Middle East Technical University

Electrical and Electronics Engineering Department

---

# -EE472: Power System Analysis II-

# -Project 1-

Newton-Raphson Load Flow Analysis

Using Phyton

Spring 2024

---

Osman Özütemiz

Due: June 12, 2024

## Introduction

In this project, we delve into power flow analysis using the Newton-Raphson iteration method in Python. Power flow analysis is vital for understanding and optimizing electrical grids, ensuring efficient energy transmission. The Newton-Raphson method, a powerful numerical technique, iteratively computes steady-state operating conditions such as voltage magnitudes and power flows. By implementing this method in Python, we aim to analyze power flow dynamics, validate results, and contribute to advancing energy system optimization techniques.

The IEEE 300-bus system is used as a test case for various power system analysis and optimization techniques, including power flow analysis, optimal power flow, transient stability analysis, and more. It serves as a standard benchmark for evaluating the performance of algorithms and methodologies in power system studies.

## IEEE 300 Bus File

The provided 300 bus file contains info about components such as generators, transformers, lines, and loads, representing a realistic power grid network.

1. **Bus Data**:

   - Bus number

   - Type of bus (3=Slack, 2=PV, 0-1=PQ)

   - Voltage magnitudes (final)

   - Voltage phase angles (final)

   - Generation and load data (Pgen, Qgen, Pload, Qload)

   - Shunt conductance G and Shunt susceptance B values.

   - MW or MVAr limits (if any)

2. **Branch Data**:

   - From bus

   - To bus

   - Type (0 = transmission line)

   - Branch parameters (r, x, b (pu))

   - Tap ratio (if applicable, if not; 0 == 1)

   - Phase shift (if applicable)

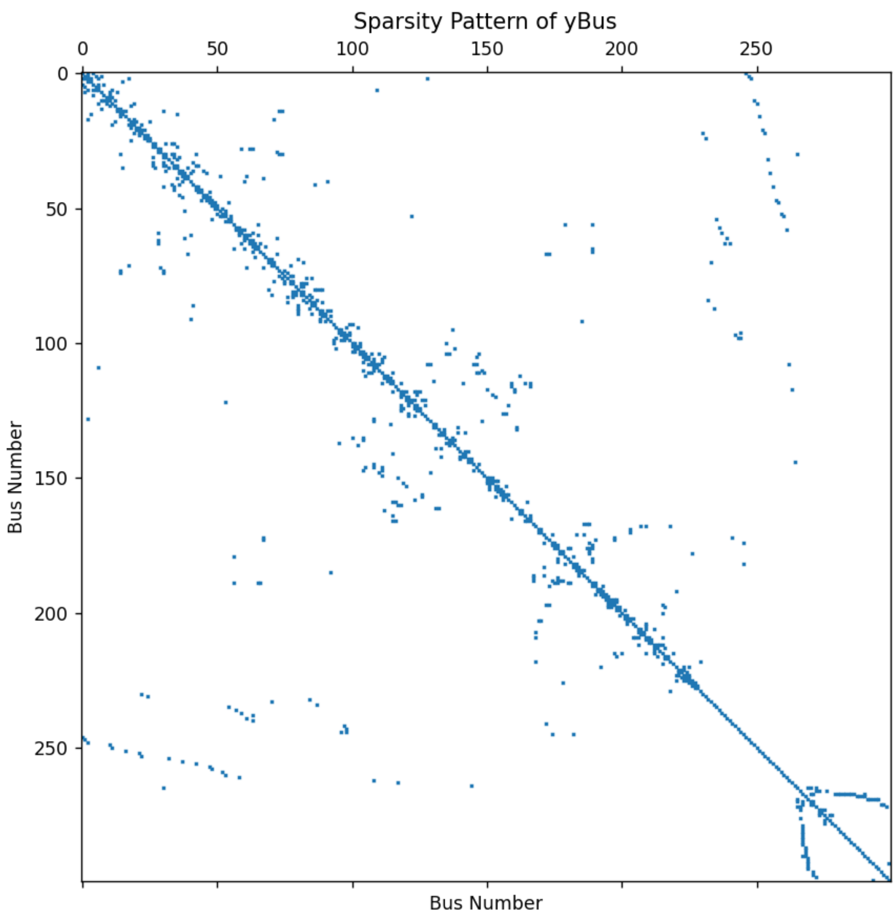
## a) Outputs of the code (ieee300cdf.txt as input file)

Output is voltage and phase angles. Total load, total generation and total loss values.

1. Sparsity pattern plot of $Y_{BUS}$, it prints Ybus matrice sparsity pattern using matplotlib library:

```
yBus_sparse = csr_matrix(yBus)

# Plotting the sparsity pattern
plt.figure(figsize=(10, 8))
plt.spy(yBus_sparse, markersize=1)
plt.title('Sparsity Pattern of yBus')
plt.xlabel('Bus Number')
plt.ylabel('Bus Number')
plt.show()
```



2. Voltage magnitude values at each bus in per unit,

**Voltage Magnitudes:**

[1.02400000e+03 1.02400000e+03 1.02400000e+03 1.02400000e+03
1.02400000e+03 1.02400000e+03 1.02400000e+03 1.00000000e+00
1.10000000e+01 1.00000000e+00 6.60000000e+01 1.10000000e+01
2.86000000e+02 6.60000000e+01 1.00100000e+03 2.86000000e+02
3.00300000e+03 1.00100000e+03 1.00000000e+00 8.00800000e+03
3.00300000e+03 1.10000000e+01 1.94480000e+04 8.00800000e+03
6.60000000e+01 4.37580000e+04 1.94480000e+04 2.86000000e+02
9.23780000e+04 4.37580000e+04 1.00100000e+03 1.84756000e+05
9.23780000e+04 3.00300000e+03 3.52716000e+05 1.84756000e+05
8.00800000e+03 6.46646000e+05 3.52716000e+05 1.94480000e+04
1.14406600e+06 6.46646000e+05 4.37580000e+04 1.96125600e+06
1.14406600e+06 9.23780000e+04 3.26876000e+06 1.96125600e+06
1.84756000e+05 5.31173500e+06 3.26876000e+06 3.52716000e+05
8.43628500e+06 5.31173500e+06 1.00000000e+00 6.46646000e+05
1.31231100e+07 8.43628500e+06 1.10000000e+01 1.14406600e+06
2.00300100e+07 1.31231100e+07 1.00000000e+00 6.60000000e+01
1.96125600e+06 3.00450150e+07 2.00300100e+07 1.10000000e+01
1.00000000e+00 2.86000000e+02 3.26876000e+06 4.43521650e+07
3.00450150e+07 6.60000000e+01 1.10000000e+01 1.00000000e+00
1.00000000e+00 1.00100000e+03 5.31173500e+06 1.00000000e+00
6.45122400e+07 4.43521650e+07 2.86000000e+02 6.60000000e+01
1.10000000e+01 1.10000000e+01 3.00300000e+03 1.00000000e+00
8.43628500e+06 1.10000000e+01 9.25610400e+07 6.45122400e+07
1.00100000e+03 2.86000000e+02 6.60000000e+01 6.60000000e+01
8.00800000e+03 1.00000000e+00 1.10000000e+01 1.31231100e+07
6.60000000e+01 1.31128140e+08 1.00000000e+00 1.00000000e+00
9.25610400e+07 3.00300000e+03 1.00100000e+03 2.86000000e+02
2.86000000e+02 1.94480000e+04 1.10000000e+01 6.60000000e+01
2.00300100e+07 2.86000000e+02 1.83579396e+08 1.10000000e+01
1.00000000e+00 1.10000000e+01 1.31128140e+08 1.00000000e+00
8.00800000e+03 1.00000000e+00 3.00300000e+03 1.00100000e+03
1.00000000e+00 1.00000000e+00 1.00100000e+03 1.00000000e+00
4.37580000e+04 6.60000000e+01 1.00000000e+00 1.00000000e+00
2.86000000e+02 3.00450150e+07 1.00000000e+00 1.00100000e+03

2.54186856e+08 6.60000000e+01 1.10000000e+01 6.60000000e+01

1.83579396e+08 1.10000000e+01 1.94480000e+04 1.10000000e+01

8.00800000e+03 3.00300000e+03 1.10000000e+01 1.10000000e+01

1.00000000e+00 1.00000000e+00 3.00300000e+03 1.10000000e+01

9.23780000e+04 2.86000000e+02 1.00000000e+00 1.00000000e+00

1.10000000e+01 1.10000000e+01 1.00100000e+03 4.43521650e+07

1.10000000e+01 3.00300000e+03 3.48330136e+08 1.00000000e+00

1.00000000e+00 1.00000000e+00 2.86000000e+02 6.60000000e+01

1.00000000e+00 1.00000000e+00 2.86000000e+02 2.54186856e+08

6.60000000e+01 4.37580000e+04 6.60000000e+01 1.94480000e+04

1.00000000e+00 8.00800000e+03 6.60000000e+01 6.60000000e+01

1.10000000e+01 1.10000000e+01 8.00800000e+03 6.60000000e+01

1.84756000e+05 1.00100000e+03 1.10000000e+01 1.10000000e+01

6.60000000e+01 6.60000000e+01 3.00300000e+03 1.00000000e+00

6.45122400e+07 6.60000000e+01 8.00800000e+03 4.72733756e+08

1.10000000e+01 1.10000000e+01 1.00000000e+00 1.00000000e+00

1.00000000e+00 1.10000000e+01 1.00100000e+03 2.86000000e+02

1.10000000e+01 1.00000000e+00 1.10000000e+01 1.00100000e+03

1.00000000e+00 3.48330136e+08 2.86000000e+02 1.00000000e+00

9.23780000e+04 2.86000000e+02 1.00000000e+00 4.37580000e+04

1.00000000e+00 1.00000000e+00 1.10000000e+01 1.00000000e+00

1.00000000e+00 1.00000000e+00 1.94480000e+04 2.86000000e+02

2.86000000e+02 6.60000000e+01 6.60000000e+01 1.94480000e+04

2.86000000e+02 3.52716000e+05 3.00300000e+03 6.60000000e+01

6.60000000e+01 2.86000000e+02 2.86000000e+02 8.00800000e+03

1.10000000e+01 9.25610400e+07 2.86000000e+02 1.94480000e+04

6.35745396e+08 6.60000000e+01 6.60000000e+01 1.10000000e+01

1.10000000e+01 1.10000000e+01 1.00000000e+00 1.00000000e+00

1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00

1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00

1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00

1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00

1.00000000e+00 6.60000000e+01 1.00000000e+00 3.00300000e+03

1.00100000e+03 6.60000000e+01 1.10000000e+01 6.60000000e+01

3.00300000e+03 1.10000000e+01 4.72733756e+08 1.00100000e+03

1.10000000e+01 1.84756000e+05 1.00100000e+03 1.10000000e+01

9.23780000e+04 1.10000000e+01 1.10000000e+01 6.60000000e+01

1.10000000e+01 1.10000000e+01 1.10000000e+01 4.37580000e+04

1.00100000e+03 1.00100000e+03 2.86000000e+02 1.00000000e+00

2.86000000e+02 1.00000000e+00 1.00000000e+00 1.00000000e+00

4.37580000e+04 1.00100000e+03 6.46646000e+05 1.02400000e+03]

3. Voltage angle values at each bus in degrees,

**Voltage Angles:**

[ 0.00000000e+00 8.89796748e+02 -2.22461826e+02 1.93466438e+03

9.71366932e+03 9.64364515e+02 1.43360687e+03 -9.29451274e+02

9.97467118e+03 4.97276419e+03 7.99395666e+03 1.76812668e+04

7.86276344e+03 1.60262886e+04 1.59590458e+04 1.63518971e+04

1.59503073e+04 1.63548954e+04 1.61059691e+04 1.62135689e+04

1.62115597e+04 1.62386272e+04 1.62279829e+04 1.62523113e+04

1.62477876e+04 1.62399752e+04 1.63577747e+04 1.63668884e+04

1.63155845e+04 1.63474663e+04 1.63500094e+04 1.63556791e+04

1.63711554e+04 1.63543420e+04 1.63594400e+04 1.63680046e+04

1.63492803e+04 1.63525185e+04 1.63601957e+04 1.63583598e+04

1.63438286e+04 1.63542129e+04 1.63534344e+04 1.63566197e+04

1.63566230e+04 1.63508220e+04 1.63535141e+04 1.63605596e+04

1.63871005e+04 1.63849775e+04 1.63831153e+04 1.63870644e+04

1.63860226e+04 1.63951429e+04 1.63657385e+04 1.63751719e+04

1.63437272e+04 1.47306007e+04 1.46954666e+04 1.63115555e+04

1.63423593e+04 1.63479758e+04 1.63269477e+04 1.63266810e+04

1.63451639e+04 1.63450889e+04 1.62893954e+04 1.63637068e+04

1.63577352e+04 1.63506419e+04 1.63377814e+04 1.63501589e+04

1.63506893e+04 1.65579757e+04 1.73948209e+04 1.63472113e+04

-9.12418217e+03 1.63466084e+04 1.63375090e+04 1.63354561e+04

1.63511520e+04 1.63350976e+04 1.63352856e+04 1.63490292e+04

1.63324976e+04 1.63601654e+04 1.63569834e+04 1.63516148e+04

1.63419076e+04 1.63420771e+04 1.63336865e+04 1.63398117e+04

1.63337547e+04 -2.99048031e+04 -1.58007448e+04 -2.98550189e+04

-2.98851627e+04 -2.98662924e+04 -6.41600700e+03 -2.99032043e+04

-2.99038966e+04 -2.98924827e+04 -2.98669822e+04 -2.20100244e+04

4.93593035e+02 5.02408650e+02 4.65705180e+02 5.08513309e+02

2.37306201e+02 1.40837936e+03 5.28062744e+02 4.47362383e+02

4.99410610e+02 5.02202143e+02 5.17979496e+02 5.09543178e+02

5.00379307e+02 4.68781530e+02 5.06440927e+02 5.12875161e+02

5.08042659e+02 -4.73021495e+01 1.63961843e+04 5.16180990e+02

5.22296582e+02 5.25434678e+02 -1.79847939e+04 7.62956098e+02

-9.10420708e+01 6.60896484e+02 5.33833024e+02 6.30275241e+02

6.09748347e+02 6.22636925e+02 6.19524835e+02 4.95076331e+02

4.96492015e+02 4.98934794e+02 4.93732358e+02 6.26341094e+02

6.42627793e+02 6.13912929e+02 6.27429383e+02 6.56252763e+02

6.63606776e+02 1.70659591e+02 5.26142842e+02 5.06727477e+02

1.85562291e+04 4.00104730e+02 4.88221917e+02 4.80139531e+02

4.93988051e+02 5.04088103e+02 9.53884972e+02 -1.61784563e+03

-1.85810707e+03 -1.27718608e+04 4.64973469e+02 5.09781029e+02

5.04332079e+02 6.23930827e+02 5.01390111e+02 4.92065302e+02

5.15221517e+02 5.16114704e+02 5.12344033e+02 1.53156384e+04

1.11469532e+04 1.12091492e+04 1.11698076e+04 1.69809288e+04

1.40667070e+04 8.66757171e+03 1.70178313e+04 1.66332741e+04

1.66534810e+04 1.42235674e+04 1.42623506e+04 1.63361343e+04

1.81506092e+06 1.64593928e+04 1.56773995e+04 1.69780709e+04

1.68050756e+04 1.67947722e+04 1.69821770e+04 1.52747565e+04

1.42233171e+04 1.62756334e+04 8.64915263e+03 1.04857383e+04

1.04762497e+04 1.04742955e+04 9.56665212e+03 9.56772605e+03

9.56850561e+03 9.56174771e+03 9.56779399e+03 1.11438577e+04

1.11439330e+04 1.11445559e+04 1.11471877e+04 1.11715606e+04

1.11467804e+04 1.11247991e+04 1.11474863e+04 1.11487202e+04

1.11424991e+04 1.11467723e+04 1.11430174e+04 1.11337039e+04

1.11477166e+04 1.11480078e+04 1.11571663e+04 1.11470437e+04

1.11483603e+04 1.10598310e+04 1.11470902e+04 1.11596352e+04

1.04763816e+04 1.29264766e+04 1.29167760e+04 1.27997015e+04

1.29108100e+04 1.22947426e+04 1.11298311e+04 1.11232747e+04

1.11264028e+04 1.11472317e+04 1.62203530e+04 1.62404480e+04

```
 1.63314325e+04  1.63389315e+04  1.63422635e+04  1.63343146e+04
 1.63234541e+04  1.63090206e+04  1.63004418e+04  1.63384668e+04
 1.63203080e+04  1.44242023e+04  5.34870347e+02  4.03477703e+02
 4.91004949e+02  1.64752783e+04 -3.33038627e+04  7.44740888e+02
-4.80010802e+02  8.12303997e+03  2.52221054e+04  1.68078638e+04
 1.62136505e+04  1.62437270e+04  1.63837427e+04  1.63588458e+04
 1.63535046e+04  1.63668734e+04  1.63902600e+04  1.63954659e+04
 1.64077328e+04  1.21863850e+04  3.43809343e+02  1.54822965e+02
 6.69904461e+02  1.63515962e+04  1.63381438e+04  1.63403618e+04
 1.63400387e+04  1.63515159e+04  1.63431270e+04  1.63419989e+04
 1.63419745e+04  1.62697082e+04  1.62658340e+04  1.62692846e+04
 1.63371990e+04  1.62677370e+04  1.62679655e+04  1.63332113e+04
 1.63339185e+04  1.63328833e+04  1.63386079e+04  1.63349177e+04
 1.63354012e+04  1.63359761e+04  1.63334359e+04  1.63378467e+04
 1.63361033e+04  1.63375616e+04  1.63401396e+04  1.63887370e+04
 1.63429510e+04  1.63198169e+04  1.63356434e+04  1.63544227e+04
 1.63392630e+04  1.63402624e+04  1.63385275e+04  1.63198361e+04]
```

**4. Solution time,**
**5. Number of iterations,**
**6. Real and reactive power losses in the system,**

Rest of the Output:

Newton Raphson Load-Flow Study

Report of Power Flow Calculations

2024-06-13 00:46:35

Number of iterations : 10

Solution time : 12.423627853393555 sec.

Total real power losses : 45257196501.9953.

Total reactive power losses: -951620309738.458.

V, theta, Pgen, Qgen, Pload, Qload results here. Line Flows

**#Line From Bus To Bus Real Reactive**

```
[[ 1.00000000e+00  3.70000000e+01  9.00100000e+03  7.93800000e-01
```

```
 -6.08580000e+00]
 [ 2.00000000e+00  9.00100000e+03  9.00500000e+03  0.00000000e+00
   0.00000000e+00]
 [ 3.00000000e+00  9.00100000e+03  9.00600000e+03  7.68285000e+00
  -1.37598300e+02]
 ...
 [ 4.09000000e+02  7.04400000e+03  4.40000000e+01  0.00000000e+00
   1.12485847e+03]
 [ 4.10000000e+02  7.05500000e+03  5.50000000e+01  0.00000000e+00
   2.27970589e+03]
 [ 4.11000000e+02  7.07100000e+03  7.10000000e+01  0.00000000e+00
   3.44800000e-01]]
[[ 1.00000000e+00  9.00100000e+03  3.70000000e+01 -1.32300000e-01
   1.01430000e+00]
 [ 2.00000000e+00  9.00500000e+03  9.00100000e+03  0.00000000e+00
   0.00000000e+00]
 [ 3.00000000e+00  9.00600000e+03  9.00100000e+03 -2.19510000e+00
   3.93138000e+01]
 ...
 [ 4.09000000e+02  4.40000000e+01  7.04400000e+03  0.00000000e+00
  -6.96062421e+06]
 [ 4.10000000e+02  5.50000000e+01  7.05500000e+03  0.00000000e+00
  -2.65084201e+07]
 [ 4.11000000e+02  7.10000000e+01  7.07100000e+03  0.00000000e+00
  -2.06880000e+00]]
```

**7. Bus ID numbers (as given in CDF file) of PV buses that are stuck to Q-limits.**

```
#buses_at_limits = []
# Check which buses are at their limits
#for i in range(len(P)):
#    if P[i] >= Pmax[i] or P[i] <= Pmin[i] or Q[i] >= Qmax[i] or Q[i] <= Qmin[i]:
#        buses_at_limits.append(i + 1)  # Adjusting to 1-based index)
#print(f"Buses at PQ limits: {buses_at_limits}")
```

## B) Test results

Test Results are provided at part a)

## c) Convergence threshold.

It is determined by “tol” in definition:

def power_flow_newton_raphson(bus_matrix_np, G, B, yBus, tol=1e-6, max_iter=20):

by default, it is 1e-6.

## d) Any computational method you used to improve computational performance.

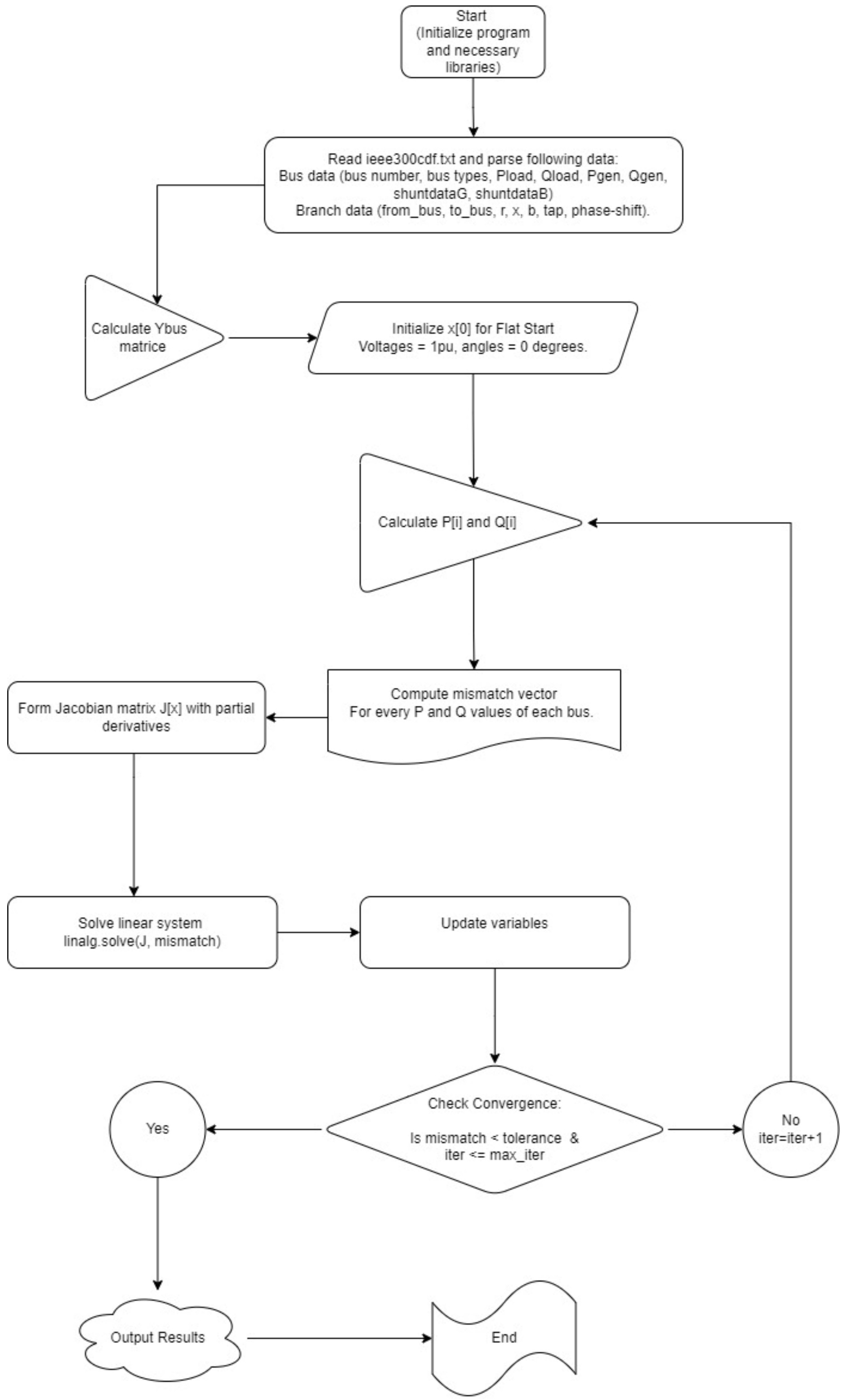
- **Sparse Matrix Representation:**

In power systems, Jacobian matrices can be very sparse (contains mostly zeros). Using sparse data in Jacobian matrices improves computational performance drastically. It reduces memory usage and reduce time for matrix operations.

- **Vectorization:**

I used Numpy to vectorize matrices. Numpy is a great library to enhance computational efficiency by **executing operations on entire arrays instead of looping through each element.**

## e) Flow chart of the algorithm:



## f) Solution time and computer details,

2024-06-13 01:28:13

Number of iterations : 10

Solution time : 12.42 sec.

2024-06-13 01:34:56

Number of iterations : 50

Solution time : 118.01 sec.

Power flow analysis converged in 10 iterations with a solution time of 12.42 seconds. Subsequently, increasing the iterations to 50 extended the solution time to 118.01 seconds. This significant increase underscores the computational intensity of brute-force methods. To optimize efficiency, decoupled power flow techniques can be explored. These methods calculate active and reactive power separately, and can improve convergence efficiency and reduce computational load, making them advantageous for large-scale power system analyses.

## g) Explanation of the solution

As discussed in the flow chart, this solution is converged to a wrong result. Because it has indexing problems. It is correct until Y bus and jacobian matrices. It calculates mismatch correct but its having error on linalg function.

The power flow analysis solution as discussed in the flow chart encounters significant problems primarily due to indexing errors and inaccuracies in linear algebra operations.

While the initial steps are all **correct**, such as calculating **mismatches**, constructing the **Y bus** matrix, and computing the **Jacobian** matrix, issues arise when accessing and manipulating array elements.

Mismatches, Ybus and J[x] are checked to be correct.

These errors lead to incorrect references in matrix operations, particularly evident during the solution's convergence check using linear algebra functions like linalg.solve. Correcting these indexing problems and ensuring proper matrix dimensions are crucial for accurately applying the Newton-Raphson method in power system analysis, ultimately yielding reliable and valid results for electrical network simulations.

## h) Any additional reasoning and comments

The issue encountered with np.column_stack from arrays representing bus numbers, voltages, angles, Ploss, and Pgen having differing sizes. This mismatch prevented straightforward stacking due to varying lengths of data entries in the output screen. To resolve this, I need to aligning data dimensions or adjusting data structures all to bus number size.

## References


- PowerFlow (Newton Raphson). (2021, August 4). Powerflow (Newton Raphson) - File Exchange - MATLAB CentralFile Exchange - MATLAB Central. https://www.mathworks.com/matlabcentral/fileexchange/60988-powerflow-newton-raphson
- Newton Raphson Power Flow Solution using MATLAB. (2010b, January 16). Newton Raphson Power Flow Solution Using MATLAB - File Exchange - MATLAB CentralFile Exchange - MATLAB Central. https://www.mathworks.com/matlabcentral/fileexchange/26391-newton-raphson-power-flow-solution-using-matlab