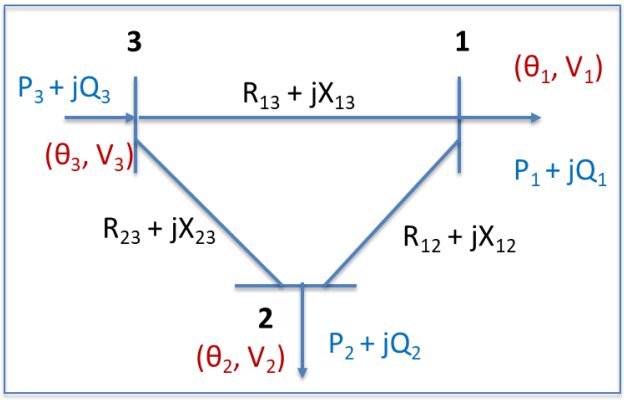
ELK14: Methods and Algorithms for Power Systems

Assignment 1: Newton-Raphson Load Flow



Network	
R_{12}	0.1
X_{12}	0.2
R_{13}	0.05
X_{13}	0.25
R_{23}	0.05
X_{23}	0.15
Loads	
P_1	-0.8
Q_1	-0.5
P_2	-0.4
Q_2	-0.5

Figure 1: Example system – all data in PU.

The bus 3 can be used as slackbus.

The calculations will be done on the example system shown in figure 1.

We will use the same test system in several assignments where the Jacobi-elements will be needed.

The slides of the lecture will be useful as a model for this assignment.

1. Calculate the base case conditions as specified in the table assuming a flat start (Angles equals zero and all bus voltages equals one). For each iteration show:
    a. The Jacobi matrix
    b. Net injections and mismatch vector (right-hand side of equation)
    c. The correction vector (angle and magnitude)
2. Increase the system load in steps of ΔP= 0.2 pu where the load increase is shared between the two buses (Bus 1 (30%) and Bus 2 (70%)) until the load flow fail to converge. Please do this for these two cases:
    a. Use previous load flow as a starting point. Plot the voltage at bus 1 and bus 2 as a function of system load
    b. Try to use flat start for each load flow. Plot the voltage at Bus 1 and Bus 2 as a function of system load
3. Increase the system load in steps of ΔQ= 0.2 pu where the load increase is shared between the two buses (Bus 1 (30%) and Bus 2 (70%)) until the load flow fail to converge.

You can use Julia, Python or Matlab. The purpose is not the check the programming skills so just put the needed effort to solve the cases and to be sure that the principles are understood and can be explained. The assignments will be highly relevant for the exam and any assignment may be requested to be explained based on your own code.

The Jacobi-matrix will be used in other assignments so it may save time later if you make it a bit general.

I recommend to make it object-oriented as later development is faster.