



**Mariana de Aragão Ribeiro Rodrigues**

**Mapping network losses and distribution line  
flows with Artificial Neural Networks**

**Dissertação de Mestrado**

Dissertation presented to the Programa de Pós-graduação em Engenharia Elétrica da PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Elétrica.

Advisor : Prof. Alexandre Street de Aguiar  
Co-advisor: Dr. Érica Telles Carlos

Rio de Janeiro  
February 2021



**Mariana de Aragão Ribeiro Rodrigues**

**Mapping network losses and distribution line  
flows with Artificial Neural Networks**

Dissertation presented to the Programa de Pós-graduação em Engenharia Elétrica da PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Elétrica. Approved by the Examination Committee.

**Prof. Alexandre Street de Aguiar**

Advisor

Departamento de Engenharia Elétrica – PUC-Rio

**Dr. Érica Telles Carlos**

Co-advisor

Departamento de Engenharia Elétrica – PUC-Rio

**Prof. Ricardo Bernardo Prada**

Departamento de Engenharia Elétrica – PUC-Rio

**Dr. Alexandre Velloso Pereira Rodrigues**

Financiadora de Estudos e Projetos – FINEP

Rio de Janeiro, February 25th, 2021

All rights reserved.

### **Mariana de Aragão Ribeiro Rodrigues**

Mariana de Aragão Ribeiro Rodrigues received her B.Sc. degree in Electrical Engineering, with emphasis on Power Systems, in 2018 from the State University of Rio de Janeiro (UERJ), Brazil. Since 2020, she has been actively participating in research projects at the Laboratory of Applied Mathematical Programming and Statistics (LAMPS) in the Department of Electrical Engineering at PUC-Rio.

#### Bibliographic data

Rodrigues, Mariana de Aragão Ribeiro

Mapping network losses and distribution line flows with Artificial Neural Networks / Mariana de Aragão Ribeiro Rodrigues; advisor: Alexandre Street de Aguiar; co-advisor: Érica Telles Carlos. – Rio de Janeiro: PUC-Rio, Departamento de Engenharia Elétrica, 2021.

v., 108 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Inclui bibliografia

1. Engenharia Elétrica – Teses. 2. Fluxo de potência;. 3. Sistemas de distribuição;. 4. *Machine Learning*;. 5. Redes Neurais Artificiais;. 6. Contratos de demanda.. I. Aguiar, Alexandre Street de. II. Carlos, Érica Telles. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título.

## Acknowledgments

I'd like to thank my family, especially my parents, Pedro and Lecy, for all their love, support, patience and advice during my entire life.

I'd also like to thank my advisor Alexandre Street, and my co-advisor Érica Telles, for all their attention and partnership during this work. Without their great orientation, this dissertation would not be possible.

I also wish to thank my professors and colleagues at UERJ, PUC-Rio and LAMPS for all their contributions during my academic life.

To the Energisa group colleagues, for providing and authorizing the use of data and information for this work.

To the CPFL Group, for technical and financial support for this work, through the Research and Development project PD-00063-3056/2019 with resources from ANEEL's RD program.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

## Abstract

Rodrigues, Mariana de Aragão Ribeiro; Aguiar, Alexandre Street de (Advisor); Carlos, Érica Telles (Co-Advisor). **Mapping network losses and distribution line flows with Artificial Neural Networks**. Rio de Janeiro, 2021. 108p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

The power flow calculation on an electric network consists of determining the network's state, power flows and electrical losses on the lines, and total losses on the feeder. In this type of problem, the system's modeling is static, and the network is represented by a set of algebraic equations and inequations. Different solution methods were proposed in the literature to perform power flow calculations. However, for distribution networks, these methods must be able to model, with sufficient details, some unique features of these systems, such as their near radial structure, the unbalanced nature of the loads, and distributed generators' insertion. Besides that, modeling the consumption pattern in distribution systems is more complex, and the line parameters are more difficult to be obtained when compared to the transmission system. Hence, applying traditional methods for power flow calculations in distribution networks may lead to divergent solutions. Within this context, this work proposes a new approach for power flow calculations in distribution systems based on Machine Learning. The proposed models use Artificial Neural Networks (ANNs) to predict the active internal losses of a distribution network and the power flows at the borders with the transmission system. Numerical simulations demonstrate the effective performance of the proposed approach, as well as its computational advantages over benchmark software programs since, once trained, ANNs can approximate power flow calculations extremely fast, as only matrix operations are needed. Moreover, the work presents an application of the ANN methodology proposed: predictions of the flows at the borders with the transmission network were used to generate optimal demand contracts for a real distribution system in Brazil.

## Keywords

Power flow; Distribution systems; Machine Learning; Artificial Neural Networks; Demand contracts.

## Resumo

Rodrigues, Mariana de Aragão Ribeiro; Aguiar, Alexandre Street de; Carlos, Érica Telles. **Mapeamento de perdas elétricas e fluxos de potência em linhas de distribuição com Redes Neurais Artificiais**. Rio de Janeiro, 2021. 108p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

O cálculo do fluxo de potência em uma rede elétrica consiste em determinar o estado da rede, os fluxos e perdas elétricas nas linhas e as perdas internas totais no sistema. Nesse tipo de problema, a modelagem do sistema é estática e a rede é representada por um conjunto de equações e inequações algébricas. Diferentes métodos de solução foram propostos na literatura para realizar cálculos de fluxo de potência. No entanto, para redes de distribuição, esses métodos devem ser capazes de modelar, com detalhes suficientes, algumas características únicas desses sistemas, como sua estrutura quase radial, a natureza desequilibrada das cargas e a inserção de geradores distribuídos. Além disso, a modelagem do padrão de consumo nos sistemas de distribuição é mais complexa e os parâmetros das linhas são mais difíceis de serem obtidos, quando comparados com o sistema de transmissão. Portanto, a aplicação de métodos tradicionais para cálculos de fluxo de potência em redes de distribuição pode levar a soluções divergentes. Nesse contexto, o presente trabalho propõe uma nova abordagem para cálculos de fluxo de potência em sistemas de distribuição, baseada em *Machine Learning*. Os modelos propostos utilizam Redes Neurais Artificiais (RNAs) para prever as perdas ativas internas de uma rede de distribuição e os fluxos de potência nas fronteiras com o sistema de transmissão. Simulações numéricas demonstram o desempenho eficiente da abordagem proposta, além de suas vantagens computacionais em relação aos *softwares* normalmente utilizados nesse tipo de estudo pois, uma vez treinadas, as RNAs podem aproximar, de modo extremamente rápido, cálculos de fluxo de potência, já que apenas operações matriciais são realizadas. Além disso, o trabalho apresenta uma aplicação da metodologia proposta: as previsões, obtidas pela RNA, para os fluxos nas fronteiras com a rede de transmissão foram utilizadas para gerar contratos ótimos de demanda para um sistema de distribuição real no Brasil.

### Palavras-chave

Fluxo de potência; Sistemas de distribuição; *Machine Learning*; Redes Neurais Artificiais; Contratos de demanda.

## Table of contents

1	Introduction	14
1.1	Contributions	15
1.2	Outline	16
2	Literature Review	17
2.1	The power flow problem	17
2.2	Main searches in the literature	22
3	Methodologies	36
3.1	Classical Iterative Methods	36
3.2	ML Techniques: Artificial Neural Networks	44
4	Case Studies	55
4.1	Analysis of DG impacts on the distribution system's active losses	55
4.2	ANN prediction of the system's active losses	66
4.3	Power flows between distribution and transmission systems	70
5	Application of ANN methodology	78
5.1	Basis of the demand contracting process	78
5.2	Strategies to obtain the optimal demand contract	79
5.3	Analysis of Results	94
6	Conclusion	99
7	References	102

## List of figures

Figure 2.1	Searches conducted on the Scopus database.	22
Figure 2.2	Analysis on the subject of power flow calculations on distribution systems.	23
(a)	Main publications on the subject.	23
(b)	Main journals on the subject.	24
(c)	Number of publications on the subject per year.	24
Figure 2.3	Most relevant keywords.	25
Figure 2.4	Number of publications on the subject per year.	28
Figure 2.5	Most relevant keywords.	29
Figure 2.6	Analysis on the subject of ANNs application for power flow calculations on distribution systems.	30
(a)	Main publications on the subject.	30
(b)	Main journals on the subject.	31
Figure 3.1	Newton's method block diagram for power flow solution.	39
Figure 3.2	Non-linear single-phase load.	41
Figure 3.3	Non-linear single-phase load represented by a Norton's equivalent.	41
Figure 3.4	Current Injection Method block diagram for power flow solution	44
Figura 3.5	Multilayer Perceptron (MLP) architecture, with one hidden layer.	46
Figura 3.6	Most commonly used activation functions	47
Figura 3.7	MSE for a set of weights obtained for a given example.	49
Figura 3.8	Local and global minimum of a function	49
Figura 3.9	Error rate measured on testing examples.	51
Figura 3.10	As the number of epochs increases, the MSE value decreases until a certain point, in which the addition of new hidden neurons contribute to further reducing the error.	52
Figure 3.11	Artificial Neural Network modelling procedure.	54
Figure 4.1	Original configuration of IEEE 34 bus system.	58
Figure 4.2	Modified configuration of IEEE 34 bus system.	58

Figure 4.3	Procedure for obtaining the DG insertion for bus 808, at day 1.	61
(a)	Calculation of parameter $\alpha$ .	61
(b)	Calculation of PV generation.	61
Figure 4.4	Comparisons between active losses with and without DG insertion on the system.	62
(a)	Losses - January	62
(b)	Losses - February	62
(c)	Losses - March	62
(d)	Losses - April	62
(e)	Losses - May	62
(f)	Losses - June	62
(g)	Losses - July	62
(h)	Losses - August	62
(i)	Losses - September	62
(j)	Losses - October	62
(k)	Losses - November	62
(l)	Losses - December	62
Figure 4.5	Procedure for obtaining the DG insertion for bus 808, at day 1.	64
(a)	Calculation of the $\alpha$ factor.	64
(b)	Hourly loads at bus 808 and obtainment of the PV generation according to the calculated values for the $\alpha$ factor and the system total load.	64
Figure 4.6	Comparisons between active losses with and without DG insertion on the system.	65
(a)	Losses - January	65
(b)	Losses - February	65
(c)	Losses - March	65
(d)	Losses - April	65
(e)	Losses - May	65
(f)	Losses - June	65
(g)	Losses - July	65
(h)	Losses - August	65
(i)	Losses - September	65
(j)	Losses - October	65
(k)	Losses - November	65
(l)	Losses - December	65
Figure 4.7	Histogram of the prediction's error for the in-sample and out-of-sample data sets.	69
(a)	In-sample prediction errors.	69
(b)	Out-of-sample prediction errors.	69
Figure 4.8	Histogram of the observed and predicted loss values for the in-sample and out-of-sample data sets. OpenDSS' results are represented in blue and ANN's predictions are illustrated in orange.	70
(a)	In-sample observed and predicted loss values.	70
(b)	Out-of-sample observed and predicted loss values.	70

Figure 4.9	System at instant $t$ .	71
Figure 4.10	Histograms of out-of-sample prediction errors for the flows on the borders.	74
(a)	Prediction Errors - Border 1	74
(b)	Prediction Errors - Border 2	74
(c)	Prediction Errors - Border 3	74
(d)	Prediction Errors - Border 4	74
(e)	Prediction Errors - Border 5	74
(f)	Prediction Errors - Border 6	74
(g)	Prediction Errors - Border 7	74
(h)	Prediction Errors - Border 8	74
(i)	Prediction Errors - Border 9	74
(j)	Prediction Errors - Border 10	74
(k)	Prediction Errors - Border 11	74
(l)	Prediction Errors - Border 12	74
Figure 4.11	Histograms of the observed and predicted borders flows for the out-of-sample data set.	76
(a)	Observed x Predicted - Border 1	76
(b)	Observed x Predicted - Border 2	76
(c)	Observed x Predicted - Border 3	76
(d)	Observed x Predicted - Border 4	76
(e)	Observed x Predicted - Border 5	76
(f)	Observed x Predicted - Border 6	76
(g)	Observed x Predicted - Border 7	76
(h)	Observed x Predicted - Border 8	76
(i)	Observed x Predicted - Border 9	76
(j)	Observed x Predicted - Border 10	76
(k)	Observed x Predicted - Border 11	76
(l)	Observed x Predicted - Border 12	76
Figure 5.1	Main stages for optimal demand contracting strategies.	80
Figure 5.2	First Selection of scenarios for Border 9.	82
Figure 5.3	Second Selection of scenarios for Border 9.	83
Figure 5.4	Third Selection of scenarios for Border 9.	83
Figure 5.5	Comparison of the optimal contract's results obtained by Organon and by the ANN for Border 10.	95
(a)	Optimal hiring result for Border 10 - Organon.	95
(b)	Optimal hiring result for Border 10 - ANN.	95

Figure 5.6 Comparison of the optimal contract's results obtained by Organon and by the ANN for Border 6. 97

(a) Optimal hiring result for Border 6 - Organon. 97

(b) Optimal hiring result for Border 6 - ANN. 97

## List of tables

Table 4.1	Training parameters for the ANN.	66
Table 4.2	Simulation times comparison for train and evaluation data sets.	67
Table 4.3	Calculated metrics for in-sample and out-of-sample predictions of system's losses.	69
Table 4.4	Training parameters for the ANN.	72
Table 4.5	Simulation times comparison for train and evaluation data sets.	72
Table 4.6	Calculated metrics for out-of-sample prediction errors for borders flows.	75
Table 5.1	Simulation times comparison for the train and evaluation data sets.	84
Table 5.2	Nomenclature adopted in the optimization model.	89
Table 5.3	Optimal demand contracts, in MW, for each of the three contract strategies considered.	93
Table 5.4	Comparison of optimal contract parameters - Border 10 (Strategy 1).	96
Table 5.5	Comparison of optimal contract parameters - Border 6 (Strategy 1).	98

## List of abbreviations

AI – Artificial Intelligence  
ANEEL – Agência Nacional de Energia Elétrica  
ANN – Artificial Neural Network  
BF method – Backward-Forward method  
CIM – Current Injection Method  
CNN – Convolution Neural Network  
CVaR – Conditional Value at Risk  
DG – Distributed Generation/Generator  
DLL – Dynamic Link Library  
DNN – Deep Neural Network  
GA – Genetic Algorithm  
HVDS – High Voltage Distribution System  
IRG – Internal Renewable Generation  
LVDS – Low Voltage Distribution System  
MVDS – Medium Voltage Distribution System  
MAE – Mean Absolute Error  
MAPE – Mean Absolute Percentage Error  
MCS – Monte Carlo Simulation  
ML – Machine Learning  
MLP – Multilayer Perceptron  
MSE – Mean Square Error  
MPP – Maximum Power Point  
MUST – Montante de Uso do Sistema de Transmissão  
ONS – Operador Nacional do Sistema  
OpenDSS – Open Distribution System Simulator  
OPF – Optimal Power Flow  
PC elements – Power Conversion elements  
PPF – Probabilistic Power Flow  
PV generation – Photovoltaic generation  
ReLU – Rectified Linear Unit  
RNN – Recurrent Neural Network  
RTP – Revisão Tarifária Periódica  
TSO – Transmission System Operator  
VaR – Value-at-Risk

# 1

## Introduction

In a power system, the determination of voltages at the buses, power flows on the lines, and electrical losses on the feeders is mandatory for obtaining the network's state and, hence, for its proper operation. Those variables can be determined through power flow calculations, which constitute a set of algebraic nonlinear equations/inequations that can be solved using either classical iterative methods or Machine Learning (ML) techniques.

Distribution systems, however, belong to a category of “ill-conditioned” power systems due to their unbalanced nature, which has been intensified, nowadays, by the growth in the installation of distributed generation units in these networks (Shirmohammadi et al., 1988; Cheng and Shirmohammadi, 1995). Thus, power flow calculations in distribution networks require modified approaches, such as the Backward-Forward Method and the Current Injection Method (Kocar et al., 2014).

The new renewable generation units being installed in the distribution networks add complexity to the grids since they may overload lines and other equipment due to the reverse flows they produce (Donnot et al., 2017; Pertl et al., 2016). Hence, they introduce a significant level of stochasticity to the system planning (Fioretto et al., 2019). Also, most of these units are customer-owned, meaning that distribution utilities do not have control over their siting locations, leading to impacts on the operation and planning of distribution systems (Barker and Mello, 2000).

Besides the aforementioned difficulties in power flow calculations and on the system planning and operation, other common challenges faced by distribution operators may include the impact on voltage conditions at customers and utility equipment. These impacts may be positive, improving power quality and reducing electrical losses, for example, or negative, when they have the opposite effect (Barker and Mello, 2000). In this work, we analyze the impacts of distributed generation in the active losses of a distribution system, which constitute an important economic indicator for the assessment of utilities (Hsu et al., 1995).

In addition to the impacts caused by the insertion of distributed generation on the grids, another critical aspect for distribution operators consists of obtaining power flows in the borders between the distribution and transmission systems. The determination of these power flows is of paramount importance for the distributor's costs due to the fact that, at each connection point, dis-

tributors must pay a tariff for the use of the transmission system, applied to an annual demand contract (Telles et al., 2018).

As mentioned, ML techniques may constitute a modern approach to power flow calculations in distribution systems. Among these techniques, Artificial Neural Networks (ANNs) consist of the main approach for power flow studies due to the following reasons: they do not require any prior knowledge of the variables related to the problem; can handle situations of incomplete information; are extremely useful for solving nonlinear problems (Fikri et al., 2018).

In the literature, ANNs are addressed to many different applications in the power flow analysis, such as: suggest corrective actions to operators, in order to prevent violations of operational limits (Donnot et al., 2017; Donnot, 2019); obtain load estimations (Cataliotti et al., 2019); estimate losses in a power system (Hsu et al., 1995; Leal et al., 2006; Chao et al., 2017; Kahef et al., 2018); estimate bus voltage magnitudes and angles (Ivanov et al., 2014); approximate and speed up calculations in Probabilistic Power Flow problems (Yang et al., 2020; Xiang et al., 2020a,b; Chatzos et al., 2020), for example.

In this work, we explore the use of ANNs as a potential power flow technique for distribution systems that can act both in predicting the system's losses and the power flows on the system's lines. Once an ANN is trained, we show that it can predict active losses or power flows much faster than classical approaches, considered, in this work, as benchmark methods, and with good accuracy. Finally, we compare the values obtained for demand contracts in the borders between transmission and distribution systems, given the power flows obtained by the benchmark methods and predicted by the trained ANN.

To that end, we propose ANN architectures that consider, as inputs, only values for generations and loads for each system's bus. This eliminates the need for topology features, such as line resistance and reactance values, which are difficult to be obtained in the case of distribution networks (Hsu et al., 1995). Numerical simulations are conducted based on a modified version of the IEEE 34-bus system containing photovoltaic generations and on a real Brazilian distribution system.

## 1.1

### Contributions

The main contributions of this work are:

1. Review classical methods for power flow calculations and their applications at the distribution level.

2. Review applications of Artificial Intelligence (AI) techniques in power systems.
3. Propose novel formulations for power flow studies at the distribution level, based on the ANNs methodology. In these studies, our goal was to train the ANNs to predict the system's losses, and power flows at the borders with the transmission system. To do that, we adopted, as case studies, the IEEE 34-bus system and a real Brazilian distribution system, respectively. The results demonstrate that, for both cases, the ANNs predictions are able to meet, with good accuracy, the values for active losses and active flows obtained by the software OpenDSS (EPRI, 2016) and Organon (HPPA, 2018), respectively (we emphasize that the use, by itself, of these programs also constitutes a contribution of this work and will be discussed in Chapters 3 and 4). Besides that, the potential reduction in the time required for traditional methods to run power flow calculations and eliminating the necessity of knowing many topology features for each operating point simulated can be highlighted.
4. Apply the proposed methodology for power flow studies in distribution systems to generate demand contracts at the borders with the transmission system, reducing the time required to simulate the scenarios considered in the optimization model that returns the optimal contracts.

## 1.2

### Outline

The remainder of this work is organized as follows: Chapter 2 presents a Literature review of methods for power flow calculations in distribution systems based on traditional methods, AI techniques, and ANNs. Chapter 3 describes the methodologies adopted by classical iterative methods, such as Newton's Method and the Current Injection Method, and modern approaches, such as ANNs. In Chapter 4, we present the systems considered for our case studies and the simulations conducted in each case, using the software OpenDSS and Organon; numerical results from the ANNs predictions are also presented. Chapter 5 details the demand contracting process adopted by distributors in Brazil, presents the optimization model employed to generate the optimal contracts, and compares the results for the demand contracts obtained from the benchmark software (Organon) with the ones obtained from the ANNs for some of the borders between the Brazilian transmission system and the distribution system considered. Finally, relevant conclusions and future studies are drawn in Chapter 6.

## 2

## Literature Review

### 2.1

#### The power flow problem

The power flow calculation on an electric network consists of determining the network's state (voltage magnitude and phase angle in all of the buses), power flow on the lines, electrical losses on the lines, and total losses on the feeder. In this type of problem, the system's modeling is static, and the network is represented by a set of algebraic equations and inequations. Therefore, the power flow calculation can be solved using computational methods specifically developed to solve the set of equations/inequations that constitute the static model of the network (Grainger and Stevenson Jr., 1994; Monticelli and Garcia, 2015).

There are two primary considerations in developing effective solutions for the power flow and system loss calculations: formulation of a mathematical description of components, such as transformers, feeders, shunt elements, and loads, and the selection of a numerical method (Chen et al., 1991).

The basic power flow equations are obtained from Kirchhoff's and Ohm's Laws, and the inequations correspond to operational constraints of the network and its components, such as generation limits and capacities of the transmission lines. In the basic formulation of the problem, for each bus, there are four associated variables, two of which must be given as an entry, and the other two must be obtained from the solution (Monticelli and Garcia, 2015):

- $V_k$ : voltage magnitude at bus  $k$ ;
- $\theta_k$ : voltage phase angle at bus  $k$ ;
- $P_k$ : liquid injection of active power at bus  $k$  (*generation – load*);
- $Q_k$ : liquid injection of reactive power at bus  $k$ .

Depending on which variables are known *a priori* and which are considered unknown, we can define three types of buses (Monticelli and Garcia, 2015):

- PQ: variables  $P_k$  and  $Q_k$  are given, while  $V_k$  and  $\theta_k$  must be calculated;
- PV: variables  $P_k$  and  $V_k$  are given, while  $Q_k$  and  $\theta_k$  must be calculated;
- V $\theta$  (slack bus): variables  $V_k$  and  $\theta_k$  are given, while  $P_k$  and  $Q_k$  must be calculated.

PQ and PV buses are used to represent load and generation buses, respectively. The slack bus gives the system's angular reference, and it is used to balance the power generation and consumption in the network, taking into account the system's losses that are unknown before the final solution of the problem (Monticelli and Garcia, 2015).

The set of equations for the power flow problem is composed by two equations for each bus, representing Kirchhoff's first law for the active and reactive components (Monticelli and Garcia, 2015):

$$P_k = \sum_{m \in \Omega_k} P_{km}(V_k, V_m, \theta_k, \theta_m) \quad (2-1)$$

$$Q_k + Q_k^{sh}(V_k) = \sum_{m \in \Omega_k} Q_{km}(V_k, V_m, \theta_k, \theta_m), \quad (2-2)$$

where:

- $k = 1, \dots, N$  (number of buses in the network);
- $\Omega_k$ : set of buses connected to bus  $k$ ;
- $V_k, V_m$ : voltage magnitudes of buses  $k$  and  $m$ ;
- $\theta_k, \theta_m$ : voltage phases of buses  $k$  and  $m$ ;
- $P_{km}$ : active power flow at branch  $k - m$ ;
- $Q_{km}$ : reactive power flow at branch  $k - m$ ;
- $Q_{km}^{sh}$ : component of reactive power injection due to the shunt element at bus  $k$ .

Equations (2-1) and (2-2) can be rewritten as:

$$P_k = V_k \sum_{m \in K} V_m (G_{km} \cos \theta_{km} + B_{km} \sin \theta_{km}) \quad (2-3)$$

$$Q_k = V_k \sum_{m \in K} V_m (G_{km} \sin \theta_{km} - B_{km} \cos \theta_{km}), \quad (2-4)$$

where  $K$  is the set of buses connected to bus  $k$  (including bus  $k$ ),  $G_{km}$  and  $B_{km}$  correspond to the real and imaginary elements of the admittance matrix, respectively.

For each bus, there are four associated variables ( $P_k$ ,  $Q_k$ ,  $V_k$  and  $\theta_k$ ). Since there are  $N$  buses in the system, we have  $2N$  equations and  $4N$  variables, some of which are previously defined, according to the bus category, resulting in a system of  $2N$  equations and  $2N$  variables unknown. However, the sets of equations (2-3) and (2-4) can be altered to remove equations referring to the slack bus since variables  $P$  and  $Q$  for this bus can be directly obtained from the calculation of the remaining variables. The resulting system has  $2N - 2N_{V\theta}$  dimensions, where  $N_{V\theta}$  is the number of slack buses in the system (usually,

there is only one slack bus). We have a similar situation for PV buses since their reactive power can also be directly obtained from the results of the remaining variables, leading us to a system of  $2N - 2N_{V\theta} - N_{PV}$  dimensions, where  $N_{PV}$  is the number of PV buses. It is important to note that PV and slack buses present limits for the injection of reactive power that must be respected in the problem's solution (Monticelli and Garcia, 2015). Therefore, the power flow problem usually takes the form of a non-convex problem, with non-convex constraints (Donnot, 2019).

Different power flow methods are used for determining the steady-state operation conditions by calculating currents, voltages, and active/reactive powers (Kahef et al., 2018). Traditionally, the Newton-Raphson and the fast decoupled power flow solution techniques, and a large number of their derivatives, were used to solve the power flow problem for “well-behaved” power systems (Shirmohammadi et al., 1988). Some of these methods will be presented in more detail in the next Chapter.

According to (Shirmohammadi et al., 1988), the scientific community has been aware of the shortcomings of those traditional methods when they were generically implemented and applied to “ill-conditioned” and/or poorly initialized power systems for a few decades. Hence, commercial power flow software always attempts to modify those algorithms to improve the solution's robustness. However, the nature of the modifications and the degree of improvement obtained vary for different methods. The Gauss-Seidel power flow technique, for example, another traditional power flow method, although very robust, can be extremely inefficient for solving large power systems.

Distribution networks, for example, due to their wide-ranging resistance and reactance values and radial structure, fall into that category of “ill-conditioned” power systems for the traditional power flow methods. Attempts to use the Newton-Raphson method for solving distribution networks are, in general, unsuccessful, leading to a divergence in the solution (Shirmohammadi et al., 1988).

Power flow methods for distribution networks must be able to model, with sufficient details, some unique features of these systems, such as: radial or near radial structure; multi-phase, unbalanced, grounded or ungrounded operation; distributed generation; multi-phase, multi-mode control distribution equipment; unbalanced distributed loads; a huge number of branches/nodes (Cheng and Shirmohammadi, 1995).

A detailed representation of a distribution network requires: the three-phase representation of the network to account for actual load unbalances; the distributed load representation along the distribution lines; the mathematical

representation of elements such as voltage regulators, loads, shunt capacitors, among others, in order to approximate their physical behavior (Shirmohammadi et al., 1988; Chen et al., 1991).

Voltage regulators can be three-phase or single-phase and are connected in the substation and/or to a specified line segment. On the other hand, loads can be connected at a node (spot loads) or assumed to be uniformly distributed along a line section (distributed loads). Loads, as well as shunt capacitor banks, may be three-phase (balanced or unbalanced), connected in wye or delta configurations, or single-phase, connected line-to-ground or line-to-line. All loads can be modeled as constant power (PQ), constant impedance (Z), or constant current (I), while capacitors are modeled as constant susceptance at nameplate rated kVAr (Kersting, 2001).

The problem of system imbalance has considerable effects on power systems studies. Although the effects of zero sequence current on protection relays and negative sequence current on motors are well known by distribution engineers, other effects such as increasing system loss, decreasing system capacity, and increasing the inductive coupling between parallel lines or feeders are sometimes overlooked (Chen et al., 1991).

Based on that, the unbalanced nature of distribution systems requires multi-phase power flow solutions to handle arbitrary network topologies and provide accurate results. This need for a detailed analysis of secondary grid systems located in dense urban areas and the modeling of distribution networks, including the sub-transmission levels, requires the use of highly efficient and large-scale system-capable methods (Kocar et al., 2014).

The increasing use of renewable sources is another important factor to be considered in power flow studies since it adds complexity for the secure operation of the transmission and distribution grids due to three main reasons:

- since those sources are less predictable, in cases of contingencies, that may be weather-related, operators must act quickly to protect the equipment and avoid lines to get overloaded, for example (Donnot et al., 2017).
- as a result of renewable energy sources' integration, the power flow in distribution networks is becoming more challenging, as reverse flows may occur (Pertl et al., 2016).
- the integration of renewable sources in sub-transmission and distribution systems has introduced great stochasticity, making load profiles much harder to be predicted and introducing significant variations in loads and generations, increasing the number of scenarios to be considered, while updating generation schedules, for example (Fioretto et al., 2019).

Therefore, modeling distributed generators (DGs) in distribution power flow algorithms has become an inevitable task. Depending on the contract and control status of a generator, it may be operated in one of the following modes (Cheng and Shirmohammadi, 1995):

- in “parallel operation” with the feeder, i.e., the generator is located near and designated to supply a large load with fixed real and reactive power output, resulting in a net effect of a reduced load at a particular location;
- to output power at a specified power factor;
- to output power at a specified terminal voltage.

In the power flow context, generation nodes in the first two cases can be well represented as PQ nodes, requiring little special treatment in the algorithm. Generation nodes in the third case, on the other hand, must be modeled as PV nodes, with special procedures needed to be performed in order to maintain its voltage magnitude, as well as to monitor its reactive power capability (Cheng and Shirmohammadi, 1995).

As mentioned, traditional power flow solution methods are not applicable to distribution systems (Shirmohammadi et al., 1988; Kocar et al., 2014). Common approaches to address this problem include iterative techniques, such as the Backward-Forward (BF) method, which does not require a matrix-based formulation (Kocar et al., 2014). In this method, the backward step estimates the power flow from the far bus, while the forward step estimates the power flow from the slack bus (Kahef et al., 2018).

The BF method was initially designed for radial networks and then extended to meshed networks at the expense of an increased number of iterations and/or more prominent divergence risks. Although the BF class of solvers is able to handle the solution in the phase domain and account for the unbalanced nature of distribution systems, it has topological limitations and lacks generality (Kocar et al., 2014).

According to (Kocar et al., 2014), another possible solution technique for the power flow problem in distribution systems is the fixed-point method, in which the nodal admittance matrix is obtained in the phase domain, and loads are initially linearized with passive resistance and inductance (RL) devices. In the consecutive iterations, power mismatches are represented with current injections populated in the current vector. This approach usually needs more iterations to converge than Newton’s method, but the advantage is that it remains efficient since the system nodal admittance matrix is not changed often. One approach for the fixed-point technique is the Current Injection Method, which consists of a Newton method describing mismatch equations

for nodal currents in terms of specified powers and voltages. This method will be described in more detail in Chapter 3.

The main difficulties in most of the power flow methods presented in the literature consist of the generalization path to arbitrary device connections and expandability (Kocar et al., 2014). In order to solve problems caused by varying demands of electricity, with complex production systems that include conventional power plants and, more recently, less predictable renewable sources, operators often recur to different approaches, such as the use of linear DC approximations for power flow calculations. However, this approach focuses only on the scenarios considered to be the most pertinent, and the simulations' fidelity may be compromised (Donnot et al., 2017; Fioretto et al., 2019).

Hence, we conducted searches on the literature to review approaches for power flow calculations in distribution systems. These searches are described next.

## 2.2

### Main searches in the literature

The goals of this Literature Review are: analyze power flow methods for distribution systems; map possible Artificial Intelligence (AI) techniques to address the previously mentioned challenges of power flow calculations in distribution networks; study applications of Artificial Neural Networks (ANNs) in distribution systems, which is the main technique discussed in this dissertation. In order to do that, three main searches on the Scopus database were conducted, with target concepts as illustrated in Figure 2.1.

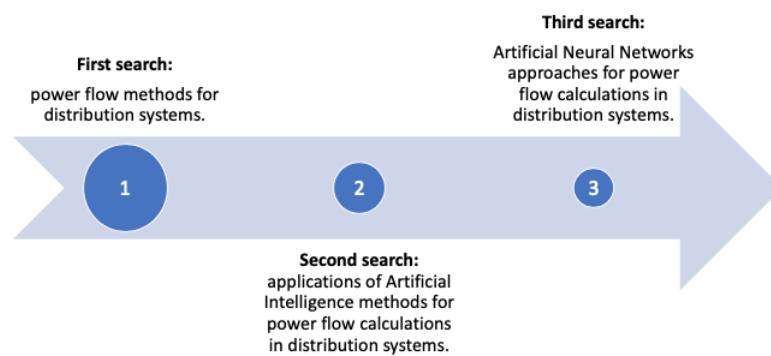


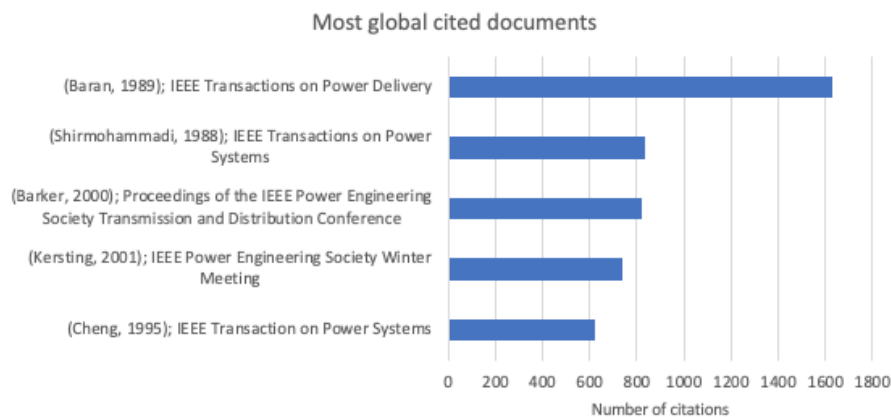
Figure 2.1: Searches conducted on the Scopus database. The Figure illustrates that the first search was broader than the second one and that the second search was broader than the third one. This way, we refined the publications obtained until our main target.

### 2.2.1

#### First Search: Power flow methods for distribution systems

For the first search, the keywords (“power flow\*” OR “load flow\*”) AND (“distribution network\*” OR “distribution system\*” OR “distribution operation\*” OR “distribution planning\*”) were applied on the database. According to (Maçaira et al., 2018), keywords should be broad enough not to restrict the number of studies and specific enough to return only the studies related to the topic.

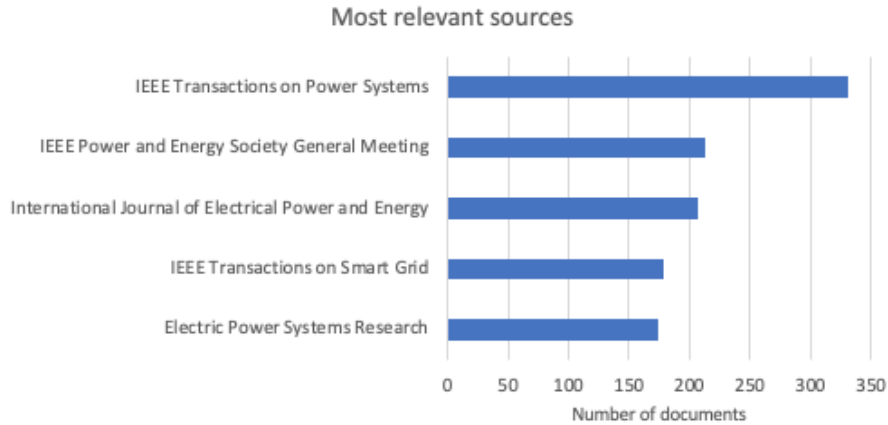
There was no time restriction for the search, and the only exclusion criteria applied were “Conference Review” documents and the language of the publications, resulting in 7075 publications written in English. Figure 2.2a presents the most relevant studies on power flow analysis in distribution systems (References (Baran and Wu, 1989; Shirmohammadi et al., 1988; Barker and Mello, 2000; Kersting, 2001; Cheng and Shirmohammadi, 1995)), while Figures 2.2b and 2.2c presents the main publication sources and the growth on the number of publications on the subject through the years, respectively, according to the data obtained from the biblioshiny application<sup>1</sup> (Aria and Cuccurullo, 2017).



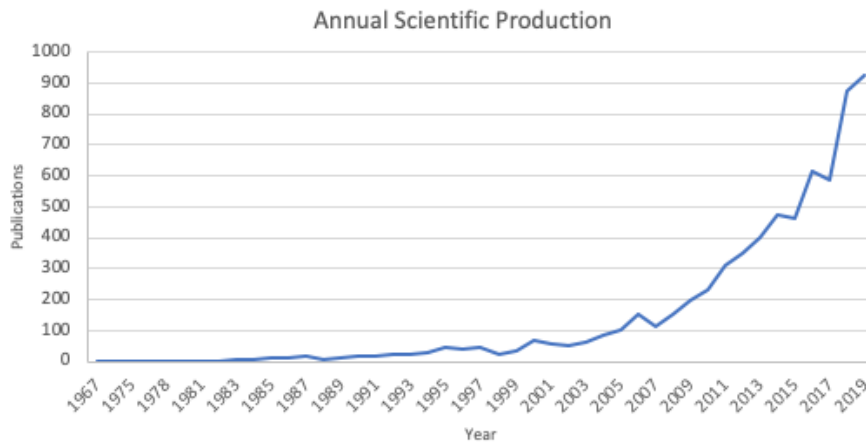
(a) Main publications on the subject.

Figure 2.2: Analysis on the subject of power flow calculations on distribution systems.

<sup>1</sup>It's worth mentioning that, although publications from the year 2020 were obtained in the search, those works were not considered on the plot of Figure 2.2c, since it deals with the scientific production of complete years and the search was conducted during the year of 2020.



(b) Main journals on the subject.



(c) Number of publications on the subject per year.

Figure 2.2: Analysis on the subject of power flow calculations on distribution systems (Continuation). Data obtained from the biblioshiny application (Aria and Cuccurullo, 2017).

Figure 2.3 indicates the authors' most used keywords. For this analysis, broad sense keywords, common to most of the publications, such as “electric load flow”, “distribution systems” and “electric power distribution”, for example, were removed.

From Figure 2.3, we can observe that the term “distributed power generation” is the most commonly used by authors in their publications due to the penetration growth of these “energy resources” in distribution networks. In recent years, much interest has been focused on the increased amounts of “renewable energy resources” on power systems. These resources have unique characteristics due to their seasonality, leading to a maximum generation potential that varies on different time scales and cannot be perfectly predicted (Barker and Mello, 2000; Ela et al., 2011).

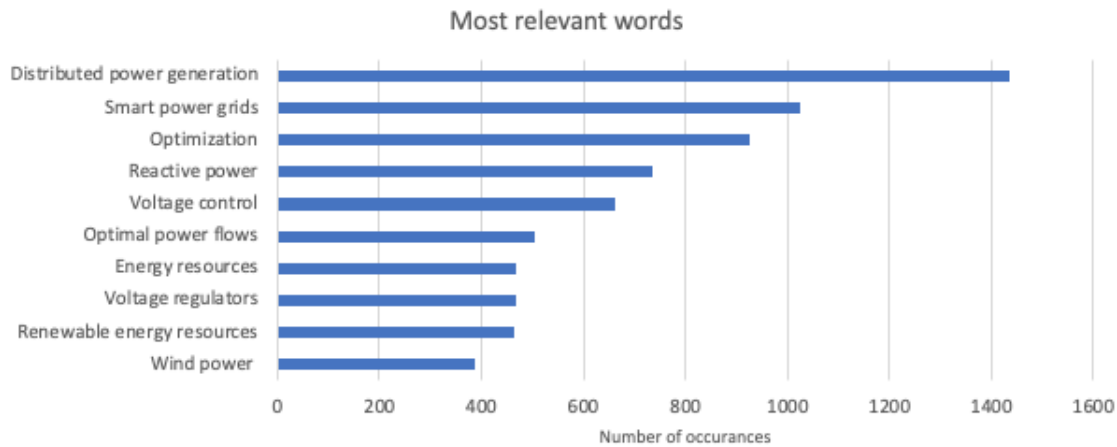


Figure 2.3: Most relevant keywords. Data obtained from the biblioshiny application (Aria and Cuccurullo, 2017).

Among the renewable energy resources, solar and “wind power” are usually the main generation sources. Wind generation typically complements solar production by generating more in the early morning and late evening and less in the middle of the day (CAISO, 2019). However, since many DGs are not utility-owned, there is no guarantee that their installation’s ideal conditions will be satisfied. This may adversely impact power system operations if specific minimum standards for control, installation, and placement are not maintained, leading to new operational challenges (Barker and Mello, 2000).

Since distribution systems were traditionally operated as radial networks, common issues associated with DGs’ penetration are related to the reverse power flows introduced by those units. Despite this power injection, flows must be kept below the maximum limits on equipment such as distribution lines and transformers. Throughout the power system, voltage levels must also be kept within nominal levels at all locations on a network. This is achieved through “reactive power” management of generators as well as by “voltage control” devices such as “voltage regulators”, transformer taps, capacitors, and reactors, for example. Systems must also be able to withstand contingency events with preventive control actions holding reserves and limiting pre-contingency flows. This way, they can survive the events, and normal operations may be completely restored shortly (Barker and Mello, 2000; Ela et al., 2011).

The aforementioned impacts on the operation of distribution systems are indicative of how the network supervision, by the distributors, will need to evolve, adopting and/or improving technologies capable of monitoring the system, providing detailed and high temporal granularity information, in addition to managing and actively controlling network devices. This evolution is linked to the concept of “smart power grids” (El-Hawary, 2014). The basis for

smart grid management is the state observability, i.e., the real-time knowledge of all the network quantities (branch currents, node voltages, and power flow), which is usually obtained by load flow algorithms, whose implementation requires an accurate model of the network and a continuous update of the input load data (Cataliotti et al., 2019). According to (Ivanov et al., 2014), smart grids should have at least the following features: two-way communication (from the consumer to the utility and from the utility to the consumer); smart metering; equipment automation and automated management; real-time monitoring and control.

Utilities that encourage DG programs may apply “optimization” techniques to obtain the optimal siting of units to enhance the system’s performance (Barker and Mello, 2000). Optimization methods can also be widely applied in power flow analysis with other purposes, for example: in network reconfiguration problems that aim to find the optimal load transfer between feeders or substations in order to reduce system’s losses and balance system’s loads while considering constraints related to voltage, lines/transformers capacities and reliability (Baran and Wu, 1989); they allow the achievement of an optimal reactive (VAR) control in order to improve voltage profiles and to minimize the system’s power losses, by proper adjustments of VAR variables, such as shunt capacitors and tap-changing transformers (Ramakrishna and Rao, 1999); they constitute the common framework to address “optimal power flow” (OPF) problems (Fioretto et al., 2019; Imen et al., 2015).

OPF problems consist of determining the generators dispatch of minimal cost that can attend the demand and the system’s physical and engineering constraints. In (Fioretto et al., 2019), the authors highlight that usually, generation schedules are updated every five minutes, but, in recent years, the integration of renewable energy in sub-transmission and distribution networks is making load profiles much harder to be predicted and is introducing significant variations in load and generation. This uncertainty leads operators to adjust generators set-points with increasing frequency to meet the demand and guarantee a safe operation of the system. However, the resolution frequency to solve OPFs is limited by their computational complexity, causing operators to use OPF approximations, such as the linear DC model, leading to solutions that may be sub-optimal, which can cause economic losses or may fail to satisfy the physical and engineering constraints (Fioretto et al., 2019).

To address those challenges, system operators may require more sophisticated methods for balancing tradeoffs between nominal performance and operational risks since penetration levels of renewable energy sources continue increasing to substantial fractions of the total supplied power, both in distri-

bution and transmission networks (Guo et al., 2019). In (Guo et al., 2019), a stochastic OPF model that aims to balance the operational costs and the total CVaR values of voltage magnitude constraints is proposed; the model considers an operational cost that captures electricity purchased by customers, excessive solar energy fed back to the utility, reactive power compensation costs and penalties for active power curtailment.

AI methods have also been applied to the OPF problem (Fioretto et al., 2019; Imen et al., 2015). The methodology proposed in (Fioretto et al., 2019), for example, approximates OPFs using Deep Neural Networks (DNNs). Since the resulting generator's set-points must satisfy the physical and engineering constraints that regulate power flows, and these constraints introduce significant difficulties for Machine Learning-based approaches, the proposed methodology exploits the problem constraints using a Lagrangian dual method.

The work presented in (Chatzos et al., 2020) demonstrates, for the first time, that DNN architectures are able to provide high-fidelity approximations for OPF problems with thousands of buses and transmission lines, expanding the results presented in (Fioretto et al., 2019). The proposed framework was evaluated on real case studies from the French transmission system and produced accurate AC-OPF approximations in the order of milliseconds.

Therefore, AI techniques constitute a modern approach to power flow analysis due to their fast calculation time (Fikri et al., 2018; Imen et al., 2015). To illustrate this, we conducted a second search on the Scopus database, described next.

### 2.2.2

#### **Second Search: Applications of AI methods for power flow calculations in distribution systems**

For the second search, we added the terms (“neural network\*” OR “artificial neural network\*” OR “deep learning” OR “machine learning”) to the keywords from the first search. The same exclusion criteria were applied, resulting in 113 publications. Figure 2.4 indicates the growth in the number of annual publications obtained from the search of the mentioned keywords. It is worth noting that the first study was published in 1990 and that a more expressive growth in the number of publications started in the year 2016, with an increase of about 44% in the annual scientific production from 2018 to 2019<sup>2</sup>.

<sup>2</sup>Once again, publications from the year 2020 were not considered on the plot of Figure 2.4, since it deals with the scientific production of complete years and the search was conducted during the year of 2020.

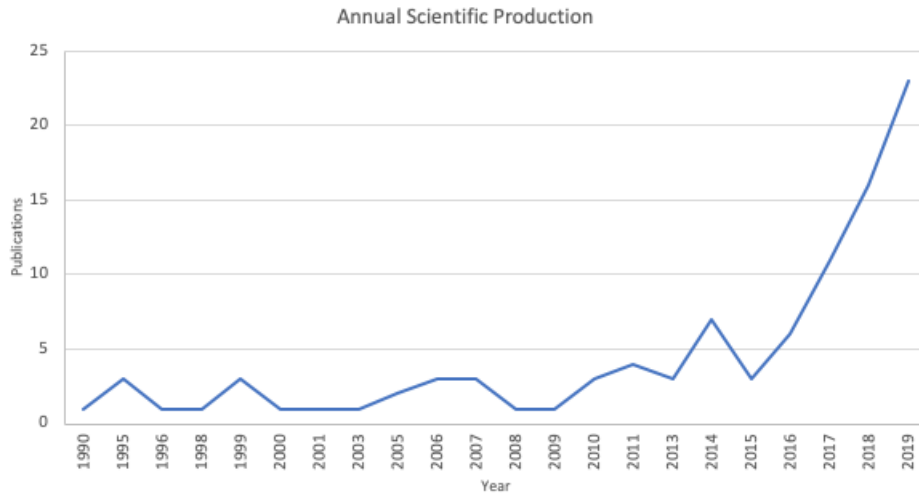


Figure 2.4: Number of publications on the subject per year. Data obtained from the biblioshiny application (Aria and Cuccurullo, 2017).

Figure 2.5 indicates the authors' most used keywords for this new search of publications. Once again, broad sense keywords, common to most of the publications, such as “electric load flow”, “neural networks” and “artificial intelligence”, for example, were removed from the analysis.

Besides some relevant words already obtained in the first search, new keywords were obtained, such as “Learning Systems”, “Genetic algorithms”, “Electric losses” and “Mathematical models”. The term “Learning Systems” refers to the three key categories of ML algorithms, described below (Misilmani and Naous, 2019):

- Supervised learning: what a correct output looks like is already known. After training the learning algorithm on a given data set, the algorithm generalizes to give accurate predictions to all possible inputs.
- Unsupervised learning: what the results should look like is unknown; the algorithm derives a structure from the data after identifying similarities in the inputs.
- Reinforcement learning: the machine receives no labeled data set. Instead, information is collected after interacting with the environment through different actions. The machine is rewarded after each action; hence, its objective is to maximize this expected average reward, where the action would become optimal.

Among the “Mathematical models” proposed by the publications, the great relevance of the keyword “Optimization” justifies the importance of the term “Genetic algorithms” in the new search. Although both ANNs and Genetic Algorithms (GAs) have biological motivations, they are very

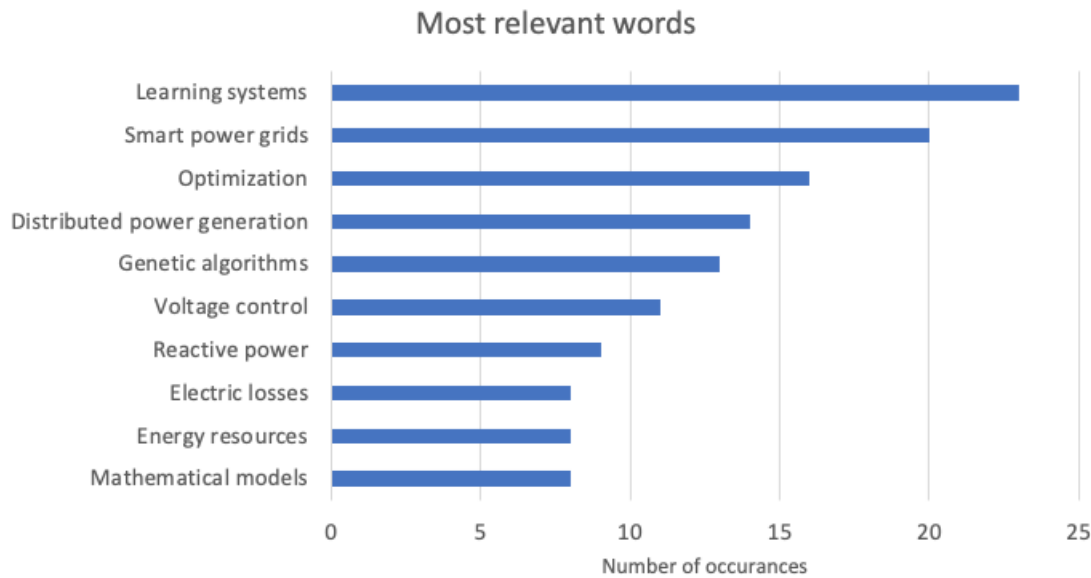


Figure 2.5: Most relevant keywords. Data obtained from the biblioshiny application (Aria and Cuccurullo, 2017).

different methods that aim to achieve distinct objectives: while ANNs can be described as computational structures inspired by the study of biological neural processing, with a learning process based on the training of a dataset and the proposal of an alternative model to describe the data (Fikri et al., 2018), GAs constitute optimization methods in which multiple search points explore optimal solution simultaneously and independently (Matsuno et al., 2003).

Also, “electric losses” in distribution systems are important economic indicators for the assessment of power supply companies since they usually appraise operating efficiency by the amount of real power loss over the whole power system. However, conventional loss analysis, using detailed system modeling, is difficult and impractical to perform due to the vast amount of data required. Therefore, Machine Learning (ML) techniques, a large subset of AI techniques, are being proposed to obtain distribution systems losses more accurately and with less effort (Hsu et al., 1995; Misilmani and Naous, 2019).

In electrical networks, bus voltage levels, load, generation, and branch power flows are interdependent variables, consisting in a problem that can be solved using the approximation capabilities of ANNs, popular tools from the ML and Computational Intelligence communities, widely and successfully employed in many fields (Cataliotti et al., 2019; Ivanov et al., 2014). In power flow analysis, ANNs play an important role due to some convenient features, such as (Fikri et al., 2018):

- they are extremely useful for solving nonlinear problems;

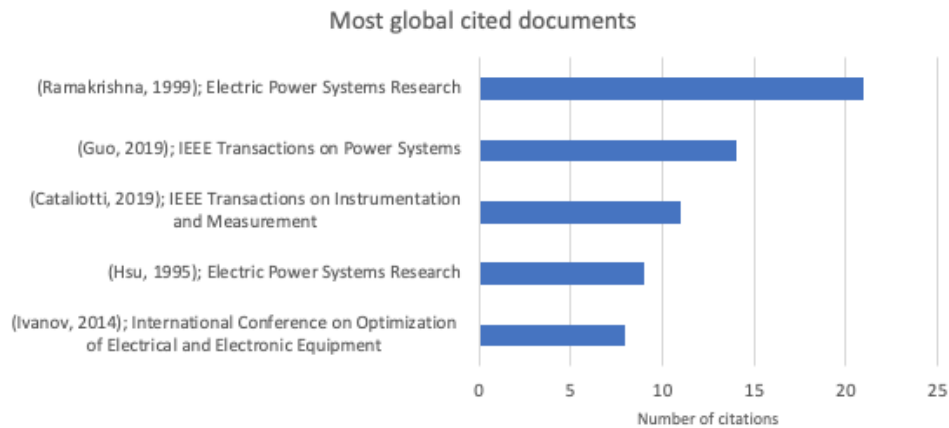
- the method is not algorithmic and does not require any prior knowledge of the variables related to the problem, such as  $P$ ,  $Q$ ,  $V$ , and  $\theta$ . Therefore, it is not necessary to make the approximations needed by traditional numerical methods;
- they can handle situations of incomplete information.

Therefore, due to the importance of this method among ML techniques for power flow analysis and, since it is the focus of this dissertation, a third search on the Scopus database was conducted, as described below.

### 2.2.3

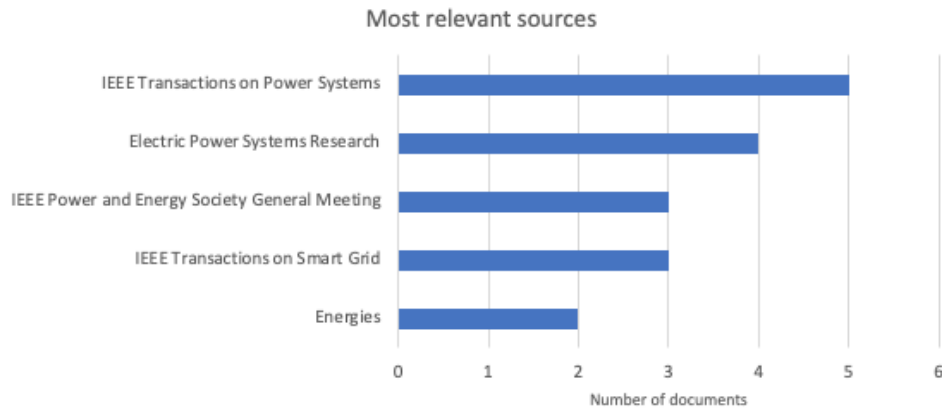
#### Third Search: ANNs approaches for power flow calculations in distribution systems

For the third search, only the subset (“neural network\*” OR “artificial neural network\*”) of the keywords from the second search was selected, resulting in 93 publications. This last search aimed to obtain the most important publications (References Ramakrishna and Rao (1999); Guo et al. (2019); Cataliotti et al. (2019); Hsu et al. (1995); Ivanov et al. (2014)) and publications sources on the subject, as depicted in Figures 2.6a and 2.6b, respectively, according to the data obtained from the biblioshiny application (Aria and Cuccurullo, 2017).



(a) Main publications on the subject.

Figure 2.6: Analysis on the subject of ANNs application for power flow calculations on distribution systems.



(b) Main journals on the subject.

Figure 2.6: Analysis on the subject of ANNs application for power flow calculations on distribution systems (Continuation). Data obtained from the biblioshiny application (Aria and Cuccurullo, 2017).

According to the literature, there are many different applications for the ANN methodology in the power flow analysis. Hsu et al. (1995) mention that, until 1995, the ANNs applications in power systems could be categorized into three main areas: regression, classification, and combinatorial optimization. However, new approaches for ANNs applications in power systems were developed through the years. For example, in (Donnot et al., 2017), ANNs were used to approximate power flows in the system's lines, and ML techniques were developed to suggest corrective actions to operators, preventing violations of thermal limits in transmission lines. In (Cataliotti et al., 2019), the authors propose a method that uses historical data to train ANNs to obtain load estimations in unavailable nodes. By using the load estimations and real available load measurements in other substations, power flows in the whole medium-voltage network could be calculated.

The establishment of models for electric loss calculations is another possible application for ANNs, overcoming the problem that traditional methods for line loss calculations depend on the grid structure and need to solve a large number of loop equations. In (Kahef et al., 2018), feed-forward networks were used to estimate losses in a system's lines. Another example is the implementation used in (Chao et al., 2017), where backpropagation neural networks were used to establish a line loss calculation model. In (Leal et al., 2006), a methodology based on ANNs was proposed to evaluate losses for each segment of distribution systems (secondary and primary networks, distribution transformers, and High-Voltage/Medium-Voltage transformers). In (Hsu et al., 1995), the authors highlight that primary and secondary conductors,

along with distribution transformers, typically contribute most to power losses in a distribution system. Besides that, variations in feeder loads and phase voltage imbalance also introduce changes in load demand and system loss. An ANN method is proposed for fast pattern recognition and regression of the feeder loss model to solve feeder loss calculations more accurately and with less effort.

Ramakrishna and Rao (1999) discuss the joint control of switched capacitors and tap-changing transformers to minimize losses while satisfying voltage constraints. The work highlights that classical techniques of Volt/VAR control in distribution systems, which include linear and nonlinear programming, are not adequate for online applications since they are inflexible with changing conditions and do not include the operator's decision-making logic. To overcome these problems, an ANN approach was proposed to solve the VAR control and design problems.

In (Pertl et al., 2016; Fikri et al., 2018; Ivanov et al., 2014), the use of an ANN eliminates the need to model the entire power system and obtain the admittance matrix to estimate bus voltages. In (Ivanov et al., 2014), an approach using Multilayer Perceptron ANN to estimate bus voltage magnitudes and angles, based on bus power injection and branch power flow measurements, is proposed. The ANN voltage estimator method does not require network data knowledge and only uses bus and branch measurements. According to the results, the ANN was able to learn the pattern of bus voltages variations with the change of hourly loads, thus replacing an iterative calculation with a more straightforward calculation. Also, in (Pertl et al., 2016), photovoltaic (PV) generations were inserted in the power system at different levels, and it was possible to observe that the estimation error increased when the PV generations reached their highest levels of penetration since those levels were not considered in the training set of the ANN.

As mentioned, the increase of uncertainties in power systems due to the integration of intermittent renewable resources has produced significant impacts on the grids' operation, planning, and control. An important tool to mitigate those impacts by quantifying how the randomness of node injection power propagates to the bus voltages, power flows, and other system operating states is the Probabilistic Power Flow (PPF). The PPF method can address random factors, such as renewable energy generation, load fluctuations, and topology features, consisting of a useful tool for analyzing power systems uncertainties (Yang et al., 2020; Xiang et al., 2020a,b).

The most common approach for the PPF problem is the Monte Carlo simulation (MCS). Although the MCS method is very accurate, it is time-

consuming since it requires solving the nonlinear power flow equations for a large number of samples. In order to improve the calculation speed of the simulation method, different solutions have been proposed in the literature, such as the use of the DC approximation, which improves the calculating speed but sacrifices the accuracy of the solutions, and the parallel computing based method that is able to calculate PPF accurately and rapidly, but is highly dependent on computational resources and, therefore, is not practical to be implemented in every power company (Xiang et al., 2020a,b). Since PPF problems subjects a large number of samples to the same computational task (the power flow calculation), one possible approach is to formulate them as ML problems (Yang et al., 2020). References (Yang et al., 2020; Xiang et al., 2020a,b) propose using DNNs to approximate and speed up the power flow calculations in PPF problems.

In (Yang et al., 2020), the authors propose an approach in which a DNN is used to approximate the power flow calculation, and its training process consists of three main steps: first of all, the branch flows are added into the objective function of the DNN as a penalty term, improving the accuracy of the approximations; second, the training speed is accelerated by removing the impact on voltage magnitudes and the relationship between reactive branch powers and phase angles, according to the physical characteristics of the transmission grid; at last, an improved initialization method for the DNN is derived to further improve the convergence efficiency.

One difficulty described in (Yang et al., 2020; Xiang et al., 2020b) is that the structure of the DNN should match the power system's size. However, with the development of renewable energy and power demand, the power grid's topology or the distribution of renewable resources may change rapidly, with new buses or branches being added to the grid. Hence, the original trained DNN may not apply to the extended system. *Transfer Learning* techniques may be adopted to address this issue and improve the scalability of DNNs for extended systems. This method applies useful knowledge from related domains to help the training of the target domains. Since the new power system is extended from the original one, power flow features are highly similar and, hence, the parameters of the original trained-DNN comprise useful knowledge for the extended system and can be used as an initialization point to train a new DNN, as it was proposed in (Xiang et al., 2020a,b).

In (Xiang et al., 2020a), two main aspects are investigated to improve the DNN performance: the construction of the feature vector that effectively characterizes the renewable energy, load, and topology and the knowledge transfer of DNN parameters to improve the training efficiency of the DNN

for evolutionary scenarios.

In (Xiang et al., 2020b), DNNs are applied to obtain power flow results, such as voltage amplitude and angle of buses and also active and reactive power flows in each branch, for different simulation cases. The networks' topology is altered from one simulation case to the other, and the authors propose using the Transfer Learning technique to train the DNNs. Two main features are studied in the application of the technique: parameters of input and output layers are transferred to initialize parts of parameters in the new DNN according to the variable with which they are connected; the remaining parameters of input and output layers are initialized based on the non-parametric estimation.

The use of ANNs in the power flow analysis has also been subject to thesis and dissertations. For example, in (Donnot, 2019) the methodology was applied to offer, in real-time, a set of actions that dispatchers could choose from in order to resolve or prevent problems caused by disturbances on the grid. At first, an artificial data set was built to perform some controlled experiments to validate the developed methodology. The main model developed consists of adapting the architecture of a neural network depending on a structural vector  $\tau$ , representing the grid's topology, possible contingencies  $z$ , and the operator's decision  $\pi$ . This way of encoding the topology forces the ANN to learn pertinent representations, allowing it to generalize to unseen structural vectors  $\tau$ . The work demonstrated that ML could allow Transmission System Operators (TSOs) to predict flows accurately on snapshots, even in unseen cases, at least in controlled experiments.

Recent developments in ML have been driven not only by the development of new learning algorithms and theory but also by the ongoing explosion in the availability of online data and low-cost computation. ML has progressed dramatically over the past two decades, evolving from academic theory to a practical technology in widespread commercial use. The past decade has seen a rapid increase in networked and mobile computing systems' ability to gather and transport large amounts of data. This phenomenon is often referred to as "Big Data". This vast amount of data makes it essential to develop scalable procedures that blend computational and statistical considerations, requiring ML solutions to the problem of obtaining useful insights, predictions, and decisions from the data sets (Jordan and Mitchell, 2015).

In this work, we propose a computational tool based on ANNs to estimate both the system's active technical losses and line flows. The predictions' accuracy is checked against traditional power flow methods, computed by commercial software such as Organon, developed by HPPA, and OpenDSS, developed by EPRI. Using this method, we were able to accurately compute losses

in distribution systems and calculate line flows much faster than traditional methods. From a methodological perspective, the proposed approach differs from most of the mentioned literature due to the fact that, since topology features are difficult to be obtained for distribution systems, the only inputs applied to the ANNs were the active loads and generations of the system, leading to an advantage to increase the scope of application of the proposed method.

Another difference is that, although Matlab is the mainly used programming language in this area, we used the Julia language to implement the proposed methodology due to some beneficial features when dealing with a large amount of data, for example: Julia is faster than Matlab; it has automatic memory management, and it has its own native ML libraries that are entirely written in Julia, so they can easily be modified by the end-user if needed.

Besides that, an additional innovation of the present work is the application of the results produced by the ANN methodology to the obtainment of optimal demand contracts for distribution systems at the borders with the transmission system.

## 3

### Methodologies

In Chapter 2, we showed that the power flow problem corresponds to a set of equations, for each bus of the system, representing Kirchhoff's first law. As depicted by equations (2-1) and (2-2), equations (2-3) and (2-4) are dependent of the unknown variables  $V_k, V_m, \theta_k$  and  $\theta_m$ . However, equations (2-3) and (2-4) are complicated nonlinear equations and it is not possible to obtain an exact analytical solution for them. Therefore, we must use an approximation technique that allows us to obtain a sufficiently accurate solution.

In this Chapter, we will present two approaches to solve the power flow calculation: the first one corresponds to classical iterative methods, while the second approach uses ML techniques to address the power flow problem.

#### 3.1

##### Classical Iterative Methods

The term “iterative method” refers to a wide range of techniques that use successive approximations to obtain more accurate solutions to a system of equations at each step. In this section, we will focus on a class of iterative methods that presents iteration-dependent coefficients (nonstationary iterative methods), more specifically, Newton's method and the Current Injection Method (Barret et al., 1994).

##### 3.1.1

###### Newton's Method

Among iterative methods, Newton's method is one of the most well-known. Generally, the method can be applied to find the solution for a equation  $f(x) = 0$ , based on the following steps (Monticelli and Garcia, 2015):

1. Arbitrate an initial condition ( $x_i = x_0$ ) and fix  $i = 0$ ;
2. Calculate  $f(x_i)$  and check the convergence. If the absolute value  $|f(x_i)|$  is equal to or smaller than a given tolerance  $tol$ , i.e., if  $|f(x_i)| \leq tol$  the procedure may be finished.
3. Linearize the function around  $(f(x_i), x_i)$  and equal the function to zero in order to establish the pace  $\Delta x_i = x_{i+1} - x_i$  and the new point  $x_{i+1}$ .
4. Set  $i = i + 1$  and return to Step 2.

We can represent the unknown variables of the power flow problem in a vectorial form, given by:

$$x = \begin{bmatrix} \theta \\ V \end{bmatrix},$$

where  $\theta$  and  $V$  corresponds to voltages angles and magnitudes, respectively. Hence, our problem is to obtain the vector  $x$  of unknown variables.

According to (Monticelli and Garcia, 2015), a subset of equations (2-3) and (2-4), corresponding to the buses where the values of  $P$  and  $V$  are previously known, can be generally represented by a function  $f(x)$ , where  $x$  is the aforementioned vector of unknown variables, given by:

$$f(x) = \begin{cases} \Delta P_k = P_k^{spe} - P_k(x), \forall k \in [\Omega_{PQ}, \Omega_{PV}] \\ \Delta Q_k = Q_k^{spe} - Q_k(x), \forall k \in [\Omega_{PQ}] \end{cases} \quad (3-1)$$

where *spe* represents *specified* power values (already known at bus  $k$ ),  $\Omega_{PQ}$  and  $\Omega_{PV}$  corresponds to the sets of PQ and PV buses on the system, respectively. We can apply Newton's method at function  $f(x)$  by expanding the function in a Taylor series around an initial point  $x_i$  and discarding the high order terms obtained by the expansion:

$$f(x) \approx f(x_i) + \left[ \frac{\partial f(x_i)}{\partial x} \right] \Delta x_i \quad (3-2)$$

By making  $f(x) = 0$ , we can obtain the correction term  $\Delta x_i$  through the solution of the system (3-3):

$$\left[ \frac{\partial f(x_i)}{\partial x} \right] \Delta x_i = -f(x_i) \quad (3-3)$$

For iteration  $i$ , we can define power mismatches at bus  $k$  as:

$$\Delta P(\theta_i, V_i) = P_k^{spe} - P_k(\theta_i, V_i) \quad (3-4)$$

$$\Delta Q(\theta_i, V_i) = Q_k^{spe} - Q_k(\theta_i, V_i), \quad (3-5)$$

Since we are dealing with vectorial functions, we can use the Jacobian matrix

$$J = \begin{bmatrix} H & N \\ M & L \end{bmatrix},$$

corresponding to the first order partial derivatives of the vectorial function  $f(x)$ , and equation (3-3) can be rewritten as:

$$\begin{bmatrix} H & N \\ M & L \end{bmatrix} \begin{bmatrix} \Delta \theta_i \\ \Delta V_i \end{bmatrix} = \begin{bmatrix} \Delta P(\theta_i, V_i) \\ \Delta Q(\theta_i, V_i) \end{bmatrix}, \quad (3-6)$$

where  $H = \partial P / \partial \theta$ ,  $L = \partial Q / \partial V$ ,  $M = \partial Q / \partial \theta$  and  $N = \partial P / \partial V$  and, since the specified values  $P^{spe}$  and  $Q^{spe}$  are constant, their partial derivatives are null.

The components of the Jacobian submatrixes  $H$ ,  $N$ ,  $M$ , and  $L$  are given by (Monticelli and Garcia, 2015):

$$H = \begin{cases} H_{km} = \frac{\partial P_k}{\partial \theta_m} = V_k V_m (G_{km} \sin \theta_{km} - B_{km} \cos \theta_{km}) \\ H_{kk} = \frac{\partial P_k}{\partial \theta_k} = -V_k^2 B_{kk} - V_k \sum_{m \in K} V_m (G_{km} \sin \theta_{km} - B_{km} \cos \theta_{km}) \end{cases}$$

$$N = \begin{cases} N_{km} = \frac{\partial P_k}{\partial V_m} = V_k (G_{km} \cos \theta_{km} + B_{km} \sin \theta_{km}) \\ N_{kk} = \frac{\partial P_k}{\partial V_k} = V_k G_{kk} + \sum_{m \in K} V_m (G_{km} \cos \theta_{km} + B_{km} \sin \theta_{km}) \end{cases}$$

$$M = \begin{cases} M_{km} = \frac{\partial Q_k}{\partial \theta_m} = -V_k V_m (G_{km} \cos \theta_{km} + B_{km} \sin \theta_{km}) \\ M_{kk} = \frac{\partial Q_k}{\partial \theta_k} = -V_k^2 G_{kk} + V_k \sum_{m \in K} V_m (G_{km} \cos \theta_{km} + B_{km} \sin \theta_{km}) \end{cases}$$

$$L = \begin{cases} L_{km} = \frac{\partial Q_k}{\partial V_m} = V_k (G_{km} \sin \theta_{km} - B_{km} \cos \theta_{km}) \\ L_{kk} = \frac{\partial Q_k}{\partial V_k} = -V_k B_{kk} + \sum_{m \in K} V_m (G_{km} \sin \theta_{km} - B_{km} \cos \theta_{km}) \end{cases}$$

Elements  $H_{kk}$ ,  $N_{kk}$ ,  $M_{kk}$ , and  $L_{kk}$  can be rewritten according to the active and reactive power injections on bus  $k$ . Hence,

$$H_{kk} = -Q_k - V_k^2 B_{kk}, \quad (3-7)$$

$$N_{kk} = V_k^{-1} (P_k + V_k^2 G_{kk}), \quad (3-8)$$

$$M_{kk} = P_k - V_k^2 G_{kk}, \quad (3-9)$$

$$L_{kk} = V_k^{-1} (Q_k - V_k^2 B_{kk}) \quad (3-10)$$

According to the expressions for the submatrixes  $H$ ,  $N$ ,  $M$  and  $L$ , if  $Y_{km} = G_{km} + jB_{km}$  is null, then the elements  $H_{km}$ ,  $N_{km}$ ,  $M_{km}$  and  $L_{km}$  are also null. Therefore,  $H$ ,  $N$ ,  $M$  and  $L$  have the same sparsity feature than the admittance matrix  $Y$ .

We can summarize the iterative process of Newton's method for power flow solution by the procedure described in Algorithm 1. Also, Figure 3.1 illustrates the process described.

**Algorithm 1** Newton's method for power flow solution

- 1: Set initial values for  $V_0$  and  $\theta_0$  ( $i = 0$ ).
- 2: Calculate power mismatches  $\Delta P(\theta_i, V_i)$  and  $\Delta Q(\theta_i, V_i)$ .
- 3: Convergence test: if all the absolute values of the calculated power mismatches are smaller than a given tolerance, the process can be ended, and we can move to the last step; otherwise, we must go to step 4.
- 4: Obtain the Jacobian matrix  $df/dx$  for  $(\theta_i, V_i)$ .
- 5: Calculate the corrections  $\Delta\theta_i$  and  $\Delta V_i$  and the new state

$$\theta_{i+1} = \theta_i + \Delta\theta_i$$

$$V_{i+1} = V_i + \Delta V_i$$

- 6: Make  $i = i + 1$  and test the convergence of the solution. In case the solution converges, follow to the next step; otherwise, return to step 2.
- 7: Calculate the remaining unknown variables of the problem.

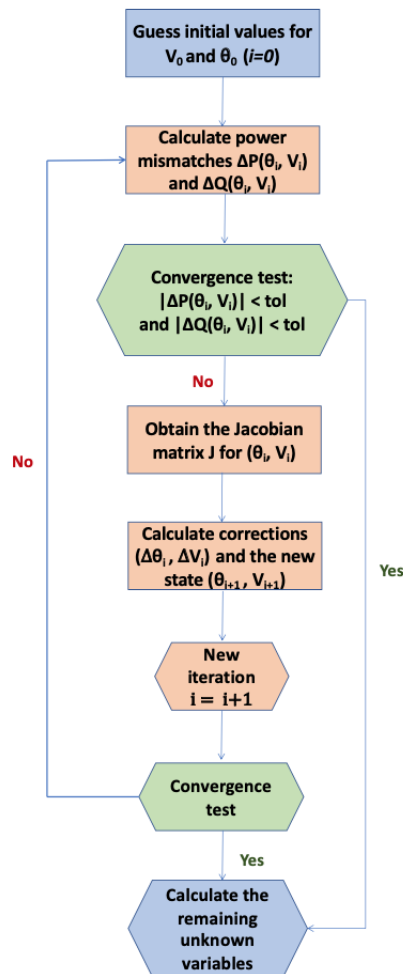


Figure 3.1: Newton's method block diagram for power flow solution.

This method is adopted by some commercial power flow software, such as Organon. However, as mentioned in Chapter 2, Newton's method is not the most appropriate technique for power flow calculations in distribution systems due to their unbalanced nature. Hence, methods that are able to provide multi-phase power flow solutions can better address this kind of problem. As an example, we present, in the next section, the Current Injection Method.

### 3.1.2

#### Current Injection Method

As mentioned in Chapter 2, the most traditional power flow method for distribution systems is the BF method. However, this technique was initially designed for radial networks and later adapted to meshed networks due to the expansion of DGs, and thus has topological limitations. Therefore, we decided to use the Current Injection Method (CIM) as our benchmark for power flow calculations in distribution systems. Another reason that motivated this choice is that CIM is already adopted by commercial power flow software, such as the Open Distribution System Simulator (OpenDSS), which allows the user to solve three-phase power flows.

In traditional power flow calculations for distribution systems, all buses, except for the slack bus, are modeled as PQ buses (Liu et al., 2016). However, power conversion (PC) elements, such as loads, DGs, and energy storages, for example, modeled as constant power (PQ) elements, constitute nonlinear components on the system that require special treatment before the power flow calculations are initialized (Rocha and Radatz, 2018).

The method models the power system based on nodal admittances: each linear internal element of the system, such as voltage regulators and capacitors, for example, has its own primitive admittance but, in order to solve the entire system, we must obtain the system's admittance matrix ( $\bar{Y}_{system}$ ) that corresponds to a sparse matrix composed by the primitive admittances of the elements (Corrêa, 2020).

However, PC elements can be modeled as current injections, depending on the voltages at each bus. For example, suppose a non-linear single-phase load connected to a distribution system at node 1 of Figure 3.2. The current flowing through this load is non-linear and can be represented as a function of the voltage applied on its terminals, i.e.,  $\dot{I}_{term} = f(\dot{V}_1)$ . Hence, current  $\dot{I}_{term}$  must not be considered as the current injected into the network (Rocha and Radatz, 2018; Corrêa, 2020).

Power flow methods may address this non-linearity through the consideration of a compensation current. Hence, to initialize the power flow calcu-

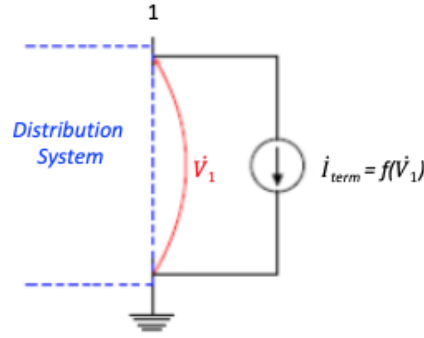


Figure 3.2: Non-linear single-phase load. Source: (Rocha and Radatz, 2018).

lations, a single-phase short-circuit is applied to the load and DG buses; PC elements are remodeled as a Norton's equivalent circuit, as depicted by Figure 3.3, composed by a constant nodal admittance  $\bar{Y}_{linear}$ , and a compensation current source  $\dot{I}_{comp}$ .  $\bar{Y}_{linear}$  is included in the system's admittance matrix  $\bar{Y}_{system}$  and represents the linear component of the load, while  $\dot{I}_{comp}$  represents the nonlinear component of the load (Rocha and Radatz, 2018; Barouche, 2017).

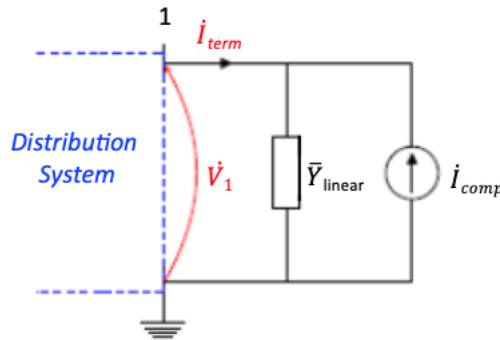


Figure 3.3: Non-linear single-phase load represented by a Norton's equivalent. Source: (Rocha and Radatz, 2018).

Therefore, the load's linear component is treated as a passive component of the network and the resulting injected current  $\dot{I}_{inj}$  on a bus can be obtained by the vectorial sum of the original current  $\dot{I}_{term}$  and the compensation current  $\dot{I}_{comp}$  on the bus (Rocha and Radatz, 2018):

$$\dot{I}_{inj} = \dot{I}_{term} + \dot{I}_{comp} \quad (3-11)$$

Current  $\dot{I}_{term}$  is responsible for the supply of the bus load and can be determined by the load's active and reactive components, i.e., it has a constant value, while current  $\dot{I}_{comp}$  can be mathematically obtained by:

$$\dot{I}_{comp} = (\dot{V}_1 \times \bar{Y}_{linear}) - \dot{I}_{term} \quad (3-12)$$

Once the system is completely modeled, we can move to the power flow solution. In the Current Injection Method, the nodal voltages are considered unknown variables (Barouche, 2017). Therefore, the first step is to obtain an initial solution to the three phases of the system's voltages (magnitude and phase angle). To do that, we can consider that, initially,  $\dot{I}_{comp} = 0$  for all PC elements, obtaining a linear system. This initial operating point is, usually, close to the final solution, which is essential for the convergence of the method (Corrêa, 2020).

By making  $\dot{I}_{comp} = 0$  and by calculating  $\dot{I}_{term}$  according to the bus active and reactive loads, we can obtain the value of  $\dot{I}_{inj}$ , according to equation (3-11), and, hence, calculate the nodal voltage  $\dot{V}_{nodal}$  by (Rocha and Radatz, 2018; Corrêa, 2020):

$$\dot{V}_{nodal} = [\bar{Y}_{system}]^{-1} \times \dot{I}_{inj} \quad (3-13)$$

After obtaining the initial voltage values, we can initiate the first iteration to calculate the compensation currents of PC elements, through equation (3-12), and, hence, the injected currents, through equation (3-11) and the new voltage values, according to equation (3-13).

With the new voltage values obtained, we can calculate the difference between this new result (voltages at iteration  $i$ ) and the previous one (voltages at iteration  $i - 1$ ) by making:

$$\Delta V = |V_{i-1}| - |V_i| \quad (3-14)$$

If the calculated value for  $\Delta V$  is greater than a given tolerance, we must continue the iterative process, adjusting the compensation current and the nodal voltage values; otherwise, the process has converged, and we can stop the iterations (Barouche, 2017).

Altogether, there are five main steps on the iterative process of the Current Injection Method (Rocha and Radatz, 2018; Barouche, 2017), described in Algorithm 2.

**Algorithm 2** Current Injection Method

- 1: Obtain Norton's equivalent for all PC elements to model the entire system and calculate the system's admittance matrix  $\bar{Y}_{system}$ . Calculate constant currents  $\dot{I}_{term}$ , according to the bus loads.
- 2: Initial guess: an initial value for the system's voltages must be considered. A good initial guess consists of a direct solution of the nodal admittance matrix  $\bar{Y}_{system}$ , considering that the compensation current  $\dot{I}_{comp}$  is null for all the PC elements on the system, i.e., only constant currents must be initially considered.
- 3: Calculation of a vector  $\dot{I}_{inj}$  of injected currents.
- 4: Once the injected current vector was obtained on the previous step, it is possible to obtain a new solution for the nodal voltages vector  $\dot{V}_{nodal}$ , given by:

$$\dot{V}_{nodal} = [\bar{Y}_{system}]^{-1} \times \dot{I}_{inj}$$

- 5: Convergence test:  $\Delta V = |V_{i-1}| - |V_i| < tol$ .  
If the test fails, the compensation current  $\dot{I}_{comp}$  must be updated for each PC element, according to:

$$\dot{I}_{comp} = (\dot{V}_1 \times \bar{Y}_{linear}) - \dot{I}_{term},$$

and we must return to step 3. Otherwise, the algorithm has converged and we can stop the iterations.

Figure 3.4 illustrates the aforementioned process.

From the main features of the Current Injection Method, we can highlight that (Barouche, 2017):

- the distribution system's admittance matrix is not modified at each iteration, which provides a great computational efficiency;
- the algorithm for the power flow calculation is straightforward and does not rely on the construction and inversion of the Jacobian matrix, like Newton's method;
- the premise of calculating nodal voltages for iteration  $i = 0$ , considering an open circuit, ensures that the voltage values at this iteration are close to the numerical solution.

Despite the presented advantages of the Current Injection Method, the need for many topology features constitutes a difficulty for distribution systems, as depicted in Chapter 2. This obstacle motivated us to pursue different approaches for dealing with power flow calculations in distribution networks. As mentioned in the literature review, ML techniques may represent an adequate path to address this problem.

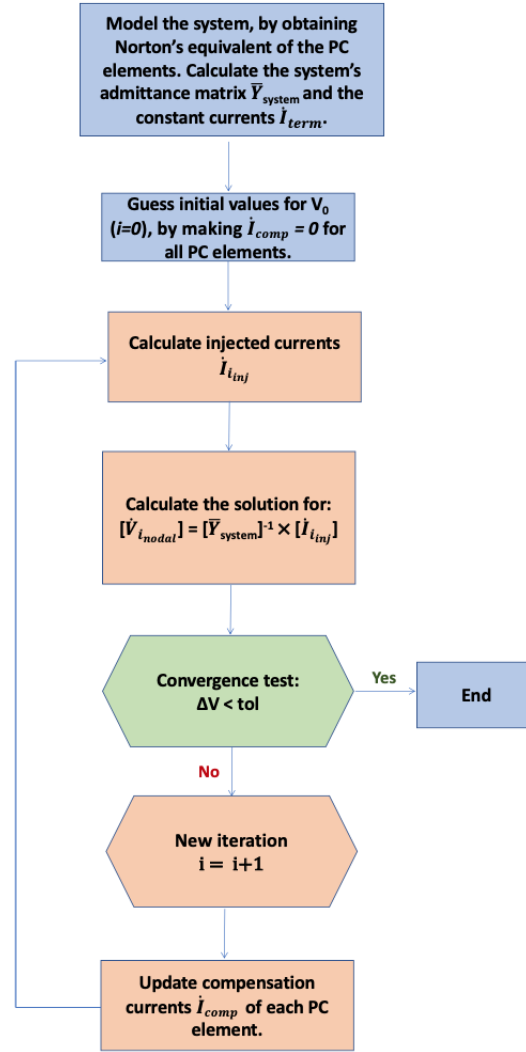


Figure 3.4: Current Injection Method block diagram for power flow solution.

### 3.2

#### ML Techniques: Artificial Neural Networks

As mentioned in Chapter 2, with the latest advances in Big Data availability, software engineering capability, and affordable high computing power, AI advancement has entered a new stage, in which major research branches include ML, a category of research and algorithms focused on finding patterns in data and using those patterns to make predictions. By using ML techniques to analyze and learn large amounts of existing data, computers can make predictions and assist in the user's decision-making process. Therefore, the goal of ML is to find the optimal mapping between input and output variables (Misilmani and Naous, 2019; Cheng and Yu, 2019).

According to (Cheng and Yu, 2019), common ML frameworks include TensorFlow, Caffe, Keras, CNTK, Torch7, Theano, and Deeplearning4J. However, for our power flow studies, we used the Julia Machine Learning library

Flux, released in 2017, due to the following main reasons: while typical ML frameworks are written in many hundreds of thousands of lines of C++, Flux is only a thousand lines of straightforward pure-Julia code; it provides a single, intuitive way to define models, with possible incorporation of different Julia libraries, making it easier to build complex data processing pipelines; it contains many useful built-in tools, but also allows the users to create their own functions, benefiting from the main advantages of the Julia language, mentioned in Chapter 2, when needed (Innes et al., 2018; Innes, 2018).

ML techniques are based on two main goals: learning (training) and generalization. In order to do that, a data set is usually split into 2 different sets (Donnot, 2019):

- **Training set:** contains the data used to compute gradients and find the optimal parameters of a model.
- **Test set:** contains data never seen during the training procedure. Therefore, these data are not used to find the optimal parameters. This set is used to report final errors on ML algorithms.

As mentioned in Chapter 2, ANNs constitute the main ML technique to address the power flow calculation problem, and, hence, they will be adopted as our benchmark.

ANNs are based on the elementary neuron, which is a mathematical model of the biological neuron. In the same way in which biological neurons, connected in neural tissues, receive stimuli, process the information according to their function, and then send a response back to the brain, artificial neurons compute an output value  $y$ , processed from an input value  $x$ , according to their processing function (Ivanov et al., 2014).

ANNs can be described as architectures composed of a single layer or a sequence of layers, each taking as inputs the results of the previous layers (Fioretto et al., 2019). When the ANN has only one layer, it is referred to as a single-layer network or *perceptron* and, in this case, a set of inputs is directly mapped to an output by the use of a generalized variation of a linear function. On the other hand, in multi-layer ANNs, the neurons are arranged in a layered fashion, in which a group of hidden layers separates the input and output layers (Aggarwal, 2018).

According to (Xiang et al., 2020a), deep models with multi-hidden layers have shown a stronger ability to extract complex features than shallow models. They can be generally categorized into three types: *convolutional neural networks* (CNNs), *recurrent neural networks* (RNNs), and fully-connected deep neural networks (also referred as *deep neural networks* (DNNs) for simplicity).

CNN is the go-to method for addressing image data; RNN is suitable to address sequence data, like speech data; DNN is suitable for addressing conventional one-dimensional data. Hence, DNNs are suitable to address the power flow calculation problem and will be the focus of our study.

Among DNNs, the most used type is the Multilayer Perceptron (MLP), simple architectures in which the neurons are arranged in the hidden and the output layers. It is common to employ two hidden layers, or even three, but, usually, no more than that. While there is no communication between neurons of the same layer, adjacent layers are fully interconnected, and each neuron-to-neuron link is associated with a weight. For example, suppose we have an MLP with a single hidden layer,  $k$  inputs, and  $n$  outputs, as illustrated in Figure 3.5. By setting the first index referring to the link's “beginning” and the second to its “end”, we can denote the weight of the link from the  $j$ -th hidden neuron to the  $i$ -th output neuron as  $w_{ji}$ , and the weight of the link from the  $k$ -th attribute to the  $j$ -th hidden neuron as  $w_{kj}$  (Fioretto et al., 2019; Ivanov et al., 2014; Kubat, 2017).

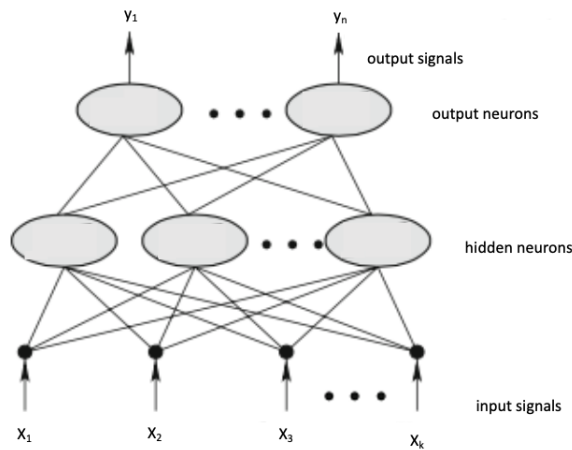


Figure 3.5: Multilayer Perceptron (MLP) architecture, with one hidden layer. Source: Adapted from (Kubat, 2017).

Hence, the input values  $x_k$  are multiplied by the weights associated with the links; the  $j$ -th hidden neuron receives, as inputs, the weighted sum  $\sum_k w_{kj}x_k$  and subjects this sum to an activation function  $f$ , which is often non-linear, obtaining  $f(\sum_k w_{kj}x_k)$ . The  $i$ -th output neuron then receives the previous value, weights it according to  $w_{ji}$  and, again, subjects it to an activation function, obtaining the  $i$ -th output signal. This process of propagating the attribute values from the network's input to its output is called *feed-forward propagation*. In essence, the two-layer MLP of Figure 3.5 calculates the following formula to obtain the output signals (Kubat, 2017):

$$y_i = f\left(\sum_j w_{ji} f\left(\sum_k w_{kj} x_k\right)\right) \quad (3-15)$$

The input and output layers' sizes are determined by the number of input and output values that describe the problem, while the number of neurons from the hidden layers is user-defined and problem-dependent (Ivanov et al., 2014). The structure of the hidden layer not only reflects the complexity of the mapping relationship between the input and the output layers but also affects the network convergence characteristics. There is no unique rule to determine the hidden layer structure for different problems (Hsu et al., 1995).

Activation functions are applied element-wise to the hidden units of an ANN, allowing the network to represent non-linear mappings between inputs and outputs. According to (Donnot, 2019), the most frequently used activation functions are the sigmoid and the Rectified Linear Unit (ReLU) functions. The sigmoid function, defined as:

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \quad (3-16)$$

was the most common non-linear function applied on hidden units of neural networks and, nowadays, has been replaced, in many applications, by the ReLU function, given by:

$$ReLU(x) = \max(0, x) \quad (3-17)$$

The ReLU function has a variation, named Leaky ReLU, that is also commonly used and it is given by:

$$LeakyReLU(x) = \max(0.1x, x) \quad (3-18)$$

Figure 3.6 illustrates those activation functions:

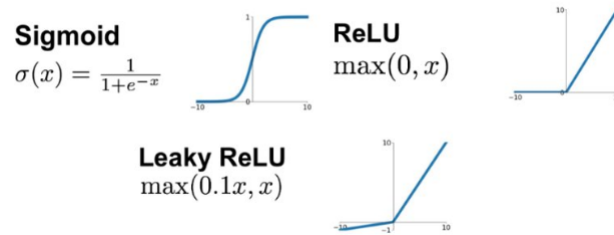


Figure 3.6: Most commonly used activation functions. Source: (Jadon, 2018).

According to (Hsu et al., 1995), key factors that strongly affect the outputs  $y$  of an ANN should be adopted as variable signals for the input layer  $x$ . When solving an approximation problem, the learning process aims to find the optimal weights so that for any input  $x$ , the output of the network will be an approximation, as close as possible, to the exact value of the corresponding  $f(x)$  (Ivanov et al., 2014).

To improve the training efficiency of ANNs, the input and output data should be preprocessed to eliminate numerical problems and the adverse influence of outlier samples in the training process (Yang et al., 2020). It is noted that the input of the samples should be normalized, whereas the output of the ANN should be denormalized to obtain the power flow solutions. The min-max normalization method uses the minimum and maximum values of samples for normalization:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}, \quad (3-19)$$

where  $x_{norm}$  is the sample value after normalization;  $x$  is the sample value that needs to be normalized;  $x_{min}$  is the minimum value of samples, and  $x_{max}$  is the maximum value of samples. The min-max normalization maps the data to the range of  $[0, 1]$ . Denormalization is the reverse process of normalization (Xiang et al., 2020a).

The optimization of the weights during the training process can be achieved, for example, through the minimization of a Mean Square Error (MSE) cost computed over a batch of past input-output observations. Specifically, we have a training set  $Z = \{z_1, \dots, z_n\}$ , where the element  $z_i$  is an input-output pair  $(x_i, y_i)$ , measured at some time  $i$  in the past, and  $n$  is the total number of such pairs (Cataliotti et al., 2019; Yang et al., 2020). The MSE is defined using the differences between the elements of the output vector and the target value (Kubat, 2017):

$$MSE = \frac{1}{n} \sum_{i=1}^n (t_i - y_i)^2 \quad (3-20)$$

When calculating the network's MSE, we have to establish, for each output neuron, the difference between its output ( $y_i$ ) and the corresponding element of the target vector ( $t_i$ ). It is worth mentioning that the minimization objective function (in the example, the MSE function) is also called *loss function* and that different loss functions can be adopted. The choice of the loss function is critical in defining the outputs in a sensitive way to the application (Aggarwal, 2018).

In the beginning, the weights are initialized to small random numbers. After this, the training examples are presented, one by one, and each of them is forward-propagated to the network's output. The discrepancy between this output and the example's target vector then tells us how to modify the weights. After the weight modification, the next example is presented. When the last training example has been reached, one *epoch* has been completed. In MLPs, the number of epochs needed for successful training can be thousands, tens of thousands, or even more (Kubat, 2017).

In Figure 3.7, the vertical axis represents the MSE, expressed as a function of the network’s weights, plotted along the horizontal axes. For graphical convenience, it is assumed that there are only two weights. The “valleys” in the error function represent the function’s *local minima*. The deepest of them is the *global minimum*. Ideally, the training procedure should be able to obtain the set of weights corresponding to the global minimum (Kubat, 2017).

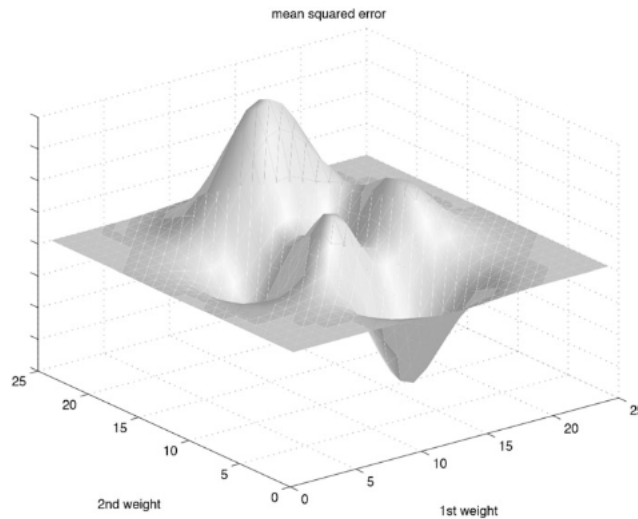


Figure 3.7: MSE for a set of weights obtained for a given example. Source: (Kubat, 2017).

With non-convex functions, such as ANNs, it is possible to have many local minima. Indeed, nearly any deep model is essentially guaranteed to have a vast number of local minima. However, for sufficiently large ANNs, most local minima have a low-cost function value, and, therefore, it is not essential to find a true global minimum. Instead, obtaining a point in parameter space with low but not minimal cost may be acceptable, as illustrated in Figure 3.8 (Goodfellow et al., 2016; Stewart, 2019).

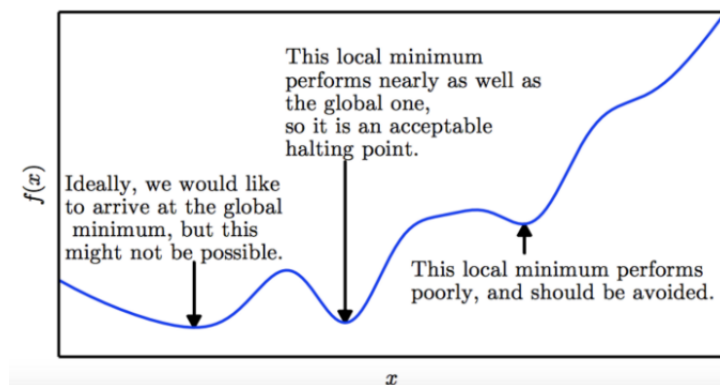


Figure 3.8: Local and global minimum of a function. Source: (Stewart, 2019).

In the case of MLPs, the problem is that the loss is a complicated composition function of the weights in previous layers, as shown by Equation (3-15). The best-known technique used to address this difficulty is the *backpropagation algorithm*, which computes the error gradients in terms of summations of local-gradient products over the various paths from a node to the output. Although this summation has an exponential number of components, it can be efficiently computed using dynamic programming. The backpropagation algorithm is a direct application of dynamic programming, and it contains two main phases (Aggarwal, 2018):

- **Forward phase:** the inputs for a training instance are fed into the ANN, resulting in a forward cascade of computations across the layers, using the current set of weights. The final predicted output can be compared to that of the training instance, and the derivative of the loss function to the output is computed. The derivative of this loss now needs to be computed with respect to the weights in all layers in the backward phase.
- **Backward phase:** the main goal is to learn the gradient of the loss function with respect to the different weights by using the chain rule. These gradients are used to update the weights.

Practical issues in ANN training may include the reaching of an undesirable local minimum by the weights. To avoid that, one possibility of timely identification of local minimum during training is to keep track of the loss function and sum it up over the entire training set at the end of each epoch. Under normal circumstances, this sum tends to go down from one epoch to another. Once it seems to have reached a plateau, where hardly any error reduction can be observed, the learning process is suspected of being trapped in a local minimum. Generally, this problem is less critical in networks with many hidden neurons (Kubat, 2017).

Another common practical issue in ANN training is overfitting, which refers to the fact that perfectly fitting a model to a particular training data set does not guarantee that it will provide good prediction performance on unseen test data. Small MLPs are usually not flexible enough to overfit, but as the number of hidden neurons increases, the network gains flexibility, and overfitting can become a real concern (Kubat, 2017; Aggarwal, 2018).

Finding the optimal structure for the trained ANN (number of hidden layers, their neuron count, and activation function) is an arduous work, since there are many possible combinations for the network's main features (Ivanov et al., 2014). If, for example, there are only one or two hidden neurons in the ANN, it will lack flexibility and will be prone to achieve a

local minimum. On the other hand, using thousands of neurons will not only increase computational costs due to the need to train so many neurons but will also cause the network to be more flexible than needed, leading to possible overfitting of the data. Therefore, some compromise between those aspects needs to be found (Kubat, 2017).

ANNs require careful design to minimize overfitting's harmful effects, even when a large amount of data is available. Since a large number of parameters may cause overfitting, different methods may be used to mitigate these impacts. Considering that smaller absolute values of the parameters tend to overfit less and since it is hard to constrain their values, the Regularization approach to reduce overfitting's impact is based on the addition of a penalty  $\lambda$ , at each parameter, to the loss function. Regularization is particularly important when the amount of available data is limited (Aggarwal, 2018).

Even though, in general, increasing the complexity of the model reduces its generalization power, it is often advisable to use more complex models with Regularization rather than simpler models without it. In the Flux package, applying Regularization to parameters is straightforward: we need to apply an appropriate regularizer to each model parameter and add it to the overall loss. Besides Regularization, other methods to reduce the impacts of overfitting may be adopted, such as: Architecture and Parameter Sharing, Early Stopping and Ensemble Methods (Aggarwal, 2018; Innes et al., 2018; Innes, 2018).

Figure 3.9 shows that the error rate measured on testing examples depends on the number of neurons in the hidden layer. The horizontal axis represents the number of hidden neurons, while the vertical axis represents the error rate measured on the testing set. As shown in the graph, larger networks exhibit lower error rates but, when they are too large, they become vulnerable to overfitting, causing the testing-set error to start growing again after a certain number of neurons. We highlight that the precise shape of the curve depends on the complexity and the training data domain. In domains where the training examples are completely noise-free, overfitting may never become an issue. (Kubat, 2017).

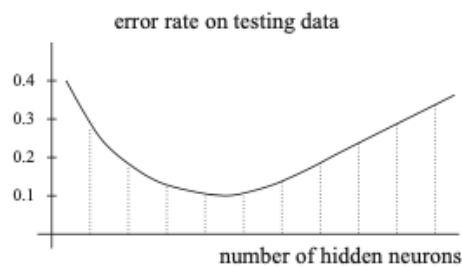


Figure 3.9: Error rate measured on testing examples. Source: (Kubat, 2017).

According to (Kubat, 2017), in the search for the appropriate network size, the main idea is to start with a minimal network that only has a few hidden neurons. After each epoch, the learning algorithm checks the loss function observed on the training set, which is likely to keep decreasing with the growing number of epochs, but only up to a certain point, in which the network's performance no longer improves, either due to insufficient flexibility or because it “fell” into a local minimum. When this is observed, a few more neurons with randomly initialized weights are added, and the training is resumed. Usually, the added flexibility makes further error reduction possible. Figure 3.10 illustrates that when the MSE does not seem to decrease, while the number of epochs is increasing, further improvement can be achieved by adding new hidden neurons (Kubat, 2017).

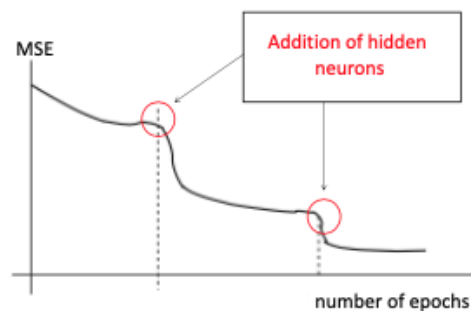


Figure 3.10: As the number of epochs increases, the MSE value decreases until a certain point, in which the addition of new hidden neurons contribute to further reducing the error. Source: Adapted from (Kubat, 2017).

If the training data set is appropriately defined in size and relevance, i.e., the provided values cover the expected variation range of the inputs, and the data sampling is appropriate, then the ANN will be able to calculate, with satisfying accuracy, the solution for the problem, for any new input for which it has not been trained, but which is in the range considered in the training (Ivanov et al., 2014). After the ANN model has been trained, the recall process becomes so fast that the time required for calculation of the outputs from the test set can be saved (Hsu et al., 1995).

According to (Maçaira et al., 2018), despite the high data processing power due to its massively distributed structure and its ability to learn and, therefore, to generalize, producing suitable outputs for inputs that were not presented during the training phase, ANNs require a high computational power and a large amount of data in the training process, which may constitute an obstacle to the implementation of this method.

To implement MLPs for power flow studies, we considered data of the systems' loads and generations as inputs of the networks. We trained the MLPs to predict the values of the systems' active power flows and active losses. Hence, for those implementations, topology features of the systems studied were not needed.

A step-by-step procedure for modelling a neural network for power flow calculations can be given by Algorithm 3 (Pertl et al., 2016).

---

**Algorithm 3** Modelling an ANN for power flow solution

---

- 1: Collect load and generation data for the system's buses (*inputs*) and the values for the system's active power flows and losses (*targets*).
  - 2: Select the type of neural network that should be implemented.
  - 3: Define the network architecture, i.e., how the computations should be performed in this network. This includes the number of neurons and hidden layers, training algorithm, training goal, among other parameters.
  - 4: Initialization of weights (done automatically by the algorithm).
  - 5: Train the network, based on the input and output data provided, considering the configuration defined in step 3.
  - 6: Apply the model on the test set, which comprises unseen values for the system's loads and generations, or retrain the ANN. Results of different training procedures can vary due to different initial values for weights.
-

Figure 3.11 illustrates the procedure described.

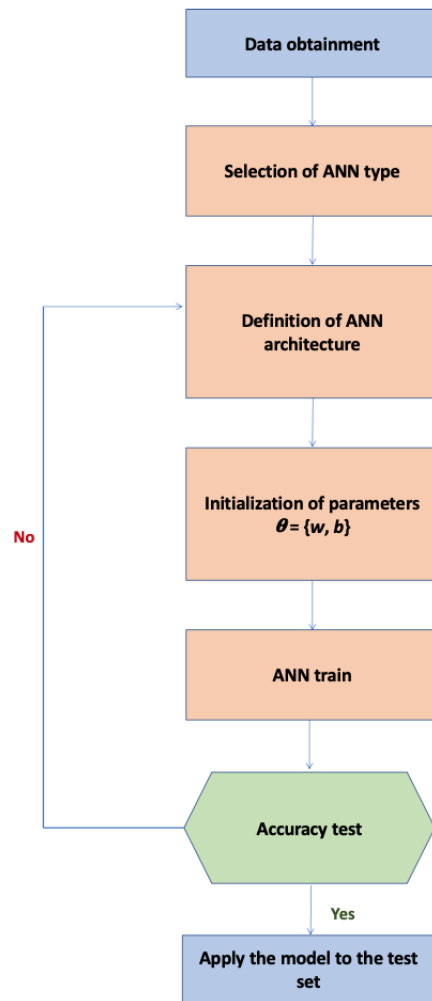


Figure 3.11: Artificial Neural Network modelling procedure.

## 4

### Case Studies

Chapters 2 and 3 presented different power flow techniques and the corresponding methodologies for their implementations. In this Chapter, we present the case studies designed to evaluate the effectiveness of those techniques in solving two different challenges faced by distribution systems: the first one aims to analyze the impacts of the DG insertion on the system's active losses, while the second aims to calculate the power flows between the distribution and the transmission systems.

For the two studies, we used the software OpenDSS, version 9.1.3.3 of 2020, and Organon, version 5.9.1 of 2018, respectively, whose methods for power flow calculations were described in Chapter 3. Based on the software results, we trained ANNs to predict each case's target values, in order to analyze the efficiency of this technique.

#### 4.1

##### **Analysis of DG impacts on the distribution system's active losses**

The goal of the distribution system is to deliver the electricity from the substation to the consumer. However, losses occur during this operation, causing utilities to search for techniques to minimize them, such as: reconfiguring the network or placing capacitors as reactive power support. Those losses can be classified as (Kahef et al., 2018):

- technical losses: related to properties of materials and their resistance to the flowing current. Thus, they occur due to the dissipated energy in equipment and conductors.
- non-technical losses: occur due to human faults or theft of electricity.

As mentioned in Chapter 2, electric losses are important economic indicators for distributors since they usually appraise operating efficiency by the amount of real power loss over the system (ANEEL, 2019).

Compared to the transmission network, the distribution system has the particularity of complex type users and being more difficult to obtain line parameters, leading to more challenging calculations to obtain system's losses, as mentioned in Chapter 2 (Chao et al., 2017). Traditional methods for calculating line losses in a distribution network are based on line current calculations, including root mean squared current method, average current method, and equivalent resistance method (Zhang et al., 2013). Other proposed

methodologies consist of improving iterative methods by matching power flow calculation methods (Chen and Guo, 2005; Ding et al., 2000). However, the difficulties of implementing these methods in actual operation rely on the fact that they need detailed line parameters for calculations, which would have to be manually collected (Chao et al., 2017).

In the Brazilian system, the National Electrical Energy Agency (ANEEL, in Portuguese “Agência Nacional de Energia Elétrica”) splits the distribution systems into three levels: High Voltage Distribution System (HVDS), Medium Voltage Distribution System (MVDS), and Low Voltage Distribution System (LVDS), according to the network’s segments, transformers, and measurement equipment. Specific methods are applied for the obtainment of electrical losses for each of these levels (ANEEL, 2018):

- network losses associated with the HVDS are obtained by the data from the measurement system;
- losses in the MVDS and LVDS are calculated by the Current Injection Method, presented in Chapter 3.

By applying the above methodology, ANEEL calculates the loss levels that are considered efficient for the distribution system. Hence, these levels are deterministic values and they are reviewed by the Agency at each Periodic Tariff Review (RTP, in Portuguese “Revisão Tarifária Periódica”), which usually occurs every 4 or 5 years, and are not altered during the current RTP cycle. The efficient loss levels calculated by ANEEL are important to the distributors because these values are contemplated in the distributor’s costs with the purchase of energy; hence, they can be passed on to the final consumers through the tariffs. Therefore, loss values greater than the efficient level represent a high additional cost that distributors cannot pass on the clients (ANEEL, 2018).

The introduction of generation sources on distribution systems can significantly impact the power flow and voltage conditions at customers and utility equipment. These impacts may manifest themselves positively or negatively, depending on the system’s operating characteristics and the DGs characteristics. Therefore, it is critical that the power system impacts are assessed accurately so that these DG units can be inserted in a manner that avoids causing degradation of power quality, reliability, and control of the distribution system (Barker and Mello, 2000).

Positive impacts are generally called “system support benefits”, and may include (Barker and Mello, 2000):

- voltage support and improved power quality;

- loss reduction;
- transmission and distribution capacity release;
- improved utility system reliability.

However, achieving these benefits is in practice much more difficult than is often realized because DG sources must be reliable, dispatchable, of the proper size, at the proper locations, and they must also meet various other operating criteria. Since many DGs are not utility-owned and are variable energy sources, such as solar and wind, there is no guarantee that these conditions will be satisfied and that the full system support benefits will be realized. In fact, power system operations may be adversely impacted by the introduction of DG if certain minimum standards for control, installation, and placement are not maintained (Barker and Mello, 2000).

Hence, a good load flow analysis software should be able to model, for example, the effects on the system's technical losses after the installation of DG units. On feeders where losses are high, a small amount of strategically placed DG with an output of just 10–20% of the feeder total demand can have a significant loss reduction benefit for the system (Barker and Mello, 2000).

In Brazil, since the methodology adopted by ANEEL is deterministic, if during an RTP cycle, the local generation experiences great variations, due to the stochastic nature of renewable DGs, there may be a divergence between the technical losses recognized by ANEEL and the ones observed by the distributors. However, the current regulation already contemplates the possibility of the injected generation being significant in relation to the loads, for the MVDS and LVDS. In this case, the distributors may submit a loss analysis for ANEEL's evaluation (ANEEL, 2018, 2019).

In this section, we will focus on the impacts on the distribution system's active losses due to DG units' introduction. The IEEE 34-bus test feeder was selected as our distribution test system. According to (Kersting, 2001; Davoudi et al., 2015), this system is a very long and lightly loaded actual feeder, located in Arizona, with a nominal voltage level of  $24.9kV$ , and it is characterized by:

- three-phase and single-phase lines;
- two in-line regulators, required to maintain an adequate voltage profile;
- an in-line transformer, reducing the voltage to  $4.16kV$ , for a short section of the feeder, downstream the transformer at bus 832;
- unbalanced loading, with both “spot” and “distributed” loads, which are assumed to be connected at the center of the line segment;
- shunt capacitors.

Figure 4.1 illustrates the feeder, with its 25 loads.

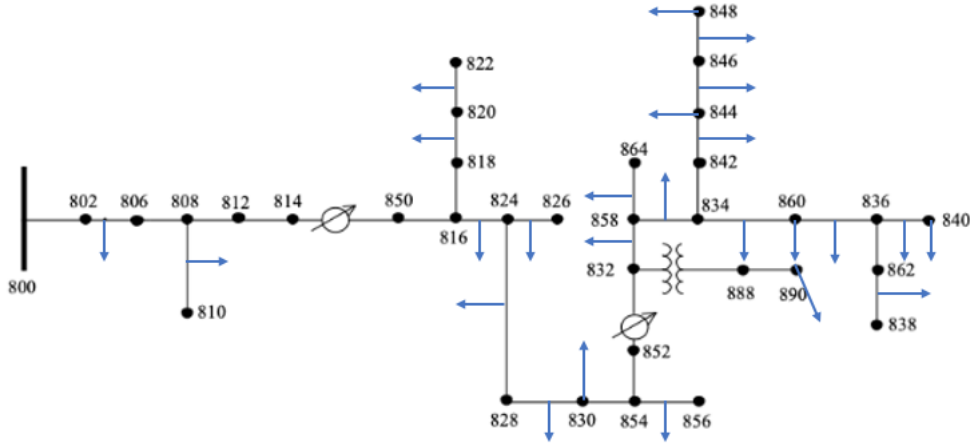


Figure 4.1: Original configuration of IEEE 34 bus system. Source: Adapted from (Panesso-Hernández et al., 2015).

In order to evaluate the impacts of DG insertion on the system's electric losses, we modified the original system of Figure 4.1, by considering only spot loads and the insertion of PV generation in six buses, operating in “parallel” with the feeder. The modified system presents 28 spot loads, as illustrates Figure 4.2.

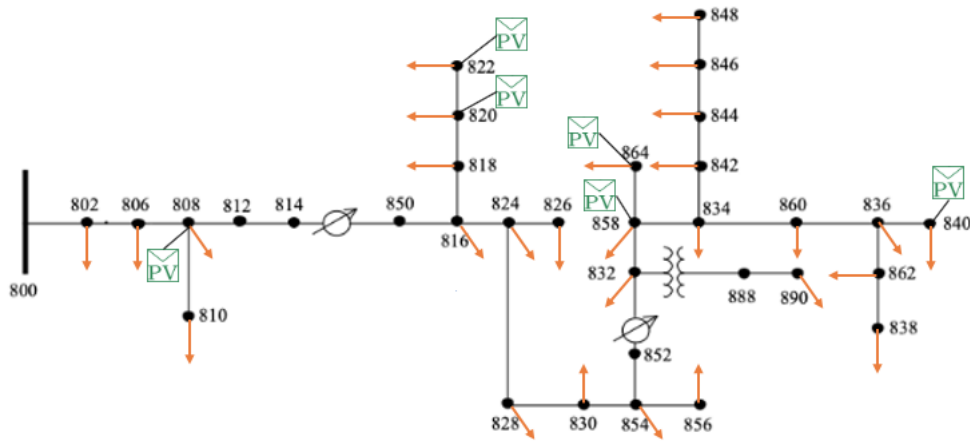


Figure 4.2: Modified configuration of IEEE 34 bus system. Source: Adapted from (Panesso-Hernández et al., 2015).

Distributed loads were converted into three-phase spot loads assuming, in all cases, a constant power factor of 0.9. For our study, we retrieved data from (Mancilla-David et al., 2020), to obtain:

- one year of loads' hourly measurements for each bus of the system. The peak loads in this profile match those of the IEEE test feeder.
- PV generation profiles, containing one year of hourly measurements, for each unit installed in the system, which were obtained based on data from one year of hourly solar spatial radiance and ambient temperatures, considering the length of feeder segments for spatio-locations. These weather data were converted into power production at Maximum Power Point (MPP) tracking, assuming the PV arrays are constructed using Canadian Solar CS6X-325P PV panels.

This section's main goal is to analyze the effects of inserting DG units in the operation of a distribution system. Next, we describe two case studies that aim to evaluate whether this type of generation is beneficial, or not, to the system.

#### 4.1.1

##### **DG insertion as a percentage of the bus load**

Once we obtained those data from (Mancilla-David et al., 2020), we inserted the DG units in the system, at different levels for each hour of the day, in a way that they would be able to meet a percentage  $\alpha$  of the corresponding bus load. Hence, each bus' hourly PV generation was calculated based on the generation profiles and the corresponding bus loads.

An essential aspect of this calculation is that the DG's insertion level  $\alpha$  must agree with the variations of the generation profiles so that, throughout the morning, these insertion levels increase until they reach the maximum generation value (which, typically, occurs at 12h or 13h) and, then, decrease, until there is no more PV generation available. Therefore, to obtain these hourly penetration levels, we calculated the maximum hourly generation value for each day of the historical data and for each bus of the system, and considered that this value corresponds to an insertion level of 100%. Hence, the remaining levels were calculated as a percentage of the maximum value obtained.

Another critical aspect observed is that the PV generation must be close enough to the load values not to limit its capacity on the load supply. In the hour of maximum PV generation, the insertion level must be 100%, so that the total bus load, for this hour, can be supplied by the DG unit. For example, consider a situation where a particular bus presents a 100kW load at 12h and the corresponding PV generation is limited to 20kW. At noon, according to our premise, the PV generation should be able to supply the total bus load, but it is limited to a much smaller value, leading us to an infeasible situation.

Hence, to adequate generation and load values, we calculated a factor  $\beta$ , for each bus, and each day, to scale the PV generation profile so that the total load average is equal to the total modified PV generation profile average.

Therefore, in order to address the aforementioned aspects, we implemented the procedure described in Algorithm 4 to insert the DG on the modified IEEE 34-bus system.

---

**Algorithm 4** DG insertion on the modified IEEE 34-bus system

---

- 1: Calculation of the daily load average at each bus ( $average_{load}$ ).
  - 2: Calculation of the daily average of the PV generation profile for each bus ( $average_{PV}$ ).
  - 3: Calculation of a factor  $\beta = average_{PV}/average_{load}$ , for each bus, at each day of the historical data.
  - 4: Division of all the daily generation profiles ( $profile_{i,t}$ ) by the  $\beta$  factor, at each bus.
  - 5: Identification of the highest value for the daily generation profile, at each bus  $i$ , given by  $max_i$ .
  - 6: Calculation of a parameter  $\alpha_{i,t} = profile_{i,t}/max_i$ , representative of the DG hourly insertion, for each bus  $i$ , in a period  $t$ .
  - 7: The PV generation, at each bus, can be obtained by the multiplication of its hourly load to the corresponding factor  $\alpha$ .
- 

To illustrate the procedure for obtaining the PV generation, Figures 4.3a and 4.3b show the values of the hourly PV profiles, already scaled by the parameter  $\beta$ , and the load values for bus 808, at the first day of the historical series. The  $\alpha$  parameter was obtained by dividing each value of the generation profile by the maximum value of the daily profile, as shows Figure 4.3a. Once the parameter  $\alpha$  was calculated, the PV generation was obtained by multiplying this factor by the corresponding hourly load, as shows Figure 4.3b. This procedure was implemented for all of the system's buses where the DG was inserted and for every day of the analyzed year.

Hour	Modified PV generation profile (kW)	$\alpha_{808}$	Hour	Load (kW)	PV Generation (kW)
1	0	0	1	3.11	0
2	0	0	2	3.13	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
8	0	0	8	4.44	0
9	4.35	0.203	9	4.36	0.88
10	10.66	0.497	10	4.49	2.23
11	17.26	0.804	11	5.48	4.41
Maximum	21.47	1	12	5.47	5.47
13	19.86	0.925	13	4.64	4.29
14	15.82	0.737	14	5.18	3.81
15	10.46	0.487	15	4.52	2.20
16	4.40	0.205	16	3.88	0.79
17	0.50	0.023	17	4.32	0.10
18	0	0	18	5.67	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
24	0	0	24	3.52	0

(a) Calculation of parameter  $\alpha$ . (b) Calculation of PV generation.

Figure 4.3: Procedure for obtaining the DG insertion for bus 808, at day 1.

Following this procedure, we could obtain a liquid hourly load for each bus by subtracting the PV generation from the original hourly load value. This procedure was implemented only for the active PV generations.

According to (Kersting, 2001), there may be a convergence problem in the power flow calculations of the IEEE 34-bus system due to the feeder's length and the unbalanced loading if traditional power flow methods are applied. Hence, to avoid convergence issues, we adopted the Current Injection Method approach, described in Chapter 3. Based on the calculated values for the system's liquid load, we performed simulations on the OpenDSS software, aiming to obtain the entire system's active electric losses, and considered these simulations results as our benchmark for this study. In order to evaluate the effects of the DG insertion on the system's active losses, the simulations were performed with and without the PV generators penetration.

From the simulations results, we observed that, for every month of the analyzed year, the active losses of the entire system decreased during the periods in which there is PV generation, as illustrates Figures 4.4a to 4.4l.

Besides that, we highlight that the most considerable losses of the system occurred during December.

For every hour of the historical data, we verified the percentage of the total load in the system supplied by the DG. Among the 8760 operating points that were studied, for the case in which the DG has the highest participation in the loads' supply, this percentage corresponds to 16.5%. Hence, the DG insertion in the study is within limits considered by (Barker and Mello, 2000) as beneficial for the decrease of the system's losses.

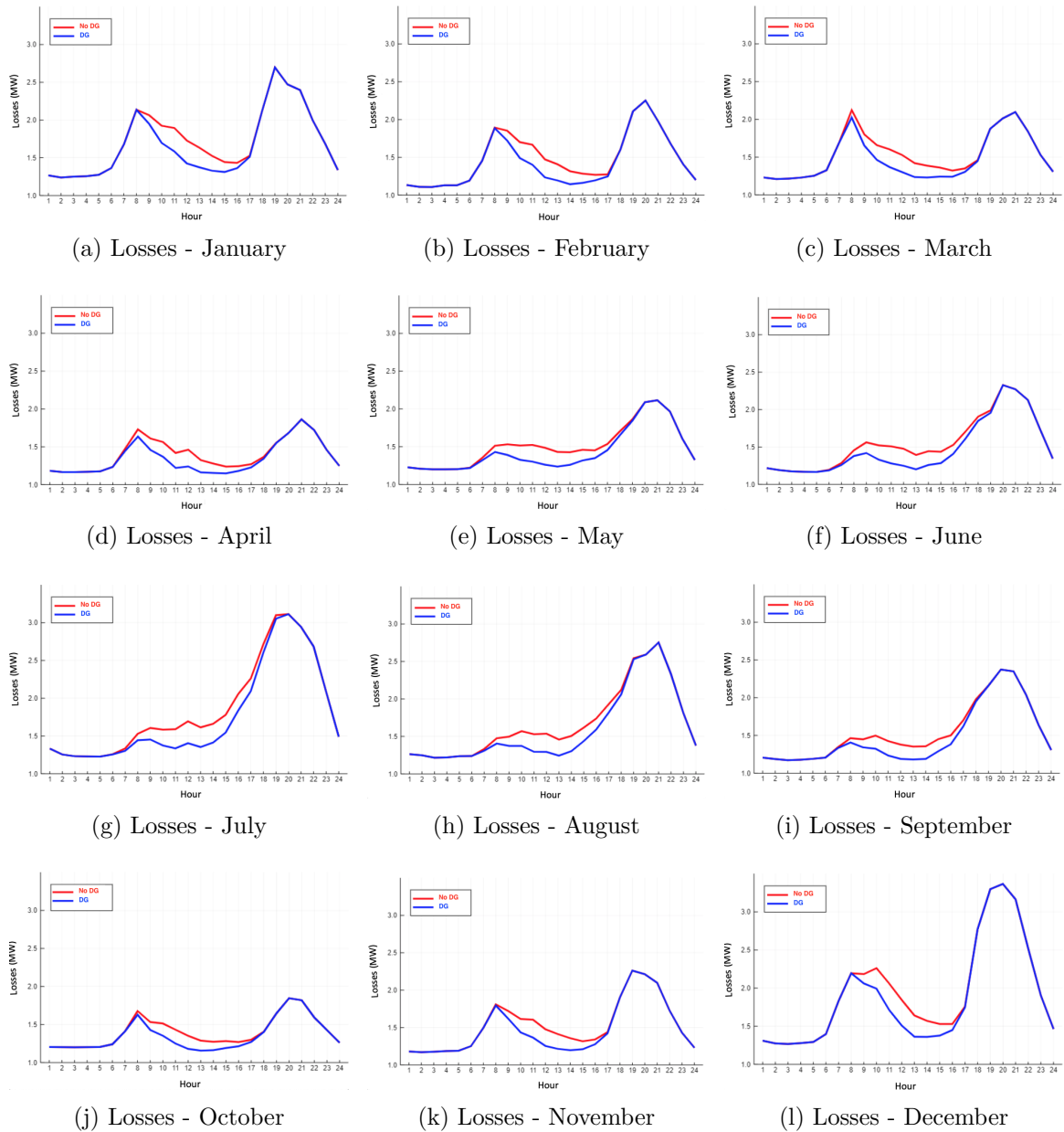


Figure 4.4: Comparisons between active losses with and without DG insertion on the system. System's losses without DG insertion are represented in red and losses with DG penetration are illustrated in blue.

### 4.1.2

#### DG insertion as a percentage of the system's total load

We simulated a new case study for the IEEE 34-bus system in which each DG unit inserted is responsible for the supply of a part of the system's total load instead of supplying a percentage of the bus load.

We applied the same procedure for obtaining the  $\alpha$  factor described in the previous section, but the PV generation was calculated based on the total system load. Therefore, we stipulated that the DG units would supply a percentage  $\alpha$  of one-third of the system's total load at each hour of the historical year. Hence, at the time period when the PV generation is at its maximum capacity ( $\alpha = 1$ ), we have six DG units generating one-third of the system's total load, i.e., the PV generation corresponds to the double of the value for the system's total demand.

We followed the same procedure described in Algorithm 4 to obtain the PV generation, adapting the load values to one-third of the system's total demand, instead of the bus load, for the calculations of the  $\beta$  factor and the PV generation. We highlight that, since we modified the load value for calculating the  $\beta$  parameter, the average of the modified PV profile is equal to the average of one-third of the system's total load for each analyzed day. Hence, the new values for the  $\alpha$  parameter are equal to the values obtained in the first study. Figures 4.5a and 4.5b present the values obtained, in this case, for the modified PV generation profile, the total load and the PV generation at bus 808 for the first year of the historical data set.

Hour	Modified PV generation profile Bus 808 (kW)	$\alpha_{808}$	Hour	Load Bus 808 (kW)	PV Generation Bus 808 (kW)
1	0	0	1	3.11	0
2	0	0	2	3.13	0
⋮	⋮	⋮	⋮	⋮	⋮
8	0	0	8	4.44	0
9	275.39	0.203	9	4.36	56.43
10	675.16	0.497	10	4.49	144.53
11	1092.78	0.804	11	5.48	294.83
Maximum	12	1359.25	1	5.47	338.20
	13	1257.27	13	4.64	294.02
	14	1001.40	14	5.18	239.43
	15	662.37	15	4.52	136.37
	16	278.27	16	3.88	50.73
	17	31.94	17	4.32	6.59
	18	0	18	5.67	0
	⋮	⋮	⋮	⋮	⋮
	24	0	24	3.52	0

(a) Calculation of the  $\alpha$  factor.(b) Hourly loads at bus 808 and obtainment of the PV generation according to the calculated values for the  $\alpha$  factor and the system total load.

Figure 4.5: Procedure for obtaining the DG insertion for bus 808, at day 1.

For every hour of the historical data, we verified the percentage of the total load in the system supplied by the DG. Among the 8760 operating points studied, for the case in which the DG has the highest participation in the loads' supply, this percentage corresponds to 200%, a much superior value than the one calculated in the first case study. This value is also much higher than the limits of DG insertion considered, in the literature, as beneficial for the system operation. The adverse effects of the high insertion of PV generation can be detected in Figures 4.6a to 4.6l, in which it is possible to observe that the system's total active losses increase in the hours of the day when there is PV generation.

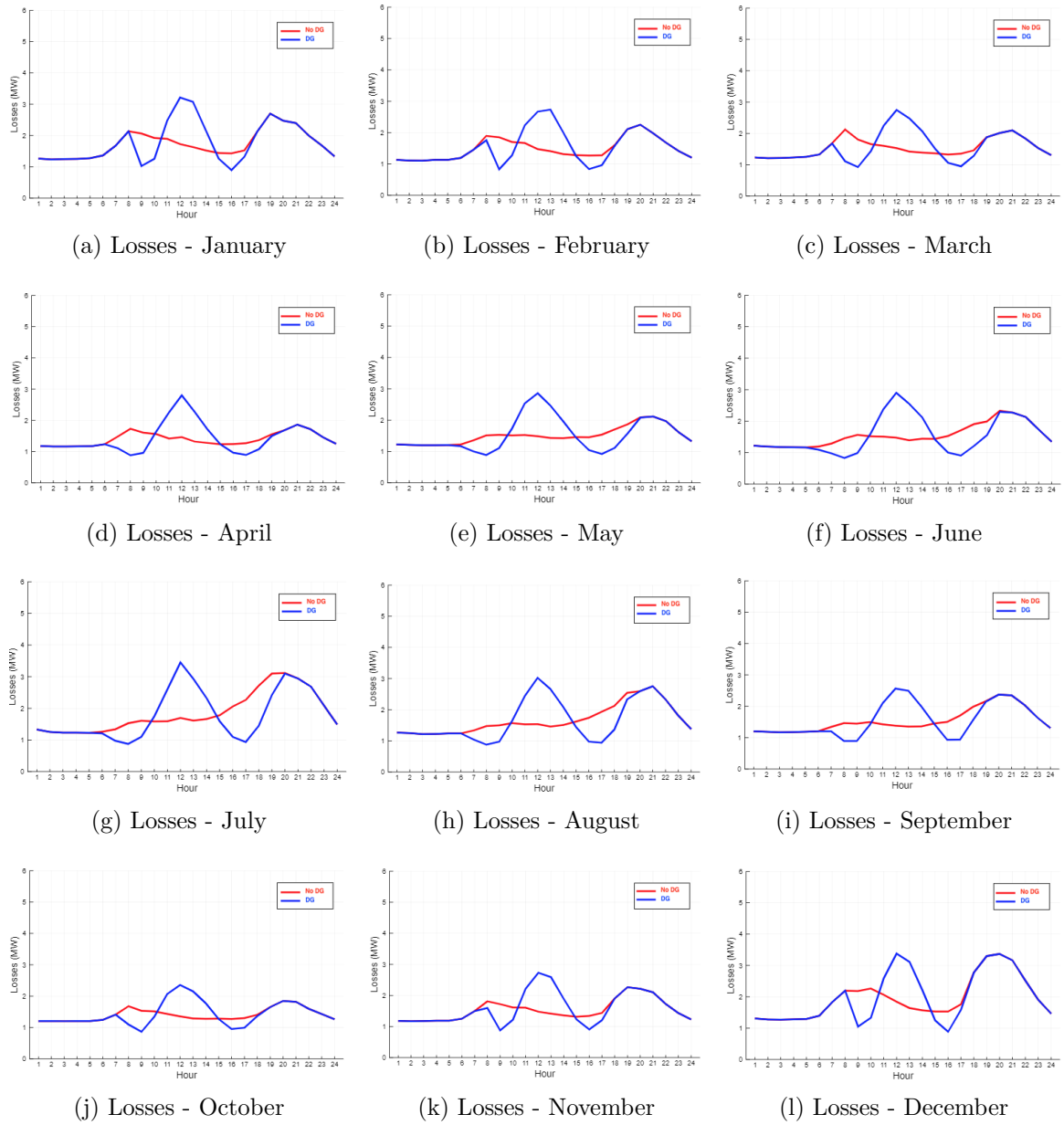


Figure 4.6: Comparisons between active losses with and without DG insertion on the system. System's losses without DG insertion are represented in red and losses with DG penetration are illustrated in blue.

From Figures 4.6a to 4.6l we can also observe that the highest values for the system's losses, in this case, occurred during July. This is due to the fact that, in this month, the PV generations are higher since it is the summer season, producing higher losses in the system.

The two case studies for the IEEE 34-bus system presented in this section were able to demonstrate the impact that DG units have on distribution systems' operation. These examples also illustrate the possibility of positive and negative effects from the DG insertion in relation to the system's losses. As mentioned by (Barker and Mello, 2000), when the distributed generation is

within acceptable limits, it has a positive impact on the system's total losses, as presented in the study of section 4.1.1. However, the last study demonstrated that, when the units are not in the proper size, they may harm the system operation regarding the active losses.

## 4.2

### ANN prediction of the system's active losses

Since the system's technical losses are an important feature for distribution utilities, as mentioned in Chapter 2, we trained an ANN, according to the simulations results obtained by OpenDSS for the system described in section 4.1.1, aiming to predict these values. As it was also pointed out in Chapter 2, our main goal in evaluating the possibility of using trained ANNs to estimate critical variables of power systems is to avoid the need to know the system's topology features for each operating point to be analyzed.

Therefore, to address this issue, we propose an ANN that requires, as inputs, only the liquid active load values at each one of the 28 buses in the system. Besides that, we also considered a bias vector, in which, initially, all elements were set to one, leading us to an ANN of 29 inputs. The historical data set, which comprises hourly measurements of loads for one year, i.e., 8760 measurements, was equally divided between training and test, that is, 50% of the data was used to train the ANN, and the remaining was used to test the trained model.

As mentioned in Chapter 3, to achieve the best network architecture possible, we started the training phase with a small network and a small number of epochs and observed the predictions generated by each configuration evaluated until the best architecture was obtained. Table 4.1 presents the ANN architecture parameters that produced the best results among the configurations evaluated.

Prediction	Hidden layers	Activation Function	Loss Function	Epochs	Batch Size
Active Losses	2	Sigmoid/ Leaky ReLU	MAE	1000	40

Table 4.1: Training parameters for the ANN.

The parameter “batch size” in Table 4.1 corresponds to the number of examples from the training data set used to estimate the error gradient. In this case, a batch size of 40 means that 40 samples from the training data set were

used to estimate the error gradient before the model weights were updated (Brownlee, 2019).

Table 4.2 shows the simulation times obtained for the train and evaluation data sets for the two methods implemented to calculate the system's active losses. We highlight that we were able to simulate the 8760 operation points in OpenDSS in such a short time due to the use of the Julia package OpenDSSDirect, which implements a direct library interface to OpenDSS (Krishnamurthy, 2020). As mentioned in Chapter 3, to implement the ANN, we used the Julia package Flux.

Data set	ANN (s)	Benchmark <sup>3</sup> (s)	Acceleration Ratio
Train	88.50	7.76	$9 \times 10^{-2}$
Evaluation	0.016	6.26	391

Table 4.2: Simulation times comparison for train and evaluation data sets.

In Table 4.2, we calculated an Acceleration Ratio that corresponds to the time required by the benchmark method (in this case, OpenDSS) to run the simulations, divided by the ANN time to generate predictions for the same data set. We can observe that the necessary time to train the ANN is greater than OpenDSS time to run the same simulations, causing the ratio to be smaller than one. However, for the evaluation data set, the ANN is much faster in predicting the system's active losses than OpenDSS, leading us to an Acceleration Ratio of 391. Therefore, even though the ANN training requires more computational time compared to OpenDSS performance, we gain a much greater time in the simulation of the test set.

Although computational time is an important aspect of the study, it is not enough to determine one technique's superiority over the other. In order to evaluate the performance of the ANN predictions, we calculated the following metrics, considering that  $y_t$  is our target value,  $\hat{y}_t$  is the predicted one, at each period  $t$ , and that we have a total of  $n$  periods:

<sup>3</sup>We highlight that, for the benchmark software, there is no need for training and evaluation phases. In the Table, the time referring to the training and evaluation sets for the software corresponds to the simulation of the operating points in each of the two sets. The division of the operating points' simulation between training and evaluation sets allowed us to effectively compare the times obtained by the benchmark method and by the ANN. The same approach was used in Tables 4.4 and 5.1.

- **MAE** (Mean Absolute Error):

$$MAE = \frac{1}{n} \sum_{t=1}^n |error_t|, \quad (4-1)$$

where  $error_t = y_t - \hat{y}_t$ .

- **MAPE** (Mean Absolute Percentage Error):

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|error_t|}{|y_t|} \quad (4-2)$$

- **Percentiles of the prediction errors** ( $P5$ ,  $P50$  e  $P95$ ).

We also calculated the above metrics according to the total active load of each period  $t$ :

- **MAE in relation to the total active load:**

$$MAE_{load} = \frac{1}{n} \sum_{t=1}^n \frac{|error_t|}{load_t}, \quad (4-3)$$

where  $load_t$  corresponds to the total active load of the internal system at time  $t$ .

- **MAPE in relation to the total active load:**

$$MAPE_{load} = \frac{1}{n} \sum_{t=1}^n \frac{\frac{|error_t|}{load_t}}{\frac{|y_t|}{load_t}} \quad (4-4)$$

- **Percentiles of the prediction errors, in relation to the total active load** ( $P5_{load}$ ,  $P50_{load}$  e  $P95_{load}$ ).

Since  $error_t$ ,  $load_t$  and  $y_t$  are all given in  $MW$ , the metrics calculated in relation to the demand are dimensionless. Besides that, from Equations (4-2) and (4-4), we can observe that the calculated values for  $MAPE$  and  $MAPE_{load}$  are equal. Table 4.3 presents the calculated metrics for the training and test sets.

From the results presented in Table 4.3, we can conclude that the ANN predictions for the system's active losses were very accurate. Figures 4.7a and 4.7b show the histograms of the obtained values for in-sample and out-of-sample prediction errors. We can observe that the most significant absolute error was  $0.0075MW$  in the evaluation set.

In-sample prediction (Training data set)			
Metric	Losses	Metric	Losses/Demand
MAE(MW)	$2.39 \times 10^{-4}$	MAE <sub>load</sub>	$3.25 \times 10^{-4}$
MAPE(%)	0.44	MAPE <sub>load</sub> (%)	0.44
P5(MW)	$-3.74 \times 10^{-5}$	P5 <sub>load</sub>	$-7.19 \times 10^{-5}$
P50(MW)	$1.63 \times 10^{-4}$	P50 <sub>load</sub>	$2.60 \times 10^{-4}$
P95(MW)	$7.29 \times 10^{-4}$	P95 <sub>load</sub>	$8.41 \times 10^{-4}$
Out-of-sample prediction (Test data set)			
Metric	Losses	Metric	Losses/Demand
MAE(MW)	$2.80 \times 10^{-4}$	MAE <sub>load</sub>	$3.51 \times 10^{-4}$
MAPE(%)	0.46	MAPE <sub>load</sub> (%)	0.46
P5(MW)	$-3.60 \times 10^{-5}$	P5 <sub>load</sub>	$-6.71 \times 10^{-5}$
P50(MW)	$1.67 \times 10^{-4}$	P50 <sub>load</sub>	$2.64 \times 10^{-4}$
P95(MW)	$8.88 \times 10^{-4}$	P95 <sub>load</sub>	$9.53 \times 10^{-4}$

Table 4.3: Calculated metrics for in-sample and out-of-sample predictions of system's losses.

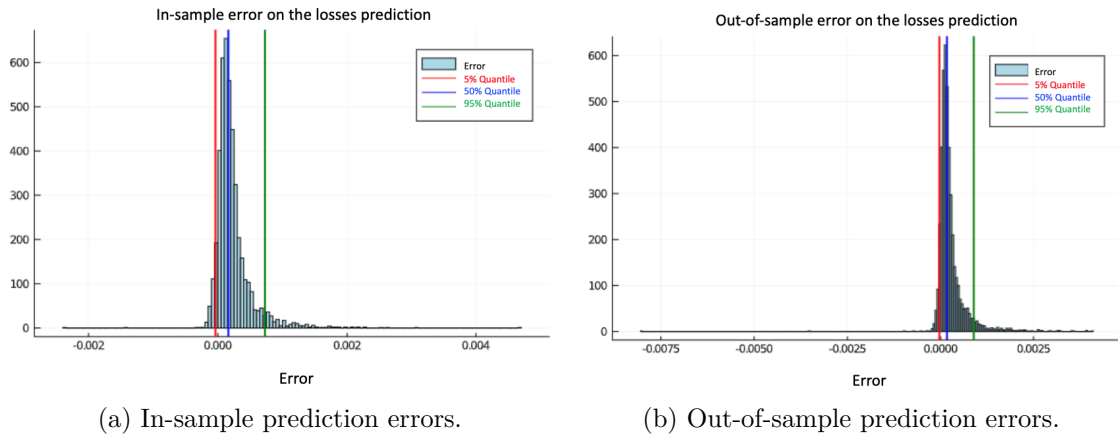


Figure 4.7: Histogram of the prediction's error for the in-sample and out-of-sample data sets. Quantiles of 5%, 50% and 95% are represented by red, blue, and green lines, respectively.

Figures 4.8a and 4.8b compare the ANN predictions to the active losses values obtained from the OpenDSS simulations, proving the remarkable capacity of the ANN to generalize to unseen operating points.

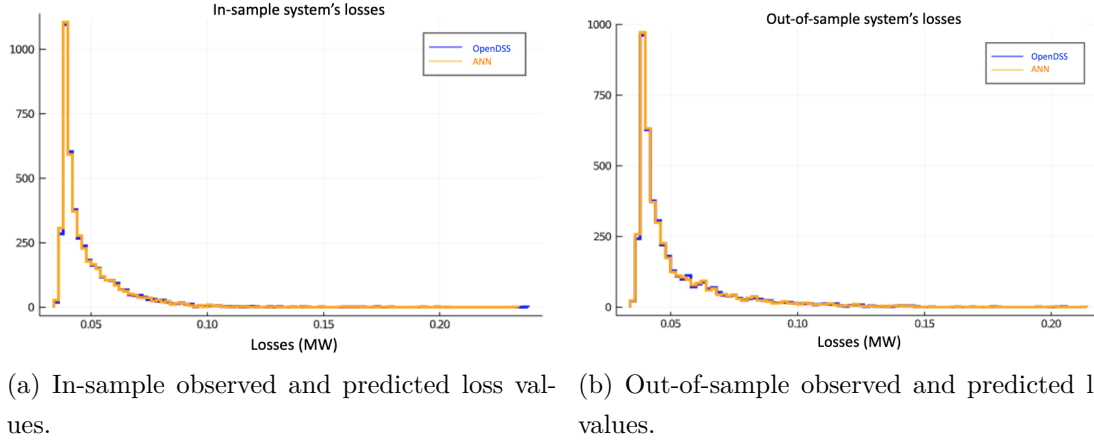


Figure 4.8: Histogram of the observed and predicted loss values for the in-sample and out-of-sample data sets. OpenDSS' results are represented in blue and ANN's predictions are illustrated in orange.

From the results presented in Table 4.3 and Figures 4.7a, 4.7b, 4.8a and 4.8b, we can conclude that the trained ANN architecture was able to meet, with good accuracy, the results from the OpenDSS simulations. Although the ANN training time was greater than OpenDSS simulation time, we can verify that, once the ANN is well trained, applying the model obtained to new inputs is practically instantaneous, as the evaluation time in Table 4.2 demonstrates.

Based on that, we can infer that, although we require knowledge of topology features from the distribution system and, possibly, a long time to train an ANN aiming to predict the system's losses, once the best model is obtained, the ANN approach may constitute an adequate option to replace applications, such as OpenDSS, when there is a need for fast results obtainment for new operating points.

Also, as mentioned in Chapter 2, the knowledge of all topology features for each operating point is a common challenge in distribution systems. In this aspect, the ANN approach offers an advantage since once the ANN is trained, not all topology features need to be available.

### 4.3

#### Power flows between distribution and transmission systems

Figure 4.9 illustrates a distribution system under analysis at time  $t \in \mathcal{T}$  and its respective power flow, with  $\mathcal{T}$  being the set of intervals of 15 minutes in the study horizon. The demands  $\{P_{j,t}\}$ ,  $j \in \mathcal{J}$ , and  $\{P_{u,t}\}$ ,  $u \in \mathcal{U}$ , correspond to power imports at border points, with  $\mathcal{J}$  and  $\mathcal{U}$  being the sets corresponding to the network's connection points with the transmission system and with other distributors, respectively (Street et al., 2020).

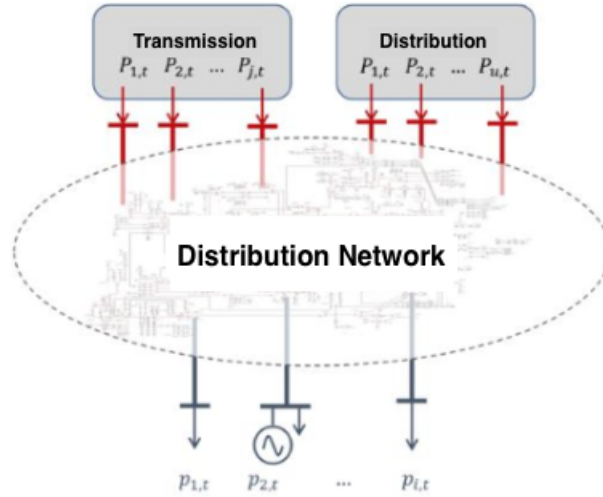


Figure 4.9: System at instant  $t$ . Source: Adapted from (Street et al., 2020).

The values verified for  $\{P_{j,t}\}$  and  $\{P_{u,t}\}$  are a consequence of the topology of the electrical system and of the operating point, defined by  $\{p_{i,t}\}$ ,  $i \in \mathcal{B}$ , with  $\mathcal{B}$  being the set of load and generation buses, or buses representing input substations for regions of final consumers (Street et al., 2020).

The main goal of the study presented in this section is to obtain, for a real distribution system in Brazil, the power flows on the borders between the network and the transmission system, i.e., variables  $\{P_{j,t}\}$  of Figure 4.9. The analyzed network corresponds to a real distribution system in Brazil, which comprises 12 borders with the transmission system. In order to obtain the flows  $\{P_{j,t}\}$ , we considered the full representation of the Brazilian electric system.

For this study, we used historical data of the distribution system selected: we considered active and reactive load variations in 46 load buses and active generation variations in seven generation buses, while the remaining loads and generations of the electric system were considered constant during the simulations. The historical data were obtained from measurements, at every 15 minutes, of the period from 01/01/2016 to 31/03/2019, totaling 113344 operating points.

Power flows on the borders were considered positive in the case of energy import from the external system. Hence, we considered the flow from the buses external to the distribution system to the internal buses as positive for our calculations. Therefore, in the case of energy export from the distribution to the transmission system, we have negative power flows.

Since the simulations were conducted considering the entire Brazilian electric system and not only the distribution network, we applied Newton's method for the power flow calculations. Hence, we considered the results obtained by the software Organon as our benchmark. We were able to simulate

this amount of operating points in Organon using a Dynamic Link Library (DLL), which implements a direct library interface of Organon in Julia. We highlight that this DLL was developed by HPPA and is not available for public access.

Once we obtained Organon results for the flows in the 12 borders between distribution and transmission systems and, aiming to predict these values without the need to conduct power flow calculations for each operating point, we designed an ANN to predict flows at the borders. This ANN was trained according to Organon simulations results.

The ANN was designed with 54 inputs, corresponding to the historical data of 46 load buses and seven generation buses, plus a bias vector, and 12 outputs, corresponding to flows in the borders with the transmission system. The data set was equally divided between training and evaluation, i.e., 50% of the historical data was used to train the ANN, and the remaining was used to evaluate the trained model. Table 4.4 summarizes the parameters adopted for the best ANN architecture obtained.

Prediction	Hidden layers	Activation Function	Loss Function	Epochs	Batch Size
Flows at Borders	2	Sigmoid/ Leaky ReLU	MSE	1000	32

Table 4.4: Training parameters for the ANN.

After training and testing the best model obtained for the ANN, we compared the simulation times for our benchmark (Organon) and the ANN. Once again, we calculated an Acceleration Ratio in order to compare both times. Table 4.5 presents the simulation times obtained and the calculated ratio.

Data set	ANN (s)	Benchmark (s)	Acceleration Ratio
Train	1924	2556	1
Evaluation	0.134	2520	18806

Table 4.5: Simulation times comparison for train and evaluation data sets. The simulations' details for the train and evaluation data sets by the benchmark method are described on the footnote on page 67.

From Table 4.5, it is possible to observe that, although the obtained times for the simulations of the training data set were fairly close, leading to an Acceleration Ratio of approximately one, for the evaluation set, the

ANN time required to predict the flows in the borders was about 18806 times smaller than the necessary time for our benchmark software to run the same simulations.

Despite the great results obtained regarding the simulation times, in order to fully evaluate the ANN's performance in predicting power flows on the 12 borders, we calculated seven metrics, described as follows. Due to the possibility of power flow inversion, we assumed that the observed values  $y_t$  correspond to positive flows. Hence, by denoting the direction of the observed flows as  $sgn_t$  and the predicted flows as  $\hat{y}_t$ , we calculated the prediction errors as:

$$sgn_t = y_t / |y_t| \quad (4-5)$$

$$error_t = sgn_t(y_t - \hat{y}_t) \quad (4-6)$$

Based on the values obtained for the prediction errors, given by Equation (4-6), we were able to calculate, for each border, the three metrics described in section 4.2: MAE, given by Equation (4-1), MAPE, given by Equation (4-2), and the Percentiles of the prediction errors ( $P5$ ,  $P50$  and  $P95$ ). Besides that, for the predictions of the flows at the borders, we also calculated:

– **MSE** (Mean Squared Error):

$$MSE = \frac{1}{n} \sum_{t=1}^n error_t^2 \quad (4-7)$$

In order to evaluate what a prediction error on the power flow in fact represents, we calculated, for each border, the average of the absolute observed flows, given by:

$$FLOW_{avg} = \frac{1}{n} \sum_{t=1}^n |y_t| \quad (4-8)$$

Based on the values obtained from equation (4-8), we calculated the MAE in relation to the average flow as:

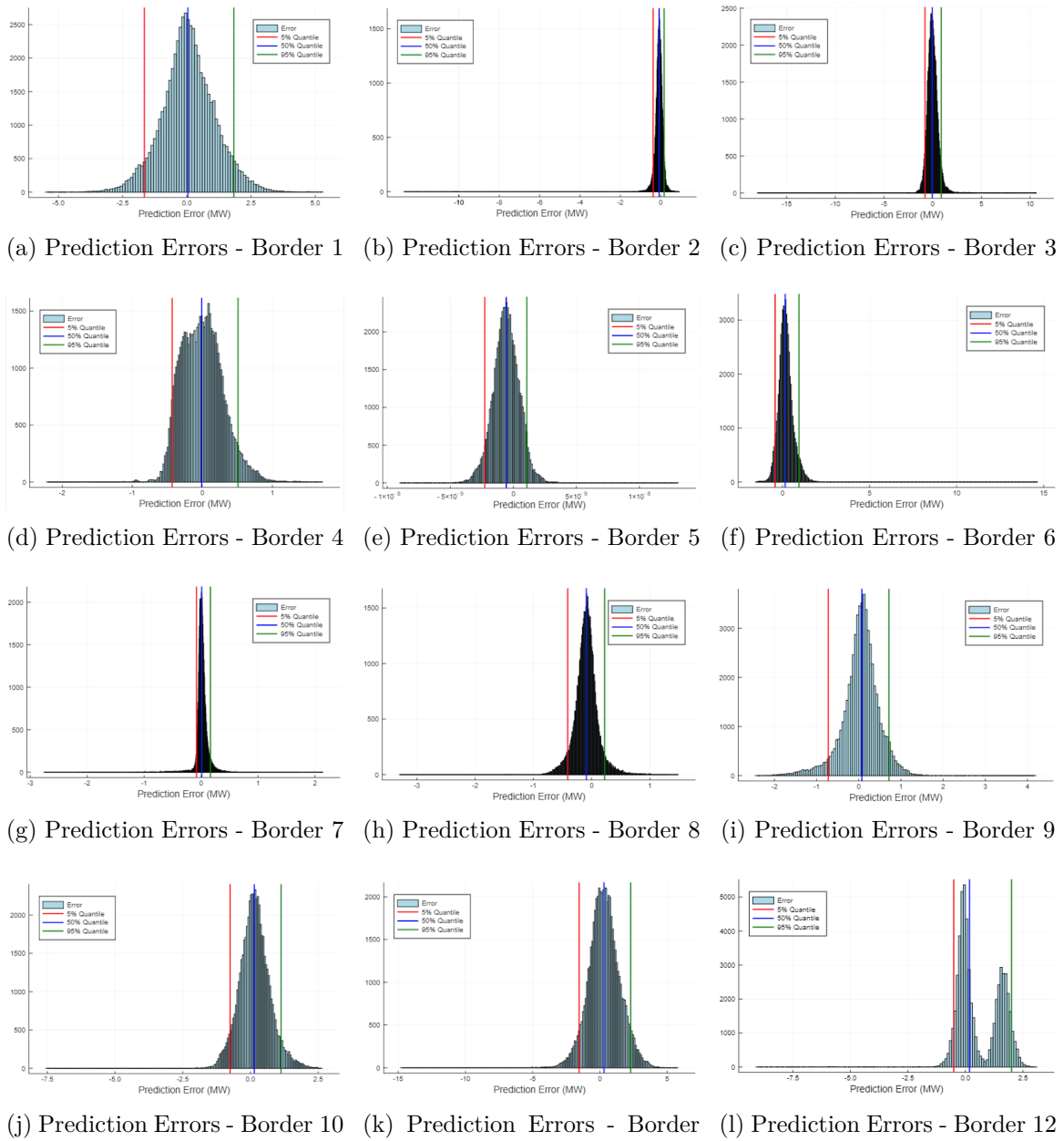
– **MAE in relation to the average flow:**

$$MAE_{rel} = \frac{MAE}{FLOW_{avg}} \quad (4-9)$$

Hence, through Equation (4-9), we can analyze the magnitude of the Mean Absolute Error calculated by Equation (4-1). For example, if we obtain, for a given border, at a given operating point, a prediction error of  $5MW$ , this error will be more significant if the average flow on the border is  $20MW$  than if it is  $200MW$ . The MAE calculation in relation to the average flow, given by Equation (4-9), provides us this sensitivity in the error analysis.

Table 4.6 summarizes the calculated metric values for each border between the distribution system and the Brazilian transmission system.

From the results presented in Table 4.6, we can observe the ANN predictions' great accuracy for the flows at the borders. Figures 4.10a to 4.10l show the histograms of the prediction errors obtained for the out-of-sample data set. We can observe that the worst predictions occurred at Border 11.



11

Figure 4.10: Histograms of out-of-sample prediction errors for the flows on the borders. Quantiles of 5%, 50% and 95% are represented by red, blue, and green lines, respectively.

Border	MAE	MAE <sub>rel</sub>	MAPE	P5	P50	P95	MSE
Border 1	0.799	$9.921 \times 10^{-3}$	0.993%	<b>-1.657</b>	0.040	1.829	1.095
Border 2	0.151	$3.445 \times 10^{-3}$	0.476%	-0.378	-0.079	0.168	0.082
Border 3	0.406	$2.874 \times 10^{-3}$	0.297%	-0.795	-0.040	0.872	0.318
Border 4	0.237	$5.033 \times 10^{-3}$	0.514%	-0.428	-0.010	0.510	0.087
Border 5	$9.348 \times 10^{-10}$	$1.902 \times 10^{-6}$	0.0002%	$-2.262 \times 10^{-9}$	$-5.668 \times 10^{-10}$	$1.071 \times 10^{-9}$	$1.392 \times 10^{-18}$
Border 6	0.337	$8.371 \times 10^{-3}$	0.933%	-0.427	0.159	0.942	0.206
Border 7	0.068	$2.314 \times 10^{-3}$	0.258%	-0.080	0.013	0.165	0.021
Border 8	0.164	$4.235 \times 10^{-3}$	0.428%	-0.411	-0.089	0.224	0.048
Border 9	0.323	$6.443 \times 10^{-3}$	0.656%	-0.723	0.073	0.714	0.195
Border 10	0.450	$5.063 \times 10^{-3}$	0.532%	-0.747	0.145	1.130	0.344
Border 11	<b>0.935</b>	<b><math>1.370 \times 10^{-2}</math></b>	<b>1.381%</b>	-1.547	<b>0.299</b>	<b>2.289</b>	<b>1.454</b>
Border 12	0.810	$1.266 \times 10^{-2}$	1.343%	-0.512	0.173	2.002	1.241
Average	0.390	$6.171 \times 10^{-3}$	0.651%	-0.642	0.057	0.904	0.424
Maximum	<b>0.935</b>	<b><math>1.370 \times 10^{-2}</math></b>	<b>1.381%</b>	<b>-1.657</b>	<b>0.299</b>	<b>2.289</b>	<b>1.454</b>

Table 4.6: Calculated metrics for out-of-sample prediction errors for borders flows.

Figures 4.11a to 4.11l compare the ANN predictions and the observed flows at the borders obtained from Organon simulations, proving, once again, the remarkable capacity of the ANN to generalize to unseen operating points.

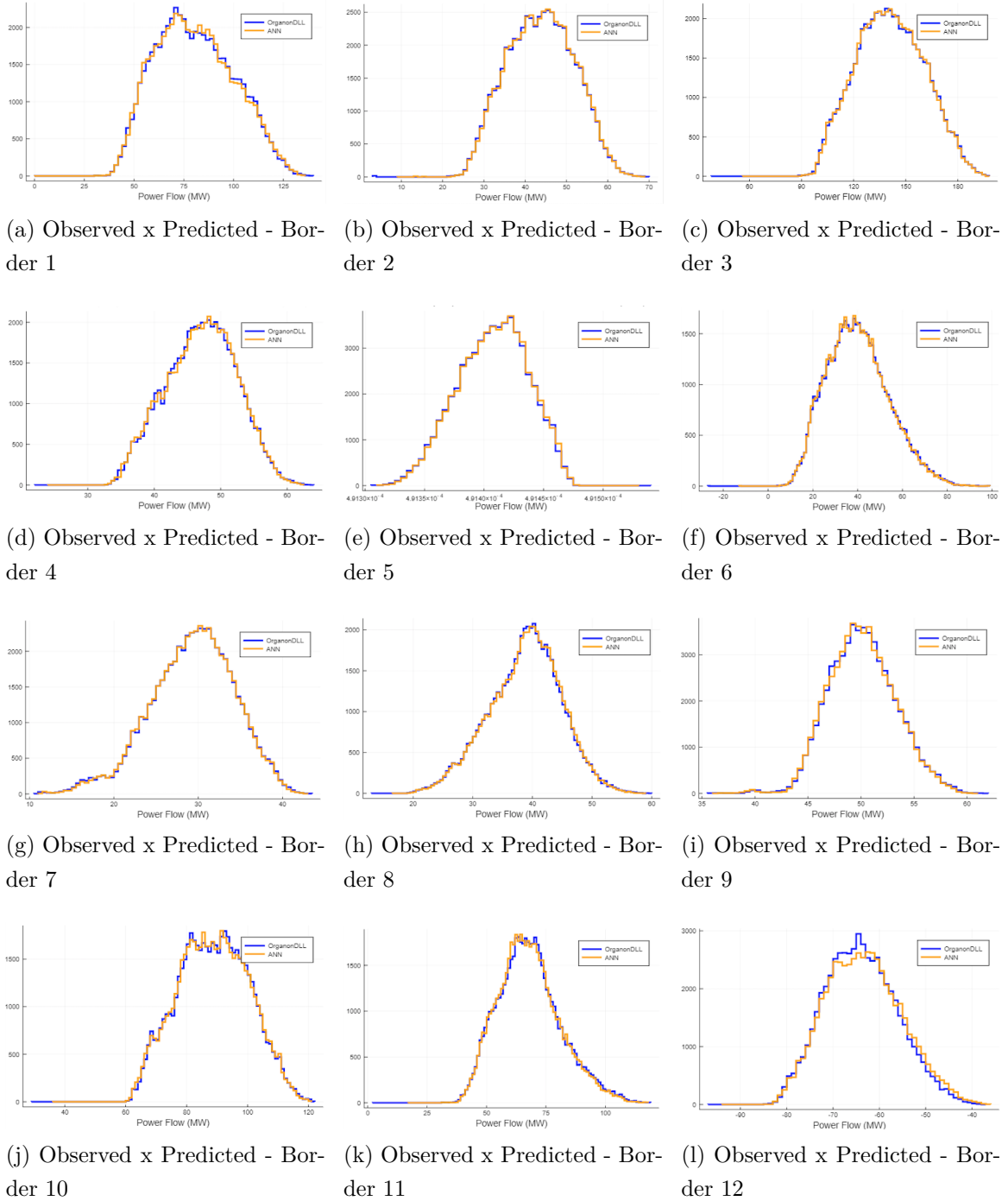


Figure 4.11: Histograms of the observed and predicted borders flows for the out-of-sample data set. Organon's results are represented in blue and ANN's predictions are illustrated in orange.

From the results obtained, we observed that Border 5 presented very small values for energy import. This is due to the fact that this border is

only used in case of contingencies, i.e., under normal operating conditions, the power flowing in this border is around zero. Besides that, on Border 12, we can observe that, in all of the operating points simulated, the power flows were negative, leading us to conclude that this border exported energy during the entire historical time that was analyzed. From the errors histograms, illustrated in Figures 4.10a to 4.10l, we can observe that the highest prediction errors occurred in a very small amount of operating points when compared to the simulated total.

Based on that and on the high Acceleration Ratio obtained for the evaluation data set, we can conclude that, for this application, the ANN approach can also be considered an adequate tool to rapidly obtain results when there is no need to run power flow calculations, or in situations in which the topology features are unavailable or incomplete.

In the Brazilian transmission system, operation and maintenance costs are shared among its users, essentially generators and distributors. Thus, at each connection point between transmission and distribution, the distributors must pay a tariff for the use of the transmission system, applied to an annual demand contract<sup>4</sup>. The definition of this contract is based on the user's maximum demand forecasts for a four-year horizon. Hence, since the contract constitutes a decision under uncertainty, the distributor's risk aversion profile must be reflected in it (Telles et al., 2018).

Therefore, the flows in the borders between the distribution and the transmission systems directly influence the demand contracting process and, consequently, the distributor's costs. The next Chapter will present an analysis of the demand contracting process and a comparison between the simulated contracts' results, according to the values of the flows at the borders obtained by Organon and predicted by the ANN.

<sup>4</sup>In the Brazilian context, this demand contract was named "Montante de Uso do Sistema de Transmissão" (MUST), defined by the regulatory rule REN666/2015 (ANEEL, 2015).

## 5

### Application of ANN methodology

In the previous Chapter, we presented the possibility to implement trained ANNs to predict power flows. For the case study considered, the ANN was trained based on the results obtained by the benchmark method and, then, we performed tests, with a different data set, in order to evaluate the ANN's predictions. We were able to observe that the predictions were accurate and the time required to run the simulations, once the ANN was appropriately trained, was much smaller than the time required by the benchmark method to perform the same calculations.

We concluded Chapter 4 mentioning that power flows in the borders between the transmission and the distribution systems are an essential feature for distributors, since they have a direct influence in the demand contracting process. Next, we will present the basis of this demand contracting process, according to proposals from the literature, and will compare the optimal contracts considering the power flow in the borders obtained by Organon and by the ANN, for the distribution system described in Chapter 4.

#### 5.1

##### Basis of the demand contracting process

According to (Telles et al., 2018) demand contracts between the distribution and the transmission systems have two main purposes: remunerate the costs of the transmission system, and provide adequate signs for the expansion of the transmission network. In this sense, the accuracy of the maximum demand forecast, made by the distributors, is essential for the planner to identify the needs for investments and reinforcements or improvements in the network to meet the demand and to ensure the reliability of the system. If the observed demand exceeds the contracted amount (*undercontracting*), in addition to paying the amount surplus, the distributor will be penalized for exposing the system to supply risk. On the other hand, if the distributor overestimates the maximum demand over the contract horizon (*overcontracting*), it may suffer penalties, in order to induce better forecasts and avoid unnecessary investments.

To determine the efficiency of a demand contract, the difference between the maximum demand observed and the value of the demand contract must be obtained, considering a tolerance. In the case of the Brazilian system, for the monthly horizon, this tolerance is 110% of the contracted value: if

the maximum demand verified is greater than this tolerance, penalties for undercontracting must be applied. On the other hand, for the annual horizon, if the maximum demand verified is less than 90% of the contracted value, the distributor is penalized for overcontracting. In both cases, the distributor cannot pass on the costs of the penalties to final consumers (Telles et al., 2018).

The contractual rules act as control parameters that the transmission planner employs to adjust the reliability implied in the forecasts that the demand contract represents. In an ideal scenario, the optimal contract is the one in which there is neither undercontracting nor overcontracting. However, often, additional costs due to undercontracting lead distributors to opt for a more conservative contract, which is based on the maximum demand verified in the previous year, in order to avoid penalties. Thus, it is of great relevance for the electricity sector to develop an optimal strategy for demand contracts, considering the tradeoff between the exposure to penalty costs and the fixed contract cost, considering the risk profile of each distributor (Telles et al., 2018).

## 5.2

### Strategies to obtain the optimal demand contract

The strategy for defining the demand contract must take into account, in addition to the contracting rules, uncertainties that affect the power flow at the distributor's connection points, such as the uncertainties of consumption and generation and the ones originated from the renewable and intermittent supply in the distributor's network. For simplicity, any renewable and intermittent injection into the distributor's network will be called Internal Renewable Generation (IRG) from now on (Telles et al., 2018).

Thus, in addition to the uncertainties related to consumer behavior and generation capacity, climate variations, such as temperature and rainfall, generally explain peaks in demand and IRG profiles. In general, when the IRG is relevant in relation to the distributor's total demand, there is a decrease in demand at the connection points. As a result, the demand contract is reduced and the expansion of the transmission system is benefited, since less investments in expansion are needed to meet the distributor's demand. Hence, to capture the benefits of IRG in the expansion of the transmission system, it is necessary to consider it in the demand hiring decision; otherwise, cases of extreme overcontracting may occur. This aspect has become increasingly important with the growth of DG units' insertion, making the correct characterization of IRG variability essential (Telles et al., 2018; Street et al., 2020).

Therefore, based on the exposed factors, the distributor's decision must

take into account both the benefits of the IRG and the costs for contract penalty. The higher the IRG, the smaller the demand contract should be, since net imports are reduced (Street et al., 2020).

In addition, aspects of the distribution systems' actual operation, such as the random occurrence of contingencies and the realization of maneuvers on the network, when pre-defined operating rules are violated, directly affect power imports at border buses. Such rules may include network security limits or serve as a contingency plan in the event of equipment failure. Thus, the distributors contracting decision strategy must consider the tradeoff between the minimum value of demand contract that avoids unnecessary investments in the transmission system, and the demand contract that minimizes penalty costs (Telles et al., 2018).

According to (Street et al., 2020), there are three main stages that should be considered in the obtainment of the optimal demand contracting strategy, as illustrates Figure 5.1. The next sections will describe these stages for the distribution system studied in Chapter 4.



Figure 5.1: Main stages for optimal demand contracting strategies.

### 5.2.1

#### First stage: Simulation of demand/generation scenarios

According to (Street et al., 2020), the first stage of the demand contracting process corresponds to the generation of a scenarios set  $\Omega$  that represent the variability in the operating points  $p_{i,t}$ ,  $\forall i \in \mathcal{B}$ ,  $t \in \mathcal{T}$ , as illustrated in Figure 4.9, produced by uncertainties regarding demand and IRG. Thus, a set of scenarios for  $i \in \mathcal{B}$  and  $t \in \mathcal{T}$  must be defined for the contracting horizon.

Load trends are influenced by several factors but, in general, follow statewide temperatures on average and the load profiles from the same period in previous months and years (CAISO, 2019). Hence, statistical methods can be used to simulate a set of scenarios, based on the history of demand and climate variables (Street et al., 2020).

For the case of the distribution system selected, we generated a set  $\Omega$  of 200 scenarios for each one of the 46 load buses and seven generation buses registered in the study. In order to estimate the models applied to create the scenarios, we used the historical data, from 01/01/2016 to 31/03/2019, for each

load/generation bus. For the scenarios generation, we adopted the methodology to simulate long-term demand scenarios in an hourly frequency, proposed in (Bodin et al., 2019; Street et al., 2020).

Once the scenarios were generated, we were able to proceed to the second stage of the demand contracting process.

### 5.2.2

#### Second stage: Power Flow Studies

Based on the load/generation values obtained for each scenario, we were able to perform power flow calculations, to obtain power imports from the transmission to the distribution system, at each border. Once again, we used Organon as our software benchmark. The flow results obtained by Organon for the scenarios generated in the First Stage correspond to hourly measurements, for 21 months (totaling 15360h).

We highlight that the necessity to simulate future scenarios representing the variability of operating points in the First Stage may result in new input (load/generation) scenarios outside the variation range from the historical data set. As mentioned in Chapter 3, ANNs can calculate, with satisfying accuracy, the solution for any new input for which it has not been trained but which is in the training's range (Ivanov et al., 2014). Hence, the previous model obtained after the ANN training with historical data in Chapter 4 could not be directly applied to the First Stage scenarios.

Aiming to train an ANN to predict the flows at the borders for the set of scenarios generated, we selected representative groups from these scenarios: from the 200 scenarios generated, 68 were selected to train the ANN, while the remaining 132 scenarios were used to evaluate the model obtained in the training phase. The three main scenarios selections performed can be described as follows:

1. **First Selection:** for each border, the 200 scenarios generated were divided into three clusters, according to the *K-means* technique. The procedure followed to determine which scenarios would be included in the training phase can be described as follows:
  - for each cluster, the scenarios liquid demand ( $d_w = \sum loads - \sum generations$ ) was calculated and the cluster's total liquid demand was obtained ( $d_c = \sum d_w$ );
  - for each border, the cluster with the highest liquid demand was selected and, from the scenarios belonging to this cluster, we selected: the scenario corresponding to the highest flow at the border;

the scenario corresponding to the highest flow at the border, for a given period, randomly selected between 1 and 15360; the scenario whose total liquid demand is the closest to the value of the average liquid demand among the scenarios; the scenario corresponding to the lowest flow at the border.

- for each border, we conducted a similar selection to the one described above, for the lowest liquid demand cluster;
- from the cluster presenting intermediate value for the liquid demand, for each border, we selected the scenario whose total liquid demand is the closest to the value of the average liquid demand among the scenarios.

Figure 5.2 illustrates the First Selection process for Border 9, for example, describing the number of scenarios per cluster, the total liquid demand for each cluster, and the corresponding selection of scenarios.

This procedure resulted in nine scenarios selected for each of the 12 borders, totaling 108 selections. However, from these scenarios, we only included in the training those that were selected more than once, resulting in the selection of 10 scenarios and the disposal of the remaining 98 scenarios.

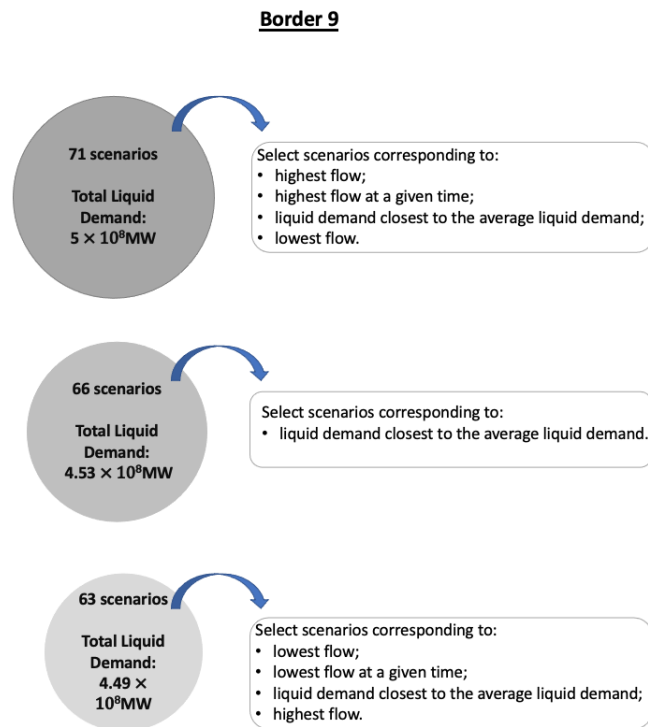


Figure 5.2: First Selection of scenarios for Border 9. The Figure illustrates that the first cluster contains a larger number of scenarios and a higher value for the total liquid demand. Similarly, the second cluster is larger and presents a higher value for total liquid demand than the last cluster.

2. **Second Selection:** for each border, we obtained the scenarios that generated the highest and the lowest observed flows. Figure 5.3 shows the procedure adopted for this selection, for Border 9.

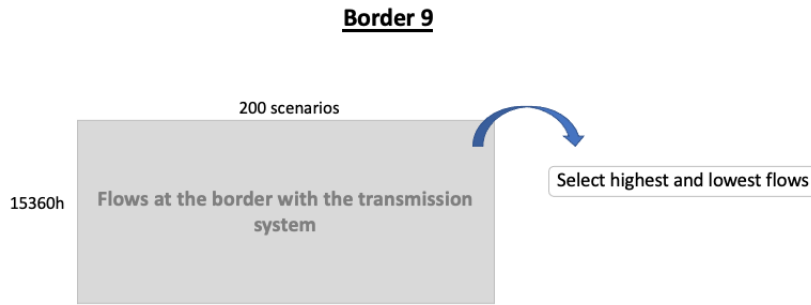


Figure 5.3: Second Selection of scenarios for Border 9.

The Second Selection resulted in 24 scenarios, two for each border. We compared these scenarios to those that were discarded in the First Selection and, once again, only the scenarios selected more than once were included in the training, resulting in 16 new scenarios.

3. **Third Selection:** for each border, we divided the power flow results into the analyzed days and obtained the maximum flow, for each of the 24h of a day, and the load/generation scenario that produced this flow. Figure 5.4 describes the process of dividing the flows at Border 9 into 24h “blocks” and selecting the highest flows for every hour of the day.

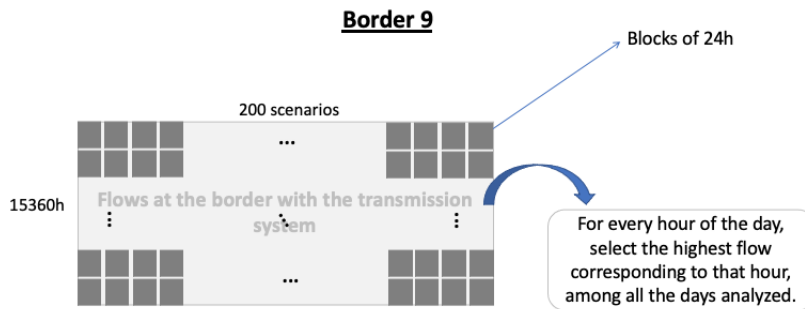


Figure 5.4: Third Selection of scenarios for Border 9.

Hence, for each border, the Third Selection resulted in 24 scenarios selected, totaling 288 scenarios for all the 12 borders. From these scenarios, we included in the training those that were not previously selected, resulting in 42 new scenarios.

The best ANN architecture obtained for the distribution system under analysis, described in Table 4.4, was applied to train the 68 scenarios selected.

In this case, we applied the Transfer Learning technique during the new training phase: we initially trained the ANN for the first scenario selected, with 30 epochs; next, for the second scenario, instead of randomly initialize the ANN's weights, we applied the model obtained from the first scenario training and so on. This way, we were able to reduce the training time and to improve the quality of the results.

After completing the training of all the scenarios, we were able to apply the final model to the test data set. Table 5.1 presents the times required by the benchmark software to run the training and test data sets simulations. It also presents the necessary time for the ANN to be trained and predict the test set's flows. The Acceleration Ratio, defined in Chapter 4, was calculated to compare both approaches' simulation times.

Data set	ANN (s)	Benchmark (s)	Acceleration Ratio
Train	1072	35445	33
Test	4.50	67876	15084

Table 5.1: Simulation times comparison for the train and evaluation data sets. The simulations' details for the train and evaluation data sets by the benchmark method are described on the footnote on page 67.

From the results presented in Table 5.1, we can observe that, in this case, both the training and evaluation times for the ANN were much shorter than the times required for our benchmark software (Organon) to perform the same calculations, with Acceleration Ratios about 33 and 15084. We highlight that, for this study, we were able to significantly reduce the training time by applying the Transfer Learning technique, previously described. Since we were not dealing with historical data, this approach allowed us to reduce the training examples presented to the ANN at each training and to set a lower number of epochs for each scenario's training since the model's weights were not randomly initialized.

Once the flows  $P_{j,t,\omega}$  at the borders between the transmission and the distribution systems were calculated by Organon and predicted by the ANN, for all the test scenarios generated at the First Stage, we were able to obtain the maximum monthly demand  $P_{j,m,a,\omega}$ ,  $\forall m \in \mathcal{M}$ ,  $a \in \mathcal{A}$  and  $\omega \in \Omega$ , where  $\mathcal{M}$  is the set of months of the year and  $\mathcal{A}$  is the set of years in the demand contract horizon, at each connection bus  $j$ . The cost of the demand contracts is a function of the maximum demand values  $P_{j,m,a,\omega}$ ; hence, these values will be adopted in the Optimization methodology, described next.

### 5.2.3

#### Third stage: Optimizing the value of the demand contract

According to (Street et al., 2020), the stochastic optimization models used in the third stage of the demand contracting process must minimize the risk measure of the cost scenarios, being subject to the constraints that define the different components of cost scenarios (fixed, variable and penalty), which translate the regulatory rules and incorporate the risk aversion profile of the distributor. In the case of the demand contract optimization, the model used corresponds to a stochastic linear optimization problem, which will be described in more detail.

As mentioned before, the optimization of the demand contract value must be performed based on regulatory standards, which define the cost to be paid by the distributors. Considering a border bus  $j$ , the total cost can be divided into two parts (Telles et al., 2018):

- **Fixed Cost** ( $c_{j,m,a}^F$ ): depends on the value of the demand contract ( $M_{j,a}$ ) and the associated tariff ( $T_{j,a}$ ). Thus, it is a deterministic component:

$$c_{j,m,a}^F = M_{j,a}T_{j,a}, \quad m \in M, a \in A \quad (5-1)$$

- **Variable Cost**: function of the set of scenarios  $\tilde{P}_{j,m,a}$  for the maximum monthly demand in a border bus  $j$ ; therefore, it is a stochastic component. This cost is related to three components, which can be obtained by subtracting  $M_{j,a}$  from the maximum demand, as follows:

1. **Cost component for maximum demand** ( $\tilde{c}_{j,m,a}^{MD}$ ): represents the rule that establishes that, in addition to the penalties, the monthly cost is defined considering the maximum between  $M_{j,a}$  and  $\tilde{P}_{j,m,a}$ . Thus, if  $\tilde{P}_{j,m,a} \leq M_{j,a}$ , only the fixed cost  $c_{j,m,a}^F$  should be paid; otherwise, there will be an additional cost  $\tilde{c}_{j,m,a}^{MD}$  applied to the difference  $\tilde{P}_{j,m,a} - M_{j,a}$  at a tariff  $T_{j,a}$ . Therefore, this cost can be described by:

$$\tilde{c}_{j,m,a}^{MD} = \max[0, \tilde{P}_{j,m,a} - M_{j,a}]T_{j,a}, \quad \forall m \in M, a \in A \quad (5-2)$$

2. **Cost component for the monthly penalty** ( $\tilde{c}_{j,m,a}^{UC}$ ): applied for undercontracting cases. If  $\tilde{P}_{j,m,a} > 1.1M_{j,a}$ , then  $\tilde{c}_{j,m,a}^{UC} > 0$ ; otherwise, this component is null. The difference  $\tilde{P}_{j,m,a} - 1.1M_{j,a}$  is valued at a tariff equal to three times  $T_{j,a}$ , i.e.:

$$\tilde{c}_{j,m,a}^{UC} = \max[0, \tilde{P}_{j,m,a} - 1.1M_{j,a}]3T_{j,a}, \quad \forall m \in M, a \in A \quad (5-3)$$

3. **Cost component for the annual penalty** ( $\tilde{c}_{j,a}^{OC}$ ): applied for cases of overcontracting. If the maximum annual demand  $\tilde{P}_{j,a}^{MAX}$  is less than 90% of the contracted value  $M_{j,a}$ , then there will be a cost of overcontracting  $\tilde{c}_{j,a}^{OC}$ ; otherwise, this cost will be zero. The difference  $0.9M_{j,a} - \tilde{P}_{j,a}^{MAX}$  is valued at a tariff equal to 12 times  $T_{j,a}$ , i.e.:

$$\tilde{c}_{j,a}^{OC} = \max[0, 0.9M_{j,a} - \tilde{P}_{j,a}^{MAX}]12T_{j,a}, \quad \forall a \in \mathcal{A} \quad (5-4)$$

Considering all the previously defined components, the total annual cost for the demand contract can be represented by:

$$\tilde{c}_{j,a}^T = \tilde{c}_{j,a}^{OC} + \sum_{m=1}^{12} (c_{j,m,a}^F + \tilde{c}_{j,m,a}^{MD} + \tilde{c}_{j,m,a}^{UC}), \quad \forall a \in \mathcal{A} \quad (5-5)$$

According to (Street et al., 2020), the “hiring policy” comprises a set of procedures and processes that must be followed in order to obtain the value referring to the power imports that must be contracted by the distributors. Thus, to define the hiring policy, the following information is required:

- corporate, regulatory and operational parameters that affect decisions and financial results;
- visions of the future, referring to uncertainties that may affect financial results, or cause limitations to the process of decision-making by distributors;
- risk profile that the distributor wishes to adopt in the contract under analysis.

Thus, there must be a process, based on quantitative methods for the obtainment of the demand contract, by border, that minimizes the expected cost by the risk profile adopted, given the initial conditions and future forecasts (Street et al., 2020).

However, quantitative models have some limitations when applied to assist the decision making process, as they aim to produce robust decisions to the uncertainties that affect the problem, characterized by the set of scenarios, obtained in the Second Stage, designed under a set of hypotheses. Such scenarios aim to map the various possible realizations of the uncertainty variables, in order to characterize the range of costs and their respective probabilities of occurrence, which can be produced by a given hiring decision. Even so, only one scenario, distinct from all of those previously considered, will actually occur. Thus, the objectives and risk metrics to be considered when optimizing the demand contract should be able to reflect risk aversion profiles

that consider the tradeoff between the possibility of occurrence of unfavorable scenarios and the inherent cost in more conservative decisions. These metrics must also be robust, that is, they must be able to produce an acceptable result, even under the occurrence of scenarios not contemplated, as long as they are similar to those used. One of the frameworks that traditionally addresses this tradeoff in optimization models under uncertainty is the *risk-averse stochastic optimization* (Street et al., 2020).

The hiring decision involves informing the National System Operator (ONS, in Portuguese “Operador Nacional do Sistema”), until the month of October of a given year, the demand value that will be contracted for the following year. Once the demand contract has been decided, the import levels and network status are observed over the contract’s months and, depending on the impacts of these uncertainty variables on the performance indicators (which may involve penalty costs, losses, network security, among others), corrective actions can be taken (Street et al., 2020).

Given the previous cost functions, one can proceed to a risk analysis that leads to decision making under uncertainty. Risk can be defined as the *probability that a given cost is greater than expected*. Thus, the more conservative a decision is, the lower the probability of the cost under analysis being higher than expected and the higher the fixed costs (Telles et al., 2018).

According to (Street et al., 2020), two metrics widely used in risk management analysis are: *Expected Value* ( $[.]$ ) and *Conditional Value at Risk* ( $CVaR_\alpha[.]$ ). The Expected Value is a measure that provides the average cost, when applied to a random variable  $\tilde{C} : \Omega \rightarrow \mathbb{R}$ , which represents the cost of a contract, for example. Therefore, this metric comprises all the cost scenarios of the sample space, corresponding to the average of the scenarios, weighted by their probabilities, that is:

$$[C] = \sum_{\omega \in \Omega} \tilde{C}(\omega) \pi_\omega, \quad (5-6)$$

where  $\pi_\omega$  represents the probability of occurrence of each scenario  $\omega$ . Therefore, the Expected Value is a risk-neutral metric, as it considers all scenarios and their respective probabilities of occurrence.

The Expected Value does not offer information about the scenarios that expose the decision maker to the worst costs. On the other hand, the  $CVaR_\alpha[.]$  corresponds to the average of the  $(1 - \alpha) \times 100\%$  worst scenarios, in terms of probability and it can be interpreted as a measure of the worst possible situation, that is, the most costly scenarios. Therefore, the CVaR and the Expected Value correspond to a more and less conservative risk profile, respectively (Street et al., 2020).

Thus, to obtain the optimal contract value  $M_{j,a}$ , for a border bus  $j$ , in a given year  $a$ , a convex combination of  $CVaR_\alpha$  and the Expected Value of the cost  $\tilde{c}_{j,a}^T$  can be applied. For this, a parameter  $\lambda \in [0, 1]$  is used, which reflects the risk aversion desired by the distributor, providing greater flexibility to the decision maker. Hence, the higher the adopted value for  $\lambda$ , the more conservative the distributor's risk aversion profile. This convex combination can be expressed by (Street et al., 2020):

$$\lambda CVaR_\alpha(\tilde{c}_{j,a}^T) + (1 - \lambda)(\tilde{c}_{j,a}^T) \quad (5-7)$$

The optimization of the demand contract will minimize the risk measure of the cost scenarios which, naturally, will avoid situations of overcontracting and, therefore, only penalties for undercontracting will be considered. Thus, an additional risk aversion parameter ( $\mu$ ), that makes it possible to limit the  $CVaR_\alpha$  of the undercontracting costs  $\tilde{c}_{j,m,a}^{UC}$ , can be considered, that is:

$$CVaR_\alpha(\tilde{c}_{j,m,a}^{UC}) \leq \mu c_{j,m,a}^F \quad (5-8)$$

The idea of the  $\mu$  parameter is to make it possible to limit the  $CVaR_\alpha$  of the monthly cost scenarios associated with undercontracting ( $\tilde{c}^{UC}$ ). In other words, it becomes possible to define, on average, a maximum amount to be paid for the most aggressive monthly penalty scenarios. For this limit value to be intuitive and easy to define, the  $\mu$  value is parameterized in relation to the fixed monthly contract cost  $c^F$ . This control is inserted by including the constraint represented by equation (5-8) in the optimization model. Thus, the distributor will be able to define, on average, the maximum monthly amount to be paid for the worst penalty scenarios, with this value parameterized in relation to the fixed cost (Telles et al., 2018).

Given the influence of the decision on the cost associated with each possible scenario, in addition to the impossibility of determining, in advance, which scenario will actually occur, it is extremely important that multiple scenarios, and their corresponding costs, be considered in the optimization problem (Street et al., 2020).

Table 5.2 summarizes the nomenclature and the corresponding definitions for the parameters, sets and variables of the optimization model. Random variables are represented by a tilde ( $\tilde{\cdot}$ ).

Parameters	
Nomenclature	Definition
$\lambda$	CVaR weight in the objective function
$\alpha$	Quantile for the CVaR metric
$\mu$	Fixed cost percentage that limits the CVaR of the penalty costs
$\pi_\omega$	Probability of occurrence for a scenario $\omega \in \Omega$
$T_{j,a}$	Transmission usage tariff at $j \in n, a \in \mathcal{A}$
Sets	
Nomenclature	Definition
$\mathcal{A}$	Set of years in the demand contract horizon
$\mathcal{M}$	Set of months of the year $\mathcal{M} = \{1, 2, \dots, 12\}$
$\Omega$	Set of $\tilde{P}_{j,m,a}$ scenarios
$n$	Set of connection buses between distribution and transmission systems
Variables	
Nomenclature	Definition
$M_{j,a}$	Value of the demand contract at $j \in n, a \in \mathcal{A}$
$P_{j,m,a,\omega}$	Monthly maximum demand at $j \in n, m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega$
$\tilde{P}_{j,a}^{MAX}$	Stochastic annual maximum demand at $j \in n, a \in \mathcal{A}$
$c_{j,m,a}^F$	Fixed cost of the demand contract at $j \in n, m \in \mathcal{M}, a \in \mathcal{A}$
$\tilde{c}_{j,a}^T$	Stochastic annual total cost for the demand contract at $j \in n, a \in \mathcal{A}$
$\tilde{c}_{j,m,a}^{MD}$	Stochastic cost for maximum demand at $j \in n, m \in \mathcal{M}, a \in \mathcal{A}$
$\tilde{c}_{j,a}^{OC}$	Stochastic cost for overcontracting at $j \in n, a \in \mathcal{A}$
$\tilde{c}_{j,m,a}^{UC}$	Stochastic cost for undercontracting at $j \in n, m \in \mathcal{M}, a \in \mathcal{A}$
$z_a$ and $\sigma_{a,\omega}$	Auxiliary variables to represent the <i>CVaR</i> operator in the objective function
$d_{m,a,\omega}^{MD}, d_{m,a,\omega}^{UC}$ and $d_{a,\omega}^{OC}$	Auxiliary variables to represent the <i>max</i> operator of the variable cost in the Optimization problem
$z_{m,a}^{UC}$ and $\sigma_{m,a,\omega}^{UC}$	Auxiliary variables to represent the <i>CVaR</i> operator in the $\mu$ parameter constraints

Table 5.2: Nomenclature adopted in the optimization model.

Next, we present the multi-period stochastic optimization model that defines the value  $M_{j,a}$  of the demand contract, for a horizon  $\mathcal{A}$  and a border bus  $j$ . In this model, each scenario  $\omega \in \Omega$  has a probability of occurrence  $\pi_\omega$  (Telles et al., 2018):

$$\min_{M_{j,a}} \sum_{a \in \mathcal{A}} \left[ \lambda \left( z_a + \frac{1}{1 - \alpha} \sum_{\omega \in \Omega} \pi_\omega \sigma_{a,\omega} \right) + (1 - \lambda) \sum_{\omega \in \Omega} \pi_\omega c_{j,a,\omega}^T \right] \quad (5-9)$$

subject to:

CVaR constraints for the objective function:

$$\sigma_{a,\omega} \geq c_{j,a,\omega}^T - z_a, \quad \forall a \in \mathcal{A}, \omega \in \Omega \quad (5-10)$$

$$\sigma_{a,\omega} \geq 0, \quad \forall a \in \mathcal{A}, \omega \in \Omega \quad (5-11)$$

Constraints on cost definitions:

$$c_{j,a,\omega}^T = c_{j,a,\omega}^{OC} + \sum_{m \in \mathcal{M}} (c_{j,m,a}^F + c_{j,m,a,\omega}^{MD} + c_{j,m,a,\omega}^{UC}), \quad \forall a \in \mathcal{A}, \omega \in \Omega \quad (5-12)$$

$$c_{j,m,a}^F = M_{j,a} T_{j,a}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A} \quad (5-13)$$

$$c_{j,m,a,\omega}^{MD} = d_{m,a,\omega}^{MD} T_{j,a}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-14)$$

$$c_{j,m,a,\omega}^{UC} = d_{m,a,\omega}^{UC} 3T_{j,a}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-15)$$

$$c_{j,a,\omega}^{OC} = d_{a,\omega}^{OC} 12T_{j,a}, \quad \forall a \in \mathcal{A}, \omega \in \Omega \quad (5-16)$$

Maximum Demand Constraints:

$$d_{m,a,\omega}^{MD} \geq P_{j,m,a,\omega} - M_{j,a}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-17)$$

$$d_{m,a,\omega}^{MD} \geq 0, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-18)$$

$$M_{j,a} \geq 0, \quad \forall a \in \mathcal{A} \quad (5-19)$$

Undercontracting constraints:

$$d_{m,a,\omega}^{UC} \geq P_{j,m,a,\omega} - 1.1M_{j,a}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-20)$$

$$d_{m,a,\omega}^{UC} \geq 0, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-21)$$

Overcontracting constraints:

$$d_{a,\omega}^{OC} \geq 0.9M_{j,a} - P_{j,a,\omega}^{MAX}, \quad \forall a \in \mathcal{A}, \omega \in \Omega \quad (5-22)$$

$$d_{a,\omega}^{OC} \geq 0, \quad \forall a \in \mathcal{A}, \omega \in \Omega \quad (5-23)$$

$\mu$  parameter constraints:

$$z_{m,a}^{UC} + \frac{1}{1-\alpha} \sum_{\omega \in \Omega} \pi_{\omega} \sigma_{m,a,\omega}^{UC} \leq \mu M_{j,a} T_{j,a}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A} \quad (5-24)$$

$$\sigma_{m,a,\omega}^{UC} \geq (d_{m,a,\omega}^{UC} 3T_{j,a}) - z_{m,a}^{UC}, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-25)$$

$$\sigma_{m,a,\omega}^{UC} \geq 0, \quad \forall m \in \mathcal{M}, a \in \mathcal{A}, \omega \in \Omega \quad (5-26)$$

In the objective function presented in (5-9), the risk measure defined in (5-7) is minimized, with the demand contract value ( $M_{j,a}$ ) being the decision variable. The CVaR formulation for minimization problems, used in equations (5-9)-(5-11) and (5-24)-(5-26), is presented in (Street, 2010). As mentioned before, constraints (5-12)-(5-16) are the definition of costs; constraints (5-17)-(5-21) represent the use of the *max* operator, as in (5-2) and (5-3), and constraints (5-24)-(5-26) include the risk parameter  $\mu$ , which limits the value of CVaR in penalty cost scenarios. We highlight that variables  $z$  and  $\sigma$  correspond to auxiliary variables used to represent the CVaR operator in the respective constraints.

The optimization model described was implemented to obtain the optimal annual demand contracts for the distribution system's connection points with the transmission network, considering the power flow results, for the scenarios in the test data set, from both Organon and the trained ANN. Different risk aversion profiles were assessed through the variation of the risk aversion parameter  $\mu$ . For the first hiring strategy, we considered a conservative profile in relation to the penalty for undercontracting, by assuming the risk parameter  $\mu = 1\%$ . We also evaluated two additional less conservative hiring strategies, setting  $\mu = 5\%$  and  $\mu = 10\%$ , respectively.

We highlight that the  $\mu$  value cannot vary broadly because the optimization of demand contracts is, essentially, a risk aversion problem; thus, if the value arbitrated for  $\mu$  is too large, constraints (5-24), (5-25) and (5-26) do not influence the optimization result. For all the contracting strategies evaluated, we adopted  $\alpha = 95\%$  for the CVaR metric. We also considered that the Expected Value and the CVaR are equally relevant to the objective function by setting  $\lambda = 0.5$ . Besides that, we considered real values for the tariff  $T_{j,a}$ , for each border, from the distribution system under analysis<sup>5</sup>.

<sup>5</sup>In general, different tariff values are considered according to the period of the day. For simplicity, in this work, we consider a single tariff value for all periods.

We excluded from the demand contracts' optimization study two borders: Border 5 was eliminated because it presents very low values for power flows and is only used in case of contingencies, and Border 12 was excluded from the study because it exported power to the transmission system, instead of importing as it was observed in Chapter 4. Table 5.3 presents the values obtained for the optimal demand contracts, considering the flows calculated by Organon and by the ANN, and the corresponding MAPE, for all hiring strategies.

From Table 5.3, we can observe that the most conservative hiring strategy (Strategy 1) resulted in the lowest average MAPE. Hence, we adopted these results as our optimal demand contracts for this distribution system. In the next section, to conclude the results of the demand contracting study, we will present additional risk and quality indicators for the annual contract, associated with the optimal results for the borders that presented the lowest and highest prediction errors for Strategy 1 (Borders 10 and 6, respectively).

Border	Strategy 1			Strategy 2			Strategy 3		
	Organon	ANN	MAPE	Organon	ANN	MAPE	Organon	ANN	MAPE
Border 1	130.50	129.71	0.61%	128.79	127.36	1.12%	126.90	125.48	1.13%
Border 2	67.13	66.26	1.31%	65.81	64.69	1.73%	64.78	63.64	1.79%
Border 3	188.53	194.28	2.96%	182.83	187.83	2.66%	180.14	184.49	2.36%
Border 4	59.50	58.88	1.05%	58.47	57.58	1.54%	58.13	57.22	1.59%
Border 6	<b>86.41</b>	<b>83.25</b>	<b>3.80%</b>	83.70	80.59	3.86%	81.95	79.13	3.56%
Border 7	46.48	46.10	0.82%	45.22	44.69	1.19%	44.39	43.90	1.12%
Border 8	62.63	61.84	1.28%	<b>60.61</b>	<b>58.25</b>	<b>4.05%</b>	<b>59.71</b>	<b>57.05</b>	<b>4.66%</b>
Border 9	58.72	60.17	2.41%	57.63	58.89	2.14%	57.30	58.51	2.07%
Border 10	<b>115.81</b>	<b>115.33</b>	<b>0.42%</b>	<b>113.06</b>	<b>112.58</b>	<b>0.43%</b>	<b>112.46</b>	<b>112.58</b>	<b>0.11%</b>
Border 11	112.61	110.19	2.20%	110.62	108.19	2.25%	109.00	106.60	2.25%
<b>Average</b>	92.83	92.60	1.69%	90.67	90.07	2.10%	89.48	88.86	2.06%

Table 5.3: Optimal demand contracts, in MW, for each of the three contract strategies considered ( $\mu$  parameter adopted as 1, 5 and 10%, respectively). For each strategy analyzed, we highlight, in bold, the minimum and maximum errors obtained for the contracts.

### 5.3

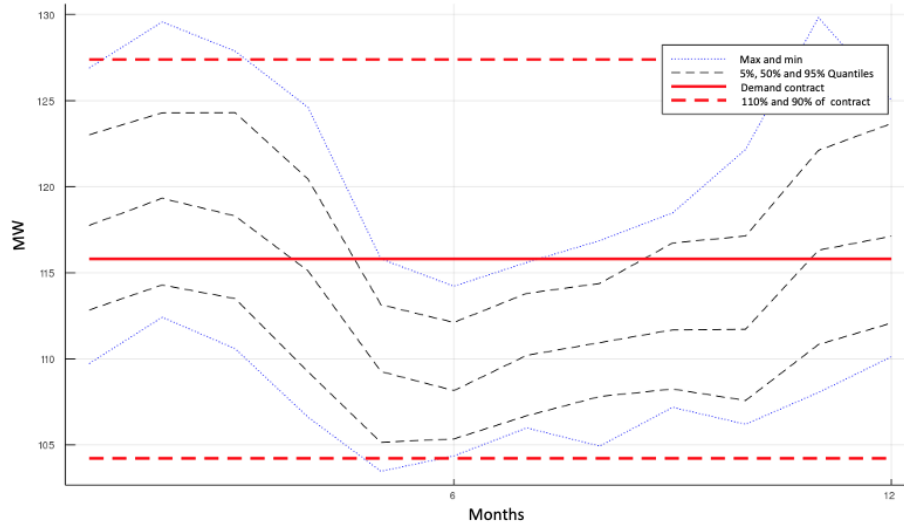
#### Analysis of Results

Once the distributor's risk profile has been defined and the optimal contracting solution has been obtained, we can evaluate this solution's performance within the uncertainty space defined for the problem through strategic risk and quality indicators of the optimal solution. Among these indicators, we highlight the following (Street et al., 2020):

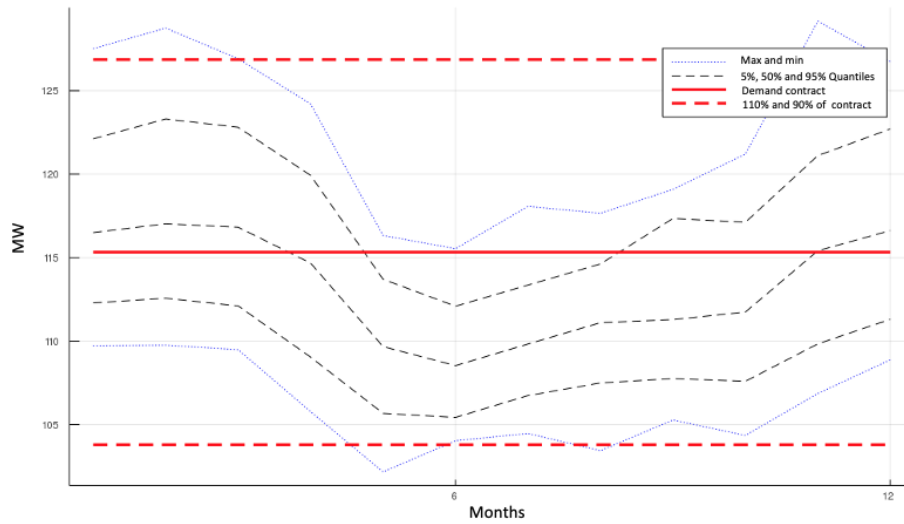
- **Probability of undercontracting:** informs, within the set of scenarios that characterize the uncertainty of the problem, the probability of undercontracting.
- **Probability of overcontracting:** informs, within the set of scenarios that characterize the uncertainty of the problem, the probability of overcontracting.
- **Average of the total cost:** reports the average of the total cost (described by Equation (5-5), including fixed, variable, and penalty costs) of the scenarios for the optimal contract. As this indicator covers all scenarios, it is risk-neutral.
- **CVaR of the total cost:** indicates the  $CVaR_{95\%}$  of the scenarios of the total cost components. Hence, this indicator corresponds to a measure aimed at more aggressive scenarios (higher cost) and, therefore, has a conservative bias.
- **Worst scenario of the total cost:** corresponds to the most conservative measure among all indicators. Informs the highest amount that the distributor can pay within the sample space defined by the scenarios set.

Aiming to evaluate the quality of the ANN's results for Strategy 1, we selected the borders that produced the lowest and the highest prediction errors (Borders 10 and 6, respectively) and compared the results obtained for the optimal contracts and for the indicators described above.

Figures 5.5a and 5.5b presents the optimal demand contract decision for the results obtained by Organon and the ANN for Border 10, respectively. Table 5.4 compares the risk and quality indicators for each of the optimal solutions obtained for Border 10.



(a) Optimal hiring result for Border 10 - Organon.



(b) Optimal hiring result for Border 10 - ANN.

Figure 5.5: Comparison of the optimal contract's results obtained by Organon and by the ANN for Border 10. In the Figures, the blue dotted lines represent the monthly maximum and minimum demand scenarios for the annual horizon; the black dashed lines correspond to the monthly 5%, 50%, and 95% quantiles for the demand scenarios; the solid red line represents the demand contract value for the annual horizon; the red dashed lines represent the 110% and 90% limits for overcontracting and undercontracting, respectively.

From the results presented in Figures 5.5a and 5.5b and Table 5.4, we can observe that none of the approaches presented any overcontracting scenarios. In the flow results obtained by Organon, three scenarios induced penalties for undercontracting while, in the ANN's results, four scenarios produced these penalties. However, for both of the methods, these penalties were provoked by scenarios with peaks in the flows at the beginning and end of the year horizon for the contract. Besides that, all the indicators calculated for the Organon

Indicator	Organon	ANN
Optimal contract (MW)	115.81	115.33
Probability of undercontracting	2.27%	3.00%
Probability of overcontracting	0%	0%
Average of the total cost (thousand R\$)	3708.72	3686.04
CVaR of the total cost (thousand R\$)	3788.38	3786.06
Worst scenario of the total cost (thousand R\$)	3824.14	3859.33

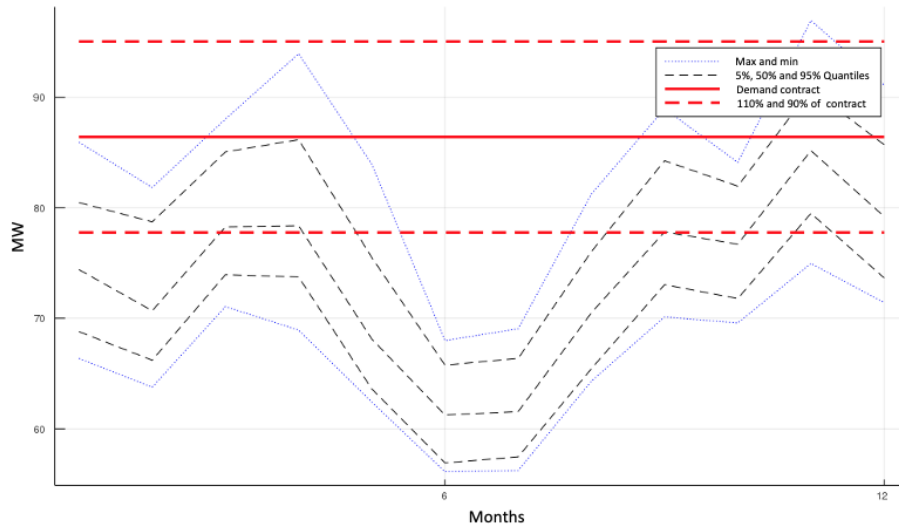
Table 5.4: Comparison of optimal contract parameters - Border 10 (Strategy 1).

and ANN results were fairly close.

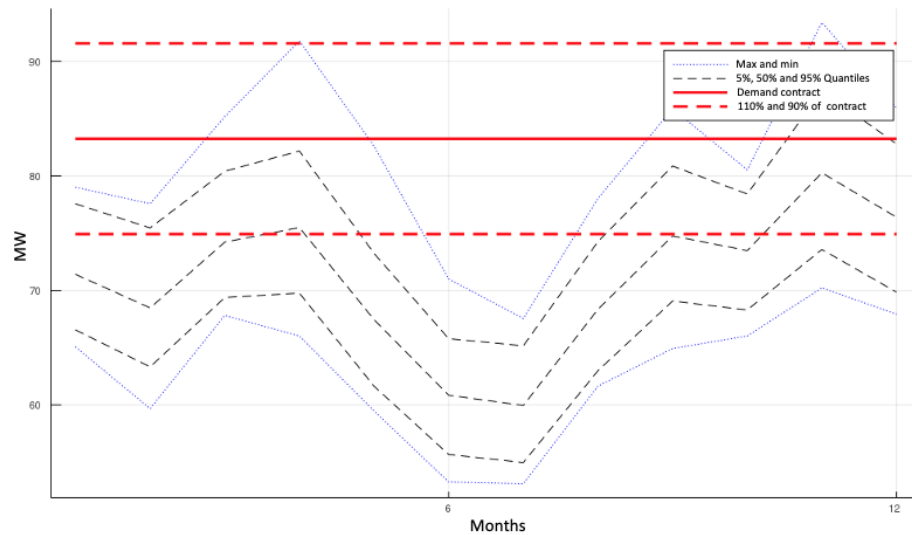
Figures 5.6a and 5.6b and Table 5.5 compare the optimal results for the demand contract obtained for Border 6. Since this border presented the higher prediction errors for the ANN approach, this example is useful for evaluating the impacts of the predictions' errors.

For this case, we can observe that the ANN produced lower results for the optimal contract, as it was also verified in the results from Border 10. However, the ANN's predictions for Border 10 were more accurate than for Border 6. Thus, in this case, a higher increase in the overcontracting and undercontracting probabilities produced by the ANN, compared to the ones obtained by Organon, is expected. Consequently, while the results obtained from Organon do not indicate penalties for overcontracting, the ANN result does. Also, the ANN predictions present two scenarios with the possibility of undercontracting, while Organon presents only one.

However, despite the increase in the probability of penalties in the ANN's result, we can observe from Table 5.5 that the ANN's costs are smaller than Organon's. This is due to the fact that the penalties from the ANN's solution constitute a low cost to the distributor, and the optimal contract obtained by this approach is smaller than the one obtained by Organon. Hence, the optimal contract obtained by the ANN produces lower costs when compared to Organon's contract for Border 6. Nonetheless, it is important to highlight that, although the costs should be minimized, the optimal contract value should always consider the distributor's risk aversion profile. Therefore, low contract values for distributors with a highly conservative profile may incur in great penalty costs.



(a) Optimal hiring result for Border 6 - Organon.



(b) Optimal hiring result for Border 6 - ANN.

Figure 5.6: Comparison of the optimal contract's results obtained by Organon and by the ANN for Border 6. In the Figures, the blue dotted lines represent the monthly maximum and minimum demand scenarios for the annual horizon; the black dashed lines correspond to the monthly 5%, 50%, and 95% quantiles for the demand scenarios; the solid red line represents the demand contract value for the annual horizon; the red dashed lines represent the 110% and 90% limits for overcontracting and undercontracting, respectively.

<b>Metric</b>	<b>Organon</b>	<b>ANN</b>
Optimal contract (MW)	86.41	83.25
Probability of undercontracting	0.76%	1.52%
Probability of overcontracting	0%	8.33%
Average of the total cost (thousand R\$)	2853.03	2751.54
CVaR of the total cost (thousand R\$)	2872.97	2810.30
Worst scenario of the total cost (thousand R\$)	2894.20	2840.99

Table 5.5: Comparison of optimal contract parameters - Border 6 (Strategy 1).

From the results obtained for the demand contracts, we can conclude that the ANN methodology proposed in Chapter 4 constitutes an adequate alternative for obtaining optimal demand contracts for distribution systems. The application presented in this Chapter for the proposed methodology was able to accurately substitute the benchmark software. The great advantage of the proposed approach relies on the Acceleration Ratios described in Table 5.1: for the first contract strategy, for example, once the ANN was appropriately trained, we were able to generate power flow results 15084 times faster than the benchmark method, with an average MAPE of 1.69%. We highlight that all the contracting strategies evaluated presented accurate results for the ANN's predictions; the First Strategy choice was mainly due to the fact that it is the most conservative one.

## 6

## Conclusion

This work has presented a Machine Learning model (implemented in Julia) to calculate active losses and power flows in a distribution system. Solution methodologies presented in the related literature are not always able to model some unique characteristics of these systems and require previous knowledge of topology features for every operating point under analysis, which is difficult to be obtained for distribution networks.

To address these issues, the proposed approach utilizes Artificial Neural Networks (ANNs) to predict the system's losses and line flows. Two case studies were conducted: the first one aimed to calculate active losses in the IEEE 34-bus system, while the second aimed to calculate power flows at the borders between a real distribution system and the Brazilian transmission network.

The implementation of this method occurs in two stages: first is the training phase, in which it is necessary to run power flow calculations, conducted by software programs, such as OpenDSS and Organon. In this stage, prior knowledge of the buses' loads and generations and topology features, such as lines and transformers parameters, are required. Once the calculations are completed, we can start training the ANN by giving, as inputs, the load/generation data and by considering as outputs the power flow results (active losses or line flows). Hence, the ANN obtains the best model to map the given inputs to the respective outputs. The training phase is the most time-consuming stage of the methodology since different ANN architectures need to be evaluated, and the ANN needs to learn the data.

Once the ANN is appropriately trained, we can move to the test phase: in this stage, the model previously obtained is applied to unseen data. Hence, the only previous knowledge required in this phase is the load/generation at each bus of the system (topology features are not required anymore), and the need to run power flow calculations is eliminated. In this phase, only matrix operations are performed, which deeply reduces computational times.

Aiming to effectively demonstrate the applicability of the proposed methodology, we presented the technique's application to the obtainment of optimal demand contracts for a real distribution system in Brazil. The demand hiring process traditionally comprises three stages: simulation of load/generation scenarios for the system's buses, power flow studies, and the contracts' optimization.

The reported numerical experience allows drawing the following main

results and insights:

1. The proposed methodology produced accurate approximations of active losses and line flows, for the case studies presented, with a very significant reduction in their computational times, as demonstrated by the Acceleration Ratios calculated. For the prediction of active losses on the IEEE 34-bus system, this ratio was 391 for the out-of-sample data set. On the other hand, for the study of line flows at the borders between a real Brazilian distribution system and the transmission network, the ratio was even more significant, and the ANN's out-of-sample predictions were obtained 18806 times faster than the benchmark software.
2. The ANN approach can be considered an effective tool to obtain results rapidly when there is no need to run power flow calculations or in situations in which the topology features are unavailable or incomplete. However, we observed that predictions for the out-of-sample data were worse than for the in-sample set since, for unseen samples, the ANN must generalize its learning from the training phase. Hence, the in-sample data set must contain enough information, with good quality, to allow the ANN to generalize to unseen data correctly.
3. For the methodology application in the demand contracting process, since it consists of a decision under uncertainty, there was a need to simulate future scenarios, representing the variability of loads and generations, outside the variation range from the historical data set. Hence, since the ANN is not able to extrapolate, with good accuracy, the solutions for new inputs outside the expected variation range, it was necessary to perform a new training for the ANN instead of directly applying the trained model obtained for the historical data set. Besides that, an additional preprocessing activity was necessary: in order to obtain accurate results for the optimal contracts, we selected a representative group of scenarios to be included in the training phase.
4. Regardless of the risk aversion profile adopted by the distributor, power flow calculations, usually performed according to classical methods, could be replaced by the ANN methodology in the demand hiring process with an appealing tradeoff between accuracy and efficiency: average errors of the order of 2% were obtained for the optimal contract values in the most conservative hiring strategy, and the computational time was reduced by 15084 times.

We believe this work motivates several possible avenues of research. For example, future studies may include: evaluating the effects of a more extensive load/generation scenarios set in the demand hiring process to the optimal contracts' results; an extension of the methodology to consider the possibility of contingencies in the system. By determining the most current contingencies in a distribution system, we would be able to train an ANN for each of these possibilities. This way, during the system's operation, given the occurrence of a contingency, the corresponding ANN would be selected, and the correct approximations for the system's active losses or line flows could be obtained; evaluating the application of the methodology to the transmission system expansion planning.

- M. Aria and C. Cuccurullo, “bibliometrix: An R-tool for comprehensive science mapping analysis,” *Journal of Infometrics*, vol. 11, no. 4, pp. 959–975, 2017.
- C. Rocha and P. Radatz, “Algoritmo de Fluxo de Potência no OpenDSS,” Universidade de São Paulo, Tech. Rep., 9 2018.
- M. Kubat, *An Introduction to Machine Learning*, 2nd ed. Springer, 2017.
- S. Jadon, “Introduction to Different Activation Functions for Deep Learning,” 2018, Available at: <https://medium.com/@shrutijadon10104776/survey-on-activation-functions-for-deep-learning-9689331ba092>.
- M. Stewart, “Neural Network Optimization,” <https://towardsdatascience.com/neural-network-optimization-7ca72d4db3e0>, 6 2019, accessed: 2020-10-15.
- A. F. Panesso-Hernández, J. Mora-Flórez, and S. Pérez-Londoño, “Complete power distribution system representation and state determination for fault location,” *Dyna*, vol. 82, no. 192, pp. 141–149, 8 2015.
- A. Street, E. Telles, G. Bodin, R. Saavedra, C. Fernandes, A. Milhorance, and A. Leite, “Incorporando a Incerteza de Contingências e Variações Climáticas na Contratação Ótima dos Montantes de Uso (MUST/D) em Redes com Forte Inserção de Geração Distribuída Renovável,” *Congresso de Inovação Tecnológica em Energia Elétrica (CITENEL)*, 2020.
- D. Shirmohammadi, H. Hong, A. Semlyen, and G. Luo, “A Compensation-based Power Flow Method for Weakly Meshed Distribution and Transmission Networks,” *IEEE Transactions on Power Systems*, vol. 3, no. 2, pp. 753–762, May 1988.
- C. Cheng and D. Shirmohammadi, “A Three-Phase Power Flow Method for Real-Time Distribution System Analysis,” *IEEE Transactions on Power Systems*, vol. 10, no. 2, pp. 671–679, May 1995.

- I. Kocar, J. Mahseredjian, U. Karaagac, G. Soykan, and O. Saad, "Multiphase Load Flow Solution for Large-Scale Distribution Systems Using MANA," *IEEE Transactions on Power Delivery*, vol. 29, no. 2, pp. 908–915, 2014.
- B. Donnot, I. Guyon, M. Schoenauer, P. Panciatici, and A. Marot, "Introducing Machine Learning for Power System Operation Support," *IREP Symposium*, 2017.
- M. Pertl, K. Heussen, O. Gehrke, and M. Rezkalla, "Voltage Estimation in Active Distribution Grids Using Neural Networks," *Technical Universtiy of Denmark*, 2016.
- F. Fioretto, T. W. Mak, and P. Van Hentenryck, "Predicting AC Optimal Power Flows: Combining Deep Learning and Lagrangian Dual Methods," 2019.
- P. Barker and R. Mello, "Determining the impact of distributed generation on power systems: Part 1 - radial distribution systems," vol. 3, 02 2000, pp. 1645 – 1656 vol. 3.
- C. Hsu, Y. Tzeng, C. Chen, and M. Cho, "Distribution feeder loss analysis by using an artificial neural network," *Electric Power Systems Research*, no. 34, pp. 85–90, 1995.
- E. Telles, R. Saavedra, G. Bodin, T. Silva, A. Milhorance, C. Fernandes, A. Street, and A. Leite, "Risk-Averse Contracting Strategy for the Transmission System Usage," *L Simpósio Brasileiro de Pesquisa Operacional*, 2018.
- M. Fikri, B. Cheddadi, O. Sabri, T. Haidi, B. Abdelaziz, and M. Majdoub, "Power Flow Analysis by Numerical Techniques and Artificial Neural Networks," *3rd Renewable Energies, Power Systems and Green Inclusive Economy*, 2018.
- B. Donnot, "Deep Learning Methods for Predicting Flows in Power Grids: Novel Architectures and Algorithms," *Université Paris-Saclay*, 2019.
- A. Cataliotti, C. Cervellera, V. Cosentino, D. Di Cara, M. Gaggero, D. Macciò, G. Marsala, A. Ragusa, and G. Tinè, "An Improved Load Flow Method for MV Networks Based on LV Load Measurements and Estimations," *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 2, pp. 430–438, 2019.

- A. Leal, J. Jardini, L. Magrini, S. Ahn, H. Schmidt, and R. Casolari, "Distribution System Losses Evaluation by ANN Approach," *2006 IEEE PES Power Systems Conference and Exposition (PSCE)*, pp. 1658–1662, 2006.
- S. Chao, L. Zhensheng, H. Jinlei, K. Zhenxing, and W. Zheng, "Line Loss Calculation in Power Distribution Network Based on Power Measurement Data and BP Neural Network," *International Conference on Power System Technology*, 2017.
- H. Kahef, K. Mahmoud, and M. Abdel-Nasser, "Power Loss Estimation in Smart Grids Using a Neural Network Model," *Aswan University*, 2018.
- O. Ivanov, M. Gavrilas, B. Neagu, and G. Asachi, "Intelligent Monitoring and Control in Transmission and Distribution Networks," *International Conference on Optimization of Electrical and Electronic Equipment*, 2014.
- Y. Yang, Z. Yang, J. Yu, B. Zhang, Y. Zhang, and H. Yu, "Fast Calculation of Probabilistic Power Flow: A Model-Based Deep Learning Approach," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2235–2244, May 2020.
- M. Xiang, J. Yu, Z. Yang, Y. Yang, H. Yu, and H. He, "Probabilistic power flow with topology changes based on deep neural network," *International Journal of Electrical Power Energy Systems*, vol. 117, May 2020.
- M. Xiang, N. Feng, W. Dai, J. Yu, Z. Yang, Y. Yang, H. Yu, and H. Xiang, "Improving the scalability of deep neural network for probabilistic power flow," *International Transactions on Electrical Energy Systems*, vol. 30, no. 5, May 2020.
- M. Chatzos, F. Fioretto, T. W. K. Mak, and P. Van Hentenryck, "High-Fidelity Machine Learning Approximations of Large-Scale Optimal Power Flow," 2020, arXiv: 2006.16356.
- EPRI, "The Open Distribution System Simulator (OpenDSS) - Reference Guide," *Technical report, EPRI - Electric Power Research Institute*, 2016.
- HPPA, "Organon User Guide, Version 9.7," *Technical report, HPPA - High Performance Power System Applications*, 2018.
- J. Grainger and W. Stevenson Jr., *Power System Analysis*, 1st ed. McGraw-Hill, 1994.

- A. Monticelli and A. Garcia, *Introdução a Sistemas de Energia Elétrica*, 2nd ed. UNICAMP, 2015.
- T. Chen, M. Chen, K. Hwang, P. Kotas, and E. Chebli, "Distribution System Power Flow Analysis - A Rigid Approach," *IEEE Transactions on Power Delivery*, vol. 6, no. 3, pp. 1146–1152, 1991.
- W. Kersting, "Radial Distribution Test Feeders," *IEEE Power Engineering Society Winter Meeting*, pp. 908–912, 2001.
- P. Maçaira, A. Thomé, F. Oliveira, and A. Ferrer, "Time series analysis with explanatory variables: A systematic literature review," *Environmental Modelling and Software*, vol. 107, pp. 199–209, 2018.
- M. Baran and F. Wu, "Network Reconfiguration in Distribution Systems for Loss Reduction and Load Balancing," *IEEE Transactions on Power Delivery*, vol. 4, no. 2, pp. 1401–1407, 1989.
- E. Ela, M. Milligan, and B. Kirby, "Operating Reserves and Variable Generation," National Renewable Energy Laboratory (NREL), Tech. Rep., 2011, NREL/TP-550051978.
- CAISO, "2019 Annual Report on Market Issues Performance," California ISO, Tech. Rep., 2019.
- M. El-Hawary, "The smart grid - State-of-the-art and future trends," *Electric Power Components and Systems*, no. 3, 2014.
- G. Ramakrishna and N. Rao, "Adaptive neuro-fuzzy inference system for volt/var control in distribution systems," *Electric Power Systems Research*, no. 49, pp. 87–97, 1999.
- L. Imen, L. Djamel, S. Hassiba, D. Abdellah, and F. Selwa, "Optimal Power Flow Study Using Conventional and Neural Networks Methods," *4th International Conference on Renewable Energy Research and Applications*, pp. 1422–1427, 2015.
- Y. Guo, K. Baker, E. Dall'Anese, Z. Hu, and T. Summers, "Data-Based Distributionally Robust Stochastic Optimal Power Flow - Part II: Case Studies," *IEEE Transactions on Power Systems*, vol. 34, no. 2, pp. 1493–1503, 2019.
- H. M. E. Misilmani and T. Naous, "Machine learning in antenna design: An overview on machine learning concept and algorithms," in *2019 International*

- Conference on High Performance Computing Simulation (HPCS)*, 2019, pp. 600–607.
- K. Matsuno, A. Ecer, J. Periaux, N. Satofuka, and P. Fox, *Parallel Computational Fluid Dynamics*. JAI Press, 2003.
- M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives and prospects,” *Science*, vol. 347, no. 6245, pp. 255–260, 7 2015.
- R. Barret, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vost, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, 1st ed. Society for Industrial and Applied Mathematics, 1994.
- J. Liu, X. Wang, W. Fang, L. Cheng, S. Niu, C. Huo, and J. Wang, “A Novel Load Flow Model for Distribution Systems Based on Current Injections,” *2016 China International Conference on Electricity Distribution*, 8 2016.
- C. S. Corrêa, “Análise do Impacto da Geração Distribuída Fotovoltaica em Sistemas de Distribuição Utilizando Múltiplos Cenários de Geração com Discretização Intra-horária,” 2020, projeto de Graduação - Universidade Federal do Rio de Janeiro.
- I. Barouche, “Análise Comparativa e Proposições de Metodologias de Cálculo de Perdas Técnicas,” Master’s thesis, Universidade Estadual de Campinas, 2017.
- L. Cheng and T. Yu, “A new generation of AI: A review and perspective on machine learning technologies applied to smart energy and electric power systems,” *International Journal of Energy Research*, vol. 43, no. 6, pp. 1928–1973, 2019.
- M. Innes, E. Saba, K. Fischer, D. Gandhi, M. C. Rudilosso, N. M. Joy, T. Karmali, A. Pal, and V. Shah, “Fashionable modelling with flux,” *CoRR*, vol. abs/1811.01457, 2018. [Online]. Available: <https://arxiv.org/abs/1811.01457>
- M. Innes, “Flux: Elegant machine learning with julia,” *Journal of Open Source Software*, 2018.
- C. Aggarwal, *Neural Networks and Deep Learning*, 1st ed. Springer, 2018.
- I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.

- ANEEL, “Perdas de Elétrica na Distribuição,” Agência Nacional de Energia Elétrica, Tech. Rep., 2019.
- K. Zhang, X. Yang, and C. Bu, “Theoretical Analysis on Distribution Network Loss Based on Load Measurement and Countermeasures to Reduce the Loss,” *Proceedings of the CSEE*, vol.33, no.1, pp.92-97, 2013.
- D. Chen and Z. Guo, “Distribution System Theoretical Line Loss Calculation Based on Load Obtaining and Matching Power Flow,” *Power System Technology*, vol.29, no.1, pp.80-84, 2005.
- X. Ding, Y. Luo, and W. Liu, “A New Practical Method for Calculating Line Loss of Distribution Network-Improved Iteration Method,” *Power System Technology*, vol.1, no.34, pp.39-42, 2000.
- ANEEL, “Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional (PRODIST) – Módulo 7: Cálculo de Perdas na Distribuição,” Agência Nacional de Energia Elétrica, Tech. Rep., 2018.
- M. Davoudi, V. Cecchi, and J. R. Agüero, “Effects of stiffness factor on bus voltage variations in the presence of intermittent distributed generation,” *North American Power Symposium (NAPS)*, 2015.
- F. Mancilla-David, A. Angulo, and A. Street, “Power Management in Active Distribution Systems Penetrated by Photovoltaic Inverters: A Data Driven Robust Approach,” *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2271–2280, 2020.
- J. Brownlee, “How to Control the Stability of Training Neural Networks with the Batch Size,” 2019, Available at: <https://machinelearningmastery.com/how-to-control-the-speed-and-stability-of-training-neural-networks-with-gradient-descent-batch-size/>.
- D. Krishnamurthy, “OpenDSSDirect.jl,” 2020, Available at: <https://dss-extensions.org/OpenDSSDirect.jl/latest/>.
- ANEEL, “Resolução Normativa 666/2015 de 29 de junho de 2015,” *Brasil*, 2015.
- G. Bodin, R. Saavedra, A. Street, and C. Fernandes, “Método de geração de cenários de longo prazo para a carga horária,” *XXV Seminário Nacional de Produção e Transmissão de Energia Elétrica (SNPTEE)*, Belo Horizonte, 2019.

- A. Street, "On the Conditional Value-at-Risk Probability-dependent Utility Function," *Theory Dec.*, 68(1-2), 2010.
- H. Gerard, E. I. R. Puente, and D. Six, "Coordination between transmission and distribution system operators in electricity sector: A conceptual framework," *Utilities Policy*, 2017.