and reactive line flow at branch *i-j*; $G_{ij}^{\phi\gamma}$ and $B_{ij}^{\phi\gamma}$ denote the conductance and susceptance between $\phi$and γ phases at branch *i-j*. Note that if $\phi = \gamma$, then $G_{ij}^{\phi\gamma} = G_{ij}$, $B_{ij}^{\phi\gamma} = B_{ij}$, they are the ϕ-phase self-conductance and self-susceptance at branch *i-j*. Generally, for balanced power systems, the coupling terms in (1) are ignored and the equations of ACPF for each phase can be simplified without the superscripts.

In the formulations above, we treat $e_i^\phi,\ f_i^\phi$ as independent variables and $P_i^\phi, Q_i^\phi, P_{ij}^\phi, Q_{ij}^\phi$ as dependent variables with respect to data mining. To facilitate the analysis of power flow equations, we transform (1) into matrix forms:

$$\begin{aligned}
p_i &= X^T\ A_i X \\
q_i &= X^T\ B_i X \\
p_{ij} &= X_{ij}^T\ A_{ij} X_{ij} \\
q_{ij} &= X_{ij}^T\ B_{ij} X_{ij} \qquad (2)
\end{aligned}$$

where p_i, q_i, p_{ij}, q_{ij} correspond to $P_i^\phi, Q_i^\phi, P_{ij}^\phi, Q_{ij}^\phi$; $A_i,\ B_i,\ A_{ij},\ B_{ij}$ are symmetrical but indefinite matrices constituted by entries of the admittance matrix, indicating that all dependent variables in power flow are nonconvex functions of the independent variables. For the balanced three-phase power systems, X denotes the voltage component vector and consists of the single-phase voltage components at all buses, $X = [e_1\ f_1, ..,\ e_n\ f_n]^T = [x_1\ x_2, ..,\ x_{2n}]^T$,

and X_{ij}denotes another voltage component vector and consists of the single-phase voltage components at buses i and j, $X_{ij} = [e_i\ f_i\ e_j\ f_j]^T = [x_{2i-1}\ x_{2i}\ x_{2j-1}\ x_{2j}]^T$. For the unbalanced three-phase power systems, X contains the three-phase voltage components at all buses, $X = [x_1^a\ x_2^a, .., x_{2n}^a, x_1^b\ x_2^b .., x_{2n}^b, x_1^c\ x_2^c, .., x_{2n}^c]^T$; X_{ij} contains the three-phase voltage components at buses i and j, $X_{ij} = [x_{2i-1}^a\ x_{2i}^a\ x_{2j-1}^a x_{2j}^a, \ldots, x_{2i-1}^c\ x_{2i}^c\ x_{2j-1}^c x_{2j}^c]^T$.

B. Existing Problems and the Proposed Solutions

The nonlinearity and nonconvexity of the three-phase ACPF (1) introduce challenges to compute optimization and control problems in power systems. Many linearization methods have been proposed to simplify the power flow models and reduce the computation burden, however, at the cost of model accuracy. To overcome these problems, given that the power flow equations are quadratic, data mining techniques can be applied to fit a quadratic convex relationship between the independent and dependent variables, provided that there are large historical databases of power system operations or enough accessible measurements. Theoretically, a convex quadratic approximation of power flows, which is nonlinear, outperforms the linear approximations in terms of model fidelity. Moreover, according to the theory of numerical optimization, a convex quadratic programming is much more computationally tractable than SDP problems.

An important step of obtaining such a DDCQA is to fit a positive semi-definite approximation of the original indefinite coefficient matrix of ACPF. Similar research can be found in other areas like analyzing the correlation matrices of financial stocks employed projection-based algorithms [40]–[42], which converted the original non-convergence problem into a convex optimization problem. Instead, in our study we take full advantage of ensemble learning techniques to approximate to the closest convex representations of power flow. The fitted DDCQA of ACPF can be further applied to OPF computation, state estimate or other cases in place of the original nonconvex model.

III. Ensemble Learning Based Convex Approximation

This section introduces the procedure for the proposed ensemble learning based DDCQA of ACPF in details. First, the nonconvex quadratic mapping in (1) is replaced by a convex quadratic mapping. Then, ensemble learning methods are introduced to infer the parameters of the proposed convex mapping. Finally, the convex mapping is further relaxed into a set of convex constraints with application in OPF.

A. Convex Mapping Representations

First, we define a convex quadratic mapping in (3) between power, i.e., p_i, q_i, p_{ij}, and q_{ij}, and voltage X, as the original power flow is represented by a quadratic formulation:

$$p_i = X^T\ A_i^p X + B_i^p X + c_i^p$$
$$q_i = X^T\ A_i^q X + B_i^q X + c_i^q$$
$$p_{ij} = X_{ij}^T\ A_{ij}^p X_{ij} + B_i^p X_{ij} + c_{ij}^p$$
$$q_{ij} = X_{ij}^T\ A_{ij}^q X_{ij} + B_i^q X_{ij} + c_{ij}^q \quad (3)$$

where the positive semi-definite coefficient matrices of the quadratic terms at bus i and branch i-j are represented by A_i^* and A_{ij}^*, respectively; the coefficient vectors of the linear terms are represented by B_i^* and B_{ij}^*, and c_i^* and c_{ij}^* are constant terms. Note that here the upper index $(*)$ represents the index set $\{p, q\}$, which corresponds to the active or reactive power. (3) are convex functions, since their coefficient matrices of the quadratic terms are at least positive semi-definite. To sum up, the target is to find convex functions to approximate the non-convex ones in (2), and (3) are inferred as the convex approximations of (2).

B. Ensemble Learning for Inferring Convex Mapping

Next, ensemble learning, a machine learning technique, is introduced to fit parameter matrices A_i^* and A_{ij}^*, vectors B_i^* and B_{ij}^*, and constants c_i^* and c_{ij}^* from historical system operation data. Bagging and Gradient boost (GB) [39], two typical ensemble learning algorithms, are of particular interest due to their strong ability to enhance basic learners to improve the model's performance.

Assume that we are given a training set including M samples $\{(X_m, Y_m)\}_{m=1}^M$. For each sample in an *n*-bus system, the vector of real and imaginary parts vector of bus voltage is denoted by $X_m = [x_{m1}, x_{m2}, .., x_{m(2n)}]$ for balanced three-phase systems, or $X_m = [x_{m1}, x_{m2}, .., x_{m(6n)}]$ for unbalanced three-phase systems. If the dependent variable Y is the active or reactive power at each bus p_i or q_i , the observation value can be depicted by $Y_m = p_{mi}$ or q_{mi}. Similarly, for p_{ij} or q_{ij}, we set $Y_m = p_{mij}$ or q_{mij}. The following illustrations of bagging and GB are all based on the dependent variable p_i as a general example. Other dependent variables follow the same procedure as p_i.

1) Bagging: Bagging, called bootstrap aggregating in some references [24], [25], is designed to improve model stability and accuracy and is applied in classification and regression analysis. As an ensemble technique, it contributes to reducing variance and avoiding overfitting through adjusting the number of bootstraps — a special case of the model averaging approach. The main purpose of bagging is to draw random samples with replacement and combine a basic learning method to train models. The specific loss function for p_i is given as the following mean squared error function:

$$L\ (p_{mi}, p_{mi}\,(X)) = \frac{1}{2}\ (p_{mi} - p_{mi}\,(X))^2 \quad (4)$$

where p_{mi} and $p_{mi}(X)$ are the observed and estimated values of p_i , respectively. The algorithm is shown below in Algorithm 1: Bagging.

2) Gradient Boosting: Gradient boosting is widely used to develop a strong learner by combining many weak learners in an iterative fashion [22], [23] for regression and classification problems. It is considered a gradient descent algorithm that can restrain the overfitting effect of the regularization parameters,

Algorithm 1: Bagging.

1. For $\boldsymbol{bt} = 1$ to $\mathbf{BT}$ where $\mathbf{BT}$ is the number of bootstraps.
   a. At the *bt*-th bootstrap, draw $\boldsymbol{M'}$ ($\boldsymbol{M'} \leq \boldsymbol{M}$) random samples with replacement.
   b. Fit a base learner $\boldsymbol{p_{mi}^{bt}}(\boldsymbol{X};\boldsymbol{\theta})$ by

$$\boldsymbol{\theta_{bt}} = \boldsymbol{arg}\min_{\boldsymbol{\theta}} \sum_{\boldsymbol{m}=1}^{\boldsymbol{M}} \boldsymbol{L}(\boldsymbol{p_{mi}^{bt}}, \boldsymbol{p_{mi}^{bt}}(\boldsymbol{X_m};\boldsymbol{\theta})) \tag{5}$$

   where $\boldsymbol{p_{mi}^{bt}}$ is the observed value of $\boldsymbol{p_i}$ at the *bt*-th bootstrap and $\boldsymbol{\theta_{bt}}$ represents the coefficient vector of $\boldsymbol{p_{mi}^{bt}}(\boldsymbol{X};\boldsymbol{\theta})$ by fitting $\boldsymbol{p_{mi}^{bt}}$. In a similar way, the polynomial regression as the basic learner is introduced to estimate all parameters in (3).

2. Output $\boldsymbol{p_{mi}^{bag}}(\boldsymbol{X})$ by averaging all bootstrap outcomes in

$$\boldsymbol{p_{mi}^{bag}}(\boldsymbol{X}) = \frac{1}{\boldsymbol{BT}} \sum_{\boldsymbol{bt}=1}^{\boldsymbol{BT}} \boldsymbol{p_{mi}^{bt}}(\boldsymbol{X}) \tag{6}$$

   where $\boldsymbol{p_{mi}^{bag}}(\boldsymbol{X})$ is the predicted value of $\boldsymbol{p_i}$ by bagging.

such as number of iterations and learning rate. The essence of gradient descent is to adjust parameters iteratively to minimize a loss function. It measures the local gradient of the loss function for a given number of iterations and takes steps in the direction of the descending gradient. Once the gradient is zero, we have reached the minimum. The detailed procedure of the algorithm is shown below in Algorithm 2: Gradient Boosting.

C. Multicollinearity and Overfitting

Multicollinearity is a situation in which two or more independent variables are strongly linearly correlated, and it can be ignored if the goal is to reach the accurate prediction of model and the model fitting is good enough [43]. Overfitting refers to a model that works well on the training dataset but poorly on the test dataset. The lack of a training dataset can also result in overfitting. In the field of machine learning, many remedies have been put forward to overcome these challenges, including removing unwanted independent variables by shrinkage and PCA-based methods [43], enlarging dataset size, and using ensemble learning [25], [26]. To some extent, removing some variables may help improve the accuracy of prediction in some linear models of power flow. However, it may not only completely change the bus number of the power system but may enlarge the model bias and contribute to poor interpretability and applicability in further optimization and control problems. Therefore, to preserve the original power system and make full use of the data-driven techniques, increasing the amount of data to learn is a sound and accessible way to relieve the fitting problem. In addition, ensemble learning is applied to avoid overfitting through regularization and resampling techniques.

Algorithm 2: Gradient Boosting.

1. Initialize the model with a constant value $\boldsymbol{p_{mi}^{0}}(\boldsymbol{X})$

$$\boldsymbol{p_{mi}^{0}}(\boldsymbol{X}) = \boldsymbol{arg}\min_{\boldsymbol{\gamma}} \sum_{\boldsymbol{m}=1}^{\boldsymbol{M}} \boldsymbol{L}(\boldsymbol{p_{mi}}, \boldsymbol{\gamma}) \tag{7}$$

   where $\boldsymbol{\gamma}$ is the initial constant vector.

2. For $\boldsymbol{t} = 1$ to $\mathbf{T}$ where T is the number of learners.
   c. Compute the negative gradient $\boldsymbol{r_t}$ by

$$\boldsymbol{r_t} = -\left[\frac{\partial \boldsymbol{L}(\boldsymbol{p_{mi}}, \boldsymbol{p_{mi}}(\boldsymbol{X}))}{\partial \boldsymbol{p_{mi}}(\boldsymbol{X})}\right]_{\boldsymbol{p_{mi}}(\boldsymbol{X})=\boldsymbol{p_{mi}^{t-1}}(\boldsymbol{X})} \tag{8}$$

   d. Fit a base learner $\boldsymbol{h_t}(\boldsymbol{X};\boldsymbol{\theta})$ by

$$\boldsymbol{\theta_t} = \boldsymbol{arg}\min_{\boldsymbol{\theta}} \sum_{\boldsymbol{m}=1}^{\boldsymbol{M}} \boldsymbol{L}(\boldsymbol{r_t}, \boldsymbol{h_t}(\boldsymbol{X_m};\boldsymbol{\theta})) \tag{9}$$

   where $\boldsymbol{\theta_t}$ presents the coefficient vector of $\boldsymbol{h_t}(\boldsymbol{X};\boldsymbol{\theta})$ by fitting $\boldsymbol{r_t}$. Here the polynomial regression is adopted to be a basic learner and fit the parameter matrices $\boldsymbol{A_i^*}$ and $\boldsymbol{A_{ij}^*}$, vectors $\boldsymbol{B_i^*}$ and $\boldsymbol{B_{ij}^*}$, and constants $\boldsymbol{c_i^*}$and $\boldsymbol{c_{ij}^*}$ in (3).
   e. Compute the learning rate β by

$$\boldsymbol{\beta_t} = \boldsymbol{arg}\min_{\boldsymbol{\beta}} \sum_{\boldsymbol{m}=1}^{\boldsymbol{M}} \boldsymbol{L}(\boldsymbol{p_{mi}}, \boldsymbol{p_{mi}^{t-1}}(\boldsymbol{X_m}) + \boldsymbol{\beta h_t}(\boldsymbol{X_m};\boldsymbol{\theta})) \tag{10}$$

   Setting a constant learning rate is also allowed. In practice, there is a common pattern that the smaller $\boldsymbol{\beta}$ is, the lower the descent increment is, and the better generalization is achieved. However, the cost of improving the generalization is the reduction of convergence speed.
   f. Update the model.

$$\boldsymbol{p_{mi}^{t}}(\boldsymbol{X}) = \boldsymbol{p_{mi}^{t-1}}(\boldsymbol{X}) + \boldsymbol{\beta_t h_t}(\boldsymbol{X}) \tag{11}$$

3. Output $\boldsymbol{p_{mi}^{T}}(\boldsymbol{X})$

D. Convex Approximation of Optimal Power Flow

After fitting the convex formulations of power flows, the data-driven convex approximation for three-phase optimal power flow consists of the objective function and constraints shown in (12) as below.

$$\begin{aligned}
\text{Minimize} & \sum_{i\in G}\sum_{\phi\in\Phi}(c_{i0} + c_{i1}p_i^{g\phi} + c_{i2}p_i^{g\phi 2}) \\
\text{s.t.} \quad & X^T A_i^{p\phi} X + B_i^{p\phi} X + c_i^{p\phi} \leq p_i^{g\phi} - p_i^{\mathrm{L}\phi} \\
& X^T A_i^{q\phi} X + B_i^{q\phi} X + c_i^{q\phi} \leq q_i^{g\phi} - q_i^{\mathrm{L}\phi} \\
& x_{2i-1}^{\phi 2} + x_{2i}^{\phi 2} \leq \bar{V}_i^{\phi 2} \\
& \underline{p}_i^{g\phi} \leq p_i^{g\phi} \leq \bar{p}_i^{g\phi}
\end{aligned}$$

$$\begin{aligned}
&\underline{q}_i^{g\phi} \le q_i^{g\phi} \le \bar{q}_i^{g\phi} \\
&p_{ij}^{\phi 2} + q_{ij}^{\phi 2} \le \bar{S}_{ij}^{\phi 2} \\
&X_{ij}^T A_{ij}^{p\phi} X_{ij} + B_{ij}^{p\phi} X_{ij} + c_{ij}^{p\phi} \le p_{ij}^{\phi} \\
&X_{ij}^T A_{ij}^{q\phi} X_{ij} + B_{ij}^{q\phi} X_{ij} + c_{ij}^{q\phi} \le q_{ij}^{\phi}
\end{aligned} \tag{12}$$

where G is the index set of generators; c_{i0}, c_{i1}, c_{i2} are cost coefficients of the *i*-th generator, and for distribution networks only c_{i1} is needed which denotes the nodal price at substation *i*; the phase subscript ϕ implies the constraints of distinct phases are considered; $p_i^{g\phi}$and $q_i^{g\phi}$ are the *i*-th generator active and reactive power of the phase ϕ; $p_i^{L\phi}$ and $q_i^{L\phi}$ are the active and reactive power load of the phase ϕ at *i*-th bus; $\underline{p}_i^{g\phi}$, $\bar{p}_i^{g\phi}$, $\underline{q}_i^{g\phi}$ and $\bar{q}_i^{g\phi}$ are the lower and upper limits of the *i*-th generator active and reactive power at the phase ϕ; $\bar{V}_i^{\phi}$ and $\bar{S}_{ij}^{\phi}$ are the upper limits of the *i*-th bus voltage and the branch *i-j* power flow at the phase ϕ. For the real and imaginary parts of bus voltage, x_{2i-1}^{ϕ} and x_{2i}^{ϕ} ($e_i^{\phi}$and f_i^{ϕ}) , the imperative constraints are added in the third inequation of (12).

The formulations (12) can also be applied to the OPF in balanced three-phase power systems only if the superscripts of the notations representing phases are ignored. And the objective function can also be to minimize the overall power loss in distribution networks.

IV. Simulation Analysis

A. Case Selection and Data Sampling

We expected to use real-world data in this research which was, unfortunately, not available at present. As an alternative, the Monte Carlo method was introduced to simulate and generate random data samples of operation measurements, including bus voltage and bus or branch active and reactive power. Different datasets were randomly sampled from diverse power networks, including IEEE 5-, 9-, 57- and 118-bus balanced transmission systems, and IEEE 34-bus unbalanced distribution system [38], [46], [47]. In these data datasets, the active and reactive power loads are randomly changed around their preset values within an interval [0.6, 1.1]. Each dataset contains up to 50000 samples to ensure that there are sufficient samples for training models. Generally, a larger sample set is needed to fit the parameters of a bigger system. It has been observed in [14], [17], and [39], [41], that the required minimum empirical sample size is at least twice or six times the number of buses for balanced or unbalanced power flows.

B. Predictive Performance Comparison

We randomly chose an equal amount of test and training datasets for each case. For instance, both the test and training sets for case 5 and case 9 contain 100 samples, even for each bootstrap in bagging. Next, we fit the DDCQA of power flow through pure polynomial regression (PR), gradient boosting (GB), and bagging. The predictive accuracy is indicated by the average root mean square error (RMSE) of the dependent variable, and the performance of different methods is compared by the test RMSE shown in Table I.

TABLE I
RMSE of All Methods on Different Cases

Case	Method	pure PR		GB		Bagging	
RMSE (10 e-05)		Test	Training	Test	Training	Test	Training
Case 5 (size=100, T=250, BT=50)	P	546.56	37.39	198.18	31.03	367.31	38.42
	Q	5477.60	215.55	1330.46	209.35	2764.26	228.99
	P_{ij}	503.06	47.88	50.85	31.16	286.67	49.96
	Q_{ij}	4453.90	251.92	199.07	88.62	2376.14	246.03
Case 9 (size=100, T=200, BT=50)	P	4.75	1.82	3.12	1.73	4.05	2.01
	Q	3.79	1.42	2.07	1.08	3.25	1.58
	P_{ij}	46.14	39.35	22.78	20.21	43.32	38.57
	Q_{ij}	93.52	77.96	84.89	68.61	86.13	79.30
Case 34 (size=250, T=200, BT=50)	P	1259.5	260.61	883.4	247.08	954.38	247.37
	Q	793.74	185.76	579.29	158.30	617.61	159.59
	P_{ij}	7330.71	2025.70	2143.99	1888.12	5773.90	1973.32
	Q_{ij}	6191.94	1621.43	1733.17	1405.64	4861.96	1527.63
Case 57 (size=200, T=200, BT=50)	P	401.14	3.07 e-10	134.31	0.33	210.74	4.76 e-10
	Q	562.07	5.07 e-10	185.44	0.08	305.51	8.58 e-10
	P_{ij}	11.18	7.53	8.54	5.36	9.39	7.07
	Q_{ij}	26.29	18.28	17.41	14.14	23.57	19.24
Case 118 (size=300, T=140, BT=50)	P	479.20	8.07 e-10	321.36	0.85	423.01	4.95 e-10
	Q	440.60	3.73 e-10	280.41	16.90	389.50	4.82 e-10
	P_{ij}	62.2	0.51	21.12	2.09	57.6	0.53
	Q_{ij}	22.12	1.87	14.89	1.37	20.40	1.91

Note that here T and BT are the maximum numbers of learners and bootstraps in GB and bagging. The unit of data above is 10e-05. From Table I, we have the following ob-servations:

- GB and bagging work better than pure PR on all cases; GB outperforms bagging and pure PR on all cases.
- The training RMSE is smaller than the test RMSE for any dependent variable, even though sometimes they seem to be close, as in case 9 and case 57.

C. Tuning Learning Parameters

Tuning engineering is a necessity in machine learning to display the model performance and the generalization ability described by the training and test RMSE. The following sections focus on the tuning process of GB and bagging, respectively, for the number of learners T and the number of bootstraps BT, and analyzes the generalization ability of ensemble learning techniques.

1) Tuning the Number of Learners T in GB: In order to determine the number of learners that effectively improve the model performance, all results of the active power injections p_i for case 5, 9, 34, 57 and 118 as paradigms are plotted in Figs. 1 –5. Each plot depicts the trend of RMSE and the logarithms of RMSE as the number of learners increases. Note that the logarithms of RMSE are used to enlarge the difference between the training and test RMSE in order to compare them. From these plots, we can observe that:

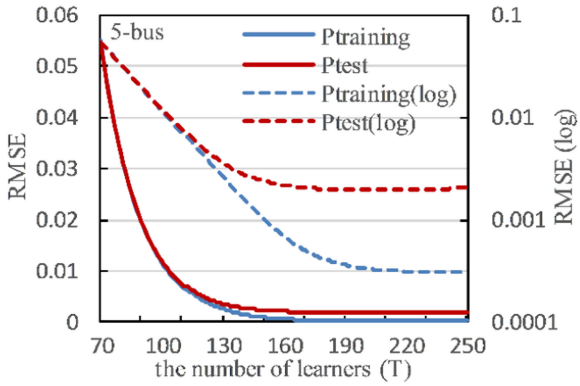

Fig. 1. Case 5: RMSE of P by boosting.

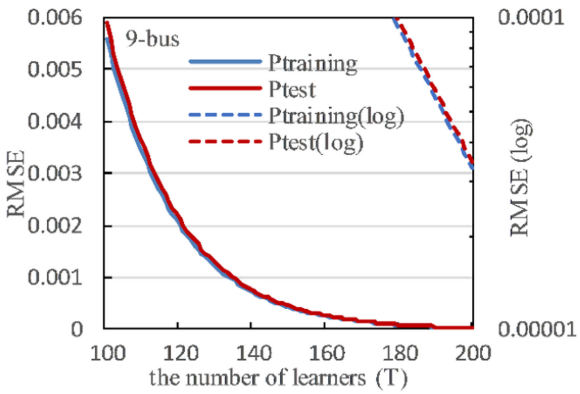

Fig. 2. Case 9: RMSE of P by boosting.

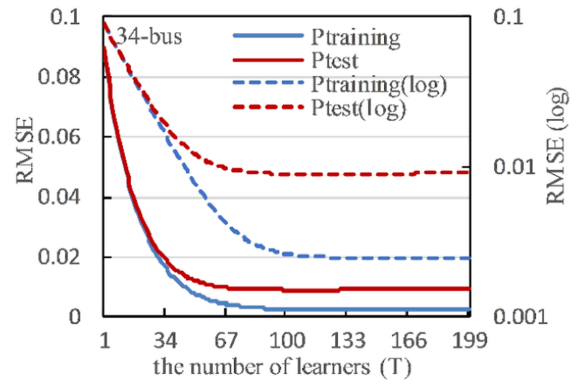

Fig. 3. Case 34: RMSE of P by boosting.

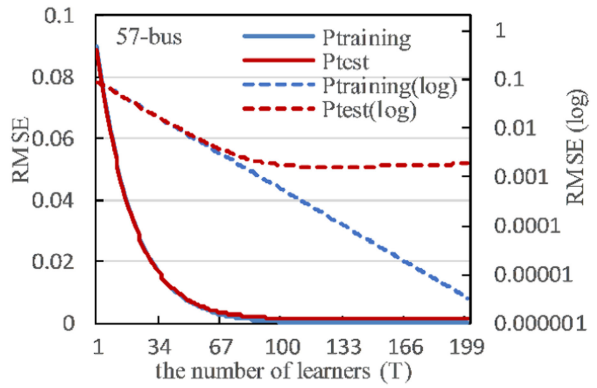

Fig. 4. Case 57: RMSE of P by boosting.

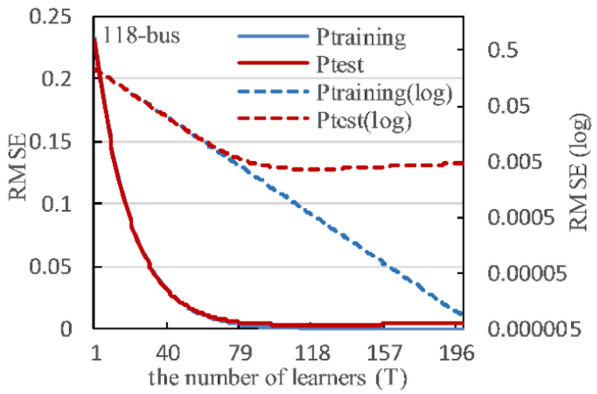

Fig. 5. Case 118: RMSE of P by boosting.

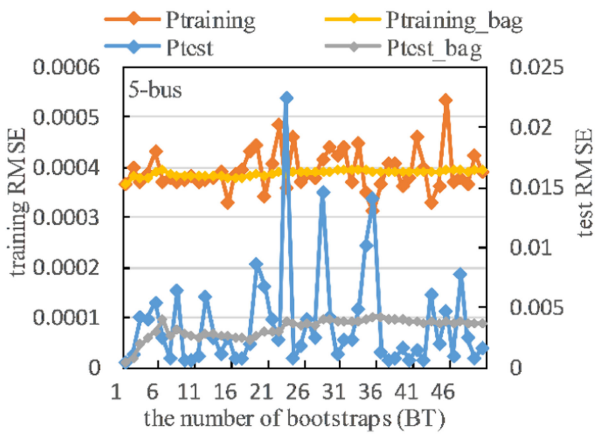

Fig. 6. Case 5: RMSE of P by bagging.

- The training and test RMSE of p_i gradually decrease to be stable with adjusting the number of learners T. Each case reaches the balance point at different numbers of learners. For case 5, the test and training RMSE tend to be constant after 150 learners have been incorporated. For case 57 (or 34) and 118, their test RMSE hardly change when T = 70 and T = 80, respectively, while their training RMSE after T = 100 start to drop slightly, with only 0.0001 or 0.001 per additional learner. Similarly, all RMSE of case 9 drops only 0.0001 with over 180 learners.
- No overfitting is observed as the test RMSE is always larger than the training one. For case 9, though the training and test RMSE seem to be very close in the same order of magnitude, the statement above still holds. Hence, GB has a good generalization ability on these cases.

2) Tuning the Number of Bootstraps in Bagging: In the same fashion, Fig. 6–10 compare the training and test RMSE of p_i on five cases before and after using bagging. Each plot has two axes, the left one for the training RMSE (red and orange lines)

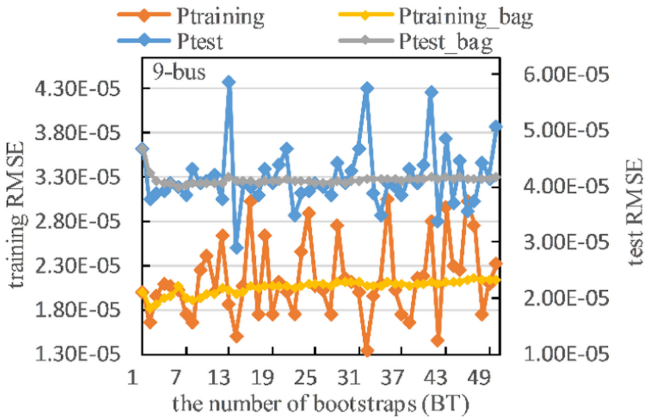

Fig. 7. Case 9: RMSE of P by bagging.

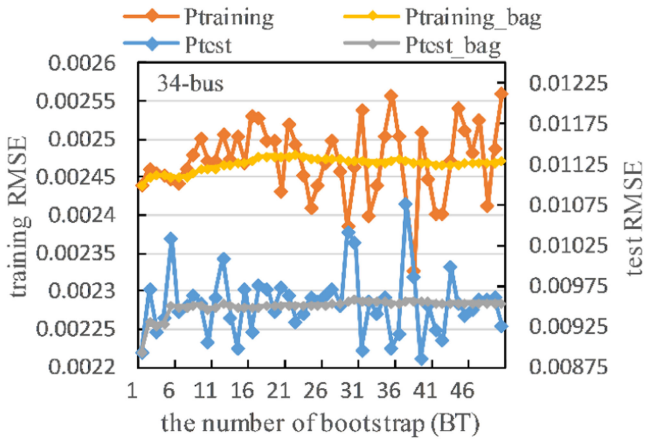

Fig. 8. Case 34: RMSE of P by bagging.

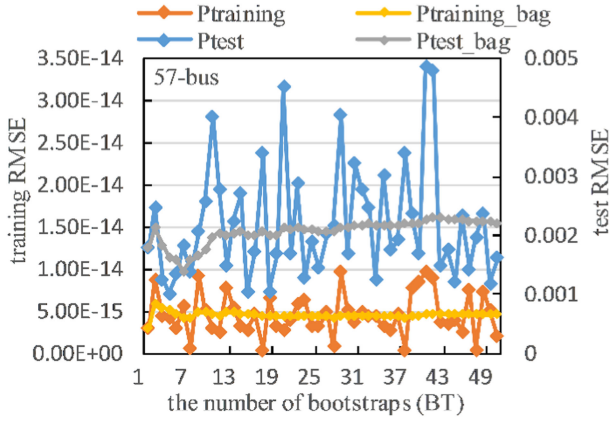

Fig. 9. Case 57: RMSE of P by bagging.

and the right one for the test RMSE (blue and gray lines). The training and test RMSE are marked by red and blue lines for each single bootstrap, and by orange and gray lines for bagging. Based on these plots, we can infer that:

- The training and test RMSE of bagging tend to be flat with slight fluctuations, given enough bootstraps. In case 5 and 34, when BT $\geq$ 30, their training and test RMSE work well. In case 9, when BT $\geq$ 15, the test and training RMSE converge to 4 e-05 and 2 e-05, respectively. Similarly, both case 57 and 118 have steady test and training RMSE after BT $\geq$ 15.
- The result of each single bootstrap distributes stochastically around the orange (gray) line, implying that single learner has its unstable weakness.
- Bagging plays an important role in averaging the variances of all single learners and avoiding overfitting. This also implies a good generalization ability for bagging.

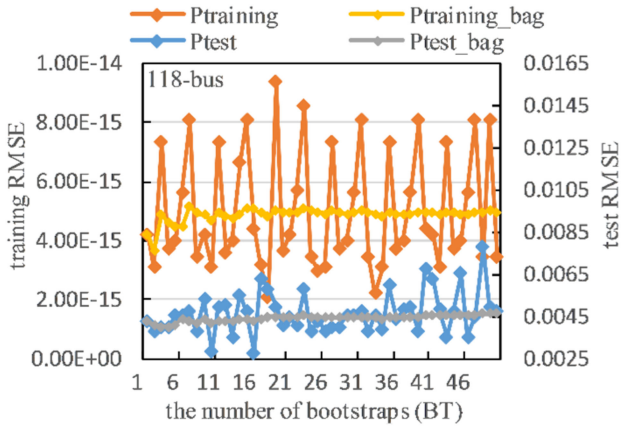

Fig. 10. Case 118: RMSE of P by bagging.

3) Generalization Ability Analysis: As we know, the generalization ability is generally referred to as the ability of how a learning method performs on the unseen (test) data, and it can be quantified by the generalization error or gap. The generalization error is usually represented by the difference between the training and test errors. To some extent, the generalization ability and the overfitting are highly correlated. In another word, the better the generalization ability, the less possibility of overfitting; a good ability of handling overfitting also indicates a good generalization ability, and vice versa [17], [43]. When tuning the parameters of boosting and bagging, both the training and test RMSE eventually become constant, given enough number of learners or bootstraps, shown in Figs. 1–10, exhibiting a good ability of avoiding overfitting. These figures also imply how the generalization error changes and finally converges across the number of learners or bootstraps.

Furthermore, taking the results of case 57 and 118 as examples, the generalization errors represented by the difference between the training and test RMSE for case 57 and 118 are plotted and marked by green lines in Figs. 11–14, respectively. From Figs. 11–14, we can observe that the generalization errors (green lines) will finally converge as the trends of training and test RMSE marked by blue and red lines gradually become flat. Note that when the training error is much smaller than the test error, the pattern of generalization error may be dominated by the test error shown above. Hence, the ensemble learning methods, GB and bagging used in this paper exhibit a good generalization ability.

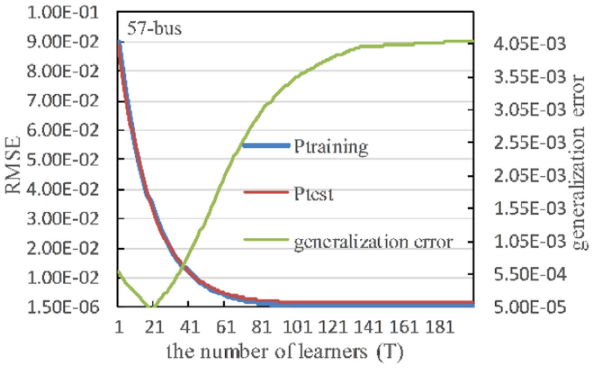

Fig. 11. Case 57: generalization error by boosting.

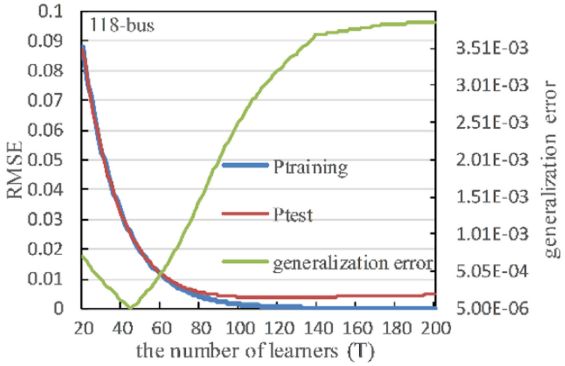

Fig. 12. Case 118: generalization error by boosting.

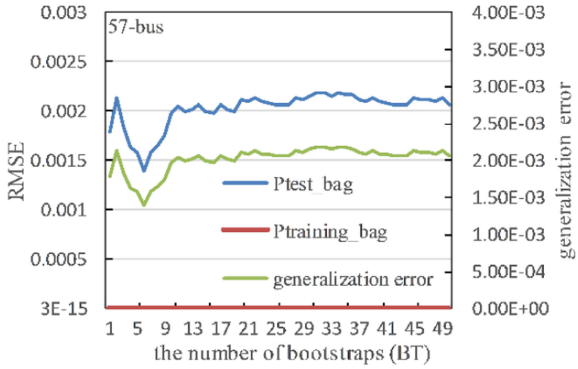

Fig. 13. Case 57: generalization error by bagging.

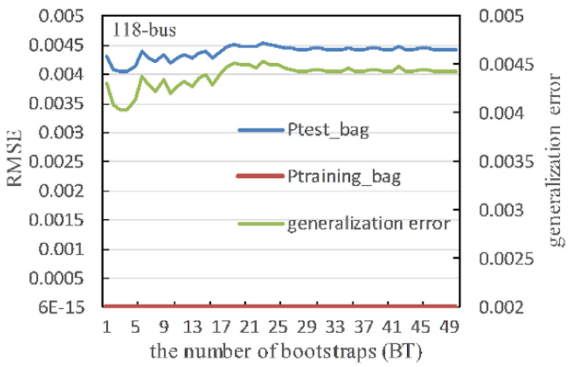

Fig. 14. Case 118: generalization error by bagging.

TABLE II
COMPARING THE OBJECTIVE VALUES OF OPF

Case	Sample Size	ACOPF	SDPOPF	DDCQAOPF
Case 5	100	17551.89 $/hr	16635.78 $/hr	17518.12 $/hr
Case 9	100	5296.69 $/hr	5297.41 $/hr	5296.70 $/hr
Case 34	300	211.30 $/hr	- $/hr	213.96 $/hr
Case 57	200	12100.86 $/hr	10458.06 $/hr	12087.96 $/hr
Case 118	300	129660.70 $/hr	129713.07$/hr	129454.02 $/hr

TABLE III
COMPARING THE OPTIMALITY GAPS

Case	Sample Size	ACOPF	SDPOPF	DDCQAOPF
Case 5	100	0%	5.21%	0.19%
Case 9	100	0%	0.01%	0.00%
Case 34	300	0%	-%	1.26%
Case 57	200	0%	13.60%	0.11%
Case 118	300	0%	0.04%	0.16%

D. Computational Evaluation of Algorithms

According to the analysis above, GB exhibits better fitting outcomes than other methods. Based on the convex models fitted by GB, the data-driven convex quadratic approximation of OPF (DDCQAOPF) is applied to compute the minimum generation cost (unit: $/hr) in all cases. In case 34, there are three sets of battery energy storages installed at buses 820, 824 and 860. The battery set at bus 820 is a single-phase source (capacity: 100kVA) and the battery sets at buses 824 and 860 are three-phase sources (single-phase capacity: 200kVA and 100kVA). The power factor of batteries is set at 0.95. The unit electrical energy costs from the substation and batteries are specified at 0.10, 0.12, 0.15, and 0.13 $/kWh. In this case study, we only consider a snapshot as the multi-period load profiles for IEEE 34-bus unbalanced system are not available. In Table II, the results of DDCQAOPF are compared with the original nonconvex ACOPF and the semidefinite programming relaxation of OPF (SDPOPF) [38] to observe the computational accuracy of algorithms. Assume that the results of ACOPF are set as the benchmarks, and the optimality gap, Err shown in Table III, is defined as

$$Err = \frac{|OV_{acopf} - OV|}{OV_{acopf}} \times 100\%$$

where OV_{acopf} is the objective value of ACOPF and OV is the objective value of SDPOPF or DDCQAOPF. For comparing the computational efficiency of diverse algorithms, the runtime (unit: second) of each algorithm in different cases is given in Table IV. Note that the calculations above for all cases are performed through Matlab, cvx package and Mosek solver.

Tables II–IV reveal that:

- For case 9 and case 118, both SDP relaxation and DDCQA work well, and their objective values are almost the same with ACOPF's.

TABLE IV
COMPARING THE RUNTIMES OF DIFFERENT METHODS

Case	Sample Size	ACOPF	SDPOPF	DDCQAOPF
		Runtime (second)		
Case 5	100	2.72	17.98	3.07
Case 9	100	2.95	19.79	3.23
Case 34	300	1.54	3.03	2.79
Case 57	200	3.43	25.11	7.65
Case 118	300	3.19	36.89	11.57

- Particularly, in some cases (case 5, case 34, case 57) that occur the inexactness of SDP relaxation, DDCQA outperforms SDP relaxation in the computational accuracy.
- A comparison of optimality gaps proves that DDCQA (0%–1.26%) performs more robustly than SDP relaxation (0.01%–13.60%) in the computational accuracy.
- The runtimes indicate that for all cases DDCQA runs more efficiently than SDP relaxation, and its runtimes on case 5 and case 9 are close to ACOPF's.
- DDCQA can be an appropriate alternative of SDP relaxation when it fails to obtain the exact solutions in case 5, case 34 and case 57.

V. CONCLUSION AND FUTURE WORK

This paper develops an ensemble learning based DDCQA for three-phase power flow. Unlike the most linear approximations of power flow that ignore the interaction terms between the bus voltages, the DDCQA retains the interaction terms and model accuracy. The proposed DDCQA of three-phase ACPF model mainly has two advantages. First, it is more computationally effective than the other available convex model—the SDP relaxation. Second, its accuracy is similar to the SDP relaxation in some cases while outperforms the later one in some other cases. More importantly, the DDCQA, which is learning-based model, has high potential in performance-improvement. On the aspect of machine learning, we introduce an emerging technique—ensemble learning algorithms — to improve the predictive performance of DDCQA. Based on these fitted convex models of power flows, a data-driven convex approximation is proposed and compared with conventional SDP relaxation. We expected to test our proposed method in real-life large-scale power systems. Unfortunately, the data of those systems is not publicly available. As empirical alternatives, IEEE standard test systems are applied in this paper, just like the most of previous research publications. The experimental analysis of IEEE standard systems shows that ensemble learning methods work better than the basic learner, gradient boosting yields the best convex model of power flows, and the proposed convex approximation is superior to SDP relaxation in the computing accuracy and efficiency. Especially, in some cases that the solutions of SDP relaxation are not exact, the proposed algorithm can be an advisable alternative. Theoretically, our approach will work for the cases to which the SDP relaxation is applicable, such as the real-life large-scale power systems. Our future work will be extended to more practical applications of DDCQA of power flows in control and optimization problems based on the data from the real-world systems, especially in large-scale systems and complex systems with uncertain intermittent renewable energy resources. Also, theoretical inference on the exactness of DDCQA and its approximation error will be discussed more in detail in the future work.

REFERENCES

[1] A. Imran *et al.*, "A new power system reconfiguration scheme for power loss minimization and voltage profile enhancement using fireworks algorithm," *Int. J. Elec. Power*, vol. 62, no. 9, pp. 312–322, Nov. 2014.

[2] O. Badran *et al.*, "Optimal reconfiguration of distribution system connected with distributed generations: A review of different methodologies," *Renew. Sust. Energ. Rev.*, vol. 73, no. 8, pp. 854–867, Feb. 2017.

[3] Z. Yang, Y. Li, and J. Xiang, "Coordination control strategy for power management of active distribution networks," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5524–5535, Sep. 2019.

[4] A. G. Garrido, A. Saez-de-Ibarra, H. Gaztañaga, A. Milo, and P. Eguia, "Annual optimized bidding and operation strategy in energy and secondary reserve markets for solar plants with storage systems," *IEEE Trans. Power Syst.*, vol. 34, no. 6, pp. 5115–5124, Nov. 2019.

[5] A. Delavari and I. Kamwa, "Sparse and resilient hierarchical direct load control for primary frequency response improvement and inter-area oscillations damping," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5309–5318, Sep. 2018.

[6] A. P. Zhou, M. Yang, Z. Wang, and P. Li, "A linear solution method of generalized robust chance constrained real-time dispatch," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 7313–7316, Nov. 2018.

[7] J. Liu, M. Kazemi, A. Motamedi, H. Zareipour, and J. Rippon, "Security-Constrained optimal scheduling of transmission outages with load curtailment," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 921–931, Jan. 2018.

[8] 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 Trans. Smart Grid*, vol. 9, no. 1, pp. 438–448, Jan. 2018.

[9] M. Chávez-Lugo, C. R. Fuerte-Esquivel, C. A. Cañizares, and V. J. Gutierrez-Martinez, "Practical security boundary-constrained DC optimal power flow for electricity markets," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3358–3368, Sep. 2016.

[10] S. M. Fatemi, S. Abedi, G. B. Gharehpetian, S. H. Hosseinian, and M. Abedi, "Introducing a novel DC power flow method with reactive power considerations," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3012–3023, Nov. 2015.

[11] Z. Yang, H. Zhong, A. Bose, T. Zheng, Q. Xia, and C. Kang, "A linearized oPF model with reactive power and voltage magnitude: A pathway to improve the MW-only DC OPF," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1734 – 1745, Mar. 2017.

[12] J. Yang, N. Zhang, C. Kang, and Q. Xia, "A state-independent linear power flow model with accurate estimation of voltage magnitude," *IEEE Trans. Power Syst.*, vol. 32, no. 5, pp. 3607–3617, Sep. 2017.

[13] Z. Li, J. Yu, and Q. H. Wu, "Approximate linear power flow using logarithmic transform of voltage magnitudes with reactive power and transmission loss consideration," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4593 – 4603, Jul. 2017.

[14] Y. X. 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.

[15] Y. X. Liu, Y. Wang, N. Zhang, D. Lu, and C. Kang, "A data-driven approach to linearize power flow equations considering measurement noise," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2576–2586, May 2020.

[16] Y. Liu, B. Xu, A. Botterud, N. Zhang, and C. Kang, "Bounding regression errors in data-driven power grid Steady-state models," *IEEE Trans. Power Syst.*, vol. 36, no. 2, pp. 1023–1033, May 2021.

[17] Y. C. Chen, A. D. Domínguez-García, and P. W. Sauer, "Measurement-based estimation of linear sensitivity distribution factors and applications," *IEEE Trans. Power Syst.*, vol. 29, no. 3, pp. 1372–1382, May 2014.

[18] Y. C. Chen, J. Wang, A. D. Domínguez-García, and P. W. Sauer, "Measurement-based estimation of the power flow jacobian matrix," *IEEE Trans. Power Syst.*, vol. 7, no. 5, pp. 2507–2515, Sep. 2016.

[19] Y. Yuan, O. Ardakanian, S. Low, and C. Tomlin, "On the inverse power flow problem," 2020, *arXiv:1610.06631*.
[20] V. Kekatos and G. B. Giannakis, "Sparse volterra and Polynomial Regression Models: Recoverability and estimation," *IEEE Trans. Signal Process,* vol. 59, no. 12, pp. 5907–5920, Dec. 2011.
[21] Y. Wi, S. Joo, and K. Song, "Holiday load forecasting using fuzzy polynomial regression with weather feature selection and adjustment," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 596–603, May 2012.
[22] R. Punmiya and S. H. Choe, "Energy theft detection using gradient boosting theft detector with feature engineering-based preprocessing," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 2326–2329, Mar. 2019.
[23] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," The Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.
[24] B. Leo, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, Aug. 1996, pp. 123–140.
[25] E. T. Iorkyase *et al.*, "Improving RF-based partial discharge localization via machine learning ensemble method," *IEEE Trans. Power Del.*, vol. 34, no. 4, pp. 1478–1489, Aug. 2019.
[26] J. Zhang, M. Wu, and V. S. Sheng, "Ensemble learning from crowds," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 8, pp. 1506–1509, Aug. 2018.
[27] N. Yurii and A. Nemirovskii, *Siam Studies Applied mathematics*. Philadelphia, PA, USA: SIAM, 1994.
[28] R. Jabr, "Radial distribution load flow using conic programming," *IEEE Trans. Power Syst.*, vol. 21, no. 3, pp. 1458–1459, Aug. 2006.
[29] X. Bai, H. Wei, K. Fujisawa, and Y. Wang, "Semidefinite programming for optimal power flow problems," *Int. J. Elect. Power Energy Syst.*, vol. 30, no. 67, pp. 383–392, 2008.
[30] Q. Li and V. Vittal. "Non-iterative enhanced SDP relaxations for optimal scheduling of distributed energy storage in distribution systems," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 1721–1732, May 2016.
[31] M. Farivar *et al.*, "Inverter var control for distribution systems with renewables," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, Oct. 2011, pp. 457–462.
[32] Q. Li, R. Ayyanar, and V. Vittal. "Convex optimization for DES planning and operation in radial distribution systems with high penetration of photovoltaic resources," *IEEE Trans. Sust. Energy*, vol. vol. 7, no. 3, pp. 985–995, Jul. 2016.
[33] H. Hijazi *et al.*, "Convex quadratic relaxations of mixed-integer nonlinear programs in power systems," 2013. [Online]. Available: http://www.optimization-online.org/DB_HTML/2013/09/4057
[34] D. Molzahn and I. Hiskens, "Sparsity-exploiting moment-based relaxations of the optimal power flow problem," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3168–3180, Nov. 2015.
[35] Q. Li and V. Vittal, "Convex hull of the quadratic branch AC power flow equations and its application in radial distribution networks," *IEEE Trans. Power Syst.*, vol.vol. 33, no. 1, pp. 839–850, Jan. 2017.
[36] J. Lavaei and S. Low, "Zero duality gap in optimal power flow problem," *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 92–107, Feb. 2012.
[37] S. Low, "Convex relaxation of optimal power flow—Part I: Formulations and equivalence," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 1, pp. 15–27, Mar. 2014.
[38] S. Low, "Convex relaxations and linear approximation for optimal power flow in multiphase radial networks," in *Proc. IEEE Power Syst. Comput. Conf.*, Aug. 2014, pp. 1–9.
[39] C. Coffrin, H. Hijazi, and P. Van Hentenryck, "The QC relaxation: A theoretical and computational study on optimal power flow," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 3008–3018, Jul. 2016.
[40] N. J. Higham, "Computing a nearest symmetric positive semidefinite matrix," *Linear Algebra. Appl.*, vol. 103, no. 6, pp. 103–118, May 1988.
[41] H. Qi and D. Sun, "A quadratically convergent newton method for computing the nearest correlation matrix," *SIAM J. Matrix Anal. Appl.*, vol. 28, no. 2, pp. 360–385, 2006.
[42] R. Borsdorf and N. J. Higham, "A preconditioned newton algorithm for the nearest correlation matrix," *IMA J. Numer. Anal.*, vol. 30, pp. 94–107, 2010.
[43] G. Casella, S. Fienberg, and I. Olkin, *An Introduction to Statistical Learning with Application in R*. New York, NY, USA: Springer, 2013.
[44] P. Kundur, *Power System Stability and Control*. New York, NY, USA: McGraw-Hill Professional, 1994.
[45] K. Purchala, L. Meeus, D. V. Dommelen, and R. Belmans, "Usefulness of DC power flow for active power flow analysis," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, 2005, pp. 454–459.
[46] B. Kocuk, S. S. Dey, and X. A. Sun, "Inexactness of SDP relaxation and valid inequalities for optimal power flow," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 642–651, Jan. 2016.
[47] Z. Y. Wang, D. S. Kirschen, and B. S. Zhang, "Accurate semidefinite programming models for optimal power flow in distribution systems," 2017, *arXiv:1711.07853*.

Ren Hu (Student Member, IEEE) received the M.S. degree in electrical engineering from the South China University of Technology, Guangzhou, China, in 2013. He is currently working toward the Ph.D. degree in electrical engineering with the University of Central Florida, Orlando, FL, USA. His research interests include machine learning and statistical inference applied in power and energy systems modeling, and optimization.

Qifeng Li (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from Arizona State University, Tempe, AZ, in 2016. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, the University of Central Florida (UCF), Orlando, FL, USA. Before joining UCF Faculty, he held a position of Postdoctoral Associate with the Department of Mechanical Engineering, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, from 2016 to 2018. His research interests include convex optimization, uncertainty-aware optimization, and nonlinear systems with applications in power and energy systems.

Feng Qiu (Senior Member) received the Ph.D. degree from the School of Industrial and Systems Engineering, the Georgia Institute of Technology in 2013. He is a Principal Computational Scientist with the Energy Systems Division, Argonne National Laboratory, Argonne, IL, USA. His current research interests include optimization in power system operations, electricity markets, and power grid resilience.