EE 5240 - Lecture 11 Mon Feb 4th / Wed Feb 6, 2019

Topics for Today:

- Questions?
- Questions/Comments on Homework #4 ?
- Building [Y] for 14-bus IEEE system.
- Useful Matlab functions: fgetl, sscanf, format '%*23c%f', spalloc, sparse, spy
- Newton Iteration, example for one-variable case
- Newton Iteration, example for two-variable case
- Loadflow Formulation: "NR Details" handout (Week 4)
- NR Algorithm implementation.

Coming up:

- More MatLab - build Jacobian, solve for $\Delta\delta$ and ΔV, iterate.
- Data structures, LU factorization, reordering to avoid zero divides and/or speed up solution.

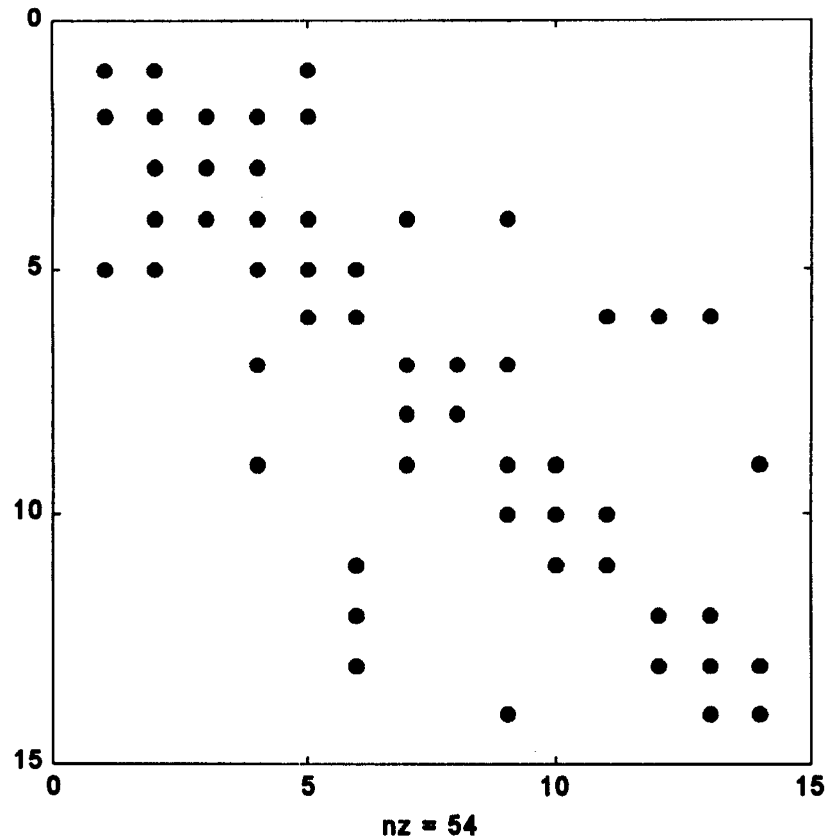
[Ybus] for 14-bus IEEE system: **IEEE14.cdf**

Produced by: lf.m
Programmed by: Bruce Mork
Date: 01 Feb 2006

(1,1)	6.0250 -19.4471i
(2,1)	-4.9991 +15.2631i
(5,1)	-1.0259 + 4.2350i
(1,2)	-4.9991 +15.2631i
(2,2)	9.5213 -30.2721i
(3,2)	-1.1350 + 4.7819i
(4,2)	-1.6860 + 5.1158i
(5,2)	-1.7011 + 5.1939i
(2,3)	-1.1350 + 4.7819i
(3,3)	3.1210 - 9.8224i
(4,3)	-1.9860 + 5.0688i
(2,4)	-1.6860 + 5.1158i
(3,4)	-1.9860 + 5.0688i
(4,4)	10.5130 -38.6542i
(5,4)	-6.8410 +21.5786i
(7,4)	0 + 4.8895i
(9,4)	0 + 1.8555i
(1,5)	-1.0259 + 4.2350i
(2,5)	-1.7011 + 5.1939i
(4,5)	-6.8410 +21.5786i
(5,5)	9.5680 -35.5336i
(6,5)	0 + 4.2574i
(5,6)	0 + 4.2574i
(6,6)	6.5799 -17.3407i
(11,6)	-1.9550 + 4.0941i
(12,6)	-1.5260 + 3.1760i
(13,6)	-3.0989 + 6.1028i
(4,7)	0 + 4.8895i
(7,7)	0 -19.5490i
(8,7)	0 + 5.6770i
(9,7)	0 + 9.0901i
(7,8)	0 + 5.6770i
(8,8)	0 - 5.6770i
(4,9)	0 + 1.8555i
(7,9)	0 + 9.0901i
(9,9)	5.3261 -24.0925i
(10,9)	-3.9020 +10.3654i
(14,9)	-1.4240 + 3.0291i
(9,10)	-3.9020 +10.3654i
(10,10)	5.7829 -14.7683i
(11,10)	-1.8809 + 4.4029i
(6,11)	-1.9550 + 4.0941i
(10,11)	-1.8809 + 4.4029i
(11,11)	3.8359 - 8.4970i
(6,12)	-1.5260 + 3.1760i
(12,12)	4.0150 - 5.4279i
(13,12)	-2.4890 + 2.2520i
(6,13)	-3.0989 + 6.1028i
(12,13)	-2.4890 + 2.2520i
(13,13)	6.7249 -10.6697i
(14,13)	-1.1370 + 2.3150i
(9,14)	-1.4240 + 3.0291i
(13,14)	-1.1370 + 2.3150i
(14,14)	2.5610 - 5.3440i

Network topology, spy(Ybus):

2



Sparsity = 54/14x14 = 27.55%

2-Variable NR

$$[A][x] = [B]$$

① $2x + y = 4$

② $2x + y^2 = 6$

②-① $y^2 - y = 2 \Rightarrow \begin{matrix} x = 1 \\ y = 2 \end{matrix}$

define $f(x,y) = 2x + y - 4 = 0$

$g(x,y) = 2x + y^2 - 6 = 0$

Taylor Expansion

iteration index

$$0 = f(x,y) = f(x^m, y^m) + \frac{\partial f(x^m, y^m)}{\partial x}(\underline{x} - x^m) + \frac{\partial f(x^m, y^m)}{\partial y}(y-$$

$$0 = g(x,y) = g(x^m, y^m) + \frac{\partial g(x^m, y^m)}{\partial x}(x - x^m) + \frac{\partial g(x^m, y^m)}{\partial y}(\underline{y} - y^m)$$

Rearranging terms:

$$\frac{\partial f^m}{\partial x}\Delta x + \frac{\partial f^m}{\partial y}\Delta y = -f^m \qquad \underbrace{f(x,y)}_{=0} - f(x^m, y^m)$$

$$\frac{\partial g^m}{\partial x}\Delta x + \frac{\partial g^m}{\partial y}\Delta y = -g^m$$

In matrix form

$$\underbrace{\begin{bmatrix} \frac{\partial f^m}{\partial x} & \frac{\partial f^m}{\partial y} \\ \frac{\partial g^m}{\partial x} & \frac{\partial g^m}{\partial y} \end{bmatrix}}_{\textbf{JACOBIAN}} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = - \begin{bmatrix} f^m \\ g^m \end{bmatrix}$$

$\Delta x = \underline{x - x^m}$

$\Delta y = y - y^m$

$f(x^m, y^m)$

$g(x^m, y^m)$

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = -\begin{bmatrix} J^m \end{bmatrix}^{-1} \begin{bmatrix} f^m \\ g^m \end{bmatrix} \cong \begin{bmatrix} x - x^m \\ y - y^m \end{bmatrix}$$

Estimate of (x, y) for next iteration:

$$\underline{\underline{\Delta x}} \cong x - \underline{\underline{x^m}} \Rightarrow x^{m+1} = x^m + \Delta x$$

$$\Delta y \cong y - y^m \Rightarrow y^{m+1} = y^m + \Delta y$$

use (x^{m+1}, y^{m+1}) for next iteration,

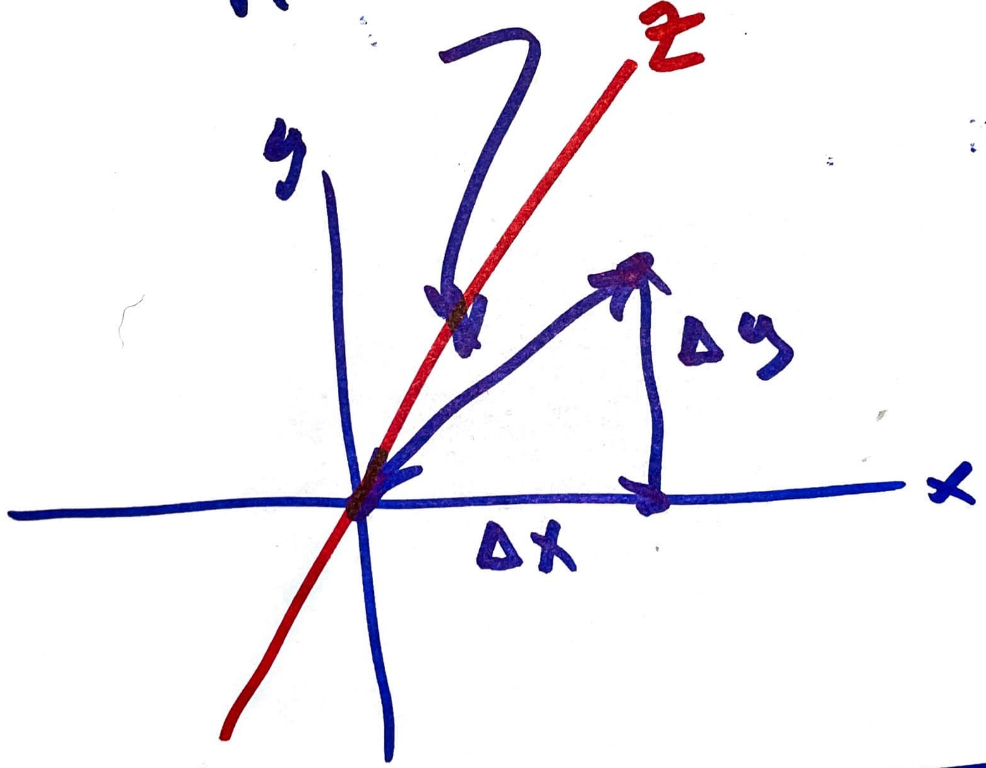
Continue until "converged".

Common tests: ① Δx & Δy both $< \epsilon$

② $\| \Delta x, \Delta y \| < \epsilon$

Norm -

$$\| \Delta x, \Delta y \| = \sqrt{\Delta x^2 + \Delta y^2}$$



For N variables? (in "N-Space") $= \sqrt{\Delta x_1^2 + \Delta x_2^2 + \ldots \Delta x_N^2}$

Norm: $\sqrt{\Delta x_1^2 + \Delta x_2^2 + \cdots \Delta x_N^2}$ (for N-variable system)

for this case,

Norm is $\| \Delta x, \Delta y \| = \sqrt{\Delta x^2 + \Delta y^2}$

Back to example:

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = - \left[J^m \right]^{-1} \begin{bmatrix} f^m \\ g^m \end{bmatrix}$$

Start: $m = 0$ ($0^{\underline{th}}$ iteration is initial guess)

$$[J] = \begin{bmatrix} 2 & 1 \\ 2 & 2y \end{bmatrix}$$

guess: $\begin{bmatrix} 0 \\ 3 \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \end{bmatrix}$

$$\left[J^0\right] = \begin{bmatrix} 2 & 1 \\ 2 & 6 \end{bmatrix} \Rightarrow \left[J^0\right]^{-1} = \begin{bmatrix} -.6 & .1 \\ .2 & -.2 \end{bmatrix}$$

$$\begin{bmatrix} f^0 \\ g^0 \end{bmatrix} = \begin{bmatrix} f(0) \\ g(3) \end{bmatrix} = \begin{bmatrix} -1 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} -.6 & .1 \\ .2 & -.2 \end{bmatrix} \begin{bmatrix} -1 \\ 3 \end{bmatrix} = \begin{bmatrix} .9 \\ -.8 \end{bmatrix} = \begin{bmatrix} x - x^0 \\ y - y^0 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} +0.9 \\ +2.2 \end{bmatrix}$$

- Next, repeat iteration using $\begin{bmatrix} x' \\ y' \end{bmatrix}$
- Test $\| \Delta x, \Delta y \| \leq \epsilon$ at each iteration.

After 3 iterations, $\underline{\epsilon < 0.001}$

$x = 1.000$
$y = 2.000$

typical for p.u. load flow.

<u>Notes</u>:

- Convergence rate not dependent on no. of variables.
- Important to make intelligent guess of initial values.

e.g.

Flat start

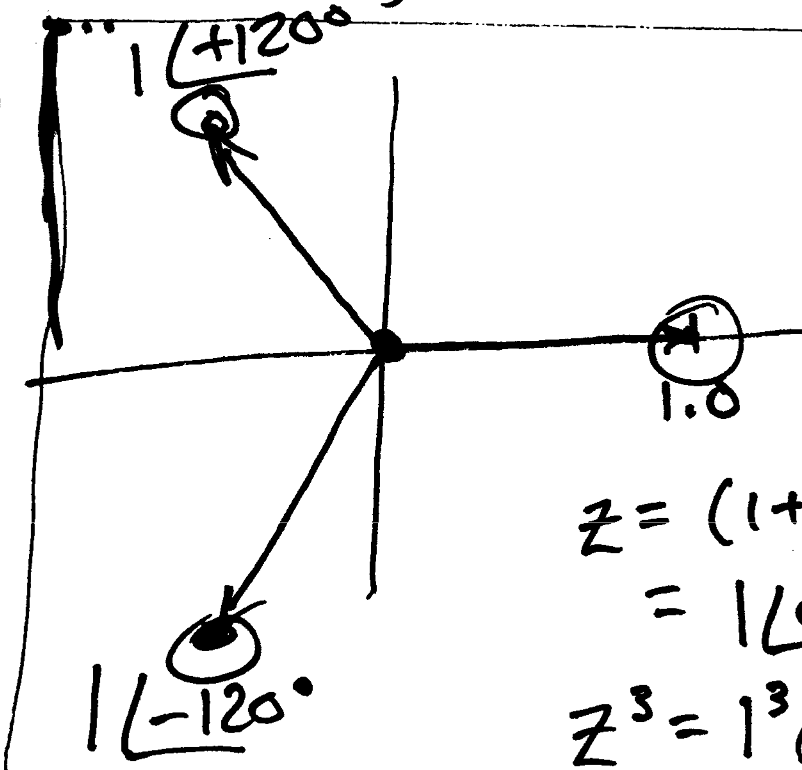
$$\left\{ \begin{array}{l} |\tilde{V}_{Bus}| = 1.0 \text{ p.u.} \\ \delta = 0^\circ \quad (\delta = \angle V_{Bus}) \end{array} \right.$$

- NR can fail to converge. Important to have correct system data.

MichiganTech Instructor: Bruce Mork Phone (906) 487-2857 Email: bamork@mtu.edu

- Possible to converge to wrong solution.

<u>Ex:</u> $z^3 = 1$



$z = (1 + j0)$
$= 1\angle{0^\circ}$

$z^3 = 1^3\angle{3 \times 0^\circ}$
$= 1\angle{0^\circ}$

Which of the 3 roots will solution converge to for an arbitrary initial value? Your guess is as good as mine. Must let NR iteration converge to find out.

<u>Note:</u> We can generate some beautiful fractal patterns in this way. Let each initial condition on the rectangular area be assigned a color depending on which root it converges to. (Each pixel = initial condition).

MichiganTech Instructor: Bruce Mork Phone (906) 487-2857 Email: bamork@mtu.edu

$$\bar{S}_{Ti} = P_{Ti} + j\,Q_{Ti}$$

$$P = |\tilde{V}_i| \sum_{n=1}^{N} |\tilde{V}_n||y_{in}| \cos(\delta_i - \delta_n - \theta_{in})$$

$$Q = |\tilde{V}_i| \sum_{n=1}^{N} |\tilde{V}_n||y_{in}| \sin(\delta_i - \delta_n - \theta_{in})$$

Assume:

$$\tilde{V}_i = |\tilde{V}_i| \angle \delta_i$$

$$\tilde{V}_n = |\tilde{V}_n| \angle \delta_n$$

$$y_{in} = |y_{in}| \angle \theta_{in}$$

For referenced current, P, Q, flow directions,

See Lectur 8 p.4.

EE 5200

Newton-Raphson Power Flow

- **Active and Reactive Power Flowing INTO bus k:** $\Rightarrow I_k = \sum_{n=1}^{N} Y_{kn} V_n$

$$S = V I^* \quad \begin{cases} P_k = |V_k| \sum_{n=1}^{N} |Y_{kn}| |V_n| \cos(\delta_k - \delta_n - \theta_{kn}) \\ Q_k = |V_k| \sum_{n=1}^{N} |Y_{kn}| |V_n| \sin(\delta_k - \delta_n - \theta_{kn}) \end{cases} \qquad \text{k=1,2,...,N}$$

$$V_k = |V_k| e^{j\delta_k}$$

$$Y_{kn} = |Y_{kn}| e^{j\theta_{kn}}$$

$$S_k = |V_k| e^{j\delta_k} \left[\sum_{n=1}^{N} |Y_{kn}| e^{j\theta_{kn}} |V_n e^{j\delta_n}| \right]^*$$

- **Let:**

$$x = \begin{bmatrix} \delta \\ V \end{bmatrix} = \begin{bmatrix} \delta_2 \\ \cdot \\ \cdot \\ \delta_N \\ V_2 \\ \cdot \\ \cdot \\ V_N \end{bmatrix} \qquad y = \begin{bmatrix} P \\ Q \end{bmatrix} = \begin{bmatrix} P_2 \\ \cdot \\ \cdot \\ P_N \\ Q_2 \\ \cdot \\ \cdot \\ Q_N \end{bmatrix} \qquad f(x) = \begin{bmatrix} P(x) \\ Q(x) \end{bmatrix} = \begin{bmatrix} P_2(x) \\ \cdot \\ \cdot \\ P_N(x) \\ Q_2(x) \\ \cdot \\ \cdot \\ Q_N(x) \end{bmatrix}$$

where V, P, and Q terms are in per unit and δ terms are in radians.

$$\begin{array}{cc} \textbf{1} & \textbf{2} \end{array}$$

$$J = \left[\begin{array}{cccc|cccc} \frac{\partial P_2}{\partial \delta_2} & \cdot & \cdot & \frac{\partial P_2}{\partial \delta_N} & \frac{\partial P_2}{\partial V_2} & \cdot & \cdot & \frac{\partial P_2}{\partial V_N} \\ \cdot & & & \cdot & \cdot & & & \cdot \\ \cdot & & & \cdot & \cdot & & & \cdot \\ \frac{\partial P_N}{\partial \delta_2} & \cdot & \cdot & \frac{\partial P_N}{\partial \delta_N} & \frac{\partial P_N}{\partial V_2} & \cdot & \cdot & \frac{\partial P_N}{\partial V_N} \\ \hline \frac{\partial Q_2}{\partial \delta_2} & \cdot & \cdot & \frac{\partial Q_2}{\partial \delta_N} & \frac{\partial Q_2}{\partial V_2} & \cdot & \cdot & \frac{\partial Q_2}{\partial V_N} \\ \cdot & & & \cdot & \cdot & & & \cdot \\ \cdot & & & \cdot & \cdot & & & \cdot \\ \frac{\partial Q_N}{\partial \delta_2} & \cdot & \cdot & \frac{\partial Q_N}{\partial \delta_N} & \frac{\partial Q_N}{\partial V_2} & \cdot & \cdot & \frac{\partial Q_N}{\partial V_N} \end{array} \right]$$

$$\begin{array}{cc} \textbf{3} & \textbf{4} \end{array}$$

- **Jacobian Entries:**

For n≠k:

$$J1_{kn} = \frac{\partial P_k}{\partial \delta_n} = |V_k||Y_{kn}||V_n| \sin(\delta_k - \delta_n - \theta_{kn})$$

$$J2_{kn} = \frac{\partial P_k}{\partial V_n} = |V_k||Y_{kn}| \cos(\delta_k - \delta_n - \theta_{kn})$$

$$J3_{kn} = \frac{\partial Q_k}{\partial \delta_n} = -|V_k||Y_{kn}||V_n| \cos(\delta_k - \delta_n - \theta_{kn})$$

$$J4_{kn} = \frac{\partial Q_k}{\partial V_n} = |V_k||Y_{kn}| \sin(\delta_k - \delta_n - \theta_{kn})$$

For n=k:

$$J1_{kk} = \frac{\partial P_k}{\partial \delta_k} = -|V_k| \sum_{\substack{n=1 \\ n \neq k}}^{N} |Y_{kn}||V_n| \sin(\delta_k - \delta_n - \theta_{kn})$$

$$J2_{kk} = \frac{\partial P_k}{\partial V_k} = |V_k||Y_{kk}| \cos\theta_{kk} + \sum_{n=1}^{N} |Y_{kn}||V_n| \cos(\delta_k - \delta_n - \theta_{kn})$$

$$J3_{kk} = \frac{\partial Q_k}{\partial \delta_k} = |V_k| \sum_{\substack{n=1 \\ n \neq k}}^{N} |Y_{kn}||V_n| \sin(\delta_k - \delta_n - \theta_{kn})$$

cos *

$$J4_{kk} = \frac{\partial Q_k}{\partial V_k} = -|V_k||Y_{kk}| \sin\theta_{kk} + \sum_{n=1}^{N} |Y_{kn}||V_n| \sin(\delta_k - \delta_n - \theta_{kn})$$

PV Buses:

Omit V_K term from x vector.

Omit the Q_K term from y vector.

In [J] omit columns corresp to $\frac{\partial}{\partial V_K}$

and omit row corresponding to partials of

row corresponding V_K

- Use Newton Raphson to solve:

Previous

New/current

1) Use P & Q equations to calculate $\Delta y(i)$:

$$\Delta y(i) = \begin{bmatrix} \Delta P(i) \\ \Delta Q(i) \end{bmatrix} = \begin{bmatrix} P - P[x(i)] \\ Q - Q[x(i)] \end{bmatrix}$$

where:

$$x(i) = \begin{bmatrix} \delta(i) \\ V(i) \end{bmatrix}$$

If Convergence criteria is met, quit.

2) Calculate the Jacobian.

iteration index = i

3) Use LU Factorization to solve:

$$\left[\begin{array}{c|c} \text{J1}(i) & \text{J2}(i) \\ \hline \text{J3}(i) & \text{J4}(i) \end{array}\right] \begin{bmatrix} \Delta\delta(i) \\ \Delta V(i) \end{bmatrix} = \begin{bmatrix} \Delta P(i) \\ \Delta Q(i) \end{bmatrix}$$

4) Compute:

$$x(i+1) = \begin{bmatrix} \delta(i+1) \\ V(i+1) \end{bmatrix} = \begin{bmatrix} \delta(i) \\ V(i) \end{bmatrix} + \begin{bmatrix} \Delta\delta(i) \\ \Delta V(i) \end{bmatrix}$$

5) Goto 1.

Calc Q at each voltage controlled bus.

- Compare to min/max limits
- If limit exceeded, set Q to limit & change to PQ bus.

~~Check Q against lim~~

- Check Q on later iterations to see if PQ can be changed back to PV bus.