

Fast Power Flow Methods

1.0 Introduction

What we have learned so far is the so-called “full-Newton-Raphson” (NR) power flow algorithm. The NR algorithm is perhaps the most robust algorithm in the sense that it is most likely to obtain a solution for “tough” problems, which are problems that start from guesses that are not close to their solution. For example, solving a large power flow case from a “flat” start is usually considered to be a “tough” problem, and as a result, it is best to do that with a NR. But NR is slow!

Often, the problem is not so “tough,” and in that case, the so-called fast decoupled (FDC) algorithm is also effective in getting the solution, there is no loss of accuracy, *and* it is much faster. A common situation where FDC is attractive is when you have solved the case, and then you want to re-solve the case (using a “hot” start) to analyze the effect of some not-so-dramatic change. Here, the fact that the problem is not so “tough” calls for relaxing solution algorithm robustness.

There are situations where speed is paramount, but accuracy is not. For example, in on-line analysis of 50,000 contingencies, we may want to only filter the contingencies that have potential to result in problems, and then perform full analysis on those. In such cases, the DC power flow is appropriate. Although DC power flow is fast and robust, it is not very accurate.

Solving the power flow equations can be computationally intensive. In these notes, we review FDC and DC power flow methods. You can see the relation between the NR and these two in Table 1, in terms of speed, accuracy, and solution robustness.

Table 1

| Solution method | Speed | Accuracy | Solution robustness |
|-----------------|-----------|-------------|---------------------|
| Newton-Raphson | Slow | Accurate | Robust |
| Fast decoupled | Fast | Accurate | Less robust |
| DC | Very fast | Approximate | Robust |

2.0 The fast decoupled power flow

The Jacobian matrix,

$$\underline{J} = \begin{bmatrix} \underline{J}^{P\theta} & \underline{J}^{PV} \\ \underline{J}^{Q\theta} & \underline{J}^{QV} \end{bmatrix} \quad (1)$$

has a special characteristic in that the elements of the off-diagonal submatrices $\underline{J}^{Q\theta}$ and \underline{J}^{PV} are usually very small relative to the elements of the diagonal submatrices $\underline{J}^{P\theta}$ and \underline{J}^{QV} (In fact, we have seen that in the first iteration of a flat start, when we assume all angles are 0, the elements of the off-diagonal matrices are all 0.) This is because every term in the expressions of the off-diagonal blocks are (a) multiplied by a “G” or (b) multiplied by a “sin”. The overall terms are small because, for transmission:

- conductance G tends to be small and
- angular differences across circuits tend to be small, resulting in small “sin” terms.

These observations are consistent with our understanding that P is not very sensitive to voltage

magnitude (i.e., small $\frac{\partial P_j(\underline{x})}{\partial |V_k|}$ and $\frac{\partial P_j(\underline{x})}{\partial |V_j|}$), and Q is

not very sensitive to angle (i.e., small $\frac{\partial Q_j(\underline{x})}{\partial \theta_k}$ and

$\frac{\partial Q_j(\underline{x})}{\partial \theta_k}$).

We can take advantage of these observations in the following way. Instead of using the exact Jacobian, *let's assume that all elements of the off-diagonal submatrices are in fact 0 and remain 0 throughout the entire NR algorithm.* In other words, let's just use the following Jacobian:

$$\underline{J} = \begin{bmatrix} \underline{J}^{P\theta} & \underline{0} \\ \underline{0}^T & \underline{J}^{QV} \end{bmatrix} \quad (2)$$

Note what this does to our update equation:

$$\underline{J}^{(j)} \Delta \underline{x}^{(j)} = - \begin{bmatrix} \underline{\Delta P} \\ \underline{\Delta Q} \end{bmatrix} \quad (3)$$

Substituting (2) into (3), we have:

$$\begin{bmatrix} \underline{J}^{P\theta} & \underline{0} \\ \underline{0}^T & \underline{J}^{QV} \end{bmatrix}^{(j)} \begin{bmatrix} \underline{\Delta \theta} \\ \underline{\Delta |V|} \end{bmatrix}^{(j)} = - \begin{bmatrix} \underline{\Delta P} \\ \underline{\Delta Q} \end{bmatrix}^{(j)} \quad (4)$$

Performing the indicated matrix multiplication, we obtain:

$$\underline{J}^{P\theta(j)} \underline{\Delta \theta}^{(j)} = -\underline{\Delta P}^{(j)} \quad (5)$$

$$\underline{J}^{QV(j)} \underline{\Delta |V|}^{(j)} = -\underline{\Delta Q}^{(j)} \quad (6)$$

Equations (5) and (6) have the following remarkable feature: *real power equations are decoupled from the voltage magnitudes, and reactive power equations are decoupled from the angles.* The implication is that either one of equations (5) and (6) may be solved independent of the other one!

Our power flow algorithm remains exactly as it was before, with the only exception being in Step 4.

1. Specify:
 - All admittance data (series Y, charging capacitance, transformer taps, & shunts)
 - P_d and Q_d for **all** buses (whether PV, PQ, or swing)
 - P_g and $|V|$ for all PV buses
 - $|V|$ for swing bus, with $\theta=0^\circ$
2. Set the iteration counter $j=0$. Use one of the following to guess the initial solution.
 - Flat Start: $V_k=1.0 \angle 0^\circ$ for all buses.
 - Hot Start: Use the solution to a previously solved case for this network.
3. Compute the mismatch vector for $\underline{x}^{(j)}$, denoted as $\underline{f}(\underline{x})$. In what follows, we denote elements of the mismatch vector as ΔP_k and ΔQ_k corresponding to the real and reactive power mismatch, respectively, for the k^{th} bus (which would not be the k^{th} element of the mismatch vector for two reasons: one reason pertains to the swing bus and the other reason to the fact that for type PQ buses, there are two equations per bus and not one). This computation will also result in all necessary calculated real and reactive power injections. Perform the following stopping criterion tests:
 - If $|\Delta P_k| < \epsilon_P$ for all type PQ & PV buses and
 - If $|\Delta Q_k| < \epsilon_Q$ for all type PQ buses,
 - Then go to step 5
 - Else
 - Go to step 4.
4. Find an improved solution as follows:
 - Evaluate the Jacobian \underline{J} at $\underline{x}^{(j)}$. Denote this Jacobian as $\underline{J}^{(j)}$
 - Solve for $\Delta \underline{x}^{(j)}$ by applying LU decomposition to:

This is only change in algorithm!!

$$\underline{J}^{P\theta(j)} \Delta \underline{\theta}^{(j)} = -\Delta \underline{P}^{(j)}$$

$$\underline{J}^{QV(j)} \Delta |V|^{(j)} = -\Delta \underline{Q}^{(j)}$$

- Compute the updated solution vector as $\underline{x}^{(j+1)} = \underline{x}^{(j)} + \Delta \underline{x}^{(j)}$.
 - Return to step 3 with $j=j+1$.
5. Stop.

How to see that the FDC algorithm is *faster* than NR?

- In NR, step 4 computes

$$\underline{J}^{(j)} \underline{\Delta x}^{(j)} = - \begin{bmatrix} \underline{\Delta P} \\ \underline{\Delta Q} \end{bmatrix}$$

The Jacobian has dimension $(2N-1-N_G)$.

- In FDC, $\underline{J}^{P\theta}$ has dimension $(N-1)$, and \underline{J}^{QV} has dimension $(N-N_G)$.

For example, if we have 20000 buses and 2000 generators, then the Jacobian in the NR has dimension of 37,999, but the FDC algorithm Jacobians will have dimensions of 19,999 and 18,000, respectively.

It is known that the speed of LU decomposition is a function, approximately linear, of the number of elements. The number of elements in NR is $(37,999)^2=1.44E9$, whereas in FDC it is $(19,999)^2+(18,000)^2=7.24E8$. Therefore, FDC will be about twice as fast per iteration as NR.

However, because the Jacobian gives the “direction” to move the solution in each iteration, **we do suffer a loss in accuracy per iteration**, and therefore we may need more iterations to obtain the final solution.

Given these two opposing forces (less time per iteration and more iterations), it is *usually* the case that FDC is between 1.5 and 2 times faster than NR.

But what about accuracy? We have said that the FDC algorithm will be less accurate per iteration. Does that imply that it will provide a less accurate solution once it stops iterating?

The answer to this question depends on the stopping criterion. Note that in the above FDC algorithm, the stopping criterion is given in Step 3, and *it is exactly the same as the stopping criterion used in the NR*. That is, both algorithms are computing the mismatch as $P_k(\underline{x}) - P_k$ and $Q_k(\underline{x}) - Q_k$, and $P_k(\underline{x})$ and $Q_k(\underline{x})$ are computed with the full real and reactive power flow equations, respectively, in both algorithms.

It is very important to recognize that *the approximation in FDC algorithm is applied to the Jacobian matrix but NOT the power flow equations used to compute the elements of the mismatch vector*.

The conclusion that we can make here is that **A POWER FLOW SOLUTION OBTAINED BY FDC IS JUST AS ACCURATE AS A POWER FLOW SOLUTION OBTAINED BY NR.**

3.0 FDC algorithm: enhancements

We may simplify the FDC algorithm still further, making it still faster (but less accurate) per iteration by working with the expressions of the Jacobian elements for $\underline{J}^{P\theta}$ and \underline{J}^{QV} .

Consider the terms $J_{jk}^{P\theta}$. If we neglect G and under small angle approximation (so that $\sin(\theta_j - \theta_k) \approx 0$ and $\cos(\theta_j - \theta_k) \approx 1$):

$$\begin{aligned} J_{jk}^{P\theta} &= \frac{\partial P_j(\underline{x})}{\partial \theta_k} = |V_j| |V_k| (G_{jk} \sin(\theta_j - \theta_k) - B_{jk} \cos(\theta_j - \theta_k)) \\ &= -|V_j| |V_k| B_{jk} \end{aligned} \quad (7)$$

Now consider the terms $J_{jj}^{P\theta}$.

$$\begin{aligned} J_{jj}^{P\theta} &= \frac{\partial P_j(\underline{x})}{\partial \theta_j} = -Q_j(\underline{x}) - B_{jj} |V_j|^2 \\ &= \left\{ - \sum_{k=1}^N |V_j| |V_k| (G_{jk} \sin(\theta_j - \theta_k) - B_{jk} \cos(\theta_j - \theta_k)) \right\} - B_{jj} |V_j|^2 \end{aligned} \quad (8)$$

Again, using small G_{jk} and small angle approximation, the above is

$$J_{jj}^{P\theta} = \frac{\partial P_j(\underline{x})}{\partial \theta_j} = \left\{ \sum_{k=1}^N |V_j| |V_k| B_{jk} \right\} - B_{jj} |V_j|^2 \quad (9)$$

We will now make use of an assumption that the voltage profile is “flat,” i.e., $|V_k| = |V_j|$. Then (9) becomes:

$$\begin{aligned}
J_{jj}^{P\theta} &= \frac{\partial P_j(\underline{x})}{\partial \theta_j} = \left\{ \sum_{k=1}^N |V_j| |V_j| B_{jk} \right\} - B_{jj} |V_j|^2 \\
&= \left\{ \sum_{k=1}^N |V_j|^2 B_{jk} \right\} - |V_j|^2 B_{jj} = |V_j|^2 \left[\left\{ \sum_{k=1}^N B_{jk} \right\} - B_{jj} \right] \quad (10)
\end{aligned}$$

Now consider the summation in the curly brackets.

$$\sum_{k=1}^N B_{jk} = B_{j1} + B_{j2} + \dots + B_{j,j-1} + B_{jj} + B_{j,j+1} + \dots + B_{jN} \quad (11)$$

Recall that from definition of Y-bus elements:

- $k \neq j$: $B_{jk} = -b_{jk} \rightarrow b_{jk} = -B_{jk}$
- $k = j$:

$$B_{jj} = b_{j1} + b_{j2} + \dots + b_{j,j-1} + b_j + b_{j,j+1} + \dots + b_{jN}$$

and using the relation from the first bullet:

$$= -B_{j1} - B_{j2} - \dots - B_{j,j-1} + b_j - B_{j,j+1} - \dots - B_{jN} \quad (12)$$

where b_j is sum of all shunt susceptance at bus j .

Substituting this last expression (12) for B_{jj} into (11), we obtain:

$$\begin{aligned}
\sum_{k=1}^N B_{jk} &= B_{j1} + B_{j2} + \dots + B_{j,j-1} \\
&\quad + \left(-B_{j1} - B_{j2} - \dots - B_{j,j-1} + b_j - B_{j,j+1} - \dots - B_{jN} \right) \\
&\quad + B_{j,j+1} + \dots + B_{jN} \quad (13) \\
&= b_j
\end{aligned}$$

Substituting (13) into (10), we obtain:

$$J_{jj}^{P\theta} = \frac{\partial P_j(\underline{x})}{\partial \theta_j} = |V_j|^2 [b_j - B_{jj}] \quad (14)$$

We could perform the subtraction in (14) using (12) to see that the term is just the negative of the sum of all non-shunt branches connected to bus j .

However, b_j is typically very small compared to B_{jj} so that neglecting b_j is quite accurate, resulting in:

$$J_{jj}^{P\theta} = \frac{\partial P_j(\underline{x})}{\partial \theta_j} = -|V_j|^2 B_{jj} \quad (15)$$

Likewise, under assumptions of flat voltage profile and small angle, we can show that:

$$\begin{aligned} J_{jk}^{QV} &= \frac{\partial Q_j(\underline{x})}{\partial |V_k|} = |V_j| (G_{jk} \sin(\theta_j - \theta_k) - B_{jk} \cos(\theta_j - \theta_k)) \\ &= -|V_j| B_{jk} \end{aligned} \quad (16)$$

$$\begin{aligned} J_{jj}^{QV} &= \frac{\partial Q_j(\underline{x})}{\partial |V_j|} = \frac{Q_j(\underline{x})}{|V_j|} - B_{jj} |V_j| \\ &= -|V_j| B_{jj} \end{aligned} \quad (17)$$

Summarizing eqs. (7), (15), (16), and (17), we have:

$$J_{jk}^{P\theta} = \frac{\partial P_j(\underline{x})}{\partial \theta_k} = -|V_j| |V_k| B_{jk} \quad (7)$$

$$J_{jj}^{P\theta} = \frac{\partial P_j(\underline{x})}{\partial \theta_j} = -|V_j|^2 B_{jj} \quad (15)$$

$$J_{jk}^{QV} = \frac{\partial Q_j(\underline{x})}{\partial |V_k|} = -|V_j| B_{jk} \quad (16)$$

$$J_{jj}^{QV} = \frac{\partial Q_j(\underline{x})}{\partial |V_j|} = -|V_j| B_{jj} \quad (17)$$

Noting that the Jacobian matrix has no equations or variables for bus 1 (the swing bus), we define the B' matrix as:

$$B' = \begin{bmatrix} B_{22} & B_{23} & \dots & B_{2n} \\ B_{32} & B_{33} & \dots & B_{3n} \\ \vdots & \vdots & \dots & \vdots \\ B_{n2} & B_{n3} & \dots & B_{nn} \end{bmatrix} \quad (18)$$

where this matrix may be obtained from the Y-bus by simply stripping off the first row and first column (assuming the swing bus is #1) and taking the imaginary part of all elements. This matrix is appropriate for writing the $J^{P\theta}$ terms of (7) and (15) in compact notation, as given by (19) below:

$$\underline{J}^{P\theta} = -[\underline{V}]B'[\underline{V}] \quad (19)$$

where

$$[\underline{V}] = \begin{bmatrix} |V_2| & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & |V_3| & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & |V_n| \end{bmatrix} \quad (20)$$

Observe that B' is pre-multiplied and post-multiplied by [V] to account for the product of two voltages in (7) and (15).

Now, regarding the J^{QV} terms...

If we could assume that we would have reactive power flow equations for all buses in the network, then we would use B' for the J^{QV} terms as well. But we do not have reactive power flow equations for the PV buses, only for the PQ buses.

To account for this, we need to eliminate the rows and columns corresponding to type PV buses from B' and from $[V]$.

Using the numbering scheme $1, \dots, N_G$, as being the voltage control buses, and bus N_G+1, \dots, N as being the type PQ buses, then eliminate row and column number $1, \dots, N_G-1$ from B' and $[V]$.

Let's refer to the resulting matrices as B'' and $[V'']$. Then (16) and (17) become:

$$\underline{J}^{QV} = -[\underline{V}'']\underline{B}'' \quad (21)$$

Summarizing, our decoupled pf equations are:

$$\underline{J}^{P\theta} = -[\underline{V}]\underline{B}'[\underline{V}] \quad (19)$$

$$\underline{J}^{QV} = -[\underline{V}'']\underline{B}'' \quad (21)$$

Now let's look at our correction formula...

Recalling eqs. (5) and (6):

$$\underline{J}^{P\theta(j)} \Delta \underline{\theta}^{(j)} = -\Delta \underline{P}^{(j)} \quad (5)$$

$$\underline{J}^{QV(j)} \Delta |\underline{V}|^{(j)} = -\Delta \underline{Q}^{(j)} \quad (6)$$

and substituting (19, 21) into (5-6), we obtain:

$$-\underline{[V]} \underline{[B']} \underline{[V]} \Delta \underline{\theta}^{(j)} = -\Delta \underline{P}^{(j)} \quad (22)$$

$$-\underline{[V'']} \underline{[B'']} \Delta |\underline{V}|^{(j)} = -\Delta \underline{Q}^{(j)} \quad (23)$$

Multiplying both sides by -1 results in:

$$\underline{[V]} \underline{[B']} \underline{[V]} \Delta \underline{\theta}^{(j)} = \Delta \underline{P}^{(j)} \quad (24)$$

$$\underline{[V'']} \underline{[B'']} \Delta |\underline{V}|^{(j)} = \Delta \underline{Q}^{(j)} \quad (25)$$

There are two more changes which prove useful in terms of capturing additional computational efficiency (more speed).

The first change is an approximation: let the second $\underline{[V]}$ in eq. (24) be the identity matrix based on the assumption that we have a “flat” voltage profile (all the same) and that all voltages are close to 1.0.

Making this change in eq. (24), our correction equations become:

$$\underline{[V]} \underline{[B']} \Delta \underline{\theta}^{(j)} = \Delta \underline{P}^{(j)} \quad (26)$$

$$\underline{[V'']} \underline{[B'']} \Delta |\underline{V}|^{(j)} = \Delta \underline{Q}^{(j)} \quad (25)$$

The second change is to pre-multiply (26) by $[\underline{V}]^{-1}$ and (25) by $[\underline{V}'']^{-1}$. This change results in

$$\underline{B}' \Delta \underline{\theta}^{(j)} = [\underline{V}]^{-1} \Delta \underline{P}^{(j)} \quad (27)$$

$$\underline{B}'' \Delta |\underline{V}|^{(j)} = [\underline{V}'']^{-1} \Delta \underline{Q}^{(j)} \quad (28)$$

Considering (27), since $[\underline{V}]$ is diagonal, $[\underline{V}]^{-1}$ is

$$[\underline{V}]^{-1} = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \dots & \mathbf{0} \\ \frac{1}{|\underline{V}_2|} & & & \\ \vdots & \mathbf{1} & \vdots & \vdots \\ \vdots & \frac{1}{|\underline{V}_3|} & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{1} \\ & & & \frac{1}{|\underline{V}_n|} \end{bmatrix} \quad (29)$$

Multiplication of (29) by the real power mismatch vector gives the right-hand-side of (27):

$$\begin{bmatrix} \mathbf{1} & \mathbf{0} & \dots & \mathbf{0} \\ \frac{1}{|\underline{V}_2|} & & & \\ \vdots & \mathbf{1} & \vdots & \vdots \\ \vdots & \frac{1}{|\underline{V}_3|} & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{1} \\ & & & \frac{1}{|\underline{V}_n|} \end{bmatrix} \begin{bmatrix} \Delta P_2 \\ \Delta P_3 \\ \vdots \\ \Delta P_n \end{bmatrix} = \begin{bmatrix} \frac{\Delta P_2}{|\underline{V}_2|} \\ \frac{\Delta P_3}{|\underline{V}_3|} \\ \vdots \\ \frac{\Delta P_n}{|\underline{V}_n|} \end{bmatrix} \quad (30)$$

A similar thing can be done for the reactive power correction equation (28):

$$[V'']^{-1} = \begin{bmatrix} \frac{1}{|V_{N_g+1}|} & 0 & \dots & 0 \\ \vdots & \frac{1}{|V_{N_g+2}|} & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{|V_n|} \end{bmatrix} \quad (31)$$

Multiplication of (31) by the reactive power mismatch vector gives the right-hand-side of (28):

$$\begin{bmatrix} \frac{1}{|V_{N_g+1}|} & 0 & \dots & 0 \\ \vdots & \frac{1}{|V_{N_g+2}|} & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{|V_n|} \end{bmatrix} \begin{bmatrix} \Delta Q_{N_g+1} \\ \Delta Q_{N_g+2} \\ \vdots \\ \Delta Q_n \end{bmatrix} = \begin{bmatrix} \frac{\Delta Q_{N_g+1}}{|V_{N_g+1}|} \\ \frac{\Delta Q_{N_g+2}}{|V_{N_g+2}|} \\ \vdots \\ \frac{\Delta Q_n}{|V_n|} \end{bmatrix} \quad (32)$$

Based on (30) and (32), eqs. (27) and (28) become:

$$\underline{B}'\underline{\Delta\theta}^{(j)} = \begin{bmatrix} \frac{\Delta P_2}{|V_2|} \\ \frac{\Delta P_3}{|V_3|} \\ \vdots \\ \frac{\Delta P_n}{|V_n|} \end{bmatrix}^{(j)} = \Delta \underline{\tilde{P}} \quad (33)$$

$$\underline{B}''\underline{\Delta|V''|}^{(j)} = \begin{bmatrix} \frac{\Delta Q_{N_g+1}}{|V_{N_g+1}|} \\ \frac{\Delta Q_{N_g+2}}{|V_{N_g+2}|} \\ \vdots \\ \frac{\Delta Q_n}{|V_n|} \end{bmatrix}^{(j)} = \Delta \underline{\tilde{Q}} \quad (34)$$

where the notation of the far right-hand-side in (33) and (34) indicates the right-hand-sides of (31) and (32).

Two comments remain:

1. Where's the speed-up? We still retain the speed up of the previous FDC algorithm, which is due to the fact that the LU-decomposition is faster per iteration as a result of the decoupling (and corresponding reduction in total matrix elements). The method described here provides additional speed-up from two sources:

- The B' -matrix need not be reevaluated in each iteration, and the B'' matrix is formed by simply deleting appropriate rows and columns from B' , and so we save the time of evaluating Jacobian matrix elements.
- Because the left-hand-side of eq. (33) is constant, we need perform LU-decomposition for this equation only once. Given the L and U factors, we need to only perform forward and backward substitution for each different right-hand-side. We are not quite as fortunate with the reactive power correction equation, (34), because there we must re-factorize each time the list of voltage control buses changes.

2. Algorithm: I indicated the power flow algorithm is exactly the same as in the NR, but there is a minor difference in that Steps 3 and 4 (page 5 above) can be alternated, as follows:

- a. Step 3a: Compute mismatch of $\Delta \underline{\tilde{P}}$ using $\Delta \underline{\theta}^{(j-1)}$ and $\Delta |\underline{V}|^{(j-1)}$.
- b. Step 4a: Solve eq. (33) for $\Delta \underline{\theta}^{(j)}$.
- c. Step 3b: Compute mismatch of $\Delta \underline{\tilde{Q}}$ using $\Delta \underline{\theta}^{(j)}$ and $\Delta |\underline{V}|^{(j-1)}$.
- d. Step 4b: Solve eq. (34) for $\Delta |\underline{V}|^{(j)}$.
- e. Step 3c: Perform stopping criterion tests:
 If $|\Delta P_k| < \varepsilon_P$ for all type PQ & PV buses and
 If $|\Delta Q_k| < \varepsilon_Q$ for all type PQ buses,
 → Then go to step 5
 Else

$$\underline{\theta}^{(j+1)} = \underline{\theta}^{(j)} + \Delta \underline{\theta}^{(j)}$$

$$\underline{V}^{(j+1)} = \underline{V}^{(j)} + \Delta \underline{V}^{(j)}$$

Return to step 3 with $j=j+1$.

The reason why this improves speed is because the update on voltages are done using a “step” based on the most recent update on angles, and this tends to reduce the necessary number of iterations.

4.0 DC Power Flow

Return to equation (33), repeated here for convenience:

$$\underline{B}'\underline{\Delta\theta}^{(j)} = \begin{bmatrix} \frac{\Delta P_2}{|V_2|} \\ \frac{\Delta P_3}{|V_3|} \\ \vdots \\ \frac{\Delta P_n}{|V_n|} \end{bmatrix}^{(j)} = \Delta \underline{\tilde{P}} \quad (33)$$

Now assume all voltages are 1.0. Then eq. (33) becomes:

$$\underline{B}'\underline{\Delta\theta}^{(j)} = \begin{bmatrix} \frac{\Delta P_2}{1.0} \\ \frac{\Delta P_3}{1.0} \\ \vdots \\ \frac{\Delta P_n}{1.0} \end{bmatrix}^{(j)} = \begin{bmatrix} \Delta P_2 \\ \Delta P_3 \\ \vdots \\ \Delta P_n \end{bmatrix}^{(j)} = \Delta \underline{P} \quad (35)$$

So, equation (35) becomes:

$$\underline{B}\underline{\Delta\theta}^{(j)} = \Delta \underline{P} \quad (36)$$

This equation, when solved just once ($j=1$) for $\Delta\underline{\theta}^{(1)}$, and with a “flat-start” solution, implies that $\underline{\theta}=\underline{0}+\Delta\underline{\theta}^{(1)}=\Delta\underline{\theta}^{(1)}$, and, if we assume that this solution is the correct one, then in other words,

$$\underline{B}'\underline{\theta} = \Delta\underline{P} = \underline{P}(\underline{0}) - \underline{P} = -\underline{P} \quad (37)$$

where $\underline{P}(\underline{0})$ are the real power flow equations for buses 2 to N evaluated at $\underline{\theta}^{(0)}=\underline{0}$ (i.e., the flat start), and \underline{P} are the real power flow injections into each bus 2 to N. Therefore,

$$-\underline{B}'\underline{\theta} = \underline{P} \quad (38)$$

This gives all of the angles for the network with a single solution to a set of linear equations. Then, the real power flows can be computed with

$$P_{kj} = B_{kj}(\theta_k - \theta_j) \quad (39)$$

which is the power flowing across a circuit connected between buses k and j under conditions of (a) neglecting resistance, (b) small angle approximation, and (c) all voltage magnitudes are 1.0.