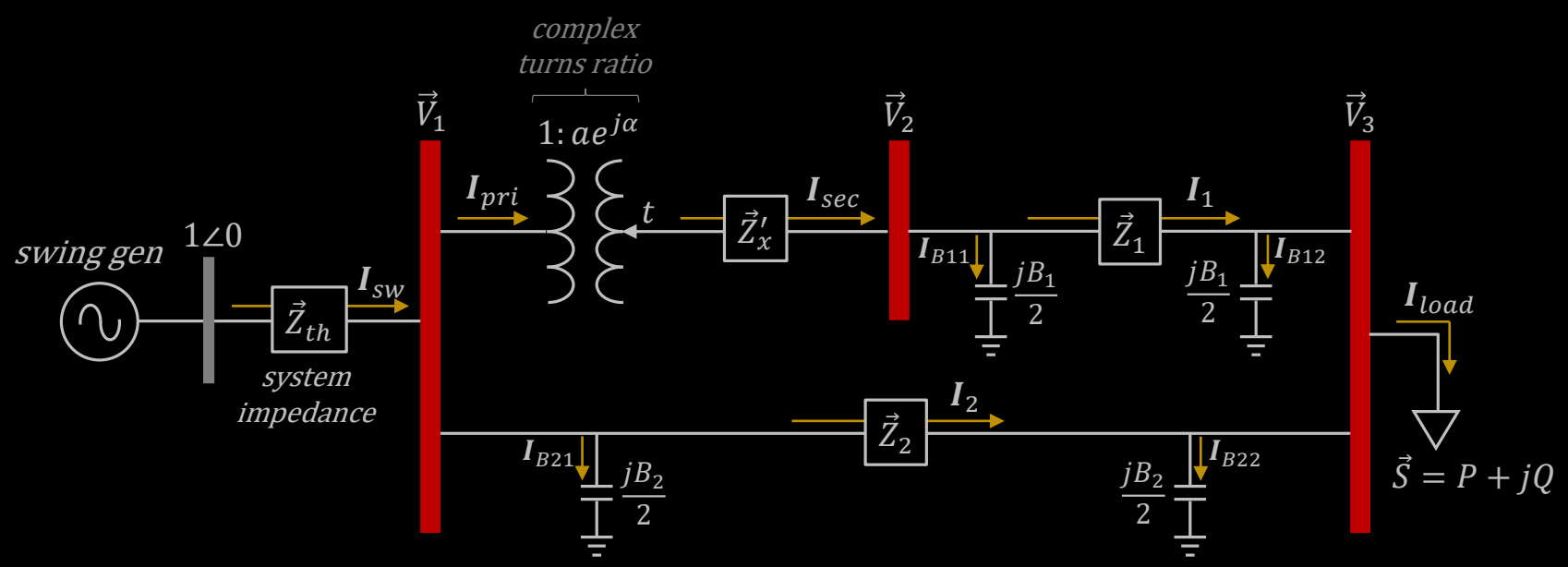
Phase Shifting Xfmr Power Flow

all values per-unit



$\alpha = pst\ phase\ shift\ angle$

$a = \sqrt{1 + [\tan\alpha]^2} = \frac{|V_{sec}|}{|V_{pri}|}$

if symmetrical pst...

$a = 1$

$t = 1 + \tan\alpha = tap\ setting$

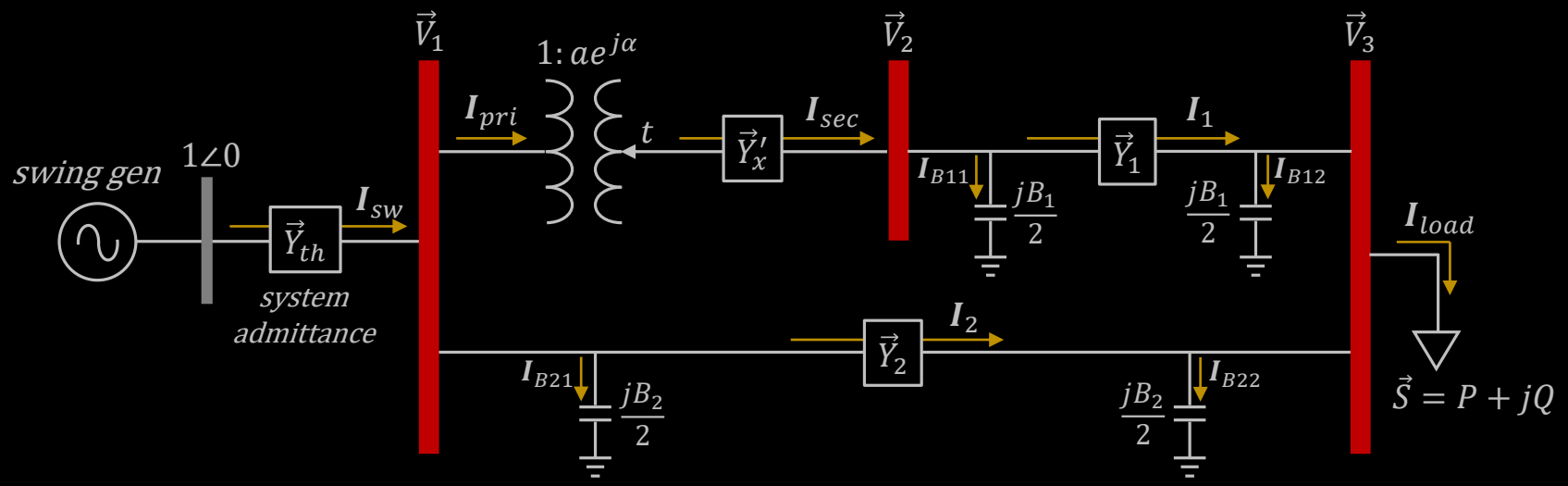
$\vec{Z}'_x = t^2 \vec{Z}_x$

KCL ...

Bus1	*Bus2*	*Bus3*
$I_{sw} = I_{pri} + I_{B21} + I_2$	$I_{sec} = I_{B11} + I_1$	$I_{load} = I_1 - I_{B12} + I_2 - I_{B22} = \frac{\vec{S}^*}{\vec{V}_3^*}$

Phase Shifting Xfmr Power Flow (cont.)

convert impedances to admittances



first look at pst current

$$\vec{V}_1 \boldsymbol{I}_{pri}^* = ae^{j\alpha}\vec{V}_1 \boldsymbol{I}_{sec}^*$$

$$\boldsymbol{I}_{pri} = ae^{-j\alpha}\boldsymbol{I}_{sec}$$

$$\boldsymbol{I}_{sec} = \left[ae^{j\alpha}\vec{V}_1 - \vec{V}_2\right]\vec{Y}_x'$$

$$\boldsymbol{I}_{pri} = ae^{-j\alpha}\left[ae^{j\alpha}\vec{V}_1 - \vec{V}_2\right]\vec{Y}_x'$$

$$\boldsymbol{I}_{pri} = \left[a^2\vec{V}_1 - ae^{-j\alpha}\vec{V}_2\right]\vec{Y}_x'$$

initially guess that...

$$\vec{V}_1 = \vec{V}_2 = \vec{V}_3 = 1\angle 0$$

then use...

$\vec{V}_2$ *and* $\vec{V}_3$

to get new $\vec{V}_1$ $\longrightarrow$

then iterate with Gauss-Seidel method

KCL @ Bus1

$$\boldsymbol{I}_{sw} = \boldsymbol{I}_{pri} + \boldsymbol{I}_{B21} + \boldsymbol{I}_2$$

$$\left[1 - \vec{V}_1\right]\vec{Y}_{th} = \left[a^2\vec{V}_1 - ae^{-j\alpha}\vec{V}_2\right]\vec{Y}_x' + \frac{jB_2}{2}\vec{V}_1 + \left[\vec{V}_1 - \vec{V}_3\right]\vec{Y}_2$$

$$\vec{Y}_{th} - \vec{V}_1\vec{Y}_{th} = a^2\vec{V}_1\vec{Y}_x' - ae^{-j\alpha}\vec{V}_2\vec{Y}_x' + \frac{jB_2}{2}\vec{V}_1 + \vec{V}_1\vec{Y}_2 - \vec{V}_3\vec{Y}_2$$

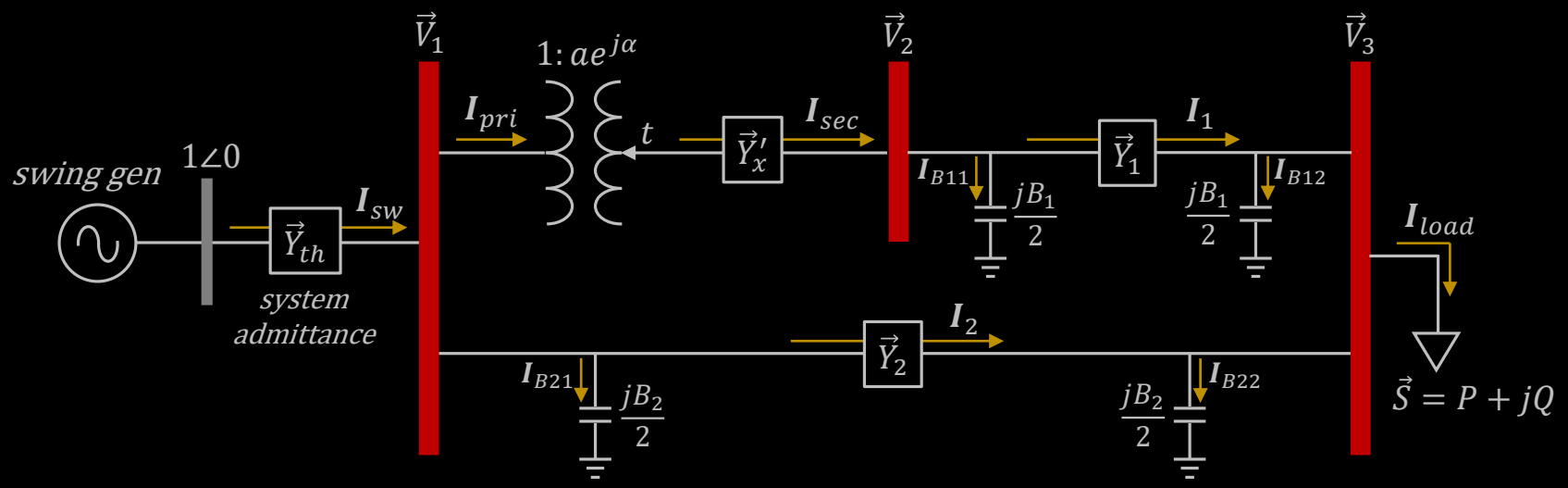
$$-\vec{V}_1\vec{Y}_{th} - \vec{V}_1\vec{Y}_2 - a^2\vec{V}_1\vec{Y}_x' - \frac{jB_2}{2}\vec{V}_1 = -ae^{-j\alpha}\vec{V}_2\vec{Y}_x' - \vec{V}_3\vec{Y}_2 - \vec{Y}_{th}$$

$$\vec{V}_1\vec{Y}_{th} + \vec{V}_1\vec{Y}_2 + a^2\vec{V}_1\vec{Y}_x' + \frac{jB_2}{2}\vec{V}_1 = ae^{-j\alpha}\vec{V}_2\vec{Y}_x' + \vec{V}_3\vec{Y}_2 + \vec{Y}_{th}$$

$$\vec{V}_1\left[\vec{Y}_{th} + \vec{Y}_2 + a^2\vec{Y}_x' + \frac{jB_2}{2}\right] = ae^{-j\alpha}\vec{V}_2\vec{Y}_x' + \vec{V}_3\vec{Y}_2 + \vec{Y}_{th}$$

$$\boxed{\vec{V}_1 = \frac{ae^{-j\alpha}\vec{V}_2\vec{Y}_x' + \vec{V}_3\vec{Y}_2 + \vec{Y}_{th}}{\vec{Y}_{th} + \vec{Y}_2 + a^2\vec{Y}_x' + \frac{jB_2}{2}}}$$

Phase Shifting Xfmr Power Flow (cont.)



KCL @ Bus2

$$\boldsymbol{I}_{sec} = \boldsymbol{I}_{B11} + \boldsymbol{I}_1$$

$$\left[ae^{j\alpha}\vec{V}_1 - \vec{V}_2\right]\vec{Y}_x' = \frac{jB_1}{2}\vec{V}_2 + \left[\vec{V}_2 - \vec{V}_3\right]\vec{Y}_1$$

$$ae^{j\alpha}\vec{V}_1\vec{Y}_x' - \vec{V}_2\vec{Y}_x' = \frac{jB_1}{2}\vec{V}_2 + \vec{V}_2\vec{Y}_1 - \vec{V}_3\vec{Y}_1$$

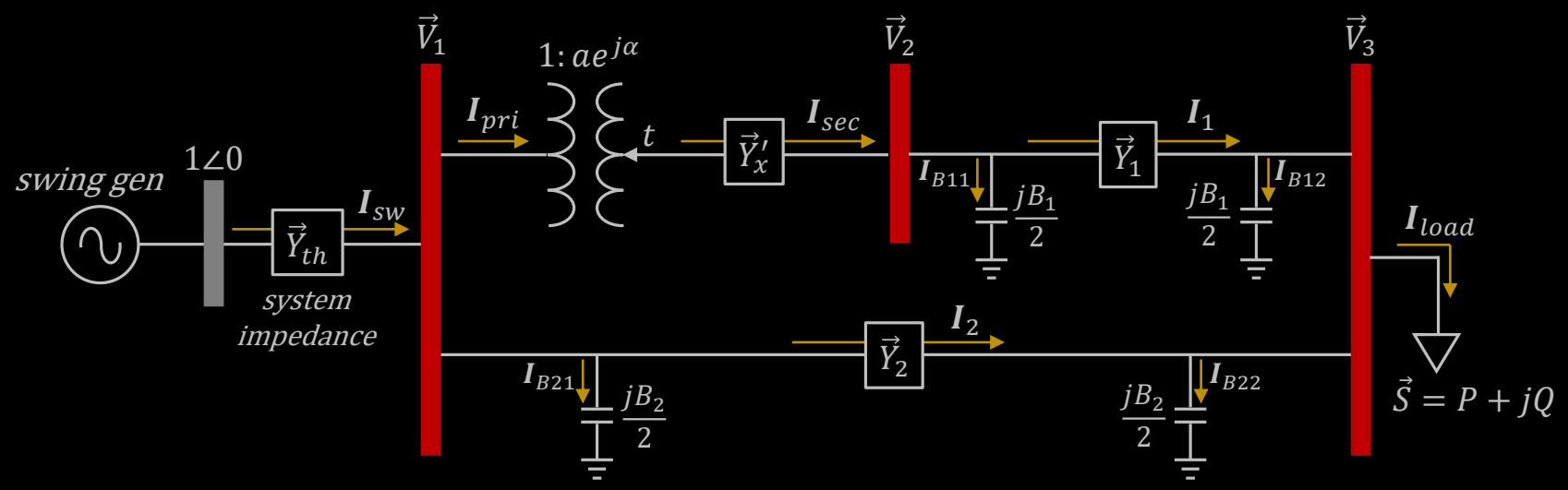
$$-\vec{V}_2\vec{Y}_x' - \vec{V}_2\vec{Y}_1 - \frac{jB_1}{2}\vec{V}_2 = -\vec{V}_3\vec{Y}_1 - ae^{j\alpha}\vec{V}_1\vec{Y}_x'$$

$$\vec{V}_2\left[\vec{Y}_x' + \vec{Y}_1 + \frac{jB_1}{2}\right] = \vec{V}_3\vec{Y}_1 + ae^{j\alpha}\vec{V}_1\vec{Y}_x'$$

use new $\vec{V}_1$ and last known $\vec{V}_3$ to get new $\vec{V}_2$ →

$$\boxed{\vec{V}_2 = \frac{\vec{V}_3\vec{Y}_1 + ae^{j\alpha}\vec{V}_1\vec{Y}_x'}{\vec{Y}_x' + \vec{Y}_1 + \frac{jB_1}{2}}}$$

Phase Shifting Xfmr Power Flow (cont.)



KCL @ Bus3

$$I_1 - I_{B12} + I_2 - I_{B22} = \frac{\vec{S}^*}{\vec{V}_3^*}$$

$$[\vec{V}_2 - \vec{V}_3]\vec{Y}_1 - \frac{jB_1}{2}\vec{V}_3 + [\vec{V}_1 - \vec{V}_3]\vec{Y}_2 - \frac{jB_2}{2}\vec{V}_3 = \frac{\vec{S}^*}{\vec{V}_3^*}$$

$$\vec{V}_2\vec{Y}_1 - \vec{V}_3\vec{Y}_1 - \frac{jB_1}{2}\vec{V}_3 + \vec{V}_1\vec{Y}_2 - \vec{V}_3\vec{Y}_2 - \frac{jB_2}{2}\vec{V}_3 = \frac{\vec{S}^*}{\vec{V}_3^*}$$

$$-\vec{V}_3\vec{Y}_1 - \vec{V}_3\vec{Y}_2 - \frac{jB_1}{2}\vec{V}_3 - \frac{jB_2}{2}\vec{V}_3 = \frac{\vec{S}^*}{\vec{V}_3^*} - \vec{V}_1\vec{Y}_2 - \vec{V}_2\vec{Y}_1$$

$$\vec{V}_3\vec{Y}_1 + \vec{V}_3\vec{Y}_2 + \frac{jB_1}{2}\vec{V}_3 + \frac{jB_2}{2}\vec{V}_3 = \frac{-\vec{S}^*}{\vec{V}_3^*} + \vec{V}_1\vec{Y}_2 + \vec{V}_2\vec{Y}_1$$

$$\vec{V}_3\left[\vec{Y}_1 + \vec{Y}_2 + \frac{jB_1}{2} + \frac{jB_2}{2}\right] = \frac{-\vec{S}^*}{\underbrace{\vec{V}_3^*}_{\text{leave conjugate here}}} + \vec{V}_1\vec{Y}_2 + \vec{V}_2\vec{Y}_1$$

use new $\vec{V}_1$ and $\vec{V}_2$ and last known $\vec{V}_3$ to get new $\vec{V}_3$

$$\boxed{\vec{V}_3 = \left[\frac{-\vec{S}^*}{\vec{V}_3^*} + \vec{V}_1\vec{Y}_2 + \vec{V}_2\vec{Y}_1\right]\left[\frac{1}{\vec{Y}_1 + \vec{Y}_2 + \frac{jB_1}{2} + \frac{jB_2}{2}}\right]}$$

pseudo code for pst power flow analysis

```
given: alpha_rad, yth, yx, y1, y2, b1, b2, s                # yth, yx, y1, y2, b1, b2, s (complex)
v1 = v2 = v3 = 1<0                                          # first guess (complex)
tol = 1e-6                                                  # assign convergence tolerance
iteration = 0                                               # initialize number of iterations
max_iterations = 99                                         # assign max iterations
converged = False                                           # set converged flag = False
while not converged and iteration < max_iterations:         # loop while not converged
    converged = True                                        # assume converged
    iteration += 1                                          # increment iteration
    new_v1 = f(v2,v3)                                       # get new v1
    new_v2 = f(new_v1,v3)                                   # get new v2
    new_v3 = f(new_v1, new_v2)                              # get new v3
    delta_v1 = abs(new_v1 - v1)                             # get magnitude of v1 change
    delta_v2 = abs(new_v2 - v2)                             # get magnitude of v2 change
    delta_v3 = abs(new_v3 - v3)                             # get magnitude of v3 change
    v1 = new_v1                                             # assign new_v1 to v1
    v2 = new_v2                                             # assign new_v2 to v2
    v3 = new_v3                                             # assign new_v3 to v3
    if delta_v1 > tol:                                      # if v1 change > tolerance…
        converged = False                                   # set convergence flag = False
    elif delta_v2 > tol:                                    # if v2 change > tolerance …
        converged = False                                   # set convergence flag = False
    elif delta_v3 > tol:                                    # if v3 change > tolerance …
        converged = False                                   # set convergence flag = False
```

```
if converged:
all bus voltage magnitudes and angles are known…
everything else can be calculated
```

ЗФЕЕ

Dedicated to Power Engineering

Questions or Comments …

contact us