https://dx.doi.org/10.22161/ijaers.5.6.25 ISSN: 2349-6495(P) | 2456-1908(O)

Power Flow Calculations by Deterministic Methods and Artificial Intelligence Method

Meriem Fikri[1], Touria Haidi[2], Bouchra Cheddadi[1], Omar Sabri[1], Meriem Majdoub[3], Abdelaziz Belfqih[3]

[1]Laboratory RITM EST, University Hassan II Casablanca, Morocco
Email: meriem.fikri@gmail.com, bcheddadi@hotmail.com, sabo1805@hotmail.com
[2] Laboratory LAGES EHTP Casablanca, Morocco
Email: touriahaidi60@gmail.com
[3]Laboratory of Electric Systems and Energy ENSEM, University Hassan II Casablanca, Morocco

Abstract— *In this paper, we will present different methods for Power Flow Calculations. First, we will describe the deterministic methods; which are Gauss-Seidel (GS) and Newton-Raphson (NR) methods, in addition to that, we will use also a Newton based method Fast Decoupled Load Flow (FDLF). Second, we have the Artificial intelligence method Neural Network (NN). Matlab programs were developed for solving Power Flow problem using GS and NR methods and regarding the ANN, we established and trained artificial neural networks models for computing voltage magnitudes and voltage phase angles. We used these methods to solve the Power Flow problem of the Institute of Electrical and Electronics Engineers (IEEE) 14 bus system. The results that we obtained were presented in graphs at the end of the paper.*
Keywords— ***Artificial Neural Networks, Fast-Decoupled Load Flow, Gauss-Seidel, Newton-Raphson, Power Flow.***

I. INTRODUCTION

Power Flow Analysis is an essential step at any electrical network analysis. Indeed, it allows us to calculate the quantities of a network under a steady state operation, namely the voltage magnitudes and voltage phase angles at any point of the network. From these, it is possible to calculate the currents in the transport lines, the transited active and reactive powers and the power losses caused during the transport of electrical energy. This analysis is very important to study, to plan and to exploit an electrical network.

The equations of Power Flow are non-linear, which require using numerical methods for solving this type of equations. In our paper, we will use Gauss-Seidel (GS) and Newton-Raphson (NR) iterative methods. Then, we have Artificial Intelligence Techniques such as Artificial Neural Networks (ANN), which we will apply to the Power Flow Analysis (PFA).

The following sections of this paper will give a preview of the iterative methods GS and NR and the Artificial Intelligence method ANN, then the solution proposed of PF problem for the Institute of Electrical and Electronics Engineers (IEEE) 14 bus system. At the end of the paper, we will make conclusions and perspectives.

II. DETERMINISTIC METHODS

2.1 Method of Gauss-Seidel

The GAUSS-SEIDEL method is one of the simplest iterative methods used to solve the problem of power flow or generally for the resolution of a very large set of nonlinear algebraic equations. At the beginning, initial values are assumed, with this values we get the first approximation, we continue with iterations until the solution converges and that is happen when the changes between a variable and its previous value is very small:

$$\Delta x_i = x_i^k - x_i^{k-1} < \varepsilon \quad (1)$$

k is the number of iteration and ε is a small value.

To speed up the convergence of the Gauss-Seidel method, we can use an acceleration factor. For example, the accelerated value of voltage at bus i and k+1 iteration is given by:

$$V_i^{(k+1)}(accelerated) = V_i^k + \alpha \Delta V_i \quad (2)$$

Where α is the acceleration factor.

This factor is used to reduce the number of iterations and also to give a precise solution. A suitable value of α for any system can be chosen by making tests on the power flow studies. A generally recommended value is α=1.6. A wrong choice of this factor may affect the convergence of the method [1] [3].

We will apply the method of Gauss-Seidel with acceleration to solve Power Flow problem.

2.1.1 Gauss-Seidel Algorithm

Step 1: Form the admittance matrix Ybus;

Step 2: Choose a tolerance value ε;

https://dx.doi.org/10.22161/ijaers.5.6.25 ISSN: 2349-6495(P) | 2456-1908(O)

Step 3: Assume the unknown values of the system; voltage magnitudes for load buses and voltage phase angles for load and generator buses;
Step 4: Start iteration;
Step 5: Compute reactive power for generator buses, compute bus voltages; for generator buses voltage magnitudes remains the same but voltage phase angles changes;
Step 6: Compute the changes in voltage magnitude value with the use of the acceleration factor ($\Delta V_i = (V_i^{(k+1)} - V_i^k)/\alpha$);
Step 7: Test if ($\Delta V_i < \varepsilon$), then stop the iteration, if not continue the iteration and repeat steps from 5 to 7.
Based on this algorithm we developed a Matlab program to solve PF problem with GSA method.

2.2 Method of Newton-Raphson

NR is the most efficient iterative method currently used for PFA. In this method, convergence is reliable and guaranteed. It's necessary to choose an assumed value near the solution to get the result very quickly if not, the method can take more time to converge [1].
The expressions of active and reactive powers at bus i:

$$P_i = |V_i| \sum_{p=1}^{n} |Y_{ip}||V_p| \cos(\delta_i - \delta_p - \gamma_{ip}) \quad (3)$$

$$Q_i = |V_i| \sum_{p=1}^{n} |Y_{ip}||V_p| \sin(\delta_i - \delta_p - \gamma_{ip}) \quad (4)$$

Pi: The calculated value of net active power entering bus i;
Qi: The calculated value of net reactive power entering bus i;
|Vi|: Voltage magnitude of bus 'i';
Yip: The parameters of line between the two buses i and p;
δi: Voltage phase angle of bus 'i';
δp: Voltage phase angle of bus 'p';
γip: The angle of the parameters of the line between the two buses 'i and p' in polar form.

We have: $\begin{bmatrix}\Delta P\\ \Delta Q\end{bmatrix} = \begin{bmatrix}J_1 & J_2\\ J_3 & J_4\end{bmatrix}\begin{bmatrix}\Delta\delta\\ \Delta V\end{bmatrix}$ (5) with: J=$\begin{bmatrix}J_1 & J_2\\ J_3 & J_4\end{bmatrix}$

ΔP: Active power mismatch;
ΔQ: Reactive power mismatch
Δδ: Voltage phase angle change;
ΔV: Voltage magnitude change;
J: Jacobian matrix.

2.2.1 Newton-Raphson Algorithm

Step 1: Formulate the admittance matrix;
Step 2: Assume the unknown values of the system;
Step 3: Start iteration;
Step 4: Calculate the powers and determine the Mismatch;
Step 5: Formulate the Jacobian matrix;
Step 6: Define differences of unknown values Δδi and Δ|Vi|;
Step 7: Edit previous approximations of δi and |Vi|;
Step 8: Check if (Mismatch < ε); then stop the iteration, if not continue the iteration and repeat steps from 4 to 8.
Based on this algorithm we developed a Matlab program to solve PF problem using NR method.

Due to the complexity of calculations in the NR method, a lot of simplifications was been used, then, as a result we have the Fast Decoupled Load Flow (FDLF) technique.

2.2.2 Method of Fast Decoupled Load Flow

It is a method based on the Newton-Raphson method. It use the decoupling that exists between the active power and the voltage phase angle, and the reactive power and voltage magnitude. This method enable us to fix the value of Jacobian during the iteration in order to avoid costly matrix decompositions. We have four hypotheses:

- The voltage magnitudes of buses are nearly equal to one per unit in a normal steady state operation;
- The susceptance is much bigger than the conductance, because the transmission lines are mostly reactive ($B_{ij} \gg G_{ij}$);
- The voltage phase angle's differences are small in a normal steady state operation;
- The reactive power consumed by the elements connected to this bus is always more than the injected reactive power at any bus when these elements are shorted to the ground ($B_{ii}V_i^2 \gg Q_i$).

With this hypotheses, we get:

$$\frac{\partial P_i}{\partial V_i} \approx 0 \ and \ \frac{\partial P_i}{\partial V_j} \approx 0 \ \Rightarrow J_2 \approx 0$$

And:

$$\frac{\partial Q_i}{\partial \delta_i} \approx 0 \ and \ \frac{\partial Q_i}{\partial \delta_j} \approx 0 \ \Rightarrow J_3 \approx 0$$

Hence,

$$\begin{bmatrix}\Delta P\\ \Delta Q\end{bmatrix} = \begin{bmatrix}J_1 & 0\\ 0 & J_4\end{bmatrix}\begin{bmatrix}\Delta\delta\\ \Delta V\end{bmatrix} \quad (6)$$

And we will rewrite the elements of Jacobian matrix. We need to find partial derivatives:

$$\frac{\partial P_i}{\partial \delta_j} \approx -B_{ii}V_i^2 \ ; \ j = i \quad (7)$$

$$\frac{\partial P_i}{\partial \delta_j} \approx -V_i V_j B_{ij} \ ; \ j \neq i \quad (8)$$

$$\frac{\partial Q_i}{\partial V_j} V_i \approx -B_{ii}V_i^2; \ j = i \quad (9)$$

$$\frac{\partial Q_i}{\partial V_j} \approx -V_i B_{ij} \ ; \ j \neq i \quad (10)$$

Combining equations (6), (7) and (8) we get:

$$\Delta P_i = -V_i \sum_{j=1}^{n} V_j B_{ij} \Delta\delta_j \Rightarrow \frac{\Delta P_i}{V_i} = -\sum_{j=1}^{n} V_j B_{ij} \Delta\delta_j$$

As: $V_i \approx 1.0\, pu$, we get: $\frac{\Delta P_i}{V_i} = -\sum_{j=1}^{n} B_{ij}\Delta\delta_j$

Or, $$\frac{\Delta P}{V} = [-B]\Delta\delta = [B']\Delta\delta \quad (11)$$

Combining equations (6), (9) and (10) we get:

$$\Delta Q_i = -V_i \sum_{j=1}^{n} B_{ij}\Delta V_j \Rightarrow \frac{\Delta Q_i}{V_i} = -\sum_{j=1}^{n} B_{ij}\Delta V_j$$

Or,

$$\frac{\Delta Q}{V} = [B'']\Delta V \quad (12)$$

With:
ΔP: Active power mismatch;

ΔQ: Reactive power mismatch
Δδ: Voltage phase angle change;
ΔV: Voltage magnitude change;
B': Constant matrix for all buses expect the slack bus, it have a dimension of (n-1) x (n-1);
B": Constant matrix for load buses, it have a dimension of (m x m);
n is the number of system's buses and m is the number of load buses.

2.2.3 Algorithm Fast Decoupled Load Flow

Step 1: Form the admittance matrix Ybus and deduce the constant matrix B' et B";
Step 2: Choose a tolerance value ε;
Step 3: Assume the unknown values of the system; voltage magnitudes for load buses and voltage phase angles for load and generator buses;
Step 4: Start iteration;
Step 5: Calculate the powers and determine the Mismatch;
Step 6: Define differences of unknown values Δδi and Δ|Vi|;
Step 7: Edit previous approximations of δi and |Vi|;
Step 8: Check if (Mismatch < ε); then stop the iteration, if not continue the iteration and repeat steps from 5 to 8.
Based on this algorithm we developed a Matlab program to solve PF problem using FDLF method.

III. ARTIFICIAL INTELLIGENCE METHOD

3.1 Artificial Neural Networks

An Artificial Neural Network (ANN) is a computing system based on biological neural networks structure. It is capable of modeling nonlinear problems. It has a lot of advantages, especially, that it can learn from observing data sets. That is why, it can be used as an estimator. For solving problems using ANN, we have to train it by presenting a history of inputs-outputs data. In general, an ANN is composed of three layers; the first one is an input layer, the second one is a hidden layer and the last one is an output layer. However, we can use more than one hidden layer and that is what we will present in this paper [4] [5] [6] [7].

3.2 Artificial Neural Network Models

In our paper we used two models of ANNs, the first one for computing voltage magnitudes for load buses and the second one for computing voltage phase angles for generator and load buses. The first model is composed of one input layer, two hidden layers and one output layer and the second one is composed of one input layer, one hidden layer and one output layer. For the both models, we used as a network type "Feed Forward Backprop", for the training function "trainlm", for the performance function "mse" and regarding the transfer functions we used "tansig" for the hidden layers and "purelin" for the output layer.
Couples of active and reactive power (inputs) were generated and by using the NR method we compute the voltage magnitudes and voltage phase angles (targets), to train our ANNs.
Mean Square Error (MSE) type performance function was used to test the ANNs models.

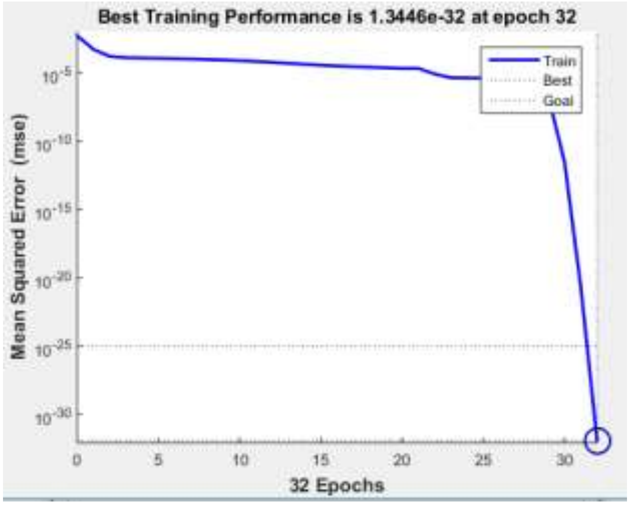

Fig.1: The performance of the ANN for voltage magnitudes.

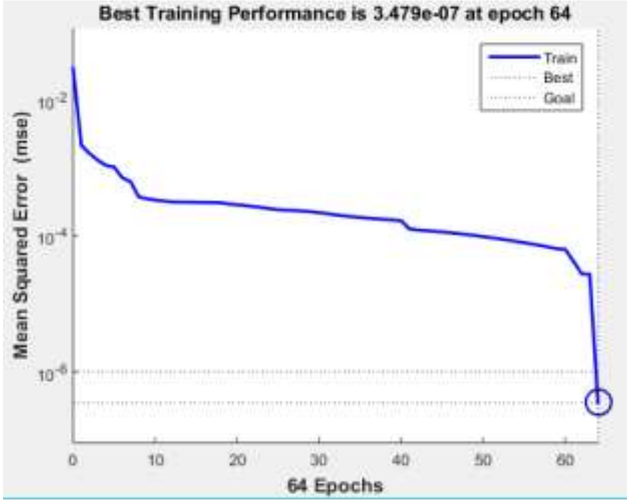

Fig.2: The performance of the ANN for voltage phase angles.

There are graphs of the MSE function representing the performance of the training of the ANN models. According to these graphs, the MSE was decreased at the end of the training phase to reach a value of 1.3446e-32 at epoch 32 for the voltage magnitude and 3.479e-7 at epoch 64 for the voltage phase angles. This means that these neural networks were well trained.

IV. APPLICATION TO IEEE NETWORK

4.1 IEEE 14 bus system

The studies were carried out on the (IEEE) 14 bus system, bus 1 is a Slack bus, buses 2 and 3 are generators buses and the remaining ones are load buses.

https://dx.doi.org/10.22161/ijaers.5.6.25 ISSN: 2349-6495(P) | 2456-1908(O)

Table.1: Values of input data in per unit (pu)

Bus number	Inputs data (power data in pu)	
	P	***Q***
1	2.3294	0.3121
2	0.1830	0.0295
3	0.9420	0.0971
4	0.4780	0.0390
5	0.0760	0.0160
6	0.1120	0.0612
7	0.0000	0.0000
8	0	0.1824
9	0.2950	0.1660
10	0.0900	0.0580
11	0.0350	0.0180
12	0.0610	0.0160
13	0.1350	0.0580
14	0.1490	0.0500

4.2 Results

The following table shows the results of voltage magnitudes by using GS method, NR methods and ANN.

Table.2: Voltage magnitudes in per unit (PU)

Bus number	Numerical Techniques and ANN for V in pu			
	NR	***FDLF***	***GS***	***ANN***
1	1.0600	1.0600	1.0600	-
2	1.0450	1.0450	1.0450	-
3	1.0100	1.0100	1.0100	-
4	1.0423	1.0423	1.0423	1.0438
5	1.0516	1.0516	1.0517	1.0530
6	1.0761	1.0761	1.0760	1.0796
7	1.0505	1.0505	1.0505	1.0549
8	1.0803	1.0803	1.0802	1.0846
9	1.0250	1.0250	1.0250	1.0461
10	1.0264	1.0264	1.0264	1.0320
11	1.0473	1.0473	1.0472	1.0519
12	1.0585	1.0585	1.0584	1.0623
13	1.0511	1.0511	1.0510	1.0551
14	1.0181	1.0181	1.0180	1.0233

The average error between NR and GS from bus 4 to bus 14 is 1.1818e-04;

And between NR and ANN from bus 4 to bus 14 is 5.40e-3.

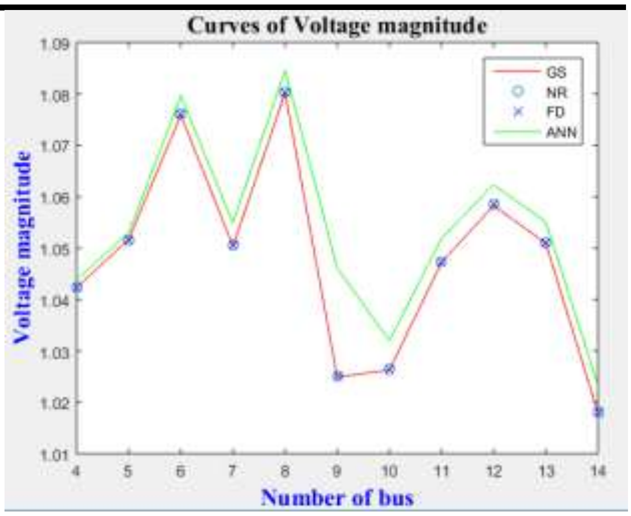

Fig. 3: Curves of voltage magnitudes.

This graph represents the voltage magnitude in each bus. The results obtained by iterative methods are almost mingled, also, the error between them and ANN is not big.

The following table shows the results of voltage phase angles by using GS method, NR methods and ANN.

Bus number	Voltage phase angles in radian (rad)			
	NR	***FDLF***	***GS***	***ANN***
1	0	0	0	-
2	-0.0859	-0.0859	-0.0857	-0.0852
3	-0.2181	-0.2181	-0.2176	-0.2176
4	-0.1836	-0.1836	-0.1831	-0.1855
5	-0.1605	-0.1605	-0.1601	-0.1609
6	-0.2574	-0.2574	-0.2567	-0.2564
7	-0.2334	-0.2334	-0.2326	-0.2340
8	-0.2334	-0.2334	-0.2326	-0.2335
9	-0.2606	-0.2606	-0.2599	-0.2448
10	-0.2649	-0.2649	-0.2641	-0.2646
11	-0.2631	-0.2631	-0.2623	-0.2632
12	-0.2720	-0.2720	-0.2713	-0.2715
13	-0.2722	-0.2722	-0.2715	-0.2712
14	-0.2837	-0.2837	-0.2829	-0.2827

The average error between NR and GS from bus 2 to bus 14 is 6.4615e-04;

And between NR and ANN from bus 2 to bus 14 is 1.8e-3.

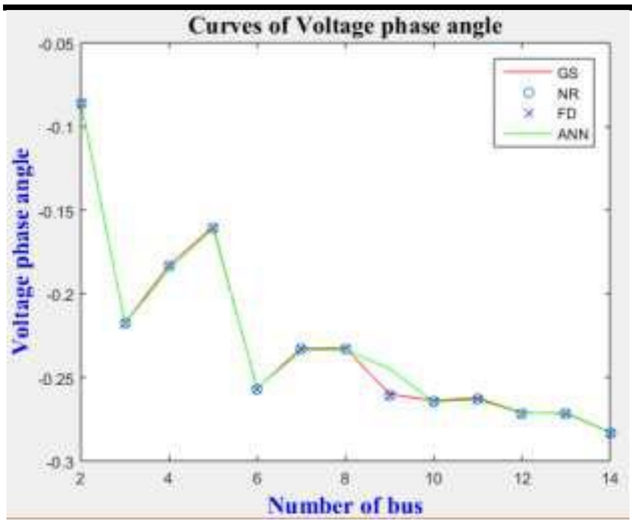

Fig. 4: Curves of voltage phase angles.

This is a graph that represents the voltage phase angles in each bus. We can notice, that the values obtained by the iterative methods and ANN are very close to each other, the error is very small between them.

V. CONCLUSION

Due to the improvement of computer technologies, different methods were developed to solve power flow problem. In this paper we treated numerical methods and artificial intelligence method.

GS's method is simple and easy to program, it is suitable for small systems, but for large systems, NR's methods have proven their robustness in front of the GS method.

The development of the networks and the transition to smart grids lead researchers in this field to depend on the artificial intelligence methods. ANNs has proven its importance in solving power flow problem.

After comparing the results of voltage magnitudes and voltage phase angles, we found that the error between the numerical methods and ANNs was very small, however, the models of the ANNs have to be improved to get better results.

The next step would be to compute the transit currents, to check that they do not exceed the limit values and to evaluate the losses, in addition to that, we can work on the optimization of the power flow path.

REFERENCES

[1] Bokka, N. (17 Dec. 2010). “Comparison of Power Flow Algorithms for Inclusion in On-Line Power Systems Operation Tools”.

[2] Kumar, N. Wangneo, R. Kalra, P.K. Srivastavaon, S.C. “APPLICATION OF ARTIFICIAL NEURAL NETWORKS TO LOAD FLOW SOLUTIONS.” 199-203.

[3] Dr. El-Shatshat, R. (1995). “Load Flow (Gauss-Seidel Method).”

[4] Rao, V. (1995). C++ Neural Networks and Fuzzy Logic. M And T Books.

[5] Paucar, V.leonardo, and Marcos J Rider. (2002). “Artificial Neural Networks for Solving the Power Flow Problem in Electric Power Systems.” Electric Power Systems Research, vol. 62, no. 2, 139-144., doi:10.1016/s0378-7796(02)00030-5.

[6] Draidi, A. (2016). “Répartition Optimale Des Puissances Utilisant Les Techniques De l’intelligence Artificielle.” Université Des Frères Mentouri Constantine Faculté Des Sciences De l’ingénieur Departement d’electrotechnique, 67-74.

[7] Müller Heloisa, H. Rider Marcos, J. Castro Carlos, A. (2010). “Artificial Neural Networks for Load Flow and External Equivalents Studies.” Electric Power Systems Research, vol. 80, no. 9, 1033-1041, doi:10.1016/j.epsr.2010.01.008.

[8] Fikri, M. (2018) “Power Flow Analysis by Numerical Techniques and Artificial Neural Networks”, unpublished.

[9] Fikri, M. (2018) “Gauss-Seidel with Acceleration, Fast-Decoupled and Artificial neural networks for Power Flow Studies”, unpublished.