OpenDSS-X Developer Guide

C++ Developer Guide

NCIM algorithm for OpenDSS-X Version 1.0

**Prepared by**

Davis Montenegro

Electric Power Research Institute, Inc.

1325 G St. NW #530

Washington, DC 20005

*Last update 03-08-2023*

Table of Contents

[1 Purpose 1](#_Toc129353327)

[2 The NCIM method 1](#_Toc129353328)

[3 Numerical example 2](#_Toc129353329)

[3.1 Applying the algorithm to the example test case 7](#_Toc129353330)

[4 References 24](#_Toc129353331)

List of Figures

[Figure 1. Test case proposed 3](#_Toc129353334)

[Figure 2. Jacobian matrix in neutral form, does not include PQ and PV bus coefficients 12](#_Toc129353335)

[Figure 3. Jacobian matrix full for iteration 1 13](#_Toc129353336)

[Figure 4. Jacobian matrix iteration 2 17](#_Toc129353337)

[Figure 5. Jacobian matrix iteration 3 19](#_Toc129353338)

[Figure 6. Jacobian matrix iteration 4 21](#_Toc129353339)

[Figure 7. Jacobian matrix iteration 5 23](#_Toc129353340)

List of Tables

[Table I. technical features of the power conversion elements in the model 3](#_Toc129353341)

[Table II. YBUS matrix for the circuit proposed 4](#_Toc129353342)

[Table III. Results for iteration 2 16](#_Toc129353343)

[Table IV. Results for iteration 3 18](#_Toc129353344)

[Table V. Results for iteration 4 20](#_Toc129353345)

[Table VI. Results for iteration 5 22](#_Toc129353346)

# Purpose

This guide presents the technical details of implementation for the N Nodes Current Injection Method (NCIM) in OpenDSS-X, this algorithm was brought into OpenDSS for helping the simulator deal with transmission system-like simulation models. NCIM is the latest version of the CIM algorithm originally developed by Martins et al [1], which later evolved into a three-phase equivalent (TCIM) [2] to finally fall into the area of multi-phase modeling [3]. This algorithm has been proposed for solving both, transmission and distribution, and given that the calculations are based on current injections similarly to the base power flow solution method in OpenDSS [4], NCIM represented the best alternative for complementing the needs of the simulator when solving transmission simulation models.

NCIM is a Newton-Raphson-based algorithm that models the system using a slack bus with other PQ and PV buses around. The conditions for the PQ buses are represented similarly to OpenDSS’ Floating point iterative (FPI) method, however, PV buses are different since the voltage and active power regulation differs from FPI by introducing the use of a decoupled-extended Jacobian matrix [5]. This method also allows to solve transmission systems using a multiphase model formulation, compatible with OpenDSS providing a solution including zero sequence components if need it. This formulation differs from traditional positive sequence solvers and opens the door for extending the simulation capabilities for transmission systems into unbalanced cases.

# The NCIM method

In OpenDSS the solution algorithm can be set with the property “Algorithm”, the algorithm must be specified before the “Solve” command to take effect. By default, the FPI algorithm is the one active. There are three different algorithms implemented in OpenDSS-X:

1. Normal: (Default) refers to the FPI algorithm.
2. Newton: Refers to the Newton algorithm implemented in OpenDSS, this algorithm is NOT Newton-Raphson since the YBus matrix is used as the Jacobian and is not refreshed based on the voltage fluctuations across the model. Instead, this algorithm reaches convergence after iterating current injections based on a static Y admittance matrix, providing small voltage corrections to the solution vector.
3. NCIM: Refers to the NCIM algorithm described above.

The syntax for setting the solution algorithm is as follows:

Set algorithm = NCIM

The following example code implemented demonstrates the inclusion of the “set algorithm=” command in a model solved with NCIM.

clear

Set DefaultBaseFrequency=60

! IEEE 4-bus test case D-Y Stepdown Balanced

! Based on script developed by Alan Dunn and Steve Sparling

new circuit.4busDYBal basekV=12.47 phases=3

! \*\*\*\* HAVE TO STIFFEN THE SOURCE UP A LITTLE; THE TEST CASE ASSUMES AN INFINITE BUS

~ mvasc3=200000 200000

! \*\*\*\* DEFINE WIRE DATA

new wiredata.conductor Runits=mi Rac=0.306 GMRunits=ft GMRac=0.0244 Radunits=in Diam=0.721

new wiredata.neutral Runits=mi Rac=0.592 GMRunits=ft GMRac=0.00814 Radunits=in Diam=0.563

! \*\*\*\* DEFINE LINE GEOMETRY; REDUCE OUT THE NEUTRAL

new linegeometry.4wire nconds=4 nphases=3 reduce=yes

~ cond=1 wire=conductor units=ft x=-4 h=28

~ cond=2 wire=conductor units=ft x=-1.5 h=28

~ cond=3 wire=conductor units=ft x=3 h=28

~ cond=4 wire=neutral units=ft x=0 h=24

! \*\*\*\* 12.47 KV LINE

new line.line1 geometry=4wire length=2000 units=ft bus1=sourcebus bus2=n2

! \*\*\*\* 3-PHASE STEP-DOWN TRANSFORMER 12.47/4.16 KV Delta-Ygrd

new transformer.t1 xhl=6

~ wdg=1 bus=n2 conn=delta kV=12.47 kVA=6000 %r=0.5

~ wdg=2 bus=n3 conn=wye kV=4.16 kVA=6000 %r=0.5

! \*\*\*\* 4.16 KV LINE

new line.line2 bus1=n3 bus2=n4 geometry=4wire length=2500 units=ft

! \*\*\*\* WYE-CONNECTED 4.16 KV LOAD

new load.load1 phases=3 bus1=n4 conn=wye kV=4.16 kW=5400 pf=0.9 model=1

! \*\*\*\* HAVE TO ALLOW P, Q TO REMAIN CONSTANT TO ABOUT .79 PU -- THIS IS ASSUMED IN TEST CASE

! \*\*\*\* DEFAULT IN DSS IS .95, BELOW WHICH IT REVERTS TO LINEAR MODEL

~ vminpu=0.75 ! model will remain const p,q down to 0.75 pu voltage

set voltagebases=[12.47, 4.16]

calcvoltagebases ! \*\*\*\* let DSS compute voltage bases

set algorithm=NCIM ! Here the algorithm gets activated

solve

…

# Numerical example

One of the most challenging parts of implementing the NCIM algorithm was given the available documentation. The equations formulated for this algorithm present discrepancies in several papers of the same authors. For this, it was necessary to request assistance from the authors themselves, who provided an example that had to be compared against the published papers to correct a couple of numerical mistakes. The purpose of the example presented in this section is to serve as reference for future implementations and bug corrections.

For the example, consider the circuit in Figure 1.

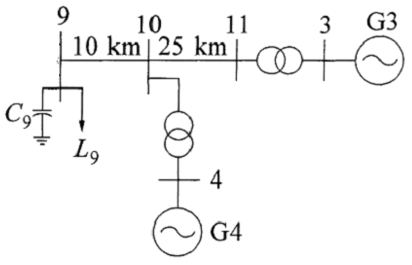


Figure . Test case proposed

The circuit in Figure 1 includes 1 load (PQ bus) and 1 generator (PV bus), the generator connected to bus 3 will be the slack bus in this case. The technical settings for the load and generator are given in Table I.

Table . technical features of the power conversion elements in the model

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Features | | |
| Load L9 | Base voltage | kV | 230 |
| Active power | kW | 1767000 |
| Reactive power | kvar | 100000 |
| Generator G4 | Base voltage | kV | 20 |
| Active power | kW | 700000 |
| Voltage set point | pu | 1.01 |
| Capacitor C9 | Base voltage | kV | 230 |
| Elastance | kvar | 350000 |

For the NCIM formulation the Y admittance matrix (YBUS) only considers the elements connected in series such as lines and transformers. Static shunt devices such as capacitors and reactors can be included in the YBUS. This is probably the main difference with FPI since it doesn’t consider Y primitives for PQ and PV buses. For this case, the YBUS matrix is as presented in Table II.

Table II. YBUS matrix for the circuit proposed

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| B3.1 | 667.23 -4655.92i | 60.89 +19.43i | 60.89 +19.43i | 0 | 0 | 0 | 0 | 0 | 0 | 0 +195.65i | 0 | 0 | 0 | 0 | 0 |
| B3.2 | 60.89 +19.43i | 667.23 -4655.92i | 60.89 +19.43i | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 +195.65i | 0 | 0 | 0 | 0 |
| B3.3 | 60.89 +19.43i | 60.89 +19.43i | 667.23 -4655.92i | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 +195.65i | 0 | 0 | 0 |
| B9.1 | 0 | 0 | 0 | 0.02 -0.18i | 0 -0.00i | 0 -0.00i | -0.02 +0.19i | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B9.2 | 0 | 0 | 0 | 0 -0.00i | 0.02 -0.18i | 0 -0.00i | 0 | -0.02 +0.19i | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B9.3 | 0 | 0 | 0 | 0 -0.00i | 0 -0.00i | 0.02 -0.18i | 0 | 0 | -0.02 +0.19i | 0 | 0 | 0 | 0 | 0 | 0 |
| B10.1 | 0 | 0 | 0 | -0.02 +0.19i | 0 | 0 | 0.03 -17.28i | 0 -0.00i | 0 -0.00i | -0.01 +0.08i | 0 | 0 | 0 +112.96i | 0 -112.96i | 0 |
| B10.2 | 0 | 0 | 0 | 0 | -0.02 +0.19i | 0 | 0 -0.00i | 0.03 -17.28i | 0 -0.00i | 0 | -0.01 +0.08i | 0 | 0 | 0 +112.96i | 0 -112.96i |
| B10.3 | 0 | 0 | 0 | 0 | 0 | -0.02 +0.19i | 0 -0.00i | 0 -0.00i | 0.03 -17.28i | 0 | 0 | -0.01 +0.08i | 0 -112.96i | 0 | 0 +112.96i |
| B11.1 | 0 +195.65i | 0 | 0 | 0 | 0 | 0 | -0.01 +0.08i | 0 | 0 | 0.01 -17.09i | 0 -0.00i | 0 -0.00i | 0 | 0 | 0 |
| B11.2 | 0 | 0 +195.65i | 0 | 0 | 0 | 0 | 0 | -0.01 +0.08i | 0 | 0 -0.00i | 0.01 -17.09i | 0 -0.00i | 0 | 0 | 0 |
| B11.3 | 0 | 0 | 0 +195.65i | 0 | 0 | 0 | 0 | 0 | -0.01 +0.08i | 0 -0.00i | 0 -0.00i | 0.01 -17.09i | 0 | 0 | 0 |
| B4.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 +112.96i | 0 | 0 -112.96i | 0 | 0 | 0 | 0 -1500.00i | 0 +750.00i | 0 +750.00i |
| B4.2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 -112.96i | 0 +112.96i | 0 | 0 | 0 | 0 | 0 +750.00i | 0 -1500.00i | 0 +750.00i |
| B4.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 -112.96i | 0 +112.96i | 0 | 0 | 0 | 0 +750.00i | 0 +750.00i | 0 -1500.00i |

The NCIM algorithm is goes as follows:

1. Calculate the YBUS matrix for the system.
2. Run a flat solution, this is, using:

Calculate the system initial voltages. The vector Inj will only have injecting currents at the slack bus, these currents are provided by the VSource element in OpenDSS. If you are in a different program, the initial voltage can be estimated as a flat start using the voltage bases of the buses. All the voltages are line to neutral. The reactive power for PV buses is equal to 0.

1. With the estimated V, calculate the current mismatches. First estimate the injection currents for the actual solution at the vector V. This solution is calculated as follows:

Reorganize the vector in its decoupled form, setting the imaginary part first and then the real part. As follows:

As a result, the vector ΔF will transform from a vector of complex into a vector of doubles.

Then, add the currents for the PQ and PV buses, for the PQ buses, the current injections will be added as:

For the PV buses the injection current is calculated as:

When including PV buses, the Jacobian matrix will include an extension for voltage and active power regulation as mentioned in [5], ending up in the following representation:

This representation suggests that the vector ΔF will be extended for including the voltage difference for the voltage regulation of the PV buses, which will serve as reference for calculating the reactive power injection/absorption required to reach the set point. Consequently, when calculating the current contribution of the PV bus it is also necessary to calculate the voltage difference ΔV using the following expression:

ΔV need to be added for each PV bus at the end of the vector ΔF aligning with each Z and X coefficient within the Jacobian matrix.

1. Evaluate the convergence criteria using vector ΔF. If the absolute value (|ΔF|) of all the elements in the vector are equal or below the convergence criteria, the solution has been found and go to step 12. Otherwise, go to step 5.
2. Calculate the Jacobian matrix, the elements of the Jacobian are based on the YBUS matrix in its decoupled form as explained in [1, 2], this is, the Jacobian matrix size is twice the size of the YBUS matrix plus the number of PV Buses multiplied by their number of phases. A cell of the Jacobian matrix based on a cell of the YBUS matrix will be represented as follows:

X and Z for the PV buses, is represented as indicated in [5].

1. Add the PQ bus coefficients to the Jacobian matrix. This only affects the diagonal of the Jacobian matrix and is calculated as follows:

The *a* and *b* coefficients are the ones in the Jacobian matrix. Don’t use the ones from the YBUS, since there can be more loads or other elements connected to the same bus, requiring this value from the Jacobian.

1. Add the PV bus coefficients to the Jacobian matrix. This only affects the diagonal of the Jacobian matrix and is calculated as follows:

The *a* and *b* coefficients are the ones in the Jacobian matrix. Don’t use the ones from the YBUS, since there can be more loads or other elements connected to the same bus, requiring this value from the Jacobian.

1. Solve the Jacobian matrix as follows:
2. Add ΔZ to the voltage vector V, for that, transform vector ΔZ into its complex equivalent:
3. Update the reactive power for PV buses using the values obtained in ΔQ.
4. Go back to step 3.
5. Power flow solution found.

## Applying the algorithm to the example test case

1. Calculate the system YBUS matrix (Done – page 1).
2. Run a flat solution for finding the initial voltage values. In OpenDSS, the slack bus represented as a Voltage source (VSource), the current injection proposed by the program for this case is as follows:

With these currents exiting the YBUS matrix, the initial voltage guess is:

|  |  |  |
| --- | --- | --- |
|  | Value | Node |
| V = | 11814.23 -1410.01i | B3.1 |
| -7128.23 -9526.41i | B3.2 |
| -4685.99 +10936.44i | B3.3 |
| 154753.03 -20681.56i | B9.1 |
| -95287.40 -123679.29i | B9.2 |
| -59465.63 +144360.85i | B9.3 |
| 149405.80 -19415.30i | B10.1 |
| -91517.16 -119681.59i | B10.2 |
| -57888.64 +139096.88i | B10.3 |
| 135924.87 -16223.25i | B11.1 |
| -82012.33 -109602.62i | B11.2 |
| -53912.45 +125826.13i | B11.3 |
| 10407.09 -7958.01i | B4.1 |
| -12095.39 -5033.81i | B4.2 |
| 1688.30 +12991.81i | B4.3 |

PV bus powers initialized as:

|  |  |
| --- | --- |
| PV bus power phase A | 2.333E+8 +0 i |
| PV bus power phase B | 2.333E+8 +0 i |
| PV bus power phase C | 2.333E+8 +0 i |

1. Calculate ΔF.

|  |  |
| --- | --- |
| ΔF = | 3745260.00 -29496700.00i |
| -27417500.00 +11504900.00i |
| 23672300.00 +17991800.00i |
| -0.00 +0.00i |
| 0.00 -0.00i |
| -0.00 -0.00i |
| 0.00 +0.00i |
| -0.00 +0.00i |
| 0.00 -0.00i |
| 0.00 +0.00i |
| 0.00 -0.00i |
| -0.00 -0.00i |
| -0.00 +0.00i |
| 0 +0.00i |
| 0 |

Calculate injected currents, first PQ buses (load) the current injections are:

Current injections for PV buses (generator):

Calculate ΔV for PV buses:

Add the injection currents from PQ and PV buses to ΔF. Make the first 6 cells of ΔF in decoupled form equal to 0 since they represent the slack bus.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ΔF = | ΔF(0) | + | Inj Currents | = |  |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | -711.344 | -711.344 |
| -1.14E-13 | 3711 | 3711 |
| -3.64E-12 | -2858.14 | -2858.14 |
| 3.64E-12 | -2471.54 | -2471.54 |
| -1.82E-12 | 3569.49 | 3569.49 |
| -3.64E-12 | -1239.46 | -1239.46 |
| 2.33E-10 | 0 | 2.33E-10 |
| 2.33E-10 | 0 | 2.33E-10 |
| 3.78E-10 | 0 | 3.78E-10 |
| -6.98E-10 | 0 | -7E-10 |
| -2.91E-11 | 0 | -2.9E-11 |
| 2.33E-10 | 0 | 2.33E-10 |
| 1.96E-10 | 0 | 1.96E-10 |
| 5.53E-12 | 0 | 5.53E-12 |
| -5.27E-10 | 0 | -5.3E-10 |
| 2.77E-10 | 0 | 2.77E-10 |
| -1.16E-10 | 0 | -1.2E-10 |
| -4.66E-10 | 0 | -4.7E-10 |
| 2.56E-09 | 10818.6 | 10818.6 |
| -1.86E-09 | -14148 | -14148 |
| 6.98E-10 | 6843.23 | 6843.23 |
| 0 | 16443.1 | 16443.1 |
| 0 | -17661.8 | -17661.8 |
| 0 | -2295.17 | -2295.17 |
| 0 | -1439.04 | -1439.04 |
| 0 | -1439.05 | -1439.05 |
| 0 | -1439.05 | -1439.05 |

1. Evaluate convergence. The convergence value for this exercise has been set to 1e-6. At this point, the maximum absolute value in ΔF is significantly larger than the convergence value, move to step 5.
2. Calculate the Jacobian matrix, the matrix is shown in Figure 2.
3. Add the PQ bus coefficients to the Jacobian matrix. The PQ bus is localized at bus B9, which within the Jacobian comprehends cells 6 to 11, this calculation only affects the diagonal therefore, only cells {6,6}, {6,7}, {7,6} and {7,7} will be altered for phase 1, {8,8}, {8,9}, {9,8} and {9,9} for phase 2 and so on covering the number of phases of the element connected at the PQ bus. For the sake of the example only the calculations for phase 1 will be presented in detail.
4. Add the PV bus coefficients to the Jacobian matrix. This step also includes the calculation of the voltage and active power regulation coefficients.

As in step 6, in this example only the calculations for phase 1 are presented in detail. The cells within the Jacobian matrix for the PV bus in this model go from cells 24 to 29. For phase 1, the coefficients are as follows:

The coefficients for voltage regulation (Z) for phase 1 will affect cells {30, 24} and {30, 25}, for phase 2 {31, 26} and {31, 27} and for phase 3 cells {32, 28} and {32, 29}. The values for phase 1 are calculated as follows:

The coefficients for active power regulation (X) for phase 1 will affect cells {24,30} and {25,30}, phase 2 cells {26, 31} and {27, 31} and for phase 3 cells {28, 32} and {29, 32}. The values for phase 1 are:

The resulting Jacobian matrix is shown at.

Table

Description automatically generated

Figure 2. Jacobian matrix in neutral form, does not include PQ and PV bus coefficients

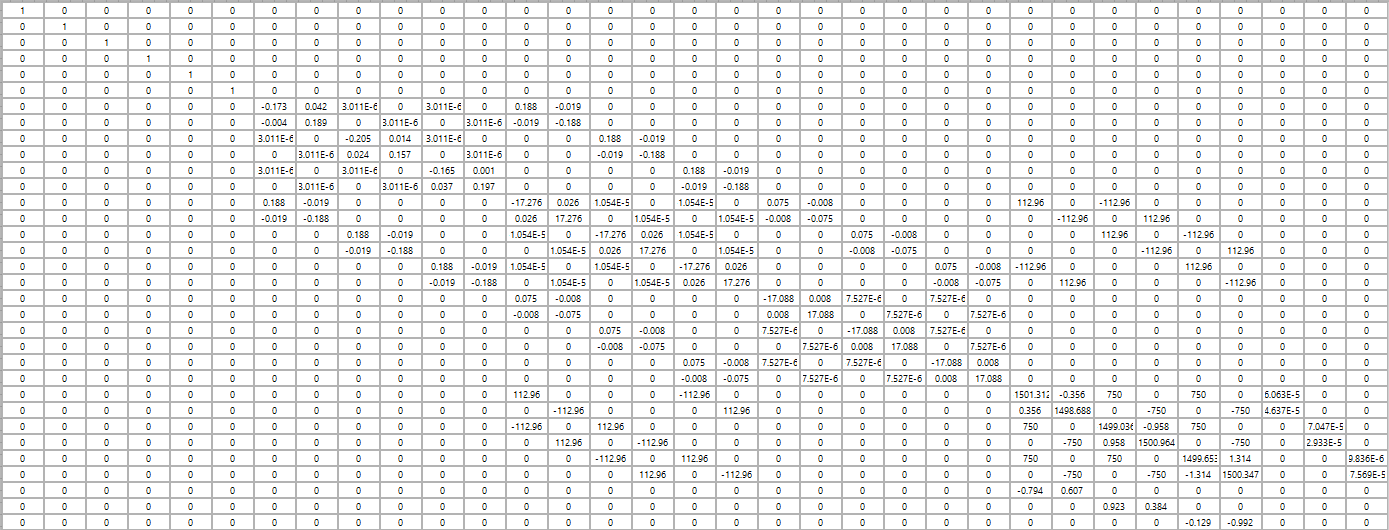


Figure 3. Jacobian matrix full for iteration 1

1. Solve the Jacobian matrix.

|  |  |
| --- | --- |
| ΔZ | 0 |
| 0 |
| 0 |
| 0 |
| 0 |
| 0 |
| 37876.3 |
| 59895.3 |
| 32932.7 |
| -62749.4 |
| -70809 |
| 2854.21 |
| 21615.5 |
| 37424.1 |
| 21602.5 |
| -37431.6 |
| -43218 |
| 7.55253 |
| 78.5291 |
| 173.724 |
| 111.185 |
| -154.87 |
| -189.714 |
| -18.8538 |
| 3245.2 |
| 1874.85 |
| 1.06149 |
| -3747.85 |
| -3246.28 |
| 1873 |
| ΔQ | -1.01E+08 |
| -1.01E+08 |
| -1.01E+08 |

1. Add ΔZ to the voltage vector V, for that, transform vector ΔZ into its complex equivalent:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| V = | 11814.23 -1410.01i | - | 0 | = | 11814.23 -1410.01i | B3.1 |
| -7128.23 -9526.41i | 0 | -7128.23 -9526.41i | B3.2 |
| -4685.99 +10936.44i | 0 | -4685.99 +10936.44i | B3.3 |
| 154753.03 -20681.56i | 37876.28 +59895.26i | 116876.76 -80576.81i | B9.1 |
| -95287.40 -123679.29i | 32932.68 -62749.43i | -128220.07 -60929.86i | B9.2 |
| -59465.63 +144360.85i | -70808.98 +2854.21i | 11343.35 +141506.63i | B9.3 |
| 149405.80 -19415.30i | 21615.53 +37424.08i | 127790.28 -56839.37i | B10.1 |
| -91517.16 -119681.59i | 21602.46 -37431.63i | -113119.62 -82249.96i | B10.2 |
| -57888.64 +139096.88i | -43217.99 +7.55i | -14670.65 +139089.33i | B10.3 |
| 135924.87 -16223.25i | 78.53 +173.72i | 135846.34 -16396.98i | B11.1 |
| -82012.33 -109602.62i | 111.18 -154.87i | -82123.51 -109447.75i | B11.2 |
| -53912.45 +125826.13i | -189.71 -18.85i | -53722.74 +125844.99i | B11.3 |
| 10407.09 -7958.01i | 3245.20 +1874.85i | 7161.89 -9832.86i | B4.1 |
| -12095.39 -5033.81i | 1.06 -3747.85i | -12096.45 -1285.96i | B4.2 |
| 1688.30 +12991.81i | -3246.28 +1873.00i | 4934.57 +11118.80i | B4.3 |

1. Update the reactive power for PV buses using the values obtained in ΔQ.

New powers for generators:

|  |  |
| --- | --- |
| PV bus power phase A | 2.333E+8 +1.01164E+8i |
| PV bus power phase B | 2.333E+8 +1.01164E+8i |
| PV bus power phase C | 2.333E+8 +1.01164E+8i |

Go back to step 3.

From this point only vector ΔF, ΔZ and V for each iteration will be presented, same for Jacobian matrices. The system converges after 5 iterations. The results presented to far apply for iteration number 1. The vector V calculated at this iteration will be the starting point for iteration 2 and so on.

Table . Results for iteration 2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ΔF |  |  | ΔZ |  | Resulting V |
| Inj | 0 |  | ΔZ | 0 |  | 11814.23 -1410.01i |
| 0 |  | 0 |  | -7128.23 -9526.41i |
| 0 |  | 0 |  | -4685.99 +10936.44i |
| 0 |  | 0 |  | 112156.95 -71357.49i |
| 0 |  | 0 |  | -117876.03 -61452.06i |
| 0 |  | 0 |  | 5719.09 +132809.50i |
| -171.721 |  | 4719.8 |  | 124089.82 -50926.44i |
| -838.766 |  | -9219.32 |  | -106148.66 -82001.73i |
| 812.251 |  | -10344 |  | -17941.16 +132928.17i |
| 270.669 |  | 522.195 |  | 135827.51 -16372.64i |
| -640.534 |  | 5624.26 |  | -82093.02 -109443.61i |
| 568.097 |  | 8697.14 |  | -53734.40 +125816.51i |
| 4.66E-10 |  | 3700.46 |  | 7139.53 -9227.34i |
| 1.75E-10 |  | -5912.93 |  | -11560.88 -1569.35i |
| -2.33E-10 |  | -6970.96 |  | 4421.36 +10796.68i |
| 2.33E-10 |  | -248.229 |  |  |
| -1.16E-10 |  | 3270.51 |  |  |
| -6.98E-10 |  | 6161.16 |  |  |
| -8.34E-11 |  | 18.8277 |  |  |
| -6.94E-12 |  | -24.3384 |  |  |
| 7.82E-11 |  | -30.4914 |  |  |
| 1.28E-10 |  | -4.13606 |  |  |
| 2.33E-10 |  | 11.6638 |  |  |
| 4.66E-10 |  | 28.4745 |  |  |
| -1477.24 |  | 22.3609 |  |  |
| 3583.07 |  | -605.513 |  |  |
| -2364.43 |  | -535.57 |  |  |
| -3070.85 |  | 283.395 |  |  |
| 3841.66 |  | 513.211 |  |  |
| -512.21 |  | 322.124 |  |  |
| ΔV | -502.611 |  | ΔQ | 2.03E+07 |  |  |
| -502.61 |  | 2.03E+07 |  |  |
| -502.614 |  | 2.03E+07 |  |  |

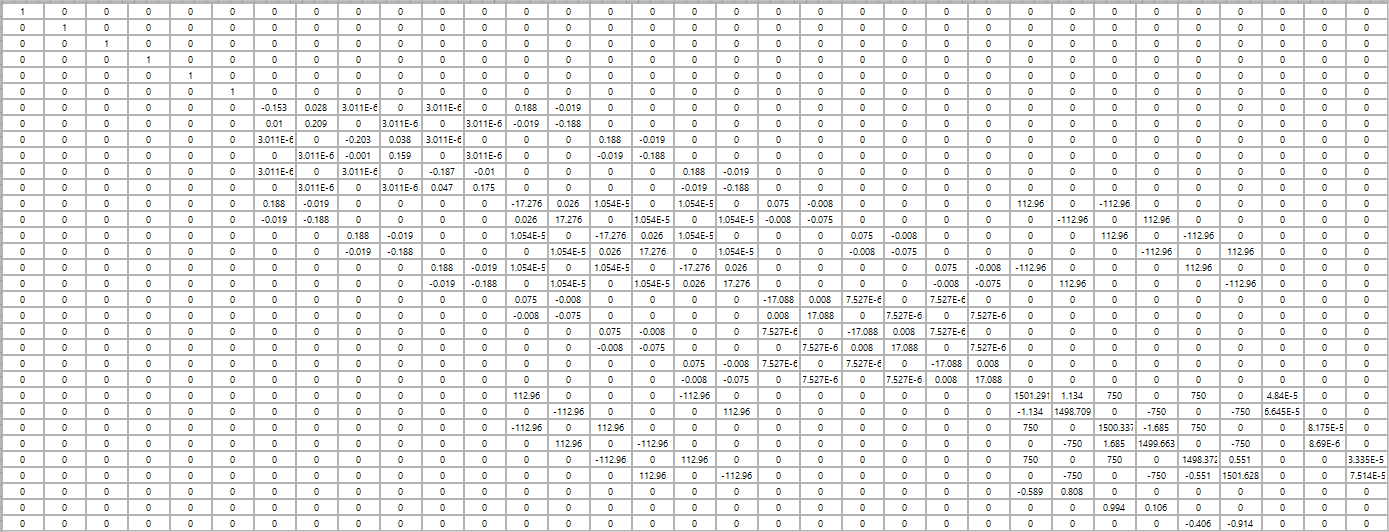


Figure 4. Jacobian matrix iteration 2

Table IV. Results for iteration 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ΔF |  |  | ΔZ |  | Resulting V |
| Inj | 0 |  | ΔZ | 0 |  | 11814.23 -1410.01i |
| 0 |  | 0 |  | -7128.23 -9526.41i |
| 0 |  | 0 |  | -4685.99 +10936.44i |
| 0 |  | 0 |  | 112007.10 -71635.40i |
| 0 |  | 0 |  | -118041.78 -61183.33i |
| 0 |  | 0 |  | 6034.69 +132818.68i |
| 8.43064 |  | 149.856 |  | 123952.49 -51114.10i |
| 22.0672 |  | 277.912 |  | -106242.52 -81788.97i |
| -23.3259 |  | 165.748 |  | -17709.98 +132903.07i |
| -3.7324 |  | -268.734 |  | 135826.99 -16373.52i |
| 14.8955 |  | -315.605 |  | -82093.53 -109442.72i |
| -18.3347 |  | -9.17761 |  | -53733.38 +125816.50i |
| 0 |  | 137.325 |  | 7121.00 -9235.48i |
| -2.33E-10 |  | 187.664 |  | -11558.65 -1549.24i |
| 0 |  | 93.8575 |  | 4437.67 +10784.70i |
| 2.33E-10 |  | -212.759 |  |  |
| 5.82E-11 |  | -231.182 |  |  |
| -2.33E-10 |  | 25.0941 |  |  |
| -1.42E-10 |  | 0.520791 |  |  |
| 9.50E-12 |  | 0.883789 |  |  |
| -1.42E-10 |  | 0.504978 |  |  |
| -7.29E-11 |  | -0.89291 |  |  |
| -2.33E-10 |  | -1.02577 |  |  |
| 0 |  | 0.009123 |  |  |
| -74.8587 |  | 18.5315 |  |  |
| -80.7945 |  | 8.13129 |  |  |
| 107.4 |  | -2.22394 |  |  |
| -24.4335 |  | -20.1145 |  |  |
| -32.5415 |  | -16.3075 |  |  |
| 105.227 |  | 11.9831 |  |  |
| ΔV | -4.90928 |  | ΔQ | 2.73E+05 |  |  |
| -4.9094 |  | 2.73E+05 |  |  |
| -4.90927 |  | 2.73E+05 |  |  |

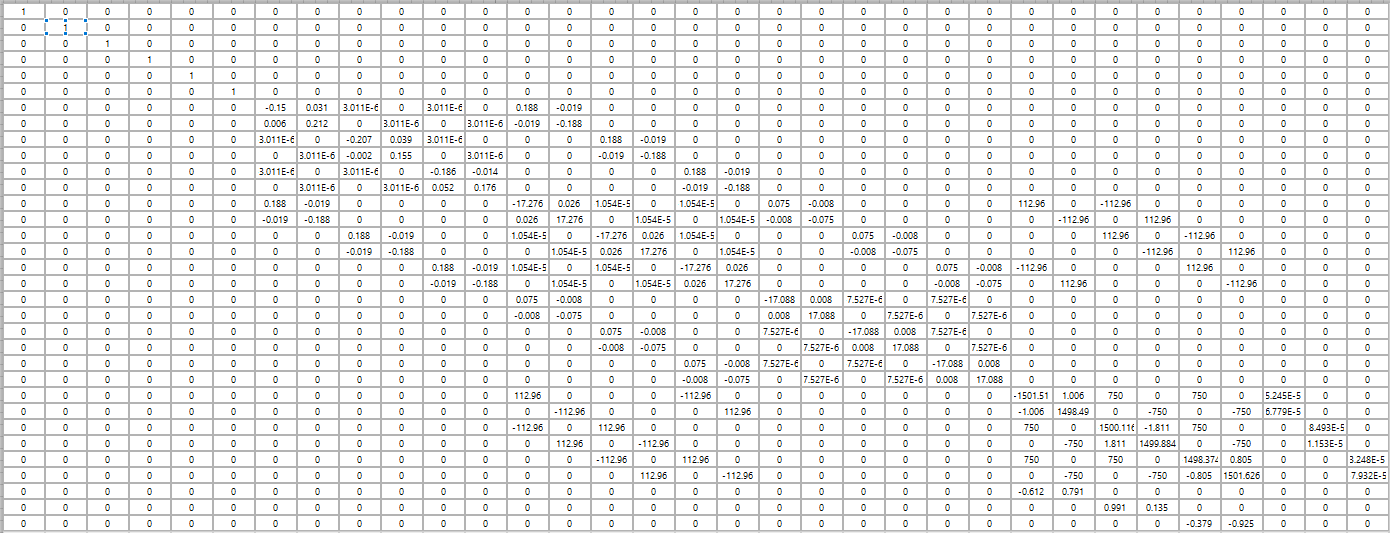


Figure 5. Jacobian matrix iteration 3

Table V. Results for iteration 4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ΔF |  |  | ΔZ |  | Resulting V |
| Inj | 0 |  | ΔZ | 0 |  | 11814.23 -1410.01i |
| 0 |  | 0 |  | -7128.23 -9526.41i |
| 0 |  | 0 |  | -4685.99 +10936.44i |
| 0 |  | 0 |  | 112007.17 -71635.03i |
| 0 |  | 0 |  | -118041.49 -61183.58i |
| 0 |  | 0 |  | 6034.33 +132818.55i |
| 0.017421 |  | -0.07115 |  | 123952.43 -51113.76i |
| -0.01798 |  | -0.3755 |  | -106242.19 -81789.08i |
| 0.006857 |  | -0.28963 |  | -17710.24 +132902.85i |
| 0.024074 |  | 0.249364 |  | 135826.99 -16373.52i |
| -0.02428 |  | 0.360771 |  | -82093.53 -109442.72i |
| -0.0061 |  | 0.126133 |  | -53733.38 +125816.50i |
| -6.98E-10 |  | 0.064413 |  | 7121.01 -9235.45i |
| 1.16E-10 |  | -0.33742 |  | -11558.63 -1549.26i |
| -5.82E-11 |  | -0.32442 |  | 4437.64 +10784.69i |
| 0 |  | 0.112926 |  |  |
| 5.82E-11 |  | 0.26001 |  |  |
| -2.33E-10 |  | 0.224491 |  |  |
| 3.72E-10 |  | 0.00043 |  |  |
| -4.05E-11 |  | -0.00145 |  |  |
| 4.68E-10 |  | -0.00147 |  |  |
| -5.77E-11 |  | 0.000354 |  |  |
| 1.16E-10 |  | 0.001043 |  |  |
| 0 |  | 0.001099 |  |  |
| -0.01863 |  | -0.00972 |  |  |
| 0.014476 |  | -0.02836 |  |  |
| -0.00322 |  | -0.0197 |  |  |
| -0.02337 |  | 0.022594 |  |  |
| 0.021851 |  | 0.029418 |  |  |
| 0.008898 |  | 0.005764 |  |  |
| ΔV | -0.01653 |  | ΔQ | 4.35E+03 |  |  |
| -0.01653 |  | 4.35E+03 |  |  |
| -0.01652 |  | 4.35E+03 |  |  |

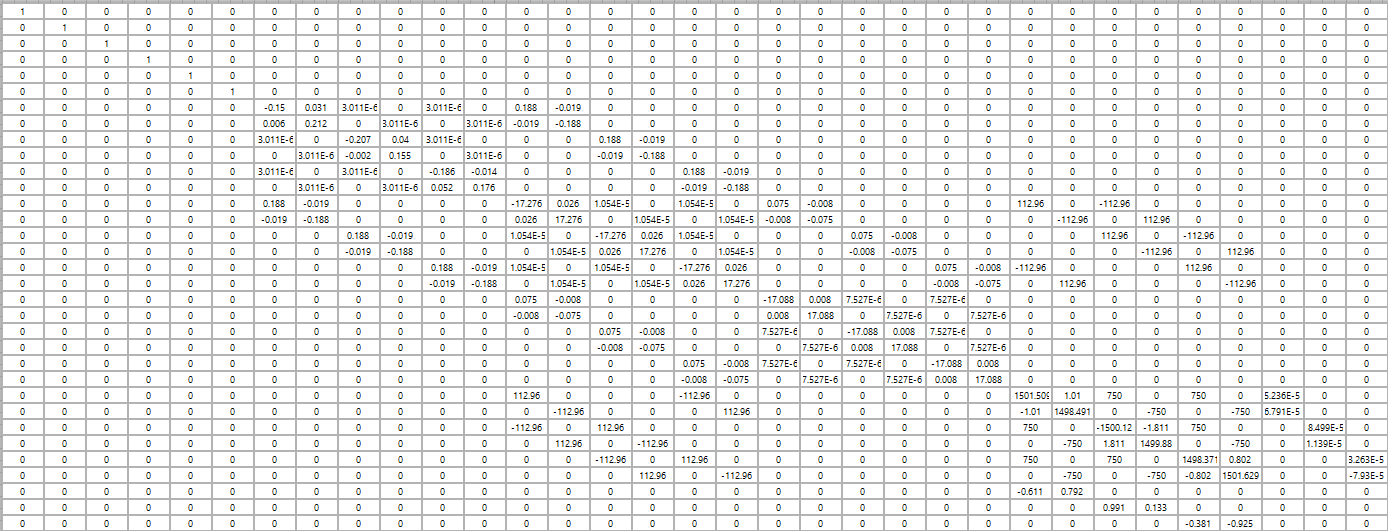


Figure 6. Jacobian matrix iteration 4

Table VI. Results for iteration 5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ΔF |  |  | ΔZ |  | Resulting V |
| Inj | 0 |  | ΔZ | 0 |  | 11814.23 -1410.01i |
| 0 |  | 0 |  | -7128.23 -9526.41i |
| 0 |  | 0 |  | -4685.99 +10936.44i |
| 0 |  | 0 |  | 112007.17 -71635.03i |
| 0 |  | 0 |  | -118041.49 -61183.58i |
| 0 |  | 0 |  | 6034.33 +132818.55i |
| 3.61E-08 |  | 4.65E-07 |  | 123952.43 -51113.76i |
| -6.63E-09 |  | 4.58E-07 |  | -106242.19 -81789.08i |
| -1.23E-08 |  | 1.66E-07 |  | -17710.24 +132902.85i |
| 3.45E-08 |  | -6.41E-07 |  | 135826.99 -16373.52i |
| -2.38E-08 |  | -6.34E-07 |  | -82093.53 -109442.72i |
| -2.79E-08 |  | 1.84E-07 |  | -53733.38 +125816.50i |
| 4.66E-10 |  | 5.40E-07 |  | 7121.01 -9235.45i |
| -5.82E-11 |  | 5.15E-07 |  | -11558.63 -1549.26i |
| 5.82E-11 |  | 1.79E-07 |  | 4437.64 +10784.69i |
| 2.33E-10 |  | -7.33E-07 |  |  |
| 5.82E-11 |  | -7.20E-07 |  |  |
| 0 |  | 2.18E-07 |  |  |
| 3.94E-10 |  | 2.12E-09 |  |  |
| 2.44E-11 |  | 2.50E-09 |  |  |
| -2.95E-10 |  | 1.13E-09 |  |  |
| 5.18E-11 |  | -3.14E-09 |  |  |
| 0 |  | -3.25E-09 |  |  |
| 0 |  | 6.43E-10 |  |  |
| -1.05E-06 |  | 6.32E-08 |  |  |
| -3.49E-08 |  | 1.49E-08 |  |  |
| 5.55E-07 |  | -1.86E-08 |  |  |
| -8.92E-07 |  | -6.32E-08 |  |  |
| 4.96E-07 |  | -4.58E-08 |  |  |
| 9.28E-07 |  | 4.79E-08 |  |  |
| ΔV | -2.68E-08 |  | ΔQ | -0.00304 |  |  |
| -2.68E-08 |  | -0.00308 |  |  |
| -2.68E-08 |  | -0.00302 |  |  |

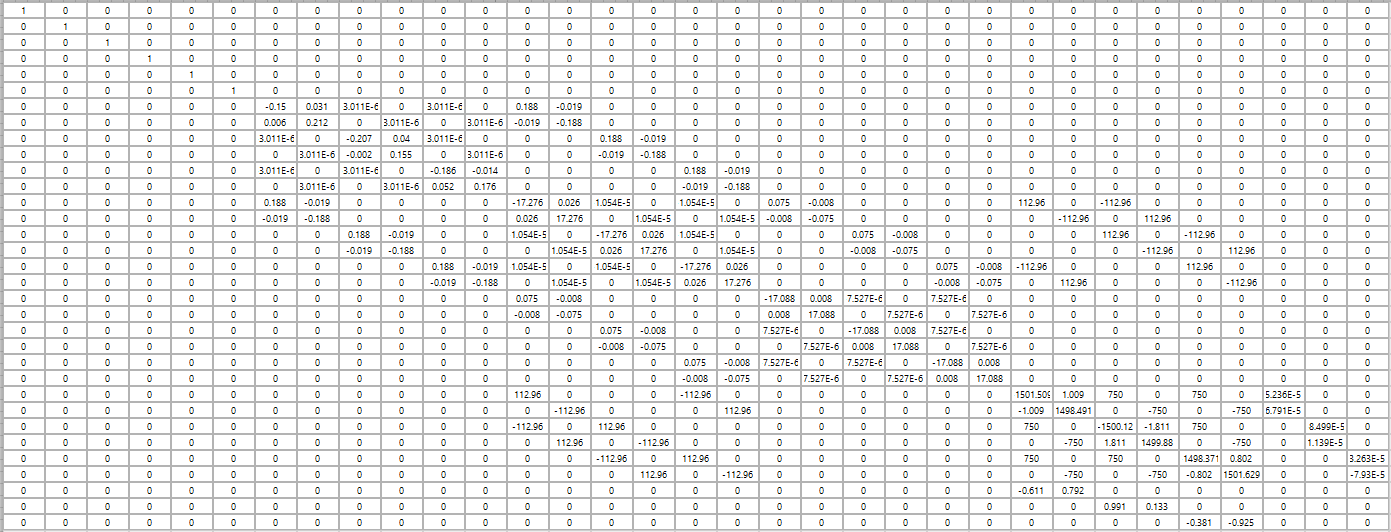


Figure . Jacobian matrix iteration 5

# References

[1] V. M. d. Costa, N. Martins, and J. L. R. Pereira, "Developments in the Newton Raphson power flow formulation based on current injections," *IEEE Transactions on Power Systems,* vol. 14, no. 4, pp. 1320-1326, 1999, doi: 10.1109/59.801891.

[2] P. A. N. Garcia, J. L. R. Pereira, S. Carneiro, Jr., V. M. da Costa, and N. Martins, "Three-phase power flow calculations using the current injection method," *IEEE Transactions on Power Systems,* vol. 15, pp. 508-514, 2000, doi: 10.1109/59.867133.

[3] D. R. R. Penido, L. R. d. Araujo, S. Carneiro Júnior, and J. L. R. Pereira, "A new tool for multiphase electrical systems analysis based on current injection method," *International Journal of Electrical Power & Energy Systems,* vol. 44, no. 1, pp. 410-420, 2013/01/01/ 2013, doi: <https://doi.org/10.1016/j.ijepes.2012.07.066>.

[4] J. Arrillaga and C. P. Arnold, *Computer Analysis of Power Systems*. Wiley, 1990.

[5] P. A. N. Garcia, J. L. R. Pereira, S. Carneiro, M. P. Vinagre, and F. V. Gomes, "Improvements in the representation of PV buses on three-phase distribution power flow," *IEEE Transactions on Power Delivery,* vol. 19, no. 2, pp. 894-896, 2004, doi: 10.1109/TPWRD.2003.820414.