Abstract—Due to the nonlinear and non-convex attributes of the optimization problems in power systems such as Optimal Power Flow (OPF), traditional iterative optimization algorithms require significant amount of time to converge for large electric networks. Therefore, power system operators seek other methods such as DC Optimal Power Flow (DC-OPF) to obtain faster results, to obtain the state of the system. However, DC-OPF provides approximated results, neglecting important features of the system such as voltages and reactive power. Fortunately, recent developments in machine learning have led to new approaches for solving such problems faster, more flexible, and more accurate. In this paper, a Deep Neural Network-based Optimal Power Flow (DNN-OPF) algorithm is implemented on small to large case studies to show the accuracy and efficiency of the ML-based algorithms. The paper provides a novel approach to classify the feasible and infeasible AC-OPF problems, and suggests a constraint-guided method, based on normalizing outputs and using particular activation functions to respect the limits of generators. Furthermore, the proposed post-processing approach guarantees the feasibility of the solutions. The suggested method is applied on IEEE24-bus, IEEE 300 bus, and PEGASE 1354 bus systems and the results show significant improvement on accuracy of the results and execution time, comparing to traditional gradient-based methods, such as Newton-Raphson and Gauss–Seidel methods.

Index Terms—Optimal Power Flow, Optimization, Machine Learning, Deep Neural Networks, AC-OPF.

I. INTRODUCTION

Optimal Power Flow (OPF) is one of the most important problems in power system operation. Independent system operators (ISOs) should run OPF problem to find generation setpoints, considering variations in demand. AC Optimal Power Flow (AC-OPF) was formulated in 1962 [1], as a nonlinear and non-convex optimization problem, considering components and constraints in the power network. Due to difficulties in solving AC-OPF, DC Optimal Power Flow (DC-OPF) was introduced as a linearized model of AC-OPF to simplify the network by assuming the voltage magnitudes as unity and neglecting resistance in transmission lines. Even though DC-OPF is faster than AC-OPF, negligence of some of the network features leads to approximated solutions with lower accuracy. Due to the remaining concerns with current approximation solution methodologies for solving AC-OPF, there is still a need to provide faster, more flexible, and more accurate solution [2].

More importantly, considering the increasing penetration of renewable energy sources, has made it challenging for system operators to maintain balance between system efficiency and reliability and therefore, OPF problems should be executed in a short time to provide the system operators with insights on the status of the power system networks.

Machine learning (ML) algorithms can be utilized to learn the non-linear dependencies in optimization problems to provide optimal or near optimal solutions in a short time [3, 4]. Such method can be used to solve AC-OPF problem generating credible results [5, 6]. Applying ML-based algorithms to solve AC-OPF problems have gained popularity among power system researchers in recent years. The authors in [7] proposed an ML method to solve AC-OPF problems by training the model with the data from the feasible solutions. In [8], the authors provide a framework to get the worst-case guarantees for the proposed DNN, using DC-OPF solution. In [9], a Quasi-Newton based ML method has been introduced to provide faster solutions than MIPS solver. However, the error of the proposed ML method is unacceptable in large case studies, which makes it impractical for real cases. Random forest is implemented as ML method for setting warm-start points for AC-OPF in [10] using the generated data of past AC-OPF runs. It implements a multi-target approach to train voltage and output power of generators only using the system loads without requiring the network parameters. In [11], a new method to model the security and stability of constraints in OPF using neural networks are are introduced which represent the security boundaries of the system. In [12], an introduction and a literature review of the challenges in power system operation and planning is provided.

The current proposed ML-based methods do not guarantee the feasibility of their solutions which is an important criterion to operationalizing such methods. Since ML algorithms produce results, regardless of their feasibility status, there should be an underlying method to investigate the feasibility of the solution, in order to enhance their credibility. Therefore, the contributions of this paper are as follows:

• First, a Deep Neural Network (DNN) algorithm is introduced to classify loads, based on yielding feasible or infeasible solutions. The added classification will support the system operators to acknowledge the feasibility of the current
load configuration before feeding it to Deep Neural Network (DNN) to predict system variables such as generator voltages and active power output.

• Second, another DNN is used to predict the desired output of the OPF problem for each feasible pair of active and reactive power demand. In this stage, a method is suggested to satisfy the constraints on generation and voltage limits within the OPF problem. To satisfy such constraints, a MinMax normalization method is used to normalize the voltage and generation outputs and furthermore, a Sigmoid function is used to generate normalized outputs within 0 and 1. The output will be later converted to pre-normalized values.

• Finally, the proposed DNN for OPF (DNN-OPF) method is benchmarked against optimization-based methods to solve AC-OPF and DC-OPF problem in order to compare its accuracy and speed.

The rest of the paper is organized as follows: In section II, a background of AC-OPF, DC-OPF, and Deep Neural Network (DNN) is introduced. In section III, a detailed explanation of the proposed DNN method for solving AC-OPF problem is provided. In section IV, three case studies are examined to evaluate the performance of the proposed DNN method. Finally, section V will provide conclusions and future directions for this research.

II. DEEP NEURAL NETWORKS FOR OPF

In this section, the formulations of the OPF problems including AC-OPF and DC-OPF are explained. A background on DNN is also provided to illustrate the benefits of its deployment to obtain solutions of OPF problem.

A. AC Optimal Power Flow

AC-OPF problem is used in power system studies to obtain generator voltages and active power output considering the limits of the system. It is formulated as a non-linear optimization problem which is categorised as NP-hard problem. A formulation of AC-OPF is shown in (1)-(6). The objective function (1) of the AC-OPF problem is to minimize the total cost of generation subject to constraints shown in (2)-(6).

The overall power system generation cost is $c(P^G)$ in which the sum of the generation cost at each bus $n$. The active and reactive power balance at each bus is shown in (2) and (3) which $p_n^D$, $q_n^D$, $p_n^G$, $q_n^G$, $v_n$, $G_{i,n}$, $B_{i,n}$, and $\Theta_{i,n}$ are the active demand, active generation, reactive demand, reactive generation, and the voltage at bus $n$, and conductance, susceptance, and the angle difference between bus $i$ and $n$.

The active power limits at each bus are depicted in (4) where $p_{n}^{G\min}$ and $p_{n}^{G\max}$ are the minimum and maximum limits of active power generation at bus $n$. The limits of reactive power at each bus are demonstrated in (5) where $q_{n}^{G\min}$, $q_{n}^{G\max}$ are the minimum limit, maximum limit, and actual value of reactive power generation at bus $n$, respectively. Lastly, (6) indicates the limits of voltages at each bus in which $v_{n}^{G\min}$, $v_{n}^{G\max}$, and $v_{n}^{G}$ are the minimum limit, maximum limit, and actual value of voltage at bus $n$, respectively.

\[
\text{minimize } c(P^G) = \sum_{n=1}^{N} c_n(p_n^G). \quad (1)
\]

s.t.: \[
\sum_{n=1}^{N} V_i V_n (G_{i,n} \cos \Theta_{i,n} + B_{i,n} \sin \Theta_{i,n}) = P_n^D - P_n^G. \quad (2)
\]

\[
\sum_{i=1}^{N} V_i V_n (G_{i,n} \sin \Theta_{i,n} - B_{i,n} \cos \Theta_{i,n}) = Q_n^D - Q_n^G. \quad (3)
\]

\[
p_{n}^{G\min} \leq p_n^G \leq p_{n}^{G\max}. \quad (4)
\]

\[
q_{n}^{G\min} \leq q_n^G \leq q_{n}^{G\max}. \quad (5)
\]

\[
v_{n}^{G\min} \leq V_n \leq v_{n}^{G\max}. \quad (6)
\]

B. DC Optimal Power Flow

DC-OPF provides linearized formulation and approximated solution to AC-OPF problem, and is widely used in practice. This linearized formulation neglects some important features of the network, as it assumes the voltages are at unity at each bus, neglects the reactive power and line resistance in the system, and considers only the reactance of the lines. Therefore, DC-OPF does not consider line losses in calculations. As a result of the underlying assumptions, the linearized DC-OPF is formulated as shown in (7)-(9) which are the objective function, active power balance, and active power limits, respectively.

\[
\text{minimize } c(P^G) = \sum_{n=1}^{N} c_n(p_n^G). \quad (7)
\]

subject to: \[
\sum_{i=1}^{N} B_{i,n} \theta_{i,n} = p_n^D - p_n^G. \quad (8)
\]

\[
p_{n}^{G\min} \leq p_n^G \leq p_{n}^{G\max}. \quad (9)
\]

C. Deep Neural Network

A DNN is a form of Artificial Neural Network (ANN) with more that one hidden layer. The number of layers depends on the complexities associated with the model. For instance, in this paper, a DNN with three hidden layers is implemented. The suggested models, both classification and prediction models, as shown in Fig 1 and Fig 2 consist of neurons and weights, $W_i$, derived from the activation functions, $f_t$ in each layer $i$, resulting the predictions in the output layer, $y$, as shown in (10). Note that the output of each individual layer is calculated by the product of the weights and the values from input layers.

\[
y = f_4(f_3(f_2(f_1(x,W_1),W_2),W_3),W_4). \quad (10)
\]
Fig. 1. DNN implemented for OPF feasibility classifier.

Fig. 2. DNN implemented for OPF solution prediction.

Such mapping of the input to the output layers is a form of matrix multiplication which can be done much faster than traditional iterative methods. Therefore, it is apparent that DNN solution method could be utilized to facilitate several runs of complex models for sensitivity analysis in a short time.

The Universal Approximation Theorem in DNN proves that DNN models could approximate any function to the desired accuracy after tuning the models by adjusting the number of neurons and the activation functions \[13,14\]. The goal of this paper is to leverage this aspect of DNN to estimate the output of OPF problem by training the models using labeled datasets, achieving faster and more accurate results than DC-OPF. It is important to note that the output resulted from DNN is not explainable and requires further studies to obtain optimal solutions. Therefore, another goal of this research is to improve the accuracy of the output and provide insights on the distance of the output from optimality \[15\].

III. METHODOLOGY

In order to achieve the goals of this research, outlined in section II, two fully connected neural networks are trained by datasets, constructed from executing AC-OPF simulations. The suggested approach benefits from two pre-processing and post-processing steps in order to achieve better results and guarantee the feasibility of solutions. As shown in Fig 3, \(P^L\) and \(Q^L\) are fed into a feasibility classifier and if the results are determined to be feasible, they are fed into a prediction algorithm to produce \(V^G\) and \(P^G\). In the post processing approach, the resulted values are studied for feasibility in power flow (PF) equations. The input layer in the provided DNN contains active and reactive loads, \(P^L\) and \(Q^L\), and the output vector contains voltage and active power generation, \(V^G\) and \(P^G\). The simulated dataset is split into three sets: training, validation, and testing. The training dataset is implemented to train DNN by calculating the weight of each neuron using back-propagation, while minimizing the loss function. Assuming the number of buses and generators to be \(N\) and \(G\), respectively, meaning that the input layer has \(2N\) nodes, indicating active and reactive loads at each bus. After manual tuning, using trial and error, it is found that three hidden layers can the most accurate results. The size and activation functions of the implemented DNN algorithm is depicted in Table I, in which characteristics of the two models, OPF feasibility and OPF solution prediction are shown.

<table>
<thead>
<tr>
<th>Input Layer</th>
<th>2N nodes</th>
<th>2N nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st hidden layer</td>
<td>6N nodes, Relu</td>
<td>6N nodes, Relu</td>
</tr>
<tr>
<td>2nd hidden layer</td>
<td>6N nodes, Relu</td>
<td>6N nodes, Relu</td>
</tr>
<tr>
<td>3rd hidden layer</td>
<td>4 nodes, Sigmoid</td>
<td>(2(G)), Sigmoid</td>
</tr>
<tr>
<td>Output layer</td>
<td>2 node, Sigmoid</td>
<td>(G) nodes, Sigmoid</td>
</tr>
</tbody>
</table>

MAE (Mean Absolute Error) is implemented as the loss function in this paper which is depicted in (11), where \(m\), \(M\), \(X_m\), \(\hat{X}_m\) are output neuron index, number of output neurons, predicted output neuron, and actual output neuron, respectively.

\[
MAE = \frac{\sum_{m=1}^{M} |X_m - \hat{X}_m|}{M}
\] (11)

In the following sub-sections, the pre-processing and post-processing steps are explained.

A. Pre-processing

After generating the data from AC-OPF simulations, pre-processing is required to convert the data to the desired format, to increase the accuracy of the results obtained from DNN. As the implemented data in this paper are simulated using traditional optimization models, it is assumed that the collected data are exact solutions.

In order to restrict the voltage and generated power remain within the required limits, a MinMax normalization method is
utilized to scale the voltage and output power values between 0 and 1. The details of the pre-processing method are displayed in Fig 4. As shown in (12) and (13), $V_n^g$, $V_n^{\hat{g}}$, $V_n^g$, $V_n^{\hat{g}}$ are the actual voltage, maximum voltage limit, minimum voltage limit, and the normalized voltage of generator at bus n. Also, $p_n^g$, $p_n^{\hat{g}}$, $p_n^g$, $p_n^{\hat{g}}$ are the actual output power, maximum output power limit, minimum output power limit, and the normalized output power of generator n.

$$
\hat{V}_n^g = \frac{V_n^g - V_n^{\hat{g}}}{V_n^g - V_n^{\hat{g}}}. \quad (12)
$$

$$
\hat{p}_n^g = \frac{p_n^g - p_n^{\hat{g}}}{p_n^g - p_n^{\hat{g}}}. \quad (13)
$$

B. Post-processing

After training the described DNN, a post-processing approach is implemented to check for the violations of the network constraints. In this step, the optimal voltage and output power values from DNN are fed into Power Flow equations using PF function in MATPOWER to investigate whether the obtained values are feasible for OPF problem. This extra steps reassures the system operators that the obtained output is feasible for the power system.

C. Case studies and data collection

The data required for the suggested method is generated using MATPOWER which is a Matlab-based open source program used to run PF and OPF, used by power system researchers to implement power system planning and operation. Three test systems, IEEE-24, IEEE-300, and PEGASE 1354 bus are used as case studies to collect data.

In order to create different scenarios for each case, the base parameters are implemented, then a random uniform value, between 0 and 1.4 is chosen for each load to be multiplied by the load. For example, the active and reactive load of bus 3 in 1354 bus case is 151 MW and 48.8 MVA. Then, a random value, e.g. 0.6 is generated to create a different load value at bus 3 for a new scenario leading to 90.6 MW and 29.28 MVA. In each scenario, a different random value is generated for each bus to create various set of loads. Note that the power factor of each load in each scenario is the same as the base.

Since the suggested approach needs two datasets and for feasibility classifier and another one for predicting network values. In order to generate data for the feasibility classifier, after generating the subset of loads, the feasibility of AC-OPF for each scenario is checked using MATPOWER, through which 2500 feasible and 2500 infeasible scenarios are generated for IEEE 24 and 300 bus system (overall 5000 scenarios). Then, this dataset is labeled as 0 for infeasible and 1 for feasible scenarios. This dataset will be used for training DNN to detect feasible and infeasible cases.

The second dataset is used to predict the results of OPF using DNN approach. To generate data, the same approach as explained for the first dataset is implemented, while only the feasible scenarios are considered. For the feasible scenarios, the voltage and output active power of the generators are collected as the output of AC-OPF. Also, DC-OPF results of the same scenarios are collected to compare with DNN. Two set of 5000 and 50000 feasible data is generated to be used in the proposed DNN method.

D. Advantages of DNN-OPF

Power system operators use DC-OPF approximation as the solution to OPF problems, as it converges faster than AC-OPF formulation. Since the power system operators need to have an estimate of the solution of the OPF problem in a short time, solving OPF problem using DNN-OPF method could provide more accurate solutions in a very short time.

The solution of DNN-OPF considers the line losses, voltages, and generators’ reactive power, which are missing in the DC-OPF formulations.
The fast and accurate solutions obtained from DNN-OPF could also support uncertainty analysis studies, such as scenario based uncertainty planning. Since this method allows the decision makers to consider the possible outcomes and their occurring probability, including the extreme conditions, the system operators can modify an uncertain parameter numerous times to study the system results for both planning and operational applications \[16\] [17].

IV. Numerical Analysis

A. Case studies description

Three case studies (IEEE 24 bus, IEEE 300 bus, and PEGASE 1354 bus system) are implemented in this paper to study the performance and accuracy of the proposed DNN-OPF. The number of buses, number of generators, and average loads in MW are shown in Table III.

<table>
<thead>
<tr>
<th>Case study</th>
<th>No. of buses</th>
<th>No. of Gen</th>
<th>Ave loads (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 24 bus</td>
<td>24</td>
<td>33</td>
<td>1,995</td>
</tr>
<tr>
<td>IEEE 300 bus</td>
<td>300</td>
<td>69</td>
<td>16,467</td>
</tr>
<tr>
<td>PEGASE 1354 bus</td>
<td>1354</td>
<td>260</td>
<td>51,141</td>
</tr>
</tbody>
</table>

B. Generators’ cost function

The cost function of each generator, \(c_n(p_n^g)\), is a quadratic function, shown in (14), in which generation cost at each bus \(c_n(p_n^g)\) is a quadratic function of \(p_n^g\) where \(c_1, c_2,\) and \(c_3\) are the coefficients of the generator cost function and \(p_n^g\) is active power output of generator \(n\).

\[
c_n(p_n^g) = c_{1n} + c_{2n} \times p_n^g + c_{3n} \times p_n^g^2 .
\] (14)

C. Feasibility Classifier

This algorithm allows the user to predict whether the current load combination will lead to a feasible or infeasible solution. The accuracy of the feasibility classifier is denoted as the ratio of correctly classified test cases relative to all cases. Table III shows the accuracy of classification method on of IEEE 24 and 300 bus systems. It can be seen that the accuracy for IEEE 24 bus, IEEE 300 bus system, and PEGASE 1354 is 98.8%, 94.0%, and 89.4% respectively.

<table>
<thead>
<tr>
<th>Case study</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 24 bus</td>
<td>98.8</td>
</tr>
<tr>
<td>IEEE 300 bus</td>
<td>94.0</td>
</tr>
<tr>
<td>PEGASE 1354 bus</td>
<td>89.4</td>
</tr>
</tbody>
</table>

D. Prediction Error and Distance to Optimality

Five type of measurements errors considered in this paper to compare the solutions of DNN-OPF to AC-OPF, which is MAE \[15\] and \[16\] that is implemented for voltage and power error as:

\[
V_{MAE} = \frac{\sum_{g=1}^{G} |V_g - \hat{V}_g|}{G} .
\] (15)

\[
P_{MAE} = \frac{\sum_{g=1}^{G} |p_g - \hat{p}_g|}{G} .
\] (16)

where \(V_g, \hat{V}_g, \) \(P_g, \) \(\hat{P}_g,\) and \(G\) are the voltages and powers obtained from DNN-OPF and AC-OPF at bus \(g\) and the number of generators, respectively.

The second error estimator is MAPE (Mean Absolute Percentage Error) shown in \[17\] which is implemented to calculate the voltage only.

\[
V_{MAPE} = \frac{1}{G} \sum_{g=1}^{G} \left| \frac{V_g - \hat{V}_g}{V_g} \right| \times 100 .
\] (17)

The main reason for choosing MAPE for voltage is that the voltage value is between 0.95 and 1.05. The difference of the predicted value and real value can be divided by the real value, but this is not the case for the output power, as the output power can be zero in some cases, leading to the denominator to be zero. Instead, RAE is used for active power error measurement.

In order to depict the active power error, another error metric called Relative Absolute Error (RAE) \[18\] is introduced as follows:

\[
P_{RAE} = \frac{\sum_{g=1}^{G} |p_g - \hat{p}_g|}{\sum_{g=1}^{G} |\hat{p}_g|} \times 100 .
\] (18)

The cost function error is defined as the total deviation of the objective function values from the one in AC-OPF, as shown in \[19\]:

\[
C_E = \left| \sum_{g=1}^{G} c_g - \hat{c}_g \right| \times 100 .
\] (19)

E. Comparison of DC-OPF and AC-OPF results

As depicted in Table IV the error of DC-OPF solver is calculated for voltage, output power, and generation cost for different case studies. The voltage error using MAPE index is 2.864% and 2.745% for IEEE 24 and 300 bus system, respectively. The error is relatively large as DC-OPF assumes that the voltages are set to 1 p.u. in all buses, while it can be any value between 0.95 and 1.05 p.u. The active power generation error, resulted from DC-OPF formulation error in MAE are 1.958, 28.85, 100.625 MW for IEEE 24, IEEE 300, and PEGASE 1354 bus system, respectively. As shown, MAE error of larger networks are significantly larger than smaller networks. The RAE errors of the active power are...
TABLE IV

<table>
<thead>
<tr>
<th>Case study</th>
<th>V (MAPE %)</th>
<th>V(MAE p.u.)</th>
<th>P(RAE% )</th>
<th>P (MAE MW)</th>
<th>C (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 24 bus</td>
<td>2.864</td>
<td>0.02949</td>
<td>0.09499</td>
<td>1.958</td>
<td>1.4515</td>
</tr>
<tr>
<td>IEEE 300 bus</td>
<td>2.745</td>
<td>0.02818</td>
<td>0.12228</td>
<td>28.85</td>
<td>2.1637</td>
</tr>
<tr>
<td>PEGASE 1354 bus</td>
<td>2.404</td>
<td>0.02461</td>
<td>0.19560</td>
<td>100.625</td>
<td>1.468</td>
</tr>
</tbody>
</table>

TABLE V

<table>
<thead>
<tr>
<th>Case study</th>
<th>V (MAPE %)</th>
<th>V(MAE p.u.)</th>
<th>P(RAE% )</th>
<th>P (MAE MW)</th>
<th>C (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 24 bus</td>
<td>0.147</td>
<td>0.00148</td>
<td>0.01365</td>
<td>0.251</td>
<td>0.186</td>
</tr>
<tr>
<td>IEEE 300 bus</td>
<td>0.552</td>
<td>0.00557</td>
<td>0.13513</td>
<td>22.422</td>
<td>7.241</td>
</tr>
<tr>
<td>PEGASE 1354 bus</td>
<td>0.501</td>
<td>0.00529</td>
<td>0.03462</td>
<td>17.697</td>
<td>0.08994</td>
</tr>
</tbody>
</table>

TABLE VI

<table>
<thead>
<tr>
<th>Case study</th>
<th>V (MAPE % correction)</th>
<th>P (MAE correction)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 24 bus</td>
<td>95%</td>
<td>87%</td>
</tr>
<tr>
<td>IEEE 300 bus</td>
<td>80%</td>
<td>21%</td>
</tr>
<tr>
<td>PEGASE 1354 bus</td>
<td>81%</td>
<td>82%</td>
</tr>
</tbody>
</table>

TABLE VII

<table>
<thead>
<tr>
<th>Case study</th>
<th>Run-time(s) 1 scenario</th>
<th>Run-time(s) 100 scenarios</th>
<th>Run-time(s) 10000 scenarios</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN</td>
<td>Matpower</td>
<td>RF</td>
<td>DNN</td>
</tr>
<tr>
<td>IEEE 24 bus</td>
<td>0.085</td>
<td>0.148</td>
<td>1.74</td>
</tr>
<tr>
<td>IEEE 300 bus</td>
<td>0.099</td>
<td>0.275</td>
<td>2.77</td>
</tr>
<tr>
<td>PEGASE 1354 bus</td>
<td>0.124</td>
<td>2.863</td>
<td>23.08</td>
</tr>
</tbody>
</table>

0.09499, 0.12228, and 0.19560% for IEEE 24, IEEE 300, and PEGASE 1354 network, respectively, which can provide a better comparison point, as it is independent of the size of the system. As the size of the network grows, RAE error tend to increase in general. Finally, the cost error of DC-OPF are 1.4515, 2.1637, and 1.468% for 24, 300, and 1354 bus system.

F. Comparison of DNN-OPF and AC-OPF results

Table VI indicates the errors obtained from the solution of DNN-OPF compared to AC-OPF. MAPE voltage error for IEEE 24, IEEE 300, and PEGASE 1354 bus networks are 0.147, 0.552, and 0.501%, respectively, while MAE error of voltage are 0.00148, 0.00557, and 0.00529 (p.u.). MAE errors of active power are 0.251, 22.762, and 17.697 MW for IEEE 24, IEEE 300, and PEGASE 1354, respectively. RAE error of active power are 0.01365, 0.13513, and 0.03462 for 24, 300, and 1354 bus system, respectively. The cost errors are 0.186, 7.241, and 0.08994% are for 24, 300, and 1354 bus system, respectively.

Comparing the errors of DC-OPF and DNN with respect to AC-OPF results, demonstrate that DNN leads to more accurate results than DC-OPF. For voltage estimation, DNN-OPF method reduces MAPE error by 95%, 80%, and 81% comparing to DC-OPF IEEE 24, IEEE 300, and PEGASE 1354, respectively. For active power prediction, DNN has less error in MAE measurement as the error correction in IEEE 24, IEEE 300, and PEGASE 1354 bus system are 87%, 21%, and 82%, respectively, as shown in Table VI.

G. Time Comparison

The time needed to reach AC-OPF solution using the proposed model and Matpower AC-OPF solver for a a test case is denoted by $\tau_{NN}$ and $\tau_{AC}$, respectively. To compare the computational run-time of two models, a Ratio Factor (RF) is defined as follows:

$$RF = \frac{\tau_{AC}}{\tau_{NN}}$$

Table VII shows the run-time of DNN-OPF solution and Matpower AC-OPF solver for different case studies. As shown, the obtained AC-OPF results are faster than Newton-Raphson method. IEEE 24 bus, IEEE 300 bus, and PEGASE 1354 bus systems have RF of 1.74, 2.77, and 23.08, respectively. RF grows larger as the number of buses in the network grows, which makes the proposed model very effective in complex and more realistic networks.

In order to depict the performance of DNN-OPF for simulation In table VII the run-time of different case studies is depicted for 1, 100, and 10000 scenarios. It is shown in Table VII that RF increases as the number of scenarios grow.
RF for 10000 scenarios are 831.9, 613.3, and 436.7 for IEEE 24 bus, IEEE 300 bus, and PEGASE 1354, respectively.

V. CONCLUSION AND FUTURE WORK

In this paper, a novel ML-based DNN algorithm was introduced to estimate the solution of AC-OPF with high accuracy and fast execution time. Two DNN models were described in which one was used to classify the feasible and infeasible results and another one was to predict the solution of AC-OPF for feasible cases. The results of the proposed model method provide faster execution time and high accuracy. Therefore, the proposed model was shown to be a perfect method replacement for widely-used DC-OPF. Moreover, it was shown that the accuracy of DNN-OPF was very close to the solution of direct AC-OPF method.

This work is planned to be expanded by deploying methods to guarantee the optimality of the solution. Also, other feedback methods will incorporated to reduce the size of the training set while producing robust solutions.

VI. ACKNOWLEDGEMENT

We acknowledge the financial support for this research which is provided by Natural Sciences and Engineering Research Council of Canada (NSERC) and Ontario Center for Innovation (OCI) to Dr. Mehrdad Pirnia.

REFERENCES


22nd Power Systems Computation Conference

Porto, Portugal — June 27 – July 1, 2022

PSCC 2022