Austin Shaw
Circuit Simulation 18-762
Amrit Pandey
Project 2 Report
4/9/2022

I. Background

The study of power flow analysis is to ensure stability of the network in question. A traditional way to accomplish this is to first find the steady state of the grid. Then, the network can be perturbed, or adjusted, to see the dependencies on specific elements and their location within the network. Of course, there are other analyses that can be performed on the objective grid. Another analysis method could just use the steady state of the grid as an initial value and model the development of the network in time. This analysis is crucial for future grid planning and security of the grid. However, both of these analyses are effectively determining the stability of the network. Stability can be determined by checking that every bus within the network is within their specific bounds. For example, an analyst would confirm the power flow through a transmission line is regulated, power ratings of transformers are maintained in order to avoid potential burn-up, generators do not draw power from the grid, buses in low voltage states do not exist, etc. In order to determine if the grid is within its acceptable limits, two items must be considered. First, the power at every bus is determined, and then bus values are compared against the tolerances at that location.

We will begin with modeling the power at each node, and we will then describe how to implement the models into a mathematical tool that will find a solution. Thus, we start with a positive power flow sequence. It is accepted within literature that positive power flow is defined as:

$$S_{bus} = VI^{*} \qquad \text{Equation 1}$$

where S_{bus} is the total power at the specific bus location, V is the complex voltage at the bus since grid circuits are in the AC domain, and I^{*} is the conjugation of the complex current flowing through the bus. Thus, we arrive at the derivation,

$$P_{bus} + jQ_{bus} = (V_R + V_I)(I_R - jI_I) \qquad \text{Equation 2}$$

where P_{bus} is the real power at the bus, Q_{bus} is the reactive power at the bus, V_R is the real voltage at the bus, V_I is the imaginary voltage at the bus, I_R is the real current flowing through the bus, I_I is the complex current flowing through the bus, and j is the imaginary number. In order to continue the methods implemented in nodal analysis, we can choose to solve Eq. 2 for currents or voltages. Going forward, we will choose currents in order to later apply KCL. Hence, Eq. 2 becomes

$$I_R + jI_I = \frac{P_{bus}+jQ_{bus}}{V_R+jV_I} \qquad \text{Equation 3}$$

However, we can recognize that there will be nonlinear terms within our models, so we will need to apply Newton Raphson to find convergent solutions. The problem arises that these functions are nonanalytic due to the introduction of complex valued terms. To amend this issue, an appropriate solution would be to separate the model across its real-valued and complex-valued terms in order to create functions that will be analytic. Thus,

$$I_R = \frac{P_{bus}V_R + Q_{bus}V_I}{V_R^2 + V_I^2} \qquad \text{Equation 4}$$

$$I_I = \frac{P_{bus}V_I - Q_{bus}V_R}{V_R^2 + V_I^2} \qquad \text{Equation 5}$$

The concept of separating Eq. 3 into Eqs. 4 and 5 will be crucial for the nonlinear models we will build in the following sections. Furthermore, our models will assume that Eqs. 4 and 5 indicate the current entering a bus.

As previously mentioned, Newton Raphson will be implemented within the models to solve the power flow problem and determine a steady-state. However, Newton Raphson by itself is not always guaranteed to converge to a solution, or an appropriate solution for the model. One example of divergence while using Newton Raphson is for the state-variables to oscillate between a set of values. This can happen when a set in the solution space has a recurrent subset of values. In other words, a set where all of the values intercommunicate with one another. If one such value from this subset is used during a Newton iteration, the algorithm will get fixed to a loop and fail to converge. Other issues include needing to initialize the algorithm sufficiently close to the solution, have a continuous space in the neighborhood of the solution, and remaining in the solution space for the problem being solved. There are proposed methods to ease convergence of Newton Raphson, in particular to physically meaningful solutions. We will begin this discussion by describing variable limiting.

Variable limiting is an important method in order to ensure that Newton Raphson remains within the appropriate solution space as long as initialization starts within this solution space. A simple overview of variable limiting would state that the set of variables is damped by controlling the difference between solutions at adjacent Newton Raphson iterations. In other words, a dampening factor, α, is applied to the difference between variables of the current iteration with the previous iteration. Then, the current solution only approaches it original value by a factor of α. As a result, it only seems appropriate to restrict α such that $0 < \alpha < 1$ so variable limiting is indeed achieved. The impact on Newton Raphson means that the number of iterations required to reach a solution is greatly increased and proportional to α. Granted, if enough domain knowledge was used to create a sufficient enough initialization for the initial conditions, a physically meaningful solution could be reached.

Variable limiting does not need to be applied the whole variable set, and it can be applied to just the most sensitive variables. For power flow analysis, bus voltages can converge to either a high voltage or low voltage state. In reality, only the high voltage solution exists, and as a result, the voltages at each bus are sensitive to the implementation of Newton Raphson. To amend this, we can apply the aforementioned α to the voltages to make sure they remain within the appropriate solution space.

One last, yet important, method that will be briefly touched upon is the implementation of homotopy methods. Homotopy methods heavily rely on the use of heuristics in order to remain within the solution space of the problem. The construct of a homotopy method is to relax the complex problem into a problem that has a trivial solution. Then, the trivial solution will be used as the starting point for finding the solution to the "slightly harder" problem. A slightly harder problem is a problem that exists along a path from the trivial solution to the original problem. The term slightly refers to taking a small step towards the original problem from the last problem. This process will be repeated until the original problem has been implemented and a solution has been found. Methods for power systems include source-stepping, power-stepping, and Tx-stepping.

A) PV Buses

First, we will begin with the model of the PV buses. As suggested by the naming scheme, PV buses are buses with independent variables of real power and voltage. These buses are also referred to as generators because their voltage and power output can be fixed within the system. However, the naming scheme may be somewhat misleading. The voltage referred to by the name PV is the "set" voltage which is otherwise represented by

$$V_{set}e^{j\theta_{ref}} = V_R + jV_I. \qquad \text{Equation 6}$$

In Eq. 6, the reference angle, θ_{ref}, is unknown which refrains us from exactly determining V_R or V_I. Furthermore, the reactive power of the generator, Q_{gen}, is unknown as well. For now, we will ignore the fact that θ_{ref} is not an independent variable. Rather, we will amend this issue later because the slack bus model is a continuation of the generator model. Thus, since I by assumption in our models represents the currents entering a bus, the generator will be modeled with $-\ I$ corresponding to the current injected into the grid. Hence, Eqs. 4 and 5 become

$$I_R = \frac{-P_{gen}V_R - Q_{gen}V_I}{V_R^2 + V_I^2}, \qquad \text{Equation 7}$$

$$I_I = \frac{-P_{gen}V_I + Q_{gen}V_R}{V_R^2 + V_I^2}. \qquad \text{Equation 8}$$

Recognizing that Eqs. 7 and 8 are nonlinear means that we cannot directly use a linear solver to determine the real and imaginary currents at a bus that contains a generator. However, we can apply Newton Raphson by finding the partial derivatives of each variable. Hence,

$$\frac{\partial I_R}{\partial V_R} = \frac{P_{gen}(V_R^2 - V_I^2) + 2Q_{gen}V_RV_I}{(V_R^2 + V_I^2)^2}, \qquad \text{Equation 9}$$

$$\frac{\partial I_R}{\partial V_I} = \frac{Q_{gen}(V_I^2 - V_R^2) + 2P_{gen}V_RV_I}{(V_R^2 + V_I^2)^2}, \qquad \text{Equation 10}$$

$$\frac{\partial I_R}{\partial Q_{gen}} = \frac{-V_I}{(V_R^2 + V_I^2)}, \qquad \text{Equation 11}$$

$$\frac{\partial I_I}{\partial Q_{gen}} = \frac{V_R}{(V_R^2+V_I^2)}. \quad \text{Equation 12}$$

Notice that $\frac{\partial I_I}{\partial V_R}$ and $\frac{\partial I_I}{\partial V_I}$ are missing from Eqs. 9-12. That is because we will give their equivalence in a briefer manner now. After derivation, we can recognize that

$$\frac{\partial I_I}{\partial V_R} = \frac{\partial I_R}{\partial V_I}, \quad \text{Equation 13}$$

$$\frac{\partial I_I}{\partial V_I} = \frac{-\partial I_R}{\partial V_R}. \quad \text{Equation 14}$$

However, we are not done with our derivations. Since Eq. 6 still describes the physics of the generator model, we will need to implement these relationships into our set of linearized equations. Hence, we will rewrite Eq. 6 from polar form into Euclidean form and recognize that this equation will also be nonlinear. Moving straight to the partials,

$$\frac{\partial F}{\partial V_R} = 2V_R, \quad \text{Equation 15}$$

$$\frac{\partial F}{\partial V_I} = 2V_I \quad \text{Equation 16}$$

where $F(V_R, V_I)$ is implied to be V_{set}^2. Together, Eqs. 10-16 represent the full set of partial derivatives required to create the Newton Raphson equation using the Taylor expansion of the generator model. However, we have not explained what the second order method for the generator will look like. For the current, this can be represented by

$$I_R^k = \frac{\partial I_R}{\partial V_R}(V_R^k) + \frac{\partial I_R}{\partial V_I}(V_I^k) + \frac{\partial I_R}{\partial Q_{gen}}(Q_{gen}^k) + I_R - [\frac{\partial I_R}{\partial V_R}(V_R) + \frac{\partial I_R}{\partial V_I}(V_I) + \frac{\partial I_R}{\partial Q_{gen}}(Q_{gen})], \quad \text{Equation 17}$$

$$I_I^k = \frac{\partial I_I}{\partial V_R}(V_R^k) + \frac{\partial I_I}{\partial V_I}(V_I^k) + \frac{\partial I_I}{\partial Q_{gen}}(Q_{gen}^k) + I_I - [\frac{\partial I_I}{\partial V_R}(V_R) + \frac{\partial I_I}{\partial V_I}(V_I) + \frac{\partial I_I}{\partial Q_{gen}}(Q_{gen})]. \quad \text{Equation 18}$$

where the superscript k represents the current iteration and variables without a superscript come from the previous iteration. Since the previous iteration is known, assumed, or initialized, variables without a superscript are numerical values. On the other hand, variables with the superscript are still unknown variables. The aforementioned naming scheme will only apply to nonlinear models such as PV and PQ buses. Now to keep bookkeeping manageable for Eqs. 17 and 18, we will then create two new variables to indicate "historical values". Thus,

$$I_R^{history} = -\ I_R + [\frac{\partial I_R}{\partial V_R}(V_R) + \frac{\partial I_R}{\partial V_I}(V_I) + \frac{\partial I_R}{\partial Q_{gen}}(Q_{gen})], \quad \text{Equation 19}$$

$$I_I^{history} = -I_I + [\frac{\partial I_I}{\partial V_R}(V_R) + \frac{\partial I_I}{\partial V_I}(V_I) + \frac{\partial I_I}{\partial Q_{gen}}(Q_{gen})]. \qquad \text{Equation 20}$$

Moving forward, the set voltage equation we will have is

$$0 = \frac{\partial F}{\partial V_R}(V_R^k) + \frac{\partial F}{\partial V_I}(V_I^k) - V_{set}^2 - [\frac{\partial F}{\partial V_R}(V_R) + \frac{\partial F}{\partial V_I}(V_I)]. \qquad \text{Equation 21}$$

Applying the same format as Eqs. 17 and 18 to Eq. 21, we recognize that

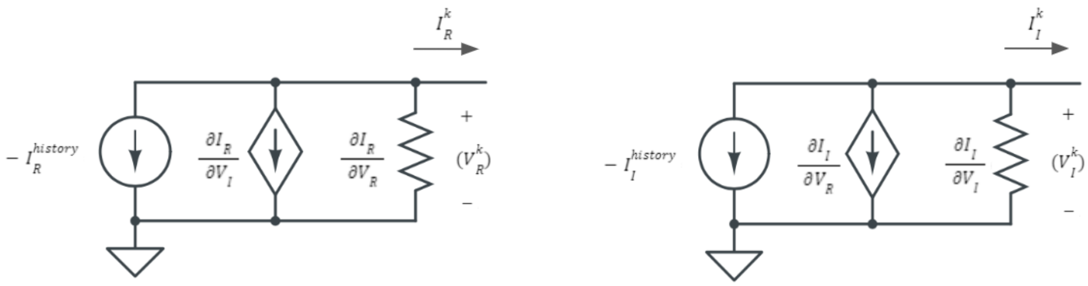
$$F^{history} = V_{set}^2 + [\frac{\partial F}{\partial V_R}(V_R) + \frac{\partial F}{\partial V_I}(V_I)]. \qquad \text{Equation 22}$$

We can represent Eqs. 17-22 as methods that would get stamped into an admittance matrix. The result is shown in Table 1.

	V_R^k	V_I^k	Q_{gen}^k	J Vector
V_R^k	$+\frac{\partial I_R}{\partial V_R}$	$+\frac{\partial I_R}{\partial V_I}$	$+\frac{\partial I_R}{\partial Q_{gen}}$	$+I_R^k + I_R^{history}$
V_I^k	$+\frac{\partial I_I}{\partial V_R}$	$+\frac{\partial I_I}{\partial V_I}$	$+\frac{\partial I_I}{\partial Q_{gen}}$	$+I_I^k + I_I^{history}$
Q_{gen}^k	$+\frac{\partial F}{\partial V_R}$	$+\frac{\partial F}{\partial V_I}$		$+F^{history}$

Table 1. Examples of stamps for an admittance matrix for the Newton Raphson method of linearizing power flow PV buses using voltages and reactive power as the steady-state variables.

From Table 1, we recognize that there are two unknown variables in the J vector. However, since we are applying KCL to each bus, all of the currents (both real and imaginary) entering or exiting the bus will sum to zero. Therefore, both I_R^k and I_I^k can be treated as zero when stamping the J vector. We will drop this notation in future example matrix stamps for other grid elements, but it is an important concept to understand nonetheless. The equivalent circuit models for PV buses using Eqs. 17-20 are shown below in Figure 1.



(a) **(b)**

Figure 1. Circuit equivalent model for the PV bus with the real current (a) and the imaginary current (b).

B) Slack Bus

The only similarity shared between the slack bus and the PV bus model is that both types are generators. Otherwise, the models and applications are different. To begin, the slack bus is a generator that is set as the reference point for a connected grid and as a consequence, the rank of the system is brought back to full rank. Without the implementation of the slack bus, the grid would not have a reference point for angles, θ, and hence, the system would have infinitely many solutions. Therefore, we arbitrarily decide on a generator that will represent a reference point and set the angle and magnitude for this bus. The result is that θ_{ref} for the system is known as well as V_{set}. Moreover, as a direct consequence of Eq. 6, V_R and V_I are also known as well. Solving Eq. 6 for V_R and V_I yields

$$V_R^{set} = V_{set} cos(\theta_{ref}), \qquad \text{Equation 23}$$

$$V_I^{set} = V_{set} sin(\theta_{ref}) \qquad \text{Equation 24}$$

where V_R^{set} and V_I^{set} indicate that these values are fixed and will not change based on the network. Now, we can model these buses as independent voltage sources since the voltages values at these buses are known and constant. The model would look like Figure 2.

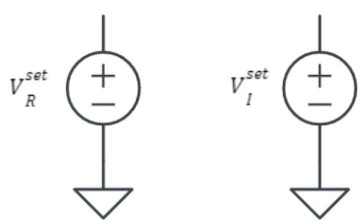


Figure 2. Circuit equivalent model for the slack bus.

Using the equivalent circuit in Figure 2 and Eqs. 23 and 24, we can create an example of the stamps for the admittance matrix as shown in Table 2.

	V_R	V_I	I_R	I_I	J Vector
V_R			$+1$		
V_I				$+1$	
I_R	$+1$				$+V_R^{set}$
I_I		$+1$			$+V_I^{set}$

Table 2. Examples of stamps for an admittance matrix for a slack bus model using modified nodal analysis of the steady-state voltages at the bus.

The implication of the one slack bus model is dangerous because the generator might require an unrealistically high real or reactive power to maintain the voltage at a singular bus. The result could be values outside of the bounds of the operation point of the generator. An example for the set of operation points for a generator is shown in Figure 3. Thus, a more realistic model would incorporate a distributed slack in which every generator shares a portion of the responsibility of the slack bus.

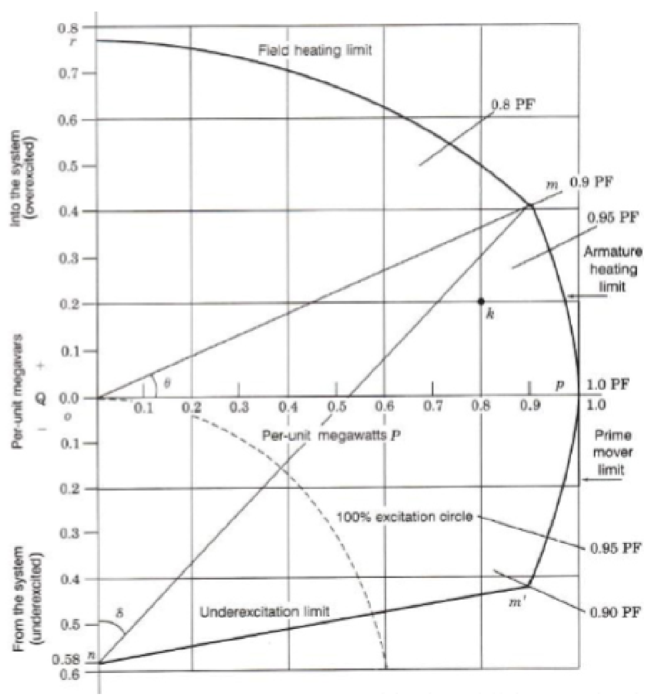


Figure 3. An example of the operation set for reactive power and real power of a generator. Adapted from Lecture 9: Traditional Power Flow Method presented by Amrit Pandey.

C) PQ Bus

The final type of bus that we need to consider is referred to as the PQ bus. Similar to how the PV bus was introduced, the PQ bus is a bus type that regulates both the real and reactive power on-site. Since power is determined at this bus location and loads have known power consumption values, PQ buses are also referred to as “loads”. Before we begin to model the PQ bus, we recognize that it will be quite similar to the PV bus. However, the steady-state variables for PQ bus only include the voltages at the bus since reactive power is given. To begin deriving the model for a PQ bus, we will need to determine the current draw by the bus in terms of power. It turns out that Eqs. 5 and 6 are the starting point for the PQ model since these equations describe the current entering a bus by convention. Furthermore, we recognize that we will have nonlinear terms because V_R and V_I are unknown. To linearize the current, we will apply Newton Raphson method and start with finding the partials of Eqs. 5 and 6. Hence,

$$\frac{\partial I_R}{\partial V_R} = \frac{P_{gen}(V_R^2 - V_I^2) + 2Q_{gen}V_R V_I}{(V_R^2 + V_I^2)^2}, \qquad \text{Equation 25}$$

$$\frac{\partial I_R}{\partial V_I} = \frac{Q_{gen}(V_I^2 - V_R^2) + 2P_{gen}V_R V_I}{(V_R^2 + V_I^2)^2}. \qquad \text{Equation 26}$$

As done before in the PV bus derivation, $\frac{\partial I_I}{\partial V_R}$ and $\frac{\partial I_I}{\partial V_I}$ are not explicitly given because there is a relationship between partials of the real and imaginary current. These relationships are

$$\frac{\partial I_I}{\partial V_R} = \frac{\partial I_R}{\partial V_I}, \qquad \text{Equation 27}$$

$$\frac{\partial I_I}{\partial V_I} = \frac{-\partial I_R}{\partial V_R}. \qquad \text{Equation 28}$$

Note that Eqs. 27 and 28 are exactly the same as Eqs. 13 and 14, further enforcing our calculation from earlier and the similar relationship between PQ and PV buses. Now, we will provide the derivation of the first-order Taylor expansion of the currents at a PQ bus in order to use Newton Raphson method. Hence,

$$I_R^k = \frac{\partial I_R}{\partial V_R}(V_R^k) + \frac{\partial I_R}{\partial V_I}(V_I^k) + I_R - [\frac{\partial I_R}{\partial V_R}(V_R) + \frac{\partial I_R}{\partial V_I}(V_I)], \qquad \text{Equation 29}$$

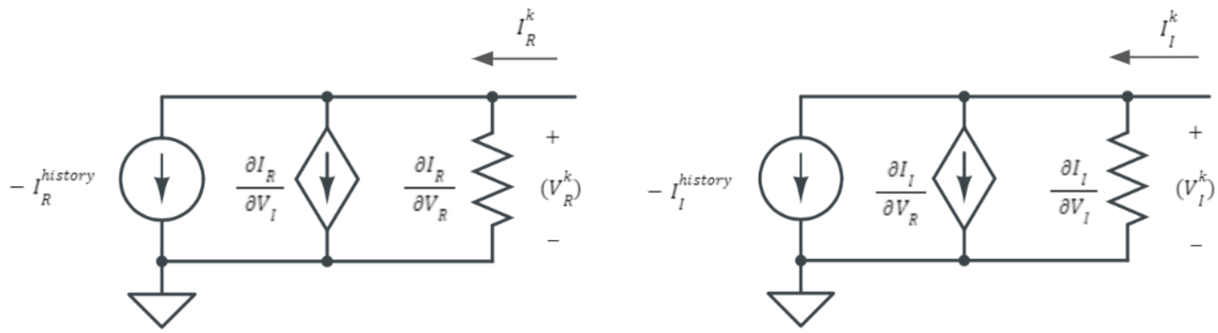
$$I_I^k = \frac{\partial I_I}{\partial V_R}(V_R^k) + \frac{\partial I_I}{\partial V_I}(V_I^k) + I_I - [\frac{\partial I_I}{\partial V_R}(V_R) + \frac{\partial I_I}{\partial V_I}(V_I)]. \qquad \text{Equation 30}$$

Lastly, we will create the historical terms,

$$I_R^{history} = -I_R + [\frac{\partial I_R}{\partial V_R}(V_R) + \frac{\partial I_R}{\partial V_I}(V_I)], \qquad \text{Equation 31}$$

$$I_I^{history} = -I_I + [\frac{\partial I_I}{\partial V_R}(V_R) + \frac{\partial I_I}{\partial V_I}(V_I)], \qquad \text{Equation 32}$$

Now that we have created a mathematical model for the PQ bus, we can create its equivalent circuit in order to better visualize the model. Thus, the equivalent circuit for the PQ bus is shown in Figure 4 below.



(a) **(b)**

Figure 4. Equivalent circuit models for the real current (a) and imaginary current (b) of a PQ bus.

Lastly, we can create a matrix of stamps for the admittance matrix in order to linearly represent our network. Table 4 shows the stamps for a PQ bus.

	V_R^k	V_I^k	J Vector
V_R^k	$+\ \frac{\partial I_R}{\partial V_R}$	$+\ \frac{\partial I_R}{\partial V_I}$	$+\ I_R^{history}$
V_I^k	$+\ \frac{\partial I_I}{\partial V_R}$	$+\ \frac{\partial I_I}{\partial V_I}$	$+\ I_I^{history}$

Table 4. Example of stamps for an admittance matrix for the Newton Raphson equivalent of the PQ bus model using bus voltages as steady-state variables.

D) Transmission Lines

In order for the system to have physical meaning, and hence a solution, the buses of the grid must be connected in some manner. Otherwise, the solution for the problem would be trivial and there would not be a question of interest. As a result, we connect to buses together by transmission lines. The model for a transmission line is a power carrying wire that has impedance. Furthermore, the model we will implement is referred to as the single π–model due to the shape of the model seen in Figure 6. A more accurate model would suggest that the impedance is infinitesimally distributed along the length of the wire, but we are just introducing the model in this report. To begin, we will start with the general model of a transmission line shown in Figure 5.

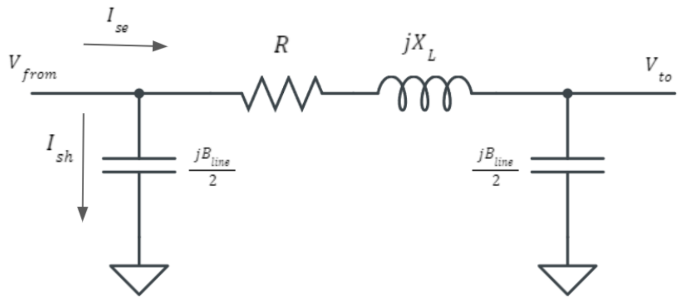


Figure 5. Physical single π–model of a transmission line.

From Figure 5 we can see that R is the line resistance, X_l is the inductive reactance of the line, and B_{line} is the total capacitive reactance of the line. Now, we can begin to create the mathematical equivalents of the transmission model. Before we begin the derivations, we will do some slight bookkeeping. During the derivations, we will allow $V^{ft} = V^{from} - V^{to}$ for convenience. Thus,

$$V_R^{ft} + jV_I^{ft} = (I_{se,R} + jI_{se,I})(R + jX_l), \qquad \text{Equation 33}$$

$$V_R^{from} + jV_I^{from} = (I_{sh,R} + jI_{sh,I})(\frac{jB_{line}}{2}), \qquad \text{Equation 34}$$

$$V_R^{to} + jV_I^{to} = (I_{sh,R} + jI_{sh,I})(\frac{jB_{line}}{2}). \qquad \text{Equation 35}$$

Recognizing that Eqs. 34 and 35 are symbolically the same, we will only further work with one of the equations due to their equivalent relationship. Solving Eq. 33 for current and separating by real and imaginary terms,

$$I_{se,R} = \frac{R}{R^2+X_L^2}(V_R^{ft}) + \frac{X_L}{R^2+X_L^2}(V_I^{ft}), \qquad \text{Equation 36}$$

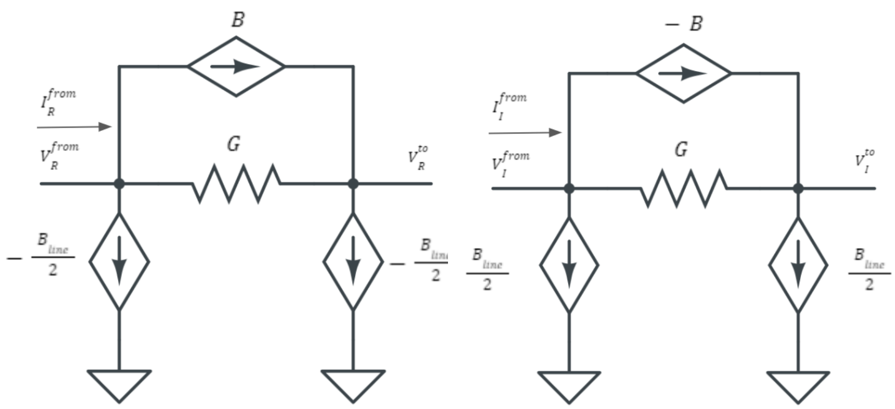
$$I_{se,I} = \frac{R}{R^2+X_L^2}(V_I^{ft}) + \frac{-X_L}{R^2+X_L^2}(V_R^{ft}). \qquad \text{Equation 37}$$

Furthermore, we will allow conductance, G, to be such that $G = \frac{R}{R^2+X_L^2}$ and susceptance, B, to be such that $B = \frac{X_L}{R^2+X_L^2}$. Now, solving Eqs. 34 and 35 for real and imaginary currents, we arrive at

$$I_{sh,R} = \frac{-B_{line}}{2}(V_I^b), \qquad \text{Equation 38}$$

$$I_{sh,I} = \frac{B_{line}}{2}(V_R^b), \qquad \text{Equation 39}$$

where b indicates either the from or to bus. Finally, we can create the equivalent circuit for the transmission line which is highly similar to Figure 5 and is shown in Figure 6.



(a) **(b)**

Figure 6. Equivalent circuit model for the single π-model representation of a transmission line with real current (a) and imaginary current (b).

Using Eqs. 36–39, we can create an example matrix of the stamps characteristic to the transmission line for an admittance matrix. These stamps are shown in Table 3.

	V_R^{from}	V_I^{from}	V_R^{to}	V_I^{to}	J Vector
V_R^{from}	$+\ G$	$+\ B - \frac{B_{line}}{2}$	$-\ G$	$-\ B$	
V_I^{from}	$-\ B + \frac{B_{line}}{2}$	$+\ G$	$+\ B$	$-\ G$	
V_R^{to}	$-\ G$	$-\ B$	$+\ G$	$+\ B - \frac{B_{line}}{2}$	
V_I^{to}	$+\ B$	$-\ G$	$-\ B + \frac{B_{line}}{2}$	$+\ G$	

Table 3. Examples of stamps for an admittance matrix for the single π-model of transmission lines using the corresponding from and to buses' voltages as steady-state variables.

E) Transformers

A vital component of the transmission grid is the transformer. Transformers use a winding ratio and magnetic fields to induce a change of voltage from one bus to another. Thus, the ratio of the voltages on both sides of a transformer are related to the intrinsic turns ratio in our derivations. For more complex models, the turns ratio does not need to be fixed and we can see this will only change the model by introducing nonlinearities. Additionally, the transformer will create a phase shift between the voltages at both buses it is connected to. For both of these instances, Newton Raphson will need to be applied to find convergence of the system. In our model, we will only consider a fixed turns ratio value and angle shift so the derivations will be linear. The model can be extended upon in the future.

To begin, we will assume that the transformer model will have impedance on the secondary side. As mentioned in transmission lines, a more accurate model would distribute this impedance along the length of the wire for the transformer. However, this implementation will be sufficient for finding steady-state values. Then, we know that

$$\frac{V^{from}}{V^{sec}} = t_r \qquad \text{Equation 40}$$

and by conservation of power,

$$\frac{I^{sec}}{I^{from}} = -t_r^* \qquad \text{Equation 41}$$

where t_r is the turns ratio of the transformer and is equal to $t_r(cos\theta_{shift} + jsin\theta_{shift})$. Recall that all values are complex in Eqs. 40 and 41 and so t_r^* will be the complex conjugate of t_r. First, we will solve for the voltage on the primary side of the transformer. Thus,

$$V_R^{from} + jV_I^{from} = t_r(V_R^{sec} + jV_I^{sec})(cos\theta_{shift} + jsin\theta_{shift}). \qquad \text{Equation 42}$$

Separating Eq. 42 into its real and imaginary voltages,

$$V_R^{from} = t_r(V_R^{sec}cos\theta_{shift} - V_I^{sec}sin\theta_{shift}), \qquad \text{Equation 43}$$

$$V_I^{from} = t_r(V_R^{sec}sin\theta_{shift} + V_I^{sec}sin\theta_{shift}). \qquad \text{Equation 44}$$

Again, to keep bookkeeping manageable for our future stamps in Table 5, we will allow $C = t_r cos\theta_{shift}$ and $S = t_r sin\theta_{shift}$. Now, we need to apply the same methodology for Equation 41 and we will have captured the physics of the transformer without impedance. This time we will solve for the current on the secondary side because the primary side of the transformer is already modeled. Hence,

$$I_R^{sec} + jI_I^{sec} = -t_r(I_R^{from} + jI_I^{from})(cos\theta_{shift} - jsin\theta_{shift}) \qquad \text{Equation 45}$$

Again, solving for real and imaginary currents yields

$$I_R^{sec} = -t_r(I_R^{from}cos\theta_{shift} + I_I^{from}sin\theta_{shift}) \qquad \text{Equation 46}$$

$$I_R^{sec} = -t_r(-I_R^{from}sin\theta_{shift} + I_I^{from}cos\theta_{shift}) \qquad \text{Equation 47}$$

Finally, we need to model the impedance of the transformer. As previously mentioned, we will isolate the impedance to the secondary side of the transformer. By letting $V^{st} = V^{to} - V^{sec}$ we will have

$$V_R^{st} + jV_I^{st} = (I_R + jI_I)(R + jX_l) \qquad \text{Equation 48}$$

where R is the resistance of the transformer and jX_l is the inductive reactive. Solving Eq. 48 in terms of current and separating along real and imaginary components, we have

$$I_R = \frac{R}{R^2+X_L^2}(V_R^{st}) + \frac{X_L}{R^2+X_L^2}(V_I^{st}), \qquad \text{Equation 49}$$

$$I_I = \frac{R}{R^2+X_L^2}(V_I^{st}) + \frac{-X_L}{R^2+X_L^2}(V_R^{st}). \qquad \text{Equation 50}$$

Eqs. 49 and 50 complete the model of the linear transformer. Following the same naming scheme as transmission lines, $B = \frac{X_L}{R^2+X_L^2}$ and $G = \frac{R}{R^2+X_L^2}$. The equivalent circuit for the model can be seen in Figure 7.

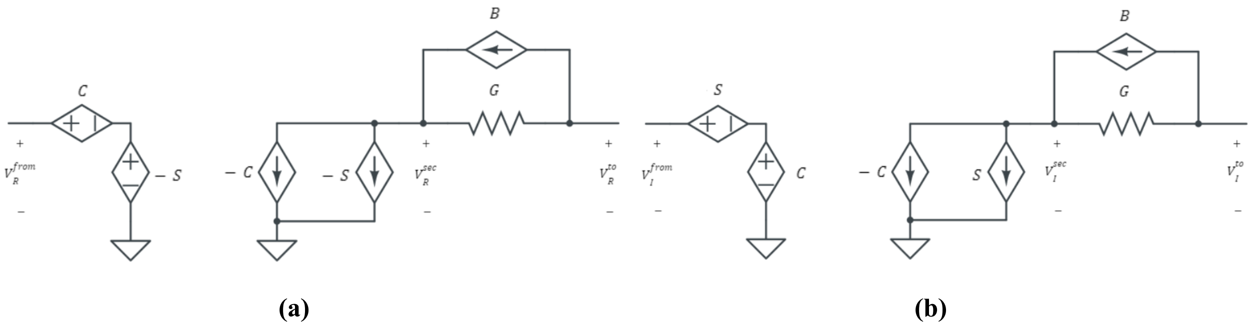


Figure 7. Equivalent circuit model for a linear transformer using current and voltage sources showing the real circuit (a) and the imaginary circuit (b).

Using Eqs. 43–50, we can create an example of the stamps for an admittance matrix as shown in Table 5.

	V_R^{from}	V_I^{from}	V_R^{to}	V_I^{to}	V_R^{sec}	V_I^{sec}	I_R^{from}	I_I^{from}
V_R^{from}							$+\ 1$	
V_I^{from}								$+\ 1$
V_R^{to}			$+\ G$	$+\ B$	$-\ G$	$-\ B$		
V_I^{to}			$-\ B$	$+\ G$	$+\ B$	$-\ G$		
V_R^{sec}			$-\ G$	$-\ B$	$+\ G$	$+\ B$	$-\ C$	$-\ S$
V_I^{sec}			$+\ B$	$-\ G$	$-\ B$	$+\ G$	$+\ S$	$-\ C$
I_R^{from}	$+\ 1$		$-\ C$	$+\ S$				
I_I^{from}		$+\ 1$	$-\ S$	$-\ C$				

Table 5. Example of stamps for an admittance matrix for the linear transformer model using bus voltages, transformer's secondary side voltage, and transformer's primary side current as steady-state variables.

Notice in Table 5 that the J vector is missing. Since we modeled the transformer as a linear device and the equivalent model in Figure 7 does not consist of independent voltage or current sources, the J vector will not have elements to be stamped into it. In other words, the linear transformer admits zeroes to the J vector.

F) Shunts

The very last model incorporated into power flow simulation is the fixed shunt model. Shunts are rather easy to derive, since we have already included them in the transmission line formulation. However, the fixed shunt model includes a resistance in parallel to the capacitor. Hence, we are implementing a capacitor bank in the system in order to store charge and maintain the voltage at a bus. As an additional advantage of the shunt, the power factor at the bus should increase because the capacitor will resist inductances to the bus which causes an increase of lag between the voltage and current. In any case, we will quickly run through the derivation. To start, the equation governing a fixed shunt is given by

$$V = I(\frac{1}{\frac{1}{R} - \frac{1}{jX_C}}) \qquad \text{Equation 51}$$

where a capacitive reactance, X_C, is added in parallel to a resistance, R. Solving for current in Eq. 51,

$$I_R + jI_I = (V_R + jV_I)(\frac{1}{R} + \frac{j}{X_C}). \qquad \text{Equation 52}$$

We will now define conductance as $G = \frac{1}{R}$ and susceptance as $B = \frac{1}{X_C}$. Then, we can separate Re and Im of Eq. 52 as

$$I_R = V_R G - V_I B, \qquad \text{Equation 53}$$
$$I_I = V_R B + V_I G. \qquad \text{Equation 54}$$

Now, we have our mathematical model for the fixed shunt, we can create the equivalent circuit model as shown in Figure 8.

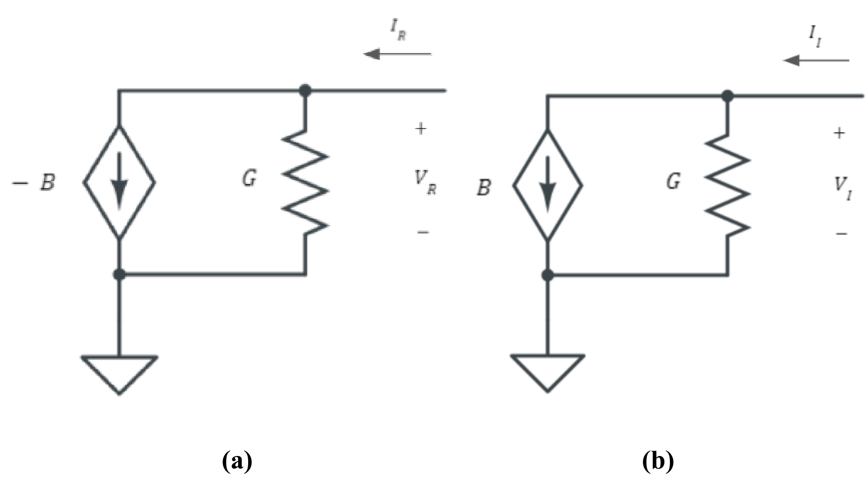


Figure 8. Equivalent circuit model of a fixed shunt depicting the real circuit (a) and the imaginary circuit (b).

Finally, using Eqs. 53 and 54, we can create the example stamping matrix for shunts as shown below in Table 6.

	V_R	V_I	J vector
V_R	$+\ G$	$-\ B$	
V_I	$+\ B$	$+\ G$	

Table 6. Example of stamps for an admittance matrix for the shunt model using the bus voltages as the steady-state variables.

II Codebase

Designs particular to my codebase include a tolerance level of 10^{-7} for changes in the state variables during iteration steps of Newton Raphson. The standard value for this codebase is typically a tolerance level of 10^{-5} but I was adjusting the error tolerance prior to submission and never returned it to its original value.

Another design feature specific to my codebase includes the initialization of steady-state variables from the prior solution of the network. Hence, the initialization vector does not necessarily begin at a flat start (initialization of 1's and 0's). Rather, the initialized vector contains steady-state values that are close to the solution of the network when the prior solution is provided.

During implementation of the solution, the attribute for VCCS equivalent of the shunt in transmissions lines (Figure 6) is defined as positive for the real current, I_R. Hence, $-\frac{B_{line}}{2}$ is defined to be positive within the transmission line class of the codebase.

Finally, the power flow tool creates an output table that is printed to screen and contains the bus voltages and their angle relative to the slack bus.

III Dense vs Sparse Matrices

I did not complete a dense to sparse matrix comparison of my codebase. However, I understand that initializing a sparse matrix upfront and using sparse linear solving methods will be significantly less computationally expensive than solving with traditional dense matrices. The significance of this difference can be seen on higher rank n matrices where the dense matrix scales by $O(n^3)$. On the other hand, sparse matrix solvers such as those used in the Scipy library are on the order of $O(n^{1.x})$ where x is a real-valued variable. The most computational exhaustive process is calculating the inverse matrix.

IV Results

A) Testcase GS-4

In testcase GS-4, there are PV buses, PQ buses, transmission lines, and a slack bus. From convergence of Newton Raphson, the maximum voltage and angle computed for this dataset was 1.020 p.u. and 1.523°. Likewise, the minimum voltage and angle recorded for the dataset was 0.969 p.u. and –1.872°. Thus, the reported voltages are within the bounds of high voltage states, approximately 0.8–1.3 p.u. Finally, the average percent difference between the reported values in Matpower and those calculated in my codebase are exactly the same. Hence, they have a 0% disagreement. The number of Newton Raphson steps required for convergence of this case was 4 iterations The results of the testcase can be seen below in Figure 9.

```
================================================================
|     Bus Data
================================================================
 Bus      Voltage          Generation             Load
  #   Mag(pu) Ang(deg)   P (MW)   Q (MVAr)   P (MW)   Q (MVAr)
----- ------- --------  --------  --------  --------  --------
    1  1.000    0.000*   186.81    114.50     50.00     30.99
    2  0.982   -0.976       -         -      170.00    105.35
    3  0.969   -1.872       -         -      200.00    123.94
    4  1.020    1.523    318.00    181.43     80.00     49.58
                        --------  --------  --------  --------
               Total:    504.81    295.93    500.00    309.86
```

```
Bus      Voltage Magnitude        Angle
=======================================
1        1.000                    0.000
2        0.982                   -0.976
3        0.969                   -1.872
4        1.020                    1.523
```

(a) **(b)**

Figure 9. Output for Testcase GS-4 from Matpower (a) and the codebase (b).

B) Testcase IEEE-14

This testcase introduces the transformer model and fixed shunts from Figures <x> in addition to the other elements found within Testcase GS-4. The maximum voltage and angle computed for this dataset was 1.090 p.u. and 0°. Likewise, the minimum voltage and angle recorded for the dataset was 1.010 p.u. and –16.034°. Thus, this dataset remains within the bounds for high voltage states, approximately from 0.8–1.3 p.u. The values calculated within the codebase and Matpower agree 100%. In other words, the average percent difference between the two computational tools was 0% different on average. Lastly, the number of Newton Raphson steps required for convergence of this case was 2 iterations. The results of this testcase can be seen below in Figure 10.

Bus #	Voltage Mag(pu)	Voltage Ang(deg)	Generation P (MW)	Generation Q (MVAr)	Load P (MW)	Load Q (MVAr)
1	1.060	0.000*	232.39	-16.55	-	-
2	1.045	-4.983	40.00	43.56	21.70	12.70
3	1.010	-12.725	0.00	25.08	94.20	19.00
4	1.018	-10.313	-	-	47.80	-3.90
5	1.020	-8.774	-	-	7.60	1.60
6	1.070	-14.221	0.00	12.73	11.20	7.50
7	1.062	-13.360	-	-	-	-
8	1.090	-13.360	0.00	17.62	-	-
9	1.056	-14.939	-	-	29.50	16.60
10	1.051	-15.097	-	-	9.00	5.80
11	1.057	-14.791	-	-	3.50	1.80
12	1.055	-15.076	-	-	6.10	1.60
13	1.050	-15.156	-	-	13.50	5.80
14	1.036	-16.034	-	-	14.90	5.00
		Total:	272.39	82.44	259.00	73.50

(a)

Bus	Voltage Magnitude	Angle
1	1.060	0.000
2	1.045	-4.983
3	1.010	-12.725
4	1.018	-10.313
5	1.020	-8.774
6	1.070	-14.221
7	1.062	-13.360
8	1.090	-13.360
9	1.056	-14.939
10	1.051	-15.097
11	1.057	-14.791
12	1.055	-15.076
13	1.050	-15.156
14	1.036	-16.034

(b)

Figure 10. Output for Testcase IEEE-14 from Matpower (a) and the codebase (b).

C) Testcase IEEE-118

This testcase contains 118 buses and utilizes the same circuit elements found within testcase IEEE-14. I did not implement a function to sort through the data and find the maximum and minimum results, so I could not report these results as easily as the previous testcases. Likewise, I did not convert the output data into a readable format for python to interpret the data and calculate the average difference between values. However, there was a disagreement between the angles calculated in Matpower against the codebase. This difference can be easily seen after briefly scanning through the outputs of both computational tools. However, the output voltages appear to agree. Lastly, the number of Newton Raphson steps required for convergence was 4 iterations.

D) Testcase ACTIVSg500

This testcase contains 500 buses and is constructed of the same power grid elements as testcase IEEE-14. Similar to Testcase IEEE-118, I did not calculate the maximum, minimum, or average difference between values. However, similar results were produced, and both power analysis tools reported the same voltages but a disagreement in angles. Lastly, the number of Newton Raphson steps required for convergence was 4 iterations.

V Reflection

During this project, I gained a deeper understanding of the implications of a power network grid and the tools required to solve such problems. Prior to taking this course, I did not have an understanding of power flow, its mechanisms, or grid analysis specifically for power transmission in regards to this project. However, I am now understanding the breadth of the field including components, problems, and solutions related to power flow. In addition, I am recognizing that there are still plenty of opportunities within the field for development and integration of new techniques. Not only that, I feel more comfortable with modeling methods used for circuit representations which is very useful for any domain within ECE.

Specific to this project, I was able to build off of example code for implementing more OOP methods. I recall one instance when I did not like the initialization function I was utilizing. Thus, I returned to this function later in the code and was able to compress the lines of code into a substantially smaller footprint with a better logical structure. This was very rewarding for me, and I think the progress can still be seen in the history of my github repository (initialize.py file if interested). I think two other significant lessons I learned was to always rederive the model so its application makes sense and to not try and create additional steady-state variables that already exist. For the prior lesson, the more times I consulted the model and derived equations for the model, the more comfortable I became implementing the model. Upfront, the models appeared cumbersome but they were actually quite manageable. For the last lesson, I really struggled using nodal analysis methods when creating the transformer model. Thus, I kept creating divergence or singular matrix issues within the codebase. Eventually after enough collaboration and reasoning of the model, I was able to implement transformers within the codebase.