3-2023

Deep Learning for Power Flow Estimation and High Impedance Fault Detection

Kun Yang
University of Denver

Deep Learning for Power Flow Estimation and High Impedance Fault Detection

Abstract

My thesis is divided into two parts.

The first part is: “Optimal Power Flow Estimation Using One-Dimensional Convolutional Neural Network [1]“. Optimal power flow (OPF) is an important research topic in power system operation and control decisions. Traditional OPF problems are solved through dynamic optimization with nonlinear programming techniques. For a large power system with large amounts of variables and constraints, the solving process would take a long time. This paper presents a new method to quickly estimate the OPF results using a one-dimensional convolutional neural network (1D-CNN). The OPF problem is treated as a high-dimensional mapping between the load inputs and the generator dispatch decisions. Therefore, through training the neural network to learn the mapping between loads and generator outputs, we can directly predict the OPF results with the load information of a system. In this paper, we built and trained a 1D-CNN to learn the mappings between system loads and generator outputs, and the 1D-CNN model was tested using IEEE 30, 57, 118, and 300 Bus systems. Extensive test and sensitivity study results have validated the effectiveness of using the 1D-CNN to estimate the OPF results. This part is from chapter 1 to chapter 6;

The second part is: “Synthetic High Impedance Fault Data through Deep Convolutional Generated Adversarial Network [2]“. High impedance faults (HIFs) have always been significant challenges in the power grids. Researchers have developed some advanced protective methods to detect the HIFs. To test and validate these methods, large amounts of HIF data are required. This paper presents a synthetic HIF data generating method using the deep convolutional generated adversarial network (DCGAN). The DCGAN includes a generator module to create synthetic HIF waveform from random noises; and a discriminator module to identify the flaws of those synthetic data, which ultimately helps improve the quality of the synthetic data created by the generator. To test the fidelity of the generated synthetic HIF data, two different HIF-detection methods have been applied. Extensive simulation results have validated the effectiveness of using the DCGAN to create synthetic HIF data. This part is from chapter 7 to chapter 11.

Document Type

Thesis

Degree Name

M.S.

Department

Electrical Engineering

First Advisor

Rui Fan

Second Advisor

Yunbo Yi

Third Advisor

Mohammad Abdul Matin

Deep Learning for Power Flow Estimation and High Impedance Fault Detection

A Thesis

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and

Computer Science

University of Denver

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Kun Yang

March 2023

Advisor: Rui Fan

Author: Kun Yang
Title: Deep Learning for Power Flow Estimation and High Impedance Fault Detection
Advisor: Rui Fan
Degree Date: March 2023

Abstract

My thesis is divided into two parts.

The first part is: **"Optimal Power Flow Estimation Using One-Dimensional Convolutional Neural Network [1]"**. Optimal power flow (OPF) is an important research topic in power system operation and control decisions. Traditional OPF problems are solved through dynamic optimization with nonlinear programming techniques. For a large power system with large amounts of variables and constraints, the solving process would take a long time. This paper presents a new method to quickly estimate the OPF results using a one-dimensional convolutional neural network (1D-CNN). The OPF problem is treated as a high-dimensional mapping between the load inputs and the generator dispatch decisions. Therefore, through training the neural network to learn the mapping between loads and generator outputs, we can directly predict the OPF results with the load information of a system. In this paper, we built and trained a 1D-CNN to learn the mappings between system loads and generator outputs, and the 1D-CNN model was tested using IEEE 30, 57, 118, and 300 Bus systems. Extensive test and sensitivity study results have validated the effectiveness of using the 1D-CNN to estimate the OPF results. This part is from chapter 1 to chapter 6;

The second part is: **"Synthetic High Impedance Fault Data through Deep Convolutional Generated Adversarial Network [2]"**. High impedance faults (HIFs) have always been significant challenges in the power grids. Researchers have developed some advanced protective methods to detect the HIFs. To test and validate these methods, large amounts of HIF data are required. This paper presents a synthetic HIF data generating method using the deep convolutional generated adversarial network (DCGAN). The DCGAN includes a generator module to create synthetic HIF waveform from random noises; and a

discriminator module to identify the flaws of those synthetic data, which ultimately helps improve the quality of the synthetic data created by the generator. To test the fidelity of the generated synthetic HIF data, two different HIF-detection methods have been applied. Extensive simulation results have validated the effectiveness of using the DCGAN to create synthetic HIF data. This part is from chapter 7 to chapter 11.

Acknowledgments

I would like to thank my supervisor Dr.Fan for his dedicated support and guidance. Dr.Fan continuously provided encouragement and was always willing and enthusiastic to assist in any way he could throughout the research thesis.

I would also like to thank Wei Gao for providing advice regarding the analysis. Finally, many thanks to all participants that took part in the study and enabled this thesis to be possible.

Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

(Part1): Power flow (PF) is one of the most important topics in the power system, as it determines the voltage magnitude and phase angle of each bus under steady state conditions. In the electricity market, the goal is even more comprehensive with minimizing the total operating cost while maintaining the system generation and demand balance, which can be further utilized by system operators to dispatch the corresponding resources. Nowadays, with the worldwide trend toward deregulation of the electricity market, the objective function of this optimization is becoming more complicated as it involves a lot of nonlinear constraints, such as generation capacity and transmission line limits, etc. Therefore, the optimal power flow (OPF) problem is becoming extremely complicated.

In recent years, a number of dedicated methods have been proposed to solve the OPF problem [7, 8]. Since the OPF is a non-convex, nonlinear and high-dimensional optimization problem, it is very difficult to be solved directly. Therefore, most methods focus on how to solve the OPF more efficiently in an iterative way. The linear programming (LP) method is a common optimization algorithm and it is widely used in many areas [7]. The LP method can solve an economic dispatch problem efficiently where only linear constraints exist. However, an OPF problem consists lots of nonlinear constraints, which make it difficult for the LP method. Another disadvantage of the LP method is that it may lead to a locally optimal solution[9]. Nonlinear programming (NLP) and quadratic programming (QP) were then proposed as they can deal with nonlinear constraints [10, 11, 12, 13, 14, 15]. A classical algorithm called primal-dual interior-point was proposed in [16] and utilized maturely in MATPOWER software [17]. However, all these methods are time-consuming for convergence especially under a large-scale system, leading to a limited application in the real power market. The other famous iterative approach - Newton method in PF calculation was also proposed and utilized in [18] and an improved Newton-Raphson algorithm in a

multi-energy system was proposed in [19], [20]due to its fast convergence speed, but the convergence is not always guaranteed. As a result, it is challenging to obtain an efficient real-time OPF result using existing optimization methods.

The goal of the OPF problem is to obtain the real and reactive power flows of each branch while minimizing the total operating cost under some system constraints. This indicates that there should be a relationship or mapping between the given system status and the OPF results. If this pattern can be learned efficiently, then the OPF problem becomes a straight-forward prediction rather than solving a complex optimization problem. Machine learning is usually used to learn the mapping between inputs and outputs. In [21, 22, 23, 24] proposed to use a machine learning approach to solve the AC-OPF problem. The inputs were the real and reactive loads at each bus, and the outputs were the real power and voltage settings of each generator. Through generating tens of thousands training data, the author successfully trained a multi-layer perceptron (MLP) model to estimate the OPF results. Similar work was also conducted in [22, 25]. The machine learning methods used in this work are based on traditional neural networks such as the MLP, which does not preserve the translation variance for the input data and may not work well for complex systems. In many research areas, the convolutional neural network (CNN) has shown its huge power in solving some formidable problems which are complex or even impossible using conventional methods, which will be continuously innovated with more state-of-the-art structures. With enough data trained by CNN, the mapping between network input and output can be learned effectively without consideration of the complicated character of the original problem.

In this paper, we propose a one-dimensional CNN (1D-CNN) based OPF estimation approach to solve the issue of time-consuming in conventional methods. The input of the 1D-CNN model is the load information at each branch, and the outputs are the generator's real and reactive power. The 1D-CNN consisted of three convolutional layers, where convolutional function, pooling function, and nonlinear activation functions were engaged. The output layer of the 1D-CNN is a dense layer that was used to predict the estimated

generator power. To train the 1D-CNN model, we automatically generated tens of thousands of training data. The proposed 1D-CNN approach was tested on IEEE 30, 57, 118, and 300 Bus test systems. Extensive simulation results have validated the effectiveness of using the 1D-CNN to estimate the OPF results.

The rest of the paper is organized as follows: Chapter 2 introduces the optimal power flow problems. In Chapter 3, introduce how to use CNN to solve the OPF problems. Chapter 4 gives the principle of this neural network based OPF estimation method and data. Then Chapter 5 shows the experimental results. Finally, the conclusions of this paper are given in Chapter 6.

(Part2): After a stable power system is established, the most significant thing is to guarantee the reliability of the constructed system, which brings the concept of power system protection. The development of variety of intelligent fault detection methods have made modern power grids much reliable. However, one existing challenge in power system protection is the high impedance fault (HIF). HIF is usually classified as a shortcircuit fault with a high impedance between the fault point and the ground, in which the short current maybe on the same level as the normal case. Therefore, the conventional overcurrent method cannot be easily applied to detecting the HIF [26]. However, because of the arcing and high-frequency components induced by the HIF, it can cause wild fires, electricity shock, and even some irreparable damage to human life. About 5% to 10% of faults happened in the distribution network are HIFs [27]; yet the true percentage tends to be higher since many HIF cases were not noticed or recorded. Therefore, solving the HIF challenge is very critical in ensuring the reliability and safety of modern power systems.

Many HIF detection methods have been proposed in recent decades. The most popular detection technique is to extract and analyze the features of the HIF waveform. This kind of method focuses on the characteristics of the HIF mathematically, such as the no, n-linearity, non-stationary nature, randomness, and asymmetry [28]. Basically, the analysis domains for

this kind of HIF detection include four categories: time domain, frequency domain, time-scale domain and time-frequency domain [29].

Time domain analysis takes advantages of the sudden variation of the HIF waveform, which focuses on the voltage or current data in the limited window. Although this method is not very accurate as it highly relies on the sampling of the data, it is still very popular for application due to the simplicity. Frequency domain analysis utilizes the frequency characteristics of the HIF waveform as the third harmonic components of HIFs can be exploited through Fast Fourier Transform (FFT) [30]. Sedighizadeh *et al.* used a high frequency method to detect the HIF by taking advantages of the rapid and sudden variation of HIF waveform [31, 32, 33, 34, 35, 36]. The challenge of the frequency domain analysis is to obtain the instantaneous frequency and localize the frequency components. Time-scale domain analysis is one of the most powerful technique for HIF detection. The well-known wavelet transform (WT) captures not only frequency information, but also time point of the specific frequency component. Siadatan *et al.* used time-frequency based algorithm to detect the HIF using real-world data under different weather conditions [37]. The challenge of the time-scale domain analysis is that determining the choice of mother wavelet used in the transform is usually difficult. Time-frequency domain analysis can be considered as an enhanced technique based on time-scale domain analysis [38]. However, the huge amount of computational efforts needed by this technique was an obstacle in previous years.

Some other techniques have also been proposed for HIF detection in recent years [39, 40, 41]. Khani *et al.* introduced a distribution network HIF detection method using an integrated enhanced particle swarm optimization-fuzzy model [40]. However, this method requires some extra information and the knowledge of large amounts of HIF data. Not to mention its implementation is relatively complex. Fan *et al.* presented a deep neural network based HIF detection method [41]. A 1-dimension convolutional neural network (CNN) was built and trained to differentiate the HIF from normal transients. This kind of method also requires a large amount of data during the training process.

The key for the aforementioned HIF fault detection methods is the HIF data. Large amounts of HIF data are either required during the training/designing process of these protective methods, or they are needed to test and validate those methods. To solve this problem, this paper presents a synthetic HIF data generating method using a deep convolutional generated adversarial network (DCGAN) [42]. DCGAN is a method of unsupervised learning, which has the advantage of no label information requirement than the supervised learning. In addition, other than traditional machine learning methods, DCGAN has two neural networks to improve the network performance through competition. Specifically, a DCGAN includes a generator module to create synthetic HIF waveform from random noises; and a discriminator module to identify the flaws of those synthetic data. Therefore, the generator and discriminator compete with each other and it results in performance improvements of both modules. To test the effectiveness of the synthetic HIF data generated by DCGAN, a third-harmonic based method in [43] and the neural network based method in [41] are applied to the generated data.

The rest of this thesis is organized as follows: in Chapter 7, the HIF data and two detection methods are briefly introduced. The detailed proposed DCGAN-based method is illustrated in Chapter 8. Chapter 9 is the research results and discussions. Finally, Chapter 10 presents the conclusions.

Chapter 2. Optimal Power Flow Problem

2.1 OPF Formulation

The general optimal power flow problem has formulation:

$$min\, f(u,x) \tag{2.1.1}$$

$$s.t.\, g_p(u,x) = 0, p = 1,2,...,n_p \tag{2.1.2}$$

$$s.t.\, h_m(u,x) \le 0, m = 1,2,...,n_m \tag{2.1.3}$$

where f(x,u) is the objective function. x and u are the [44, 45, 46, 47, 48, 49] . The different type of buses have different decision and state variables:

1) Load bus: decision variables are the real and reactive powers; State variables are voltage magnitude and angle.

2) Generation bus: decision variables are generated real power and voltag e magnitude; State variables are generated reactive power and voltage angle.

3) Reference bus: decision variables are voltage magnitude and angle; State variables are generated real and reactive power.

2.2 Newton OPF Scheme

The state variables and decision variables are described in the section above, to solve the problem, we first set up the Lagrangian:

$$L(x,u,\lambda) = f(u,x) + \sum_{i=1}^{m} \lambda_i g_i (x,u \qquad) \tag{2.2.1}$$

Where λ is the vector of the Lagrage multipliers. Then we make use of the fact that a necessary condition for a minimum is that:

$$\nabla L(x,u,\lambda) = 0 \tag{2.2.2}$$

For simplicity define

$$z = [x,u,\lambda]^T \tag{2.2.3}$$

The goal then is to solve for a z^* that minimizes the Lagrangian and hence solves:

$$\nabla L(z^*) = 0 \tag{2.2.4}$$

To solve this equation, first express $\nabla L(z^*)$ with is Taylor expansion about some known(and presumably non-optimal) point z:

$$\nabla L(z^*) = \nabla L(z) + \nabla^2 L(z) \triangle z + \textit{higher order terms} \tag{2.2.5}$$

with $\triangle z = z^* - z$. If we ignore the higher order terms, we can directly solve for $\triangle z$

$$\triangle z = -\left[\nabla^2 L\,(z)\right]^{-1} \nabla L\,(z) \tag{2.2.6}$$

Since the higher order terms were ignored, $z+\triangle z$ is only an approximation of z^*. Hence we need to slove iteratively:

$$z^{(k+1)} = z^{(k)} - \left[\nabla^2 L\left(z^{(k)}\right)^{-1}\right] \nabla L\left(z^{(k)}\right) \tag{2.2.7}$$

To avoid having to continually write the gradient symbol, define:

$$h(z) = \nabla L(z) \tag{2.2.8}$$

$$W\,(z) = \nabla^2 L(z) \tag{2.2.9}$$

Chapter 3. Proposed Approach

In the traditional calculation method of optimal power flow, an initial guess is made for state variables and decision variables, and then the optimal solution is obtained through iteration. The disadvantage of this solution method is that with the increase of the number of buses and generators, the amount of calculation will become huge. In the real world, grid operators must solve OPF for the entire electric grid every 5 minutes, considering the changes in the power generation and loads. Therefore, it is very challenging to fulfill the computing requirement in such a short time.

Therefore, in this paper, we propose to use a machine learning based approach to directly predict the OPF results without solving the optimization problem. The objective is to train a neural network to learn the mapping between the system information (as inputs) and the OPF results (as outputs). The machine learning method used in this study is the 1D-CNN approach.

3.1 1D-CNN Model

CNN is a deep neural network model that is widely used in the fields of image and natural language processing. In 1999, Lecun [50] proposed a gradient-based back propagation algorithm for document recognition. In that neural network, the convolutional layer played a crucial role. CNN is a typical supervised learning method. It is suitable for identifying simple patterns in the data, and then using these simple patterns to generate more complex patterns at a higher level with the increasing computing power, some large CNN networks began to show great advantages in the field of images, therefore most of the existing applications of CNN are two-dimensional. However, CNN can also be applied to one-dimensional data, such as time-series waveform or vector inputs. In the OPF problem,

because the power system load information can be treated as a vector, using 1D-CNN could be a feasible approach.

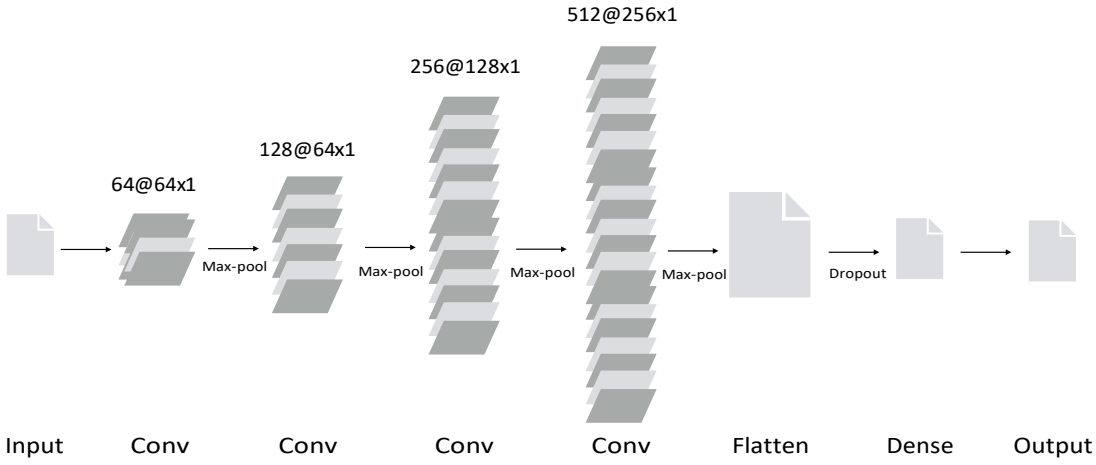


Figure 3.1: Proposed 1D-CNN model for OPF estimation.

The proposed 1D-CNN model is shown in Fig 3.1. As is known to all, CNN is often used for two-dimensional or higher-dimensional data processing, for the one-dimensional convolution layer, the biggest difference between the main convolution layer and the two-dimensional convolution layer is that the convolution kernel of the one-dimensional convolution layer becomes one-dimensional. That is to say, local one dimensional sequence segments(i.e., sub-sequences) are extracted from the sequence according to a certain size window, and then dot product with weights, then output is a part of the new sequence. Then one-dimensional pooling layer was added to the model, one-dimensional pooling layer extracts one-dimensional sequence segments from input and outputs their maximum value or average value to reduce the length of one-dimensional input. The purpose of the pooling layer is to blur the result of convolution, and summarize the statistical features in the local region and the over-fitting is avoided by dimensionality reduction. The “he uniform” initialization was adopted in every convolutional layer, which helps find a good variance for the distribution from the initially drawn parameters. This variance is adapted to the activation function used and is derived without explicitly considering the type of the distribution [51, 52, 53, 54, 55]. In the process of network training, the gradient is easy to

disappear (the gradient is extremely close to 0) and the gradient explosion (the gradient is extremely large), resulting in most of the gradients obtained by backpropagation are ineffective or counterproductive. The researchers hope to have a good weight initialization method: when the network is propagated forward or back, the output of the convolution and the gradient of the forward transmission are relatively stable.

Using the proposed 1D-CNN model in Fig. 1, the OPF problem was turned into a regression task, where a mapping between the input and output is the key issue. In other words, we were estimating the OPF results (generator output $P_{G,i}$and $Q_{G,i}$) from load information ($P_{L,i}$and $Q_{L,i}$). The remaining voltage magnitude and angles can be directly solved with the estimated OPF results, thus they are not included here. In conclusion, the input to the 1D-CNN model are the real and reactive loads at each bus$X$ = $[P_{L,1},\cdots ,X_{L,N},Q_{L,1},\cdots ,Q_{L,N}]$, and the output are the generator real and reactive power Y = $[P_{G,1},\cdots ,P_{G,M},Q_{G,1},\cdots ,Q_{G,M}]$.The objective of the 1D-CNN is to learn the mapping: $X \rightarrow Y$ that minimize the mean-squared error between the predicted generator output $\hat{Y}$ and the optimal generator setting (true value) Y.

3.2 Test IEEE Systems and Data

To train and test the proposed 1D-CNN approach, we have chosen several test systems with small, medium, and large scales. Specifically, the IEEE 30, 57, 118, and 300 Bus test systems were used in this study. For each IEEE test system with a load distribution of x, we randomly sampled the load distribution x with a range of $[(1-\delta)x,(1+\delta)x]$. To test the impact of data variation range (δ) on the accuracy of OPF estimation, we specifically selected two different to be 10% and 30%. In other words, there were two data-sets for each IEEE test system, one has a range of 90% to 110% of its original load distribution, the other has a range of 70% to 130% of its original load distribution. In addition, for each of the IEEE test systems (and each 10% and 30% data variation), we generated around 200,000 solved OPF cases, which gave us enough data for training and testing the 1D-CNN performance. Note

that not all the data were used to train the 1D-CNN model, and the training data-set and testing data-set were separate. In fact, when training the 1D-CNN model for each IEEE test system, we tested the impact of different numbers of training data on the accuracy of the proposed approach. Details were included in the following section.

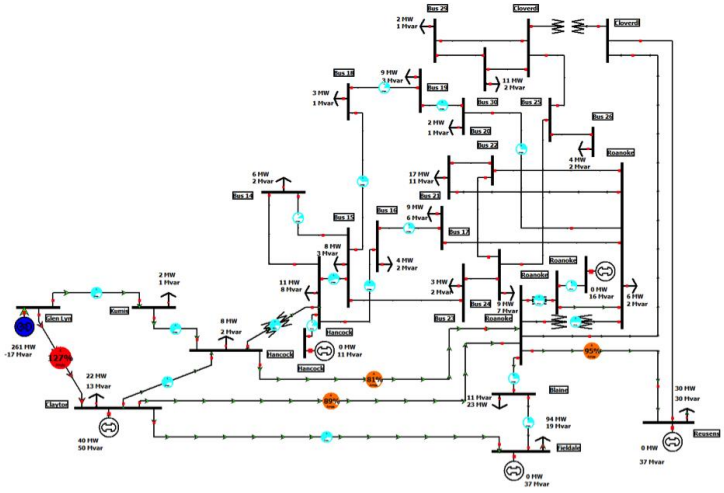


Figure 3.2: IEEE test case: 30-bus system[3]

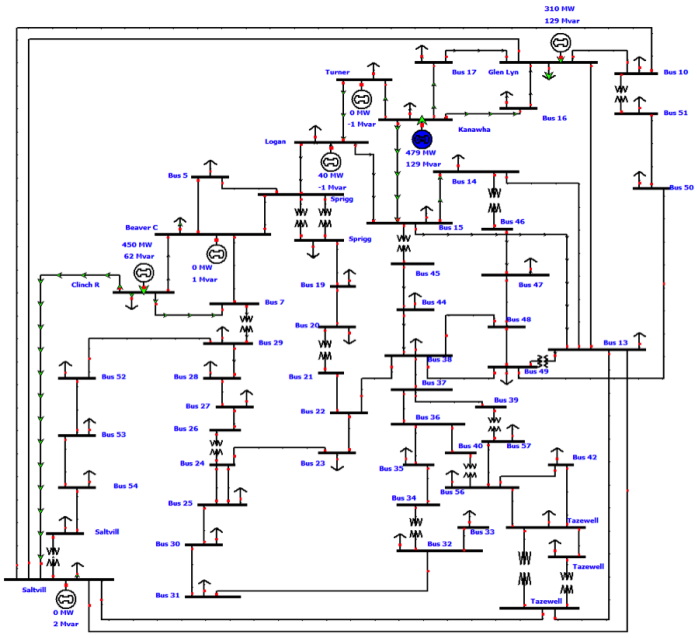


Figure 3.3: IEEE test case: 57-bus system[4]

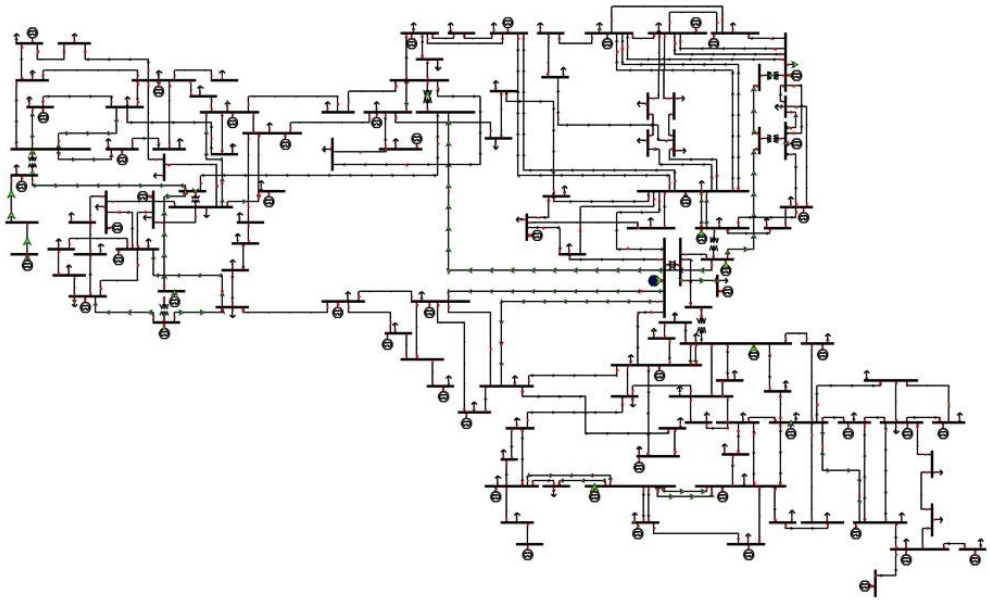


Figure 3.4: IEEE test case: 118-bus system[5]

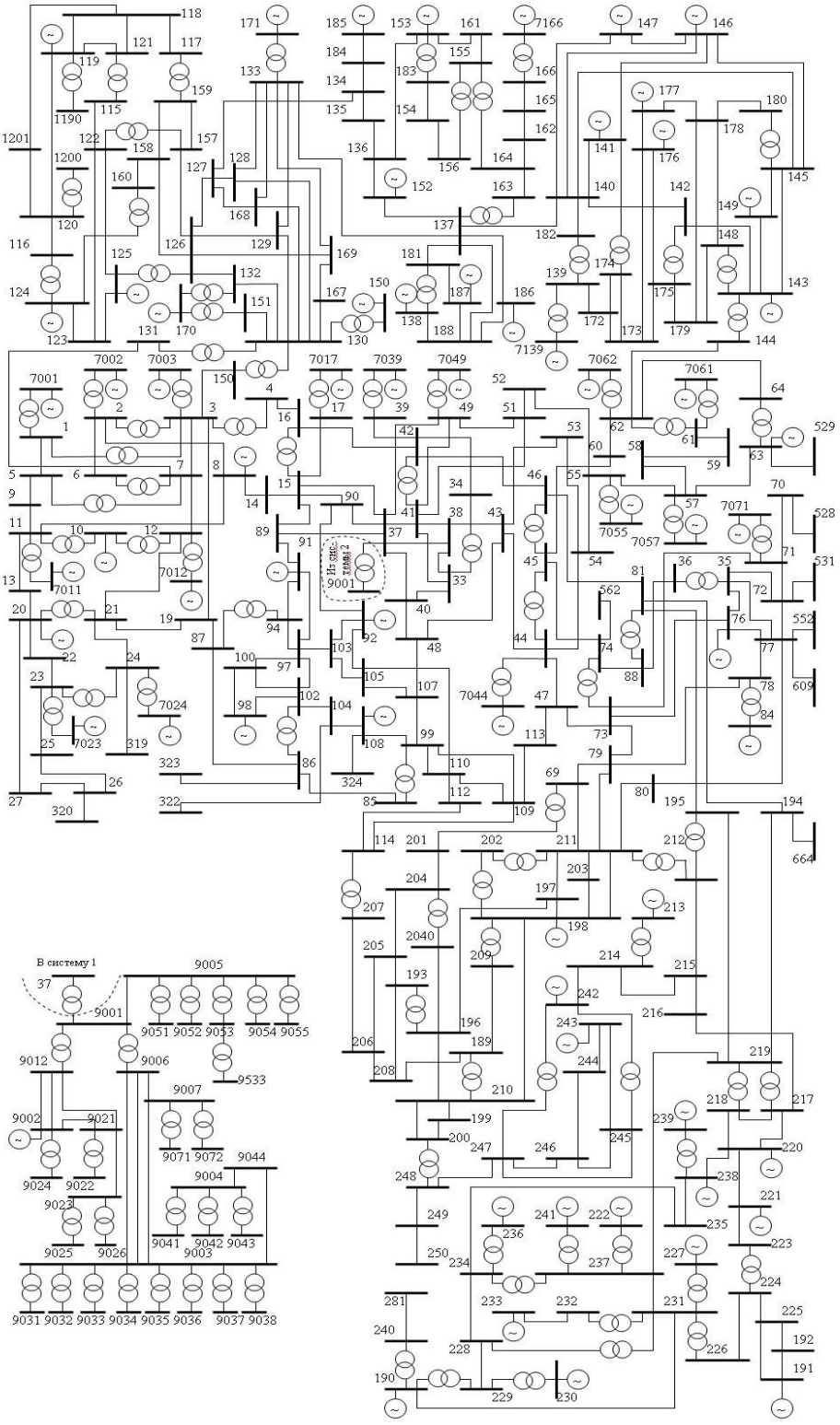


Figure 3.5: IEEE test case: 300-bus system[6]

Chapter 4. Results and Discussions

To test the performance of the proposed 1D-CNN approach, a traditional machine learning method was also implemented for comparison. The traditional machine learning method used a three-layer MLP model with 128, 64, 32 neurons for the respective layers. The average absolute mismatches of the OPF results were used as the criterion for evaluating the performances of the proposed 1D-CNN and transitional machine learning methods.

4.1 Test Result of 1D-CNN and MLP

In the first study, we trained the 1D-CNN and MLP models with 180,000 data for each IEEE test system. These data had a variation range [δ] of 10%. An example of the training losses for the 1D-CNN and MLP for the IEEE 57-Bus system is shown in Fig. 2. From the figure we could see that at the beginning the losses were very high for both models, however, the losses decreased as the training epochs increased. After a few hundred of training, the losses stabilized, which meant the training was completed. It was noticed that the 1D-CNN had a lower loss than the MLP method. Therefore, the 1D-CNN method provided better OPF estimation results than the MLP method.

For each IEEE test system, separate 2,000 data were used to test the performance of the well-trained 1D-CNN and MLP models. The average absolute mismatches of the OPF results for the 1D-CNN and MLP predictions were summarized in Table I. From the comparison, we could see that the 1D-CNN had a higher accuracy in predicting the
OPF results than the MLP method, which was in consistent with the training results

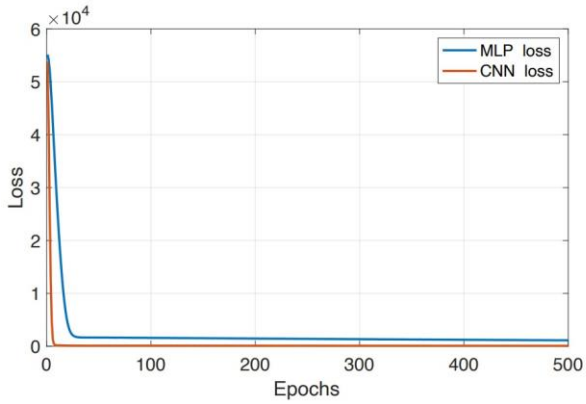


Figure 4.1: Training losses for the 1D-CNN and MLP methods

Table 4.1: 1D-CNN and MLP result comparison

Case ID	CNN MR	MLP MR
case30	1.16%	2.25%
case57	0.29%	2.57%
case118	0.91%	3.15%
case300	1.81%	7.26%

showed in Fig. 2. The MLP method worked well for small systems, but the mismatches were high for the large systems. In contrast, the average mismatches for the 1D-CNN method stayed low regardless of the size of the system. Therefore, the proposed 1D-CNN approach could effectively predict the OPF results.

4.2 10% Versus 30% Variation

The previous results were based on OPF results with a 10% variation range of the IEEE test system. In this study, we would like to test the impact of load variation ranges on the performance of the 1D-CNN and MLP methods. Specifically, we compared the OPF prediction results of a 10% variation range with a 30% variation range for each IEEE test system. The number of training data for each IEEE test system and each variation range was still 180,000 pieces. Similarly, separate 2,000 data were used to test the performance of the

well-trained 1D-CNN and MLP models for each case. The results of the comparison are shown in Table 4.2.

Note that the Case ID contained detailed information about the test case. For example, "case118 30%" means the test system is IEEE 118-Bus, and the load variation is from 70% to 130%. From Table 4.2 we could see that the load variation range indeed Table 4.2: Results of 10% vs 30% variation using 1D-CNN and MLP methods

Case ID	CNN MR	MLP MR
case30 10%	1.16%	2.25%
case30 30%	2.46%	3.57%
case57 10%	0.29%	2.57%
case57 30%	0.79%	5.14%
case118 10%	0.91%	3.15%
case118 30%	1.25%	4.25%
case300 10%	1.81%	7.26%
case300 30%	2.55%	11.79%

had an impact on the accuracy of the OPF prediction results. The average mismatch of a 30% variation range was higher than the mismatch of a 10% variation range for all the IEEE test systems, no matter whether the machine learning methods were used. The results were as expected because both 1D-CNN and MLP methods had to learn a wider range of outputs for the 30% variation situation. However, compared with the MLP results, the 1D-CNN still presented a much higher OPF prediction accuracy. The average mismatches for the 1D-CNN method were below 3% in all the cases. Therefore, the proposed 1D-CNN method could be used to accurately predict OPF results.

Chapter 5. Sensitivity Study on Training Data Size

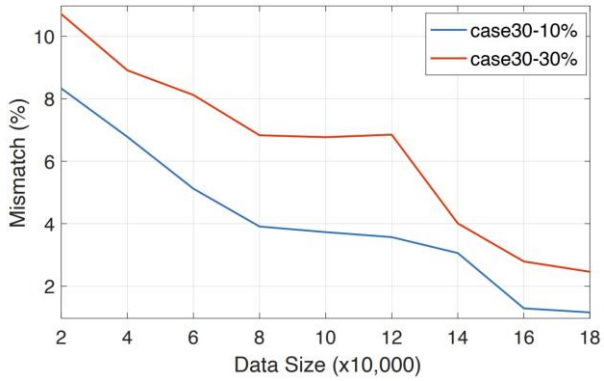


Figure 5.1: Sensitivity studies for IEEE 30-Bus system with different training data size.

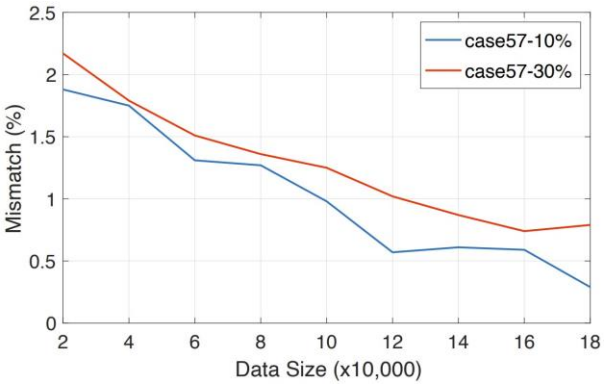


Figure 5.2: Sensitivity studies for IEEE 57-Bus system with different training data size.

It is well known that, machine learning methods require a large amount of training data to achieve a good performance. In this study, we investigated the impact of training data size on the performance of the proposed 1D-CNN method. Specifically, for each

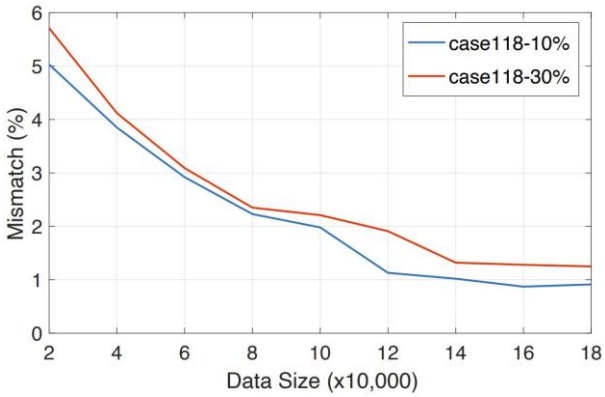


Figure 5.3: Sensitivity studies for IEEE 118-Bus system with different training data size.

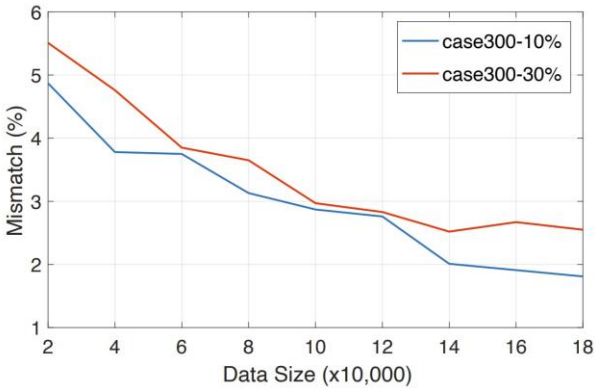


Figure 5.4: Sensitivity studies for IEEE 300-Bus system with different training data size.

IEEE test system, we randomly selected 20,000, 40,000, 60,000, 80,000, 100,000, 120,000, 140,000, 160,000, and 180,000 pieces of training data to train the 1D-CNN model. After the 1D-CNN model was well trained, similarly, separate 2,000 data were used to test the performance. Again, we used the average mismatch to evaluate the performance. The results of the 1D-CNN prediction mismatches for IEEE 30-, 57-, 118-, and 300-Bus systems were shown in Fig.5.1 to Fig.5.4, respectively.

For either 10% or 30% load variation, we could see an obvious decline of the mismatch when the training data size increased for any IEEE test system. The result was reasonable as the increase in training data size would enhance the performance of machine learning methods. We also noticed that the mismatch of a 30% load variation was still higher than the 10% load variation situation, which was consistent with the conclusions we found in

Table II. We also noted that the results of the IEEE 30-Bus system were interesting, as its mismatch was very high (around 10%) when the training data size is small, while the mismatch could be significantly decreased with more training data. If we set a 2% mismatch as the goal, the required training data sizes were 150,000 for the 30-Bus system, 20,000 for the 57-Bus system, 100,000 for the 118-Bus system, and 140,000 for the 300-Bus system. The general trend was that the large the test system, the more training data were required to achieve good performance for the 1D-CNN method.

Chapter 6. Conclusions for OPF Estimation

In this thesis, we proposed a novel 1D-CNN based approach for predicting the OPF results. Compared with traditional machine learning method such as the MLP, the proposed 1D-CNN approach have been proven to be more accurate. The superiority of the 1D-CNN method was validated using IEEE 30-, 57-, 118-, and 300-Bus with different load variations. Test results find that the increase in load variation will decrease the 1DCNN performance. A sensitive study has also shown that the accuracy of predicting OPF would significantly increase with more training data. The general trend was that the large the test system, the more training data were required to achieve good performance for the 1D-CNN method.

In future work, we would explore methods to increase the performance of the 1D-CNN approach in situations with larger load variation and fewer training data.

Chapter 7. HIF Data and Detection Methods

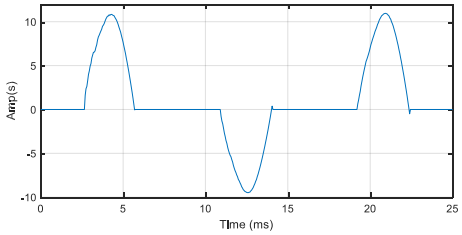
7.1 HIF Data

In this thesis, we used the HIF data in paper [41]. The nonlinear HIF data well represented the arcing effect and asymmetry characteristics. The pure fault current at the fault point is shown in Fig. 7.1(a). The fault current will only conduct when the phase voltage is higher than certain level; thus, it only occurs around the peak phase values. The current monitored by the distribution overcurrent relays are the line current at the terminal of feeders. During the HIF scenario, the measured current by the relays is shown in Fig. 7.1(b). The original sinusoidal current waveform is distorted around the peaks (when the HIF occurs). However, the HIF does not significantly increase the magnitude of the measured current. As a result, the overcurrent relays cannot detect the existence of the HIF.

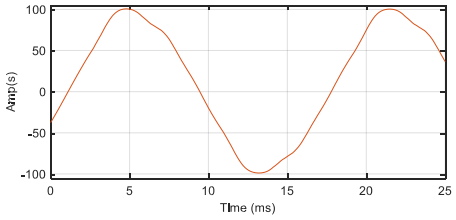
In order to accurately detect HIFs, two protective techniques are introduced: the third-harmonic-based method in [43] and the neural network-based method in [41]. We will also use these two methods to test the fidelity of the generated synthetic HIF data.

7.2 Third Harmonics-Based HIF Detection Method

One of the HIF features is that HIF owns higher second, third and fifth harmonic components than other harmonics. Therefore, this feature can be fully utilized in HIF detection. In addition, because the third harmonic component is dominant in all the harmonics, for easy implementation only the third harmonic component is used to detect



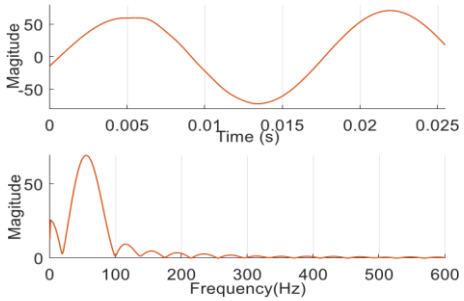
(a) Fault points current



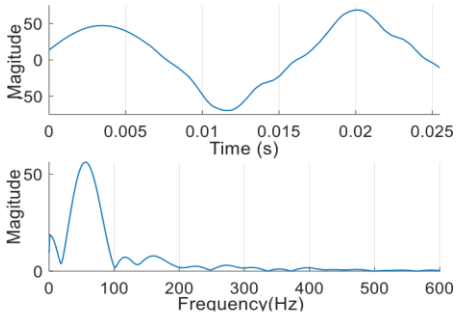
(b) Faultline current

Figure 7.1: The HIF data at fault point and line terminal

HIF in many cases. In this paper, we also use the difference in the third harmonic component between the normal transients and HIF data as a criterion of detecting HIFs.



(a) FFT analysis for normal transient



(b) FFT analysis for HIF

Figure 7.2: Example of FFT analysis for normal and HIF waveform

To find out the appropriate threshold of the third harmonic component, we randomly selected some HIF data and normal transients from capacitor switching, load changes, motor starts and transformer tap changes. Then, FFT analysis is applied to obtain the harmonic components of both normal and fault waveform. An example of the FFT analysis is shown in Fig. 7.2. Note that the third harmonic component in the HIF data is higher than that in the normal transient waveform. We tested 10 random normal transient samples and 10 random HIF data samples. The FFT analysis results are shown in Table 7.1.

Table 7.1: Percentage of third harmonics for normal and fault cases

Samples	Normal case	Fault case
1	7.04%	14.01%
2	6.98%	14.88%
3	6.93%	13.60%
4	8.81%	13.29%
5	4.99%	11.34%
6	6.53%	15.85%
7	8.77%	13.62%
8	6.55%	14.64%
9	8.39%	12.32%
10	6.34%	14.26%

Results show that there is a clear boundary between the normal cases and fault cases. In the random normal transient samples, the third harmonic components are less than 9%; in contrast, the third harmonic components are larger than 12% for the HIF cases. Therefore, it is reasonable to use 10% or 11% as the boundary between the normal transient and fault cases.

7.3 Neural Network Based HIF Detection Method

The fundamental theory of using neural network based method to detect HIF is to train a network to find out the mapping between input waveform and the corresponding labels ("fault" or "normal"). In this paper, we used the trained CNN model in [41] to test if the synthetic HIF data generated by our proposed method are qualified as "real HIF data". The well-trained CNN consists of four CNN-layers and one dense layer.

The activation functions are rectified linear unit (ReLU) and sigmoid functions for the hidden layers and the output layer, respectively. The trained network has an accuracy of 99.5% to differentiate HIF data from normal transients like load change, switching, capacitor bank change, etc. Therefore, this network can be a good judge to test the generated synthetic HIF data.

Chapter 8. Proposed Synthetic HIF Data Generation Approach

In this thesis, we use a special generative adversarial network (GAN) to generate synthetic HIF data from scratches. GAN is a typical unsupervised learning method. It uses the powerful nonlinear fitting ability of neural network to learn the nonlinear mapping from an arbitrary prior noise distribution to the real data distribution, so as to enable the generator to generate realistic samples. A GAN has two neural network modules called discriminator and generator. The generator module creates synthetic data from random noises; and the discriminator module identifies the flaws of those synthetic data. The discriminator and generator of a traditional GAN are both made up of multi-layer perceptron networks (MLP). However, its training process is not stable and unexpected gradient explosion happened a lot. In this paper, we proposed to use a deep convolutional generative adversarial network, which is known as DCGAN. The difference between traditional GAN and DCGAN is that the DCGAN integrates the CNN into a traditional GAN. In other words, at least one MLP network is replaced by the CNN. Therefore, the DCGAN leverages the translation variance and feature capturing advantages of the CNN for a better training and testing performance.

In this paper, we use a DCGAN where both the discriminator and generator are made up of CNN. The discriminator CNN has two convolutional layers and one dense layer, and the sizes of each layer are 40, 80 filters and 128 neurons, respectively. The generator CNN has one dense layer and three reversal convolutional layers, and the sizes are 960 neurons, 60, 100 and 1 filters, respectively. The activation functions of

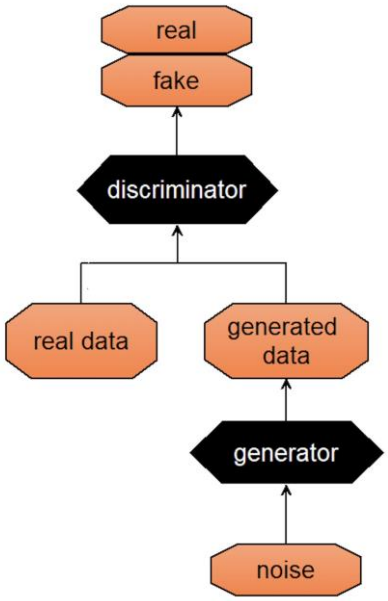


Figure 8.1: The structure of proposed DCGAN

the hidden layers of the discriminator network is ReLU. The ReLU can better capture relevant features and avoid the gradient diminishing problem, resulting in a more stable and efficient training. The output layer activation function of the discriminator network is sigmoid, which classifies the inputs into two categories: 1 (real) and 0 (fake). The activation functions of the hidden layers of the generator network is leaky-ReLU, which is very similar to the ReLU. The only difference is that the gradient for $x < 0$ in ReLU is zero while it is 0.1 in leaky-ReLU. Using leaky-ReLU could result in a slightly better performance than ReLU, but the computational burden is heavier. Because training a generator network is usually more difficult than training the discriminator network, the leaky-ReLU activation is used here in spite of the heavier computational burden.

The structure of the proposed DCGAN model is shown in Fig. 8.1. The generator will create some synthetic HIF data from random noises and try to match the real HIF data distribution. The generated synthetic data, together with real HIF data will be treated as inputs to the discriminate module, where the discriminator tries to tell the real data from synthetic (or fake) data. The DCGAN trains both the discriminator and generator through

competitions between the two modules. Specifically, the objective function of the discriminator module is

$$\max_{D} E_{x\,Pr}[logD(x)] + E_{x\,Pg}[log(1 - D(x))] \tag{8.0.1}$$

where D stands for the discriminator. Therefore, the discriminator will be optimized during the training process to find the best performance of differentiating the real and synthetic (fake) data.

In contrast, the objective function of the generator module is

$$\min_{G} E_{x\,Pr}[logD(x)] + E_{x\,Pg}[log(1 - D(x))] \tag{8.0.2}$$

where G stands for the generator. Therefore, the generator will be optimized during the training process to minimize the probability that the generated synthetic (fake) data being identified.

The combined objective function of a DCGAN is

$$\max_{G}\min_{D} E_{x\,Pr}[logD(x)] + E_{x\,Pg}[log(1 - D(x))] \tag{8.0.3}$$

The interpretation of the above objective function is that the discriminator will be optimized to find the flaws of the generator as much as possible; while the generator will be optimized to fool the discriminator as much as possible. During this competition process between the two modules, the discriminator will become better and better in identifying real HIF data from synthetic (fake) data. In return, the generator will be forced to become better and better at creating synthetic (fake) data that have almost the same distribution as real HIF data. Finally, when both modules of the DCGAN are well-trained, we can easily use the generator module to create large amounts of "high-fidelity" synthetic HIF data.

Chapter 9. Results and Discussions

9.1 Generated Synthetic HIF Data

The proposed DCGAN network reached a steady performance after 2,500 iterations. It took about 30 minutes to well train the DCGAN network when using an GeForce RTX 2070 GPU. We randomly generated 800 synthetic HIF data, and one example is shown in Fig. 9.1. The generated HIF data looks very like the real HIF data shown in
Fig. 7.1(b). However, to determine whether the synthetic HIF data can be treated as "real" HIF data and used for research purpose, we need to test the synthetic data with the two HIF detection methods.

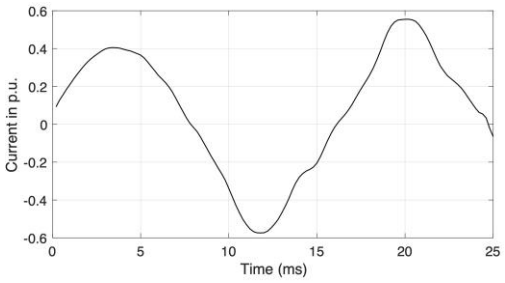


Figure 9.1: An example of generated synthetic HIF waveform

9.2 Test Results of Two HIF Detection Methods

First, the third harmonic-based HIF detection method is applied. We conducted an FFT analysis on all the 800 randomly generated synthetic HIF data, and the distribution of the third harmonic component of the 800 data is shown in Fig. 9.2. Note that the majority of the

800 synthetic data have a higher-than 10% third harmonic components, indicating that most of the generated synthetic HIF data are classified as “real HIF data” by this HIF detection method. We did some further statistical analysis with the 800 synthetic data and the results are shown in Table 9.1. The mean and median third harmonic components are 15.36% and 15.6%, respectively. If choosing 10% as the threshold to differentiate HIF data and normal transients, 787 out of all the 800 generated synthetic data can be classified as “real HIF data”. If using a threshold of 11%, then 768 out of all the 800 generated synthetic data can be classified as “real HIF data”.

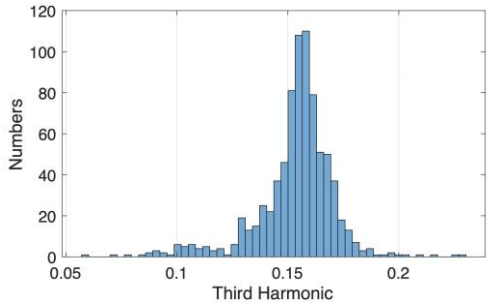


Figure 9.2: The distribution of the third harmonic component

Table 9.1: Statistics of 800 generated synthetic HIF data

3rd **Harmonic Component**	**Value**
Mean	15.36%
Median	15.6%
# > 10%	787
# > 11%	768

Then, the neural network based HIF detection method is applied to determine if the synthetic HIF data are qualified. We randomly generated three groups of synthetic HIF data, and each group contains 800 pieces of data. Note that the generator of the DCGAN network takes random noises as inputs; thus, the generated HIF data are also random. When using the neural network based HIF classifier in [41], the output is either “Positive” (means the input is HIF) or “Negative” (means the input is not HIF). The results of the neural network based method are shown in Table 9.2. All of the three groups of synthetic data show very good performance of the DCGAN. Averagely about 781 out of the 800 randomly generated

synthetic HIF data are classified as “real HIF data”, which represents a very high percentage (97.6%).

Based on the waveform of the synthetic data shown in Fig. 9.1, and the extensive test

Table 9.2: Accuracy Rate of Generated Fault Data Recognition

Synthetic Data	Positive	Negative	Accuracy
Group I	783	17	97.8%
Group II	786	14	98.3%
Group II	774	26	96.8%
Average	781	19	97.6%

results of two different HIF detection methods, we can see that most of the generated synthetic HIF data can be treated as “real HIF data”. Therefore, using the DCGAN to generate HIF data is very promising in solving the data limitation problem of the HIF related researches.

Chapter 10. Conclusions for HIF Detection

This thesis presented a synthetic HIF data generation approach using the DCGAN. The DCGAN contains two CNN modules: the discriminator and generator. The competition between the two modules results in high-fidelity synthetic HIF data. The generated synthetic data have been tested using the third harmonic based and neural network based HIF detection methods. Extensive results have demonstrated that the synthetic HIF data generated by the DCGAN can be well qualified as “real HIF data”.

Chapter 11. Overall Conclusions and Future Work

In the thesis, I explored two deep learning applications in the power flow estimation and high impedance fault detection.

Through the experiments of these two parts, it is proved that deep learning has strong advantages in power flow estimation and high Impedance fault detection, which is obviously better than traditional methods. In the experiment, the convenience and accuracy of deep learning have been confirmed. Because the model of deep learning is like human brain, it will become more accurate and efficient in continuous learning.

In future work, we would explore more deep learning application scenarios, In the field of power systems, there are countless application scenarios where machine learning can be used to work, like the CNN, RNN and some others could be used in sequence analysis, voltage detection, high voltage line inspection, fault identification, peak and valley prediction and so on.

I believe in the near future, with the increase of computing power and the evolution of models, machine learning will be even more brilliant in this field.

Bibliography

[1] Kun Yang, Wei Gao, Rui Fan, Tianzhixi Yin, and Jianming Lian. Synthetic high impedance fault data through deep convolutional generated adversarial network.

[2] Kun Yang, Wei Gao, and Rui Fan. Optimal power flow estimation using onedimensional convolutional neural network. In *2021 North American Power Symposium (NAPS)*, pages 1–6, 2021.

[3] ILLINOIS Information Trust Institute. Illinois center for a smarter electric grid (icseg), ieee 30-bus system. 1961.

[4] ILLINOIS Information Trust Institute. Illinois center for a smarter electric grid (icseg), ieee 57-bus system. 1961.

[5] ILLINOIS Information Trust Institute. Illinois center for a smarter electric grid (icseg), ieee 118-bus system. 1961.

[6] Yu-Shuai Li, Da-Zhong Ma, Hua-Guang Zhang, and Qiu-Ye Sun. Critical nodes identification of power systems based on controllability of complex networks. *Applied Sciences*, 5(3):622–636, 2015.

[7] O. Alsac, J. Bright, M. Prais, and B. Stott. Further developments in lp-based optimal power flow. *IEEE Transactions on Power Systems*, 5(3):697–711, 1990.

[8] M.A. Abido. Optimal power flow using particle swarm optimization. *International Journal of Electrical Power Energy Systems*, 24(7):563–571, 2002.

[9] LineRoald and G¨oran Andersson. Chance-constrained ac optimal power flow: Reformulations and efficient algorithms. *IEEE Transactions on Power Systems*, 33(3):2906–2918, 2018.

[10] Danny Pudjianto, S. Ahmed, and Goran Strbac. Allocation of var support using lp and nlp based optimal power flows. 2002.

[11] Brian Stott, J.L. Marinho, and O. Alsac. Review of linear programming applied to power system rescheduling. pages 142 – 154, 06 1979.

[12] K.R.C. Mamandur and R.D. Chenoweth. Optimal control of reactive power flow for improvements in voltage profiles and for real power loss minimization. *Power Apparatus and Systems, IEEE Transactions on*, PER-1:3185 – 3194, 08 1981.

[13] S. Ahmed, G. Strbac, L. Yao, A. Dixon, A. Chebbo, A. Ekwue, and D.T.Y. Cheng. Method for green field security-constrained allocation of reactive support. *Generation, Transmission and Distribution, IEE Proceedings-*, 146:65 – 71, 02 1999.

[14] Hua Wei, H. Sasaki, Junji Kubokawa, and R. Yokoyama. An interior point nonlinear programming for an interior point nonlinear programming for optimal power flow problems with a novel data structure. ieee trans power syst. *Power Systems, IEEE Transactions on*, 13:870 – 877, 09 1998.

[15] Gerald F. Reid and Lawrence Hasdorff. Economic dispatch using quadratic programming. *IEEE Transactions on Power Apparatus and Systems*, PAS-92(6):2015–2023, 1973.

[16] Rabih A. Jabr. A primal-dual interior-point method to solve the optimal power flow dispatching problem. *Optimization and Engineering*, 4:309–336, 2003.

[17] Ray Daniel Zimmerman, Carlos Edmundo Murillo-S´anchez, and Robert John Thomas. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on Power Systems*, 26(1):12– 19, 2011.

[18] Mohamed Ebeed, Salah Kamel, and Francisco Jurado. Chapter 7 - optimal power flow using recent optimization techniques. In Ahmed F. Zobaa, Shady H.E. Abdel

Aleem, and Almoataz Youssef Abdelaziz, editors, *Classical and Recent Aspects of Power System Optimization*, pages 157–183. Academic Press, 2018.

[19] Yushuai Li, David Wenzhong Gao, Wei Gao, Huaguang Zhang, and Jianguo Zhou. Double-mode energy management for multi-energy system via distributed dynamic event-triggered newton-raphson algorithm. IEEE Transactions on Smart Grid, 11(6):5339–5356, 2020.

[20] Yushuai Li, David Wenzhong Gao, Wei Gao, Huaguang Zhang, and Jianguo Zhou. A distributed double-newton descent algorithm for cooperative energy management of multiple energy bodies in energy internet. *IEEE Transactions on Industrial Informatics*, 17(9):5993–6003, 2021.

[21] Neel Guha, Zhecheng Wang, Matt Wytock, and Arun Majumdar. Machine learning for ac optimal power flow. *ArXiv*, abs/1910.08842, 2019.

[22] Jubeyer Rahman, Cong Feng, and Jie Zhang. Machine learning-aided security constrained optimal power flow. *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pages 1–5, 2020.

[23] Stephen M. Frank and Steffen Rebennack. A primer on optimal power flow: Theory, formulation, and practical examples. 2012.

[24] James Edward King, Samuel C. E. Jupe, and Philip C. Taylor. Network statebased algorithm selection for power flow management using machine learning. *IEEE Transactions on Power Systems*, 30:2657–2664, 2015.

[25] Tianyu Zhao, Xiang Pan, Minghua Chen, Andreas Venzke, and Steven H. Low. Deepopf+: A deep neural network approach for dc optimal power flow for ensuring feasibility. *2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–6, 2020.

[26] W Beaty. Ieee issues high-impedance fault detection report. *Electric Light and*

Power, 74(10), 1996.

[27] B Don Russell. Detection of downed conductors on utility distribution systems. Institute of Electrical and Electronics Engineers, 1989.

[28] Tammy Gammon and John Matthews. The historical evolution of arcing-fault models for low-voltage systems. In *1999 IEEE Industrial and Commercial Power Systems Technical Conference*, pages 6–pp. IEEE, 1999.

[29] Mark Adamiak, Craig Wester, Manish Thakur, and Chuck Jensen. High impedance fault detection on distribution feeders. *GE Industrial solutions*, 2006.

[30] LD Mitchell. Improved methods for the fast fourier transform (fft) calculation of the frequency response function. 1982.

[31] M Sedighizadeh, A Rezazadeh, and Nagy I Elkalashy. Approaches in high impedance fault detection a chronological review. *Advances in Electrical and Computer Engineering*, 10(3):114–128, 2010.

[32] N. Zamanan and J.K. Sykulski. Modelling arcing high impedances faults in relation to the physical processes in the electric arc. 01 2006.

[33] D. Hou and Normann Fischer. Deterministic high-impedance fault detection and phase selection on ungrounded distribution systems. *IEEE Industrial Commercial Power Systems Technical Conference*, pages 1–10, 01 2007.

[34] RE LEE and LA KILAR. Summary and status-report on research to detect and de-energize high impedance faults on 3-phase, 4-wire distribution circuits. *IEEE Transactions on Power Apparatus and Systems*, 99:14–14, 01 1980.

[35] Keng-Yu Lien and Shi-Lin Chen. Self-tuning of fault detection threshold for high impedance fault detection. 5:277–285, 11 1998.

[36] Mehdi El-Hami. Distribution system fault location technique utilizing high frequency spectrum of fault current. 01 2023.

[37] A Siadatan, H Kazemi Karegar, and V Najmi. New high impedance fault detection. In *2010 IEEE International Conference on Power and Energy*, pages 573–576. IEEE, 2010.

[38] Carl Benner, Pat Carswell, and B Don Russell. Improved algorithm for detecting arcing faults using random fault behavior. *Electric Power systems research*, 17(1):49–56, 1989.

[39] Y. Liu, A. P. S. Meliopoulos, R. Fan, L. Sun, and Z. Tan. Dynamic state estimation based protection on series compensated transmission lines. *IEEE Transactions on Power Delivery*, 32(5):2199–2209, 2017.

[40] Milad Khani, Reza Ghazi, and Behnam Nazari. Decision support system for optimal location of hifds in real distribution network using an integrated epso-fuzzy ahp model. *IET Generation, Transmission & Distribution*, 14(9):1616–1626, 2020.

[41] Rui Fan and Tianzhixi Yin. Convolutional neural network and transfer learning for high impedance fault detection. *arXiv preprint arXiv:1904.08863*, 2019.

[42] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David WardeFarley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[43] V Torres, JL Guardado, HF Ruiz, and S Maximov. Modeling and detection of high impedance faults. *International Journal of Electrical Power & Energy Systems*, 61:163–172, 2014.

[44] Xiang Pan, Tianyu Zhao, and Minghua Chen. Deepopf: Deep neural network for dc optimal power flow, 2019.

[45] Benjamin Karg and Sergio Lucia. Efficient representation and approximation of model predictive control laws via deep learning. 06 2018.

[46] Andreas Kettner and Mario Paolone. On the properties of the power systems nodal admittance matrix. *IEEE Transactions on Power Systems*, PP, 07 2017.

[47] Lanchao Liu, Amin Khodaei, Wotao Yin, and Zhu Han. A distribute parallel approach for big data scale optimal power flow with security constraints. pages 774–778, 10 2013.

[48] Stephen Frank, Ingrida Steponaviˇce˙, and Steffen Rebennack. Optimal power flow: a bibliographic survey ii. *Energy Systems*, 3, 09 2012.

[49] Stephen Frank, Ingrida Steponaviˇce˙, and Steffen Rebennack. Optimal power flow: a bibliographic survey i. *Energy Systems*, 3, 09 2012.

[50] Yann LeCun, Patrick Haffner, L´eon Bottou, and Yoshua Bengio. Object recognition with gradient-based learning. In David A. Forsyth, Joseph L. Mundy, Vito di Gesu, and Roberto Cipolla, editors, *Shape, Contour and Grouping in Computer Vision*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pages 319–345. Springer Verlag, 1999. International Workshop on Shape, Contour and Grouping in Computer Vision ; Conference date: 26-05-1998 Through 29-05-1998.

[51] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015.

[52] R.K. Srivastava, Jonathan Masci, S. Kazerounian, Faustino Gomez, and J. Schmidhuber. Compete to compute. *Advances in Neural Information Processing Systems*, 01 2013.

[53] Forest Agostinelli, Matthew Hoffman, Peter Sadowski, and Pierre Baldi. Learning activation functions to improve deep neural networks. 12 2014.

[54] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. 09 2014.

[55] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander

Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115, 09 2014.