

# Learning-based AC-OPF Solvers on Realistic Network and Realistic Loads

Tsun Ho Aaron Cheung

School of Data Science, City University of Hong Kong, HKSAR, China

Email: aaronthcheung@gmail.com

**Abstract**—Deep learning approaches for the Alternating Current-Optimal Power Flow (AC-OPF) problem are under active research in recent years. A common shortcoming in this area of research is the lack of a dataset that includes both a realistic power network topology and the corresponding realistic loads. To address this issue, we construct an AC-OPF formulation-ready dataset called TAS-97 that contains realistic network information and realistic bus loads from Tasmania’s electricity network. We found that the realistic loads in Tasmania are correlated between buses and they show signs of an underlying multivariate normal distribution. Feasibility-optimized end-to-end deep neural network models are trained and tested on the constructed dataset. Trained on samples with bus loads generated from a fitted multivariate normal distribution, our learning-based AC-OPF solver achieves 0.13% cost optimality gap, 99.73% feasibility rate, and 38.62 times of speedup on realistic testing samples when compared to PYPOWER.

**Index Terms**—AC-OPF, realistic dataset, deep learning, feasibility optimization

## I. INTRODUCTION

The Alternating Current-Optimal Power Flow (AC-OPF) problem aims to minimize the generator dispatch costs while meeting the load demands in an AC power network and satisfying physical and network constraints. It is a problem faced daily by power companies that, if solved reliably in near real-time, could result in tremendous savings in operation costs. The non-convexity of this optimization problem prevents any quick solutions using conventional optimization methods. Despite the difficulty, researchers and industry practitioners have been actively searching for better solutions to the problem in the last decades. Recently, deep learning methods have shown promising results to solve AC-OPF problems with their capability to quickly map network loads to dispatch solutions. Several literatures have shown neural networks can produce similar solutions to conventional solvers in a fraction of the conventional computation time. Most literatures perform numerical experiments on an existing power network. However, the network loads used for testing in their experiments are sampled from estimated probability distributions which could be a poor representation of real network loads. Whether learning-based solvers are capable of real network topology and its corresponding real network loads remains a question, of which a solid answer is needed for the industry to better evaluate their effectiveness.

We summarize our main contributions in the following. First, we integrate multiple data sources into one coherent realistic dataset that contains all the required information for

AC-OPF formulation. The constructed dataset is an AC-OPF formulation-ready dataset that represents a realistic network topology and includes months of the corresponding realistic bus load data with a short sample time interval. Second, we perform numerical experiments to show that given training samples that have a close resemblance to the realistic testing samples, learning-based AC-OPF solvers can achieve a small optimality gap, high feasibility rate, and two orders of magnitude faster than a conventional AC-OPF solver.

## II. THE AC-OPF FORMULATION

Define the following symbols:

Symbol	Definition
$\mathcal{N}$	Set of buses
$\mathcal{G}$	Set of P-V buses
$\mathcal{D}$	Set of P-Q buses
$\mathcal{E}$	Set of branches
$P_{Gi}$	Active power generation on bus $i$ , $i \in \mathcal{N}$
$P_{Di}$	Active power load on bus $i$ , $i \in \mathcal{N}$
$Q_{Gi}$	Reactive power generation on bus $i$ , $i \in \mathcal{N}$
$Q_{Di}$	Reactive power load on bus $i$ , $i \in \mathcal{N}$
$V_i$	Complex voltage on bus $i$ , $i \in \mathcal{N}$
$X^{\min}$	Minimum operation limit of quantity $X$
$X^{\max}$	Maximum operation limit of quantity $X$
$S_{ij}$	Apparent power on branch $(i, j) \in \mathcal{E}$
$y_{ij}$	Complex admittance on branch $(i, j) \in \mathcal{E}$
$\text{Re}\{z\}$	Real part of complex number $z$
$\text{Im}\{z\}$	Imaginary part of complex number $z$
$z^*$	Conjugate of complex number $z$

The AC-OPF problem can be formulated as follows [1]:

$$\min \sum_{i \in \mathcal{N}} C_i(P_{Gi}), \quad (1)$$

$$\text{s.t.} \quad \sum_{(i,j) \in \mathcal{E}} \text{Re}\{V_i(V_i^* - V_j^*)\} = P_{Gi} - P_{Di}, \quad (2)$$

$$\sum_{(i,j) \in \mathcal{E}} \text{Im}\{V_i(V_i^* - V_j^*)\} = Q_{Gi} - Q_{Di}, \quad (3)$$

$$P_{Gi}^{\min} \leq P_{Gi} \leq P_{Gi}^{\max}, i \in \mathcal{N}, \quad (4)$$

$$Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max}, i \in \mathcal{N}, \quad (5)$$

$$|V_i|^{\min} \leq |V_i| \leq |V_i|^{\max}, i \in \mathcal{N}, \quad (6)$$

$$|V_i(V_i^* - V_j^*)y_{ij}^*| \leq S_{ij}^{\max}, (i, j) \in \mathcal{E}, \quad (7)$$

$$\text{var.} \quad P_{Gi}, Q_{Gi}, V_i, i \in \mathcal{N}.$$

Equation (1) is the total generation cost that we want to minimize. Equations (2) and (3) are the power balance

equations. Equations (4) to (7) are the operation constraints of the buses and transmission lines. This optimization problem is nonconvex because Equations (2), (3) and (7) are nonconvex constraints.

### III. LITERATURE REVIEW

#### A. Research of learning-based AC-OPF solvers

[2] categorizes learning-based AC-OPF solvers into either end-to-end models where mapping to OPF solutions are directly learned, or hybrid models where the neural network outputs are used as inputs to a conventional optimization solver. Fig. 1 shows the flowchart of the two models. According to its survey, end-to-end models have relatively short prediction time but high constraint violation rates, while hybrid models have relatively long prediction time but low constraint violation rates.

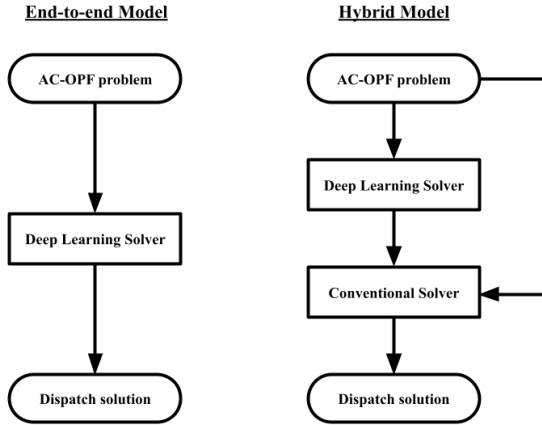


Fig. 1. Flowchart of end-to-end model and hybrid model

Works from [1]–[5] adopt the end-to-end approach. [1] tackles the feasibility drawback of end-to-end models by introducing constraint violation loss to the objective function in training. [2] attempts to leverage local information from the power grid topology by adopting a learning-based model with a graph neural network. [3] improves the scalability of learning-based AC-OPF solvers by reducing the problem size through load embedding. [4] proposes a physics-informed neural network that is capable of training on collocation points without traditional OPF ground truths. [5] improves the scalability of learning-based AC-OPF solvers by dividing a power network into islands and using a smaller neural network on each island.

Works from [6]–[8] adopt the hybrid approach. [6] improves the practicality of learning-based AC-OPF solvers in a safety-critical setting by developing a neural network that can classify the safeness of dispatch setpoints. [7] uses a 1-D convolutional neural network to output the voltage and phase angle as “hot-start” conditions for a conventional AC-OPF solver. [8] classifies the active set of constraints at optimality with which simple manipulations give the DC-OPF solutions.

#### B. Research Limitation

To facilitate discussion in this paper, we define the following terms. Realistic networks are networks that are a representation of an existing power network. Realistic loads are historical loads in a realistic network. Realistic samples are samples that contain realistic loads. Synthetic loads are bus loads that are generated by sampling from estimated distributions. Synthetic samples are samples that contain synthetic loads.

[1]–[8] all perform numerical experiments on realistic networks. However, the testing dataset used in their experiment are synthetic samples. [1]–[3], [5] use uniform distribution with mean set as one reference load data to generate synthetic loads. [4], [6] use the latin hypercube sampling method [9] for synthetic sampling while [7], [8] use the normal distribution.

Regardless of the distribution family used in generating the synthetic samples, [1]–[4], [6]–[8] share the same two limitations. First, they estimate the distribution parameters from a limited sample size, therefore the distribution parameter estimates may be inaccurate. Second, they have assumed the distributions of bus loads are independent of each other, which is not a necessarily true property of realistic loads. [5] removes the assumption of independent bus loads by including shared geographical and weather factors among nearby buses. Nonetheless, it still estimates the mean parameter of the uniform distribution for bus load from only one reference bus load data.

Because of the above limitations, the testing performance from [1]–[8] on synthetic testing dataset may not reflect the model performance on realistic samples. Our work directly uses a realistic testing dataset to avoid these limitations. Table I shows a comparison between different works.

TABLE I  
SUMMARY OF LITERATURE REVIEW

Literature	Realistic network	Testing samples	Distribution estimated from > 20 samples	Bus load correlation
[1]–[4]				
[6]–[8]	✓	synthetic	✗	✗
[5]	✓	synthetic	✗	✓
This work	✓	realistic	-	-

### IV. DATA SOURCES FOR DATASET CONSTRUCTION

#### A. Realistic loads

Historical substation loads from Australia’s National Electricity Market (NEM) 960-bus power network is publicly available [10]–[21]. The NEM network interconnects five regional electricity market jurisdictions (NEM regions) in Australia – Queensland, New South Wales (including the Australian Capital Territory), Victoria, South Australia, and Tasmania. Table II shows a summary of the available load data in the NEM region.

Only 7 out of 12 distribution companies in the NEM region have provided both active and reactive substation loads. There is insufficient real load data for AC-OPF study for the NEM

TABLE II  
NEM LOAD DATA SUMMARY

NEM Region	Company Name	Load Type		Period	
		Active	Reactive	From	To*
NSW	Ausgrid	✓	✗	2005	2021
NSW	Endeavour	✓	✗	2010	2020
NSW	Essential Energy	✓	✗	2005	2018
NSW	Evoenergy	✓	✗	2004	2021
VIC	CitiPower	✓	✓	2010	2020
VIC	Powercor	✓	✓	2010	2020
VIC	Jemena	✓	✓	2008	2019
VIC	United Energy	✓	✗	2011	2020
QLD	Energex	✓	✓	2012	2020
QLD	Ergon	✓	✓	2012	2021
SA	SAPN	✓	✓	2012	2021
TAS	Tasnet	✓	✓	2004	2018

\*As of 1<sup>st</sup> March 2022

network as a whole. To ensure our data's truthfulness to reality, we require a part of the NEM network where all its distribution companies have provided both active and reactive load data, and whose connection to the rest of NEM network can be justifiably neglected. QLD, SA, and TAS networks all satisfy the first requirement. The TAS network also satisfies the second requirement in the period from 21<sup>st</sup> December 2015 to 13<sup>rd</sup> June 2016 because its high-voltage direct current interconnector to the Australian mainland was disconnected in that period due to a hardware problem. Therefore, realistic load data in the TAS network [21] in the disconnected period is used to construct our dataset.

### B. Realistic network topology

Dataset [22] includes topological information of the TAS network, e.g. minimum dispatch and registered capacity of generators, resistance and reactance and susceptance of transmission lines, location and voltage of buses, etc.. This dataset would be used to formulate the constraints in Equations (4) and (6). 200MW is added to P-V bus 676's active generation capacity because of the temporary capacity boosting implementation for dealing with the crisis of interconnector outage [23]. The generation cost function for each generator required in Equation (1), the reactive generation limits required in Equation (5), as well as the branch flow limits required in Equation (7) are not publicly available. Their estimation would be explained in the next section.

## V. CONSTRUCTION OF TAS-97 DATASET

Each power station and substation is simplified as one node [22], which translates to the 97 buses in our dataset. The constructed dataset is named TAS-97 to indicate the real-world network it represents and the number of buses it contains.

### A. Assigning loads to TAS postal areas

Only P-Q loads from 1<sup>st</sup> January 2016 00:30 to 1<sup>st</sup> June 2016 00:00 are considered to avoid any system instability near the beginning and the end of the outage period. Each load datum is 30 minutes apart, so in total  $24\text{hours} \times 2\text{data/hour} \times 152\text{days} = 7296$  data are considered.

### Algorithm 1: P-Q bus load assignment

---

**input :** Voronoi cell  $c_i$  for P-Q bus  $i$ ,  
Postal area polygons  $[A_1, \dots, A_K]$ ,  
Postal area active loads  $[P_{A1}, \dots, P_{AK}]$ ,  
Postal area reactive loads  $[Q_{A1}, \dots, Q_{AK}]$

**output:**  $P_{Di}, Q_{Di}$   
 $P_{Di} := 0; Q_{Di} := 0;$   
**for**  $k := 1$  **to**  $K$  **do**  
    Assign intersection area  $a_k := c_i \cap A_k;$   
    Assign area ratio  $r_k := a_k/A_k;$   
    Update  $P_{Di} := P_{Di} + r_k P_{Ak};$   
    Update  $Q_{Di} := Q_{Di} + r_k Q_{Ak};$   
**end**

---

Every Tasmanian region has a postal code. The data given in [21] include the substation name, which can be used to find the postal area in which the substation locates. A mapping of substation name to the located postal code would be constructed. Time series active and reactive load of individual postal areas are obtained using the constructed mapping. The load data from substations 'Huon River' and 'Newton' are not available for reasons of confidentiality. Their active and reactive loads would be taken as the average of the neighboring substations. Loads of 'Huon River' are taken as the average loads from substations 'Electrona', 'Kermadie', 'Kingston', 'Knights Road', 'Summerleas Zone'. Loads of 'Newton' are taken as the average loads from substations 'Queenstown', 'Rosebery', 'Savage River', 'Trial Harbour Zone'.

### B. Assigning loads to P-Q buses

The exact coordinates for each P-Q bus are known from [22]. Their locations are used to draw a Voronoi Diagram where each Voronoi cell corresponds to one P-Q bus. This Voronoi Diagram is overlayed on the Tasmanian postal area polygons [24]. For each P-Q bus cell  $c_i$  we calculate its intersection area with every postal area. Then, for each postal area that has an intersection with  $c_i$ , it gives out a portion of its active and reactive loads to  $c_i$  with the portion as the ratio of the intersection area over the overall postal area. The operation to assign active and reactive loads to one bus at one time point is given in Algorithm 1. Fig 2 illustrates an example of assigning active and reactive loads from one postal area to the neighboring P-Q buses. After assignment, all P-Q loads are linearly scaled such that their sum matches the historical Tasmanian regional load from [25].

The TAS network has one wind P-V bus. Since wind generators are highly dependent on weather, predicting their optimal generation values has minimal practical value. Instead, it is regarded as a negative P-Q bus with its active loads set as the negative of its historical active generations, which can be obtained from [26]. Its reactive generation is approximated by multiplying its active generation by a sample from the sample normal distribution of  $Q_{Di}/P_{Di}$  on the same time interval,

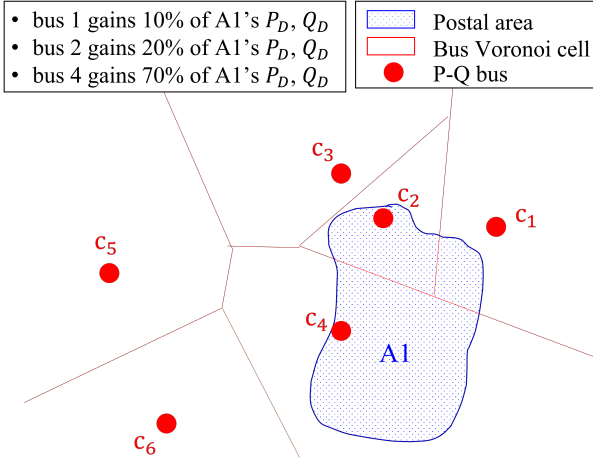


Fig. 2. Example of assigning loads from one postal area to P-Q buses

where  $i \in \mathcal{N}$ . As a negative P-Q bus, its reactive load is the negative of this reactive generation.

Data smoothing is performed on the constructed time-series load data for each bus as a means to remove any possible outliers. A moving average of a window of 13 is applied to the data. 3 shows that the smoothed active load of bus number 3 maintains its overall shape in the time-series plot. 7284 realistic load data remains after the data smoothing.

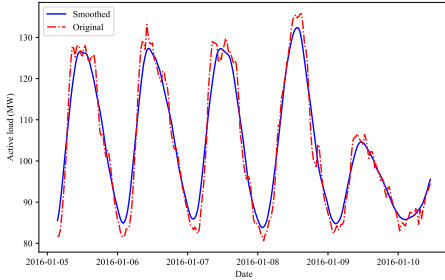


Fig. 3. Active load of bus number 3

### C. Calculating dispatch ground truths in TAS-97

The ground truths of optimal dispatch at each network load data point are obtained using PYPOWER [27]. The TAS network topology information provided by [22] provides most information required for the AC-OPF formulation. Nonetheless, the dataset does not include the required information of generation cost functions, reactive generation limits, and branch flow limits. Their values are estimated as follows.

1) *Cost function for large hydroelectric generators:* Hydroelectric generators with active power capacity  $P_{G_i} \geq 100\text{MW}$  would have linear cost functions of  $C_i = 6.724778P_{G_i}$ . This function references bus 30 in the *pglib-opf* test case *pglib\_opf\_case39\_epri* [28], which is a hydroelectric generator in a real power network in New England.

2) *Cost function for small hydroelectric generators:* Hydroelectric generators with active power capacity  $P_{G_i} < 100\text{MW}$  would have linear cost functions of  $C_i = 10.087167P_{G_i}$ . This function is the cost function of larger hydroelectric generators scaled by 1.5, a factor taken from a report [31] written on hydroelectric power cost modeling in Tasmania.

3) *Cost function for natural gas generators:* For natural gas generators, their cost function are taken as the cost function from natural gas generators with similar capacity in the *pglib-opf* test case *pglib\_opf\_case240\_pserc* [28], which is the representation of a real network in the United States [29].

4) *Reactive generation limits:* For each generator  $i$ , the lower limit of reactive generation  $Q_{G_i}^{\min}$  is taken as  $Q_{G_i}^{\min} = -0.5P_{G_i}^{\max}$ . The upper limit of reactive generation  $Q_{G_i}^{\max}$  is taken as  $Q_{G_i}^{\max} = 0.5P_{G_i}^{\max}$ . This estimation references the real-world *pglib-opf* test case *pglib\_opf\_case240\_pserc* [28]. For bus 6, the upper and lower limits are doubled since it has been found that only in this arrangement PYPOWER can successfully solve all the load samples.

5) *Branch flow limits:* Branch flow limits are initially relaxed to high levels where branch flow constraint violations would be impossible. Then, the relaxed AC-OPF formulation and the realistic load samples are run with PYPOWER solver to obtain optimal dispatches under relaxed branch flow constraints. We assume that the branch flow limit  $S_{ij}^{\max}$  for branch  $ij$  increases with its maximum experienced flow in relaxed scenarios. Then, we set the branch flow limits using the following rule: if branch  $ij$ 's maximum experienced flow is the  $k^{\text{th}}$  quantile among all maximum experienced flows,  $S_{ij}^{\max}$  is set to the  $k^{\text{th}}$  quantile of branch flow limits in the *pglib-opf* test case *pglib\_opf\_case89\_pegase*. *pglib\_opf\_case89\_pegase* is the representation of a small part of the European power network. This network is chosen for reference since it has a comparable size of buses and generation capacity to TAS-97.

P-V bus generation initial points for the PYPOWER OPF solver *runopf* are set as the historical generation values from [26].

### D. Summary of TAS-97

Fig. 4 illustrates the construction process of TAS-97 dataset. It is an AC-OPF formulation-ready dataset that represents the Tasmanian power network from 1<sup>st</sup> January 2016 03:30 to 31<sup>st</sup> May 2016 21:00. The network has 78 P-Q buses, 19 P-V buses, and 126 branches. Generator types include hydroelectricity, natural gas, and wind. Accompanying this network topology is 7284 bus load-optimal dispatch data pairs. The P-Q bus loads are correlated. Fig. 5 shows positive correlation between bus 696 and 697 (correlation coefficient is 0.86), and negative correlation between bus 3 and 781 (correlation coefficient is -0.13). The dataset is available on GitHub [30].

## VI. LEARNING-BASED MODELS

### A. Model architecture

Model architectures would be built upon the architecture DeepOPF of [1], which showcases a feasibility-optimized end-to-end model that has high prediction accuracy, no violation

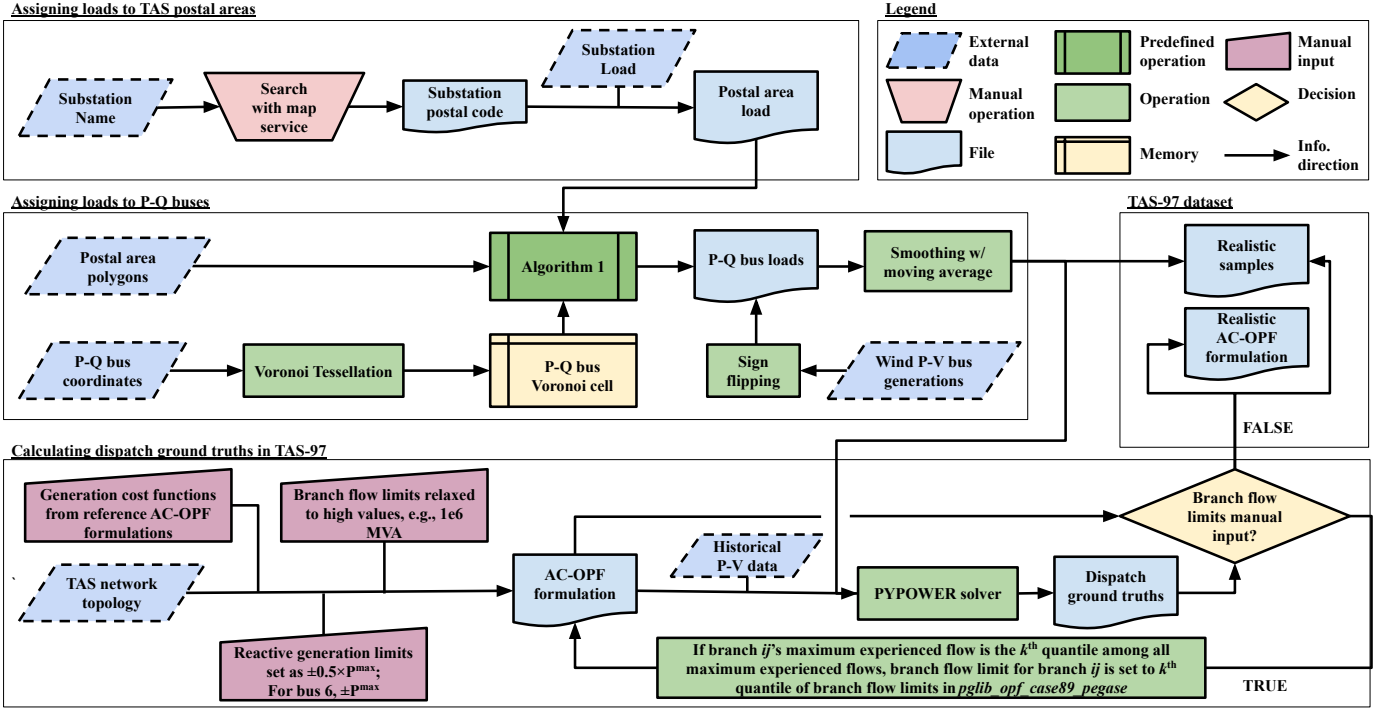


Fig. 4. Flowchart of the construction of TAS-97 dataset

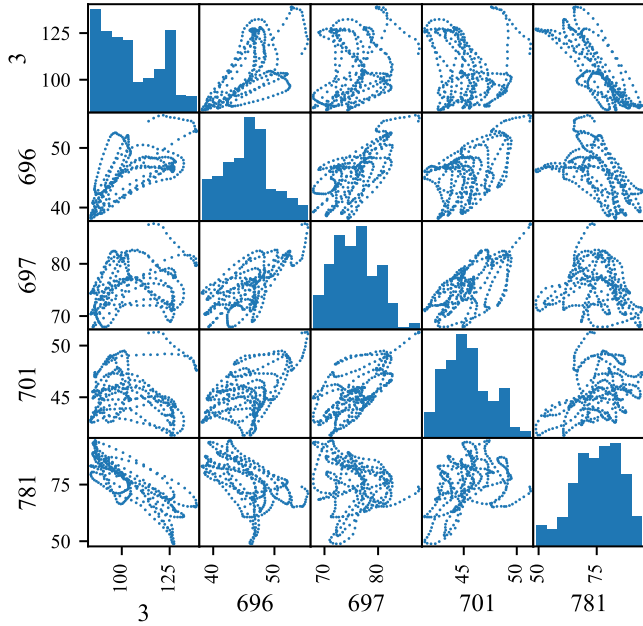


Fig. 5. Scatterplot matrix of  $P_{di}$  of selected P-Q buses in TAS-97

rate, and two orders of magnitude of speedups on realistic networks and synthetic loads.

Define the following symbols:

Symbol	Definition
$P_{G0}$	Active power generation on slack bus
$Q_{G0}$	Reactive power generation on slack bus
$ V_i $	Voltage magnitude on bus $i$ , $i \in \mathcal{D}$
$ V_0 $	Voltage magnitude on slack bus
$\mathcal{G}'$	Set of P-V buses that are not slack bus
$(\mathbf{P}_D)_D$	Vector of $\{P_{Dj} : j \in \mathcal{D}\}$
$(\mathbf{Q}_D)_D$	Vector of $\{Q_{Dj} : j \in \mathcal{D}\}$
$(\mathbf{P}_G)_{\mathcal{G}'}$	Vector of $\{P_{Gj} : j \in \mathcal{G}'\}$
$(\mathbf{Q}_G)_{\mathcal{G}'}$	Vector of $\{Q_{Gj} : j \in \mathcal{G}'\}$
$( \mathbf{V} )_D$	Vector of $\{ V_j  : j \in \mathcal{D}\}$
$( \mathbf{V} )_{\mathcal{G}'}$	Vector of $\{ V_j  : j \in \mathcal{G}'\}$
$\theta_i$	Phase angle on bus $i$
$(\Theta)_D$	Vector of $\{\theta_j : j \in \mathcal{D}\}$
$(\Theta)_{\mathcal{G}'}$	Vector of $\{\theta_j : j \in \mathcal{G}'\}$
$\text{card}(X)$	Cardinality of set $X$
$\hat{y}$	Prediction of quantity $y$
$y^*$	Ground truth of quantity $y$
$\cdot \parallel$	Vector concatenation operator

Our model is the zero-order optimized model proposed in [1]. The model consists of a feedforward neural network and a power-flow solver. The feedforward neural network  $\mathcal{F}$  maps  $(\mathbf{P}_D)_D \parallel (\mathbf{Q}_D)_D$  onto  $(|\mathbf{V}|)_{\mathcal{G}'} \parallel (\mathbf{P}_G)_{\mathcal{G}'}$ . The feedforward process for each hidden layer is:

$$\mathbf{h}_i(\mathbf{x}_i) = \sigma'(\mathbf{W}_i \mathbf{x}_i + \mathbf{b}_i) \quad (8)$$

where  $\sigma'(\cdot)$  is the ReLU activation function,  $\mathbf{x}_i$  is the input

vector to the hidden layer,  $\mathbf{W}_i$  is the trainable connection weights,  $\mathbf{b}_i$  is the trainable bias. The network output ground truths are linearly scaled within their respective operating limits. The sigmoid activation function is adopted for the output layer such that the network outputs are always within the operating limits. The feedforward process of the output layer is:

$$\mathbf{o}(\mathbf{x}_o) = \sigma(\mathbf{W}_o \mathbf{x}_o + \mathbf{b}_o) \quad (9)$$

where  $\sigma(\cdot)$  is the sigmoid activation function,  $\mathbf{x}_o$  is the input vector to the last layer,  $\mathbf{W}_o$  is the trainable connection weights,  $\mathbf{b}_o$  is the trainable bias. Connected, the forward propagation of the neural network is:

$$\mathcal{F}(\mathbf{P}_D, \mathbf{Q}_D) = \mathbf{o}(\mathbf{h}_n(\cdots \mathbf{h}_2(\mathbf{h}_1(\mathbf{P}_D \parallel \mathbf{Q}_D)))) \quad (10)$$

where  $n$  is the number of hidden layers in the neural network.

The power-flow solver uses the neural network outputs as independent variables to calculate the dependent variables  $P_{G0} \parallel (\mathbf{Q}_G)_{\mathcal{G}} \parallel (\Theta)_{\mathcal{D}} \parallel (|\mathbf{V}|)_{\mathcal{D}} \parallel (\Theta)_{\mathcal{G}'}$ . PYPOWER's *runpf* method would be used. Since the solver is based on Newton's method, different initial points for the dependent variables would affect the performance of the power-flow solver. The initial points are set as the average value of the dependent variables in the training data. Fig. 6 illustrates the architecture of our model.

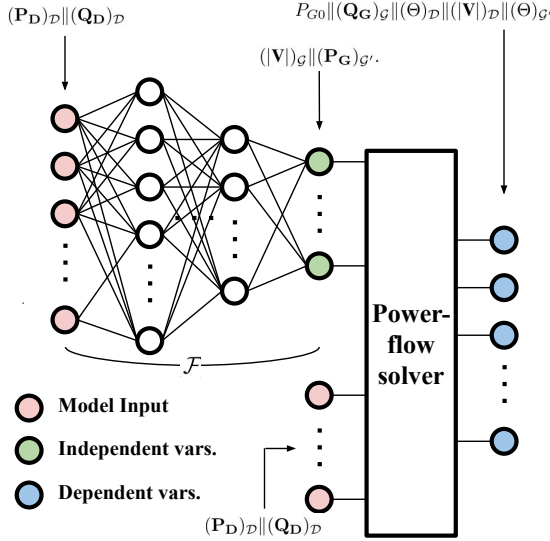


Fig. 6. Model architecture

### B. Model training scheme

In training the neural network in our model, the training scheme of [1] is adopted. That is, prediction loss  $\mathcal{L}_{pred}$  from deviations between the neural network outputs and ground truths, as well as the penalty loss  $\mathcal{L}_{pen}$  from operation constraints violations, are used. The prediction loss function is defined as the mean squared error:

$$\mathcal{L}_{pred} = (1/\text{card}(\mathcal{F}))(\hat{\mathcal{F}} - \mathcal{F}^*)^T(\hat{\mathcal{F}} - \mathcal{F}^*) \quad (11)$$

where  $\hat{\mathcal{F}}$  is the prediction from  $\mathcal{F}$ ,  $\mathcal{F}^*$  is the vector of ground truths  $(\mathbf{P}_G)_{\mathcal{G}'}^* \parallel (|\mathbf{V}|)_{\mathcal{G}}^*$  from a training sample. The gradients  $\nabla \mathcal{L}_{pred}$  w.r.t.  $\mathbf{W}_i$  and  $\mathbf{b}_i$  are calculated by first computing the gradient w.r.t.  $\hat{\mathcal{F}}$ , then applying chain rule on each  $\mathbf{W}_i$  and  $\mathbf{b}_i$ .

The penalty loss function is the deviation of the model prediction from the feasible region bounded by Equations (4) to (7). Let  $y^{\max}$ ,  $y^{\min}$  be the operation limits of quantity  $y$ , penalty loss on a prediction  $\hat{y}$  is

$$p(\hat{y}) = \max\{\hat{y} - y^{\max}, 0\} + \max\{y^{\max} - \hat{y}, 0\} \quad (12)$$

and the total penalty loss on the prediction model is

$$\begin{aligned} \mathcal{L}_{pen} = & \frac{1}{\text{card}(\mathcal{E})} \sum_{(i,j) \in \mathcal{E}} p(\hat{S}_{ij}) + \\ & \frac{1}{\text{card}(\mathcal{D})} \sum_{i \in \mathcal{D}} p(|\hat{\mathbf{V}}_i|) + \\ & \frac{1}{\text{card}(\mathcal{G})} \sum_{i \in \mathcal{G}} p(\hat{Q}_{Gi}) + p(\hat{P}_{G0}) + p(\hat{Q}_{G0}). \end{aligned} \quad (13)$$

$\mathcal{L}_{pen}$  does not depend on  $\hat{\mathcal{F}}$ , so a two-point zero-order optimization technique [33] is used to compute the gradient:

$$\hat{\nabla} \mathcal{L}_{pen}(\hat{\mathcal{F}}) = \frac{\text{card}(\mathcal{F})}{2\delta} \mathbf{v}(\mathcal{L}_{pen}(\hat{\mathcal{F}} + \mathbf{v}\delta) - \mathcal{L}_{pen}(\hat{\mathcal{F}} - \mathbf{v}\delta)) \quad (14)$$

where  $\delta$  is a small smoothing parameter and  $\mathbf{v}$  is a vector that has the same dimension as  $\mathcal{F}$  and is randomly sampled on the unit ball.

## VII. EXPERIMENTS AND RESULTS

### A. Experimental setup

Experiments are run on Windows 10 with quad-core (i7-10510U@1.80GHz) CPU with 8GB RAM. The model has 3 hidden layers with 512, 256, 128 hidden units. Both layer weight and bias initialization are sampled from the uniform distribution  $\text{Unif}(-\sqrt{1/k}, \sqrt{1/k})$ , where  $k$  is the number of hidden units in the layer. Training epochs is set to 200. The learning rate decays from  $10^{-4}$  at epoch 1 to  $10^{-9}$  at epoch 200. The learning rate is evenly spaced on the log scale from epoch 1 to epoch 200. Our empirical experience on TAS-97 shows including penalty gradient when the epoch is still low brings unstable penalty gradients that impair both the prediction accuracy and constraint satisfaction. Therefore the penalty gradient is only introduced after epoch 100. The penalty gradient is weighted by a 0.1 factor as suggested in [1].

To investigate the effect of sampling methods, we experiment on different sampling schemes.

1) *Testing on all 7284 realistic samples:* We train three models on 65556 synthetic samples. Model 1 trains on samples synthesized from independent uniform distributions with bounds set as each bus's historical load limits. Model 2 trains on samples synthesized from independent normal distributions fitted with each bus's realistic loads. Model 3 trains on samples synthesized from a multivariate normal distribution fitted with all buses' realistic loads.

TABLE III  
TESTING RESULTS

Experiment	Sampling scheme		$\eta_{opt}$ (%)	$\eta_{fea}$ (%)	$\Delta_{PG}$ (p.u.)	$\Delta_{QG}$ (p.u.)	$\Delta_{ V }$ (p.u.)	$\Delta_S$ (p.u.)	$t_0$ (ms)	$t_M$ (ms)	$\eta_{sp}$
	Training	Testing									
1	65556, synthetic (uniform)	7284	2.89	94.10	0.014	0.000	0.000	0.000	855.51	20.26	42.23
2	65556, synthetic (normal)	7284	0.16	94.99	0.018	0.001	0.000	0.000	855.51	22.36	38.26
3	65556, synthetic (multivariate normal)	7284	0.13	99.73	0.0005	0.000	0.000	0.000	855.51	22.15	38.62
4	65556, synthetic (uniform)	3642	2.02	90.06	0.023	0.000	0.000	0.000	836.13	20.32	41.15
5	65556, synthetic (normal)	3642	0.99	92.30	0.027	0.001	0.000	0.000	836.13	17.55	47.64
6	65556, synthetic (multivariate normal)	3642	0.88	99.75	0.018	0.0004	0.000	0.000	836.13	22.69	36.85
7	3642, realistic	3642	2.58	94.73	0.029	0.003	0.000	0.000	836.13	26.80	31.20

2) *Testing on the last half of the realistic samples (3642 samples)*: Experiments 4,5,6 use the models used in Experiments 1,2,3 respectively. Experiment 7 trains on the first half of the realistic samples (3642 samples).

Model performance is evaluated by the following metrics:

- 1) Optimality loss  $\eta_{opt}$ : Average relative deviation between the model's and PYPOWER's dispatch cost.
- 2) Feasibility rate  $\eta_{fea}$ : Percentage of feasible predictions of the model. A prediction is considered to be feasible if there are no constraint violation from Equations (4) to (7).
- 3) Degree of deviation: Distances between a violated decision variable and the boundary in per-unit system for Equations (4) to (7).  $\Delta_{PG}$ ,  $\Delta_{QG}$ ,  $\Delta_{|V|}$ ,  $\Delta_S$  denote the degree of violation of active generation, reactive generation, voltage magnitude, and branch flow respectively.
- 4) Speedup factor  $\eta_{sp}$ : The average ratio of the prediction time  $t_M$  taken by the model to the prediction time  $t_0$  taken by PYPOWER.

Table III shows the experimental results.

#### B. Effect of the quality of training samples

Experiments 4 to 7 illustrate how the quality of training samples affects the model performance on realistic testing samples. The quality of training samples are listed below in descending order.

1) *Realistic training samples*: While realistic training samples in Experiment 7 are expected to have the closest resemblance to the realistic testing samples, the model performance is not the best because of the limited training sample size. However, even with a small training size, the testing performance is better than Experiments 4 and 5, showcasing the quality of the samples.

2) *Multivariate normal synthetic samples*: Fig. 5 implies a multivariate normal probability distribution for the bus loads. Samples synthesized from a fitted multivariate normal distribution would have a close resemblance to the realistic testing samples, leading to the best testing performance in Experiment 6.

3) *Normal synthetic samples*: Training samples from Experiment 5 disregard the correlation between bus loads but have a properly selected distribution family, therefore the model performance is worse than Experiment 6 but better than Experiment 4.

4) *Uniform synthetic samples*: Training samples from Experiment 4 disregard both the correlation and have a poorly selected distribution family, therefore the model performance is the worst among Experiments 4 to 7.

#### C. Learning-based OPF solver on realistic network and realistic loads

Experiment 3 shows a promising result of using learning-based models on a realistic network and realistic loads. Both the optimality gap and degree of deviation are small, and the feasibility rate is high.

There are violations on active generation constraints in Experiment 3. Note that the neural network's sigmoid output activation function already forces the active generations from all generators except the slack generator to be within operational limits, therefore the degree of violation  $\Delta_{PG}$  comes only from the slack generator. With this small deviation, in practice we can make use of charging and discharging batteries in the slack generator substation to make sure the slack generator does not exceed its generation limits.

## VIII. CONCLUSIONS

We construct a dataset that represents the power network in Tasmania and included 7284 samples of realistic load data. It has been found that the realistic bus loads exhibit signs of a multivariate normal distribution. We train and test a feasibility-optimized end-to-end learning model using the dataset. Training on samples synthesized from a fitted multivariate normal load distribution, our model achieves 0.13% cost optimality gap, 99.73% feasibility rate, and 38.62 times of speedup on realistic testing samples when compared to PYPOWER.

## REFERENCES

- [1] X. Pan, M. Chen, T. Zhao, and S. H. Low, "DeepOPF: A Feasibility-Optimized Deep Neural Network Approach for AC Optimal Power Flow Problems," 2021. [Online]. Available: arXiv:2007.01002v5
- [2] T. Falconer, L. Mones, "Leveraging power grid topology in machine learning assisted optimal power flow," 2021. [Online]. Available: arXiv:2110.00306v
- [3] T. W. K. Mak, F. Fioretto, P. V. Hentenryck, "Load Embeddings for Scalable AC-OPF Learning," 2021. [Online]. Available: arXiv:2101.03973v1
- [4] R. Nellikkath, S. Chatzivasileiadis, "Physics-Informed Neural Networks for AC Optimal Power Flow," 2021. [Online]. Available: arXiv:2110.02672v1
- [5] M. Chatzos, T. W. K. Mak, and P. V. Hentenryck, "Spatial Network Decomposition for Fast and Scalable AC-OPF Learning," *IEEE Trans. Power Sys.*, doi: 10.1109/TPWRS.2021.3124726



- [6] A. Venzke, S. Chatzivasilieadis, "Verification of Neural Network Behaviour: Formal Guarantees for Power System Applications," *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 383-397, Jan. 2021.
- [7] L. Chen, J. E. Tate, "Hot-Starting the Ac Power Flow with Convolutional Neural Networks," 2020. [Online]. Available: [arXiv:2004.09342v1](https://arxiv.org/abs/2004.09342v1)
- [8] D. Deka, S. Misra, "Learning for DC-OPF: Classifying active sets using neural nets," 2019. [Online]. Available: [arXiv:1902.05607v1](https://arxiv.org/abs/1902.05607v1)
- [9] M. D. McKay, R. J. Beckman, and W. J. Conover, "Comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239-245, 1979.
- [10] *Distribution zone substation data*, Ausgrid. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.ausgrid.com.au/Industry/Our-Research/Data-to-share/Distribution-zone-substation-data>
- [11] *Distribution Annual Planning Report*, Endeavour Energy. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.endeavourenergy.com.au/modern-grid/creating-the-modern-grid/network-planning/distribution-annual-planning-report>
- [12] *Standardised Essential Energy Zone Substation Load Data*, Australia's National Energy Analytics Research Program. Accessed: Feb. 22, 2022. [Online]. Available: <https://near.csiro.au/assets/216328c8-9026-4810-9859-3a7149711ad6>
- [13] *Zone substation data*, Evoenergy. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.evoenergy.com.au/about-us/about-our-network/zone-substation-data>
- [14] *CitiPower Zone Substation Reports*, CitiPower and Powercor Australia. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.powercor.com.au/industry-partners/renewable-generation/network-data/citipower-zone-substation-reports/>
- [15] *Powercor Zone Substation Reports*, CitiPower and Powercor Australia. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.powercor.com.au/industry-partners/renewable-generation/network-data/powercor-zone-substation-reports/>
- [16] *Zone substation information*, Jemena. Accessed: Feb. 22, 2022. [Online]. Available: <https://jemena.com.au/industry/electricity/zone-substation-information>
- [17] *Regulatory reports - Zone substation information*, United Energy Limited. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.unitedenergy.com.au/industry/mdocuments-library/>
- [18] *Zone substation load data reports*, Energex. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.energex.com.au/about-us/company-information/our-network/zone-substation-load-data-reports>
- [19] *Zone substation data reports*, Ergon. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.ergon.com.au/network/contractors-and-industry/design-and-construction-contractors/zone-substation-data-reports>
- [20] *Resource Library*, SA Power Networks. Accessed: Feb. 22, 2022. [Online]. Available: <https://www.sapowernetworks.com.au/resource-library>
- [21] *Substation load information*, Tasmanian Networks Pty Ltd. Accessed: Mar. 1, 2022. [Online]. Available: <https://www.tasnetworks.com.au/poles-and-wires/planning-and-upgrades/planning-our-network>
- [22] A. Xenophon, D. Hill, "Open grid model of Australia's National Electricity Market allowing backtesting against historic data," *Scientific Data*, Art. no. 180203, Oct. 2018, doi: 10.1038/sdata.2018.203
- [23] E. Gramenz, "Tasmanian energy crisis: More diesel generators fired up to boost power grid," *ABC News (Australian Broadcasting Corporation)*, Mar. 31, 2016.
- [24] *Population estimates by Statistical Area Level 2, 2001 to 2020, in GeoPackage*, Australian Bureau of Statistics, Mar. 2021. [Online]. Available: <https://www.abs.gov.au/statistics/people/population/regional-population/2019-20>
- [25] *Aggregated Price and Demand Data - Historical*, Australian Energy Market Operator. Accessed: Feb. 22, 2022. [Online]. Available: <https://aemo.com.au/energy-systems/electricity/national-electricity-market-nem/data-nem/aggregated-data>
- [26] *Actual Generation and Load Data*, Australian Energy Market Operator. Accessed: Feb. 22, 2022. [Online]. Available: <https://aemo.com.au/energy-systems/electricity/national-electricity-market-nem/data-nem/market-management-system-mms-data/generation-and-load>
- [27] "PYPOWER," 2018, <https://pypi.org/project/PYPOWER/>.
- [28] *Benchmarks for the Optimal Power Flow Problem*, IEEE Power and Energy Society Task Force on Benchmarks for Validation of Emerging Power System Algorithms. Accessed: Mar. 18, 2022. [Online]. Available: <https://github.com/power-grid-lib/pglib-opf>
- [29] F. Munoz, "A scalable solution framework for stochastic transmission and generation planning problems," *Computational Management Science*, Mar. 2015, doi: 10.1007/s10287-015-0229-y
- [30] *TAS-97*, AaronTHCheung: May 2, 2022. [Online]. Available: <https://github.com/AaronTHCheung/TAS-97>
- [31] Hydro-Electric Corporation, "Pumped Hydro Cost Modelling," Entura, ABN48 072 377 158, Dec. 2018.
- [32] Energy and Mining Sector Board, "Technical and Economic Assessment of Off-grid, Mini-grid and Grid Electrification Technologies," World Bank Grp., Tech. Rep. 43099rev, Dec. 2007.
- [33] A. Agarwal, O. Dekel, and L. Xiao, "Optimal algorithms for online convex optimization with multi-point bandit feedback," in *Proc. COLT*, A. T. Kalai and M. Mohri, Eds. Haifa, Israel: Omnipress, 2010, pp. 28-40.