Simplified Model and Improvement on Simulation Time

The tools under development for distribution system planning work together to provide useful information. As the size and number of tools increases, so too does the time required to gather and process the information. The *Iterative Static Hosting Capacity Analysis* is only one part of this process and, as mentioned previously, requires 103 hours to analyze six feeders for an average runtime of over 17 hours per feeder.

Utilities are continuing to face an increase in the rate of PV interconnection requests. According to a June 2016 salesforce pull, the Duke Energy territory has over 1600 connection requests. In order to prepare the distribution system for the large influx of PV interconnections, the utility needs a fast, accurate, and effective tool to analyze feeders for necessary system upgrades. With 1000s of substations, an average of 17 hours per feeder for a single step in the process does not correspond to a speedy analysis.

The speed of a program generally depends on the available hardware, the number and efficiency of operations, and the size of the data being operated on by the program. Hardware is expensive, ties up a valuable resource, and may increase additional costs to the utility. Software efficiency improvements lead to relatively small decreases in runtime. In order to effectively decrease the runtime, it was necessary to greatly reduce the size of the data. This necessity lead to the introduction of a reduced system model.

The overall goal of a system reduction is to decrease the number of buses and lines; however, the system needed to remain electrically equivalent to the original, detail model. Analyzing the *LinesByPhase* OpenDSS file showed that many of the adjacent lines had line spacing and conductor types which exactly matched each other. In order to reduce the feeder, adjacent lines with matching characteristics were combined by summing their lengths to form one long line with the same line characteristics. Each change of conductor type or spacing caused the creation of a new line.

For the reduction of Feeder 4, new unique bus IDs were assigned to the beginning and end of each line. Collapsing the lines and assigning these new IDs meant that the *Loads* file no longer allocated any load to the circuit since it referenced bus IDs that did not exist. Therefore, the *Loads* file also needed to be reduced. Each new line represents the combination of a number of adjacent lines, thereby representing a list of adjacent bus IDs. The loads associated with each of these bus IDs were summed for each new line to create a spot load. This spot load was then centered to reflect the relative load density along the line. Moreover, a load placed at 0.8 times the length of a new line reflects that there is a greater load density closer to the ending bus ID of the new line. Each of these new spot loads split the new lines into two lines with equal characteristics but different lengths to represent the load placement.

The *LinesByPhase* and *Loads* OpenDSS files for Feeder 4 were reduced by hand. A new topography was developed in order to better showcase the reduction method. This new topography is shown in Figure X. Notice how the loads are positioned to reflect the load density as discussed earlier. Also note that the voltage regulators and capacitors were maintained in this model. Equipment such as these would also force the start of a new line in the line reduction method presented.

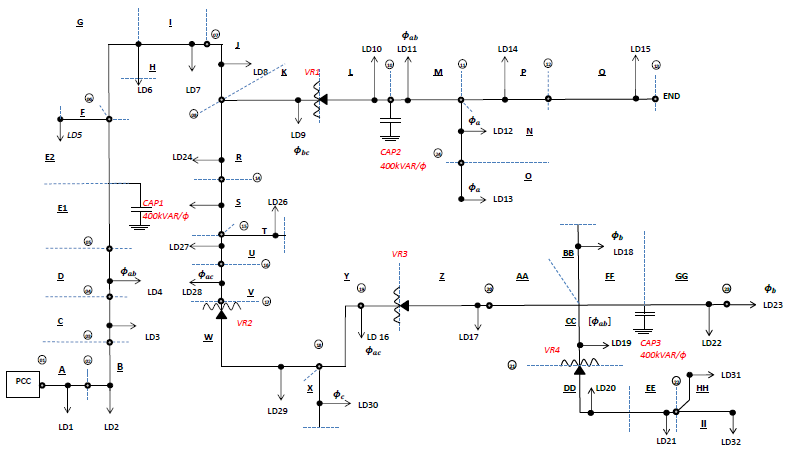
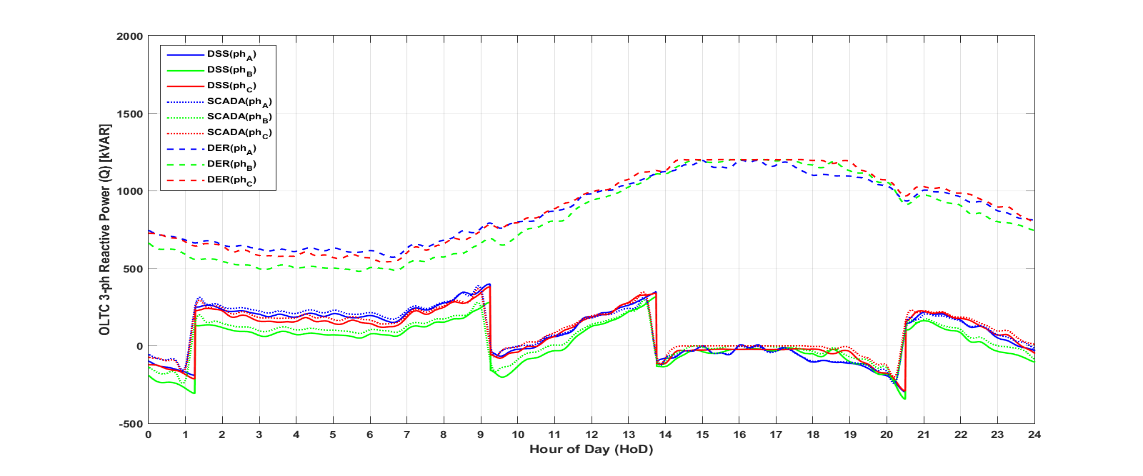


Figure X: Reduced Feeder 4 Topography

The requirements of the reduction were that it reduced the runtime and maintained the electrical equivalency of the detailed feeder model. In order to prove the latter, a QSTS analysis was performed on the reduced model and compared to the historical data. The result of interest from this analysis was the reactive power, which is shown in Figure Xa. The historical data is represented by the marker labeled SCADA while the results of the QSTS utilizing OpenDSS for powerflow with the reduced model is represented by the marker labeled DSS.

Visually, the reduced model appears closely match the historical data. In order to quantify how close the reduced model matches the historical, the the difference between the two was plotted in Figure Xb. As a reminder, the historical data was interpolated to 1 minute data, from 10 or 15 minute data, in order to time synchronize the data. Interpolated data does not perfectly predict the actual data. For example, if a measurement is taken at 0900 and another at 0915 indicates that a capacitor operates, there is no way of knowing at exactly what time that capacitor operated. So the interpolated historical data showcases capacitor operations on either 10 or 15 minute intervals. For the QSTS analysis, MATLAB is communicating with OpenDSS every minute of simulation. This degree of clarity allows the OpenDSS model to capture the capacitor operation at the time of operation, rather than a few minutes later. This difference in clarity is shown by the spikes seen in Figure Xb. The average percent error for both the real and reactive power was just over 1%. It is assumed the maximum error of the reactive power is due to interpolated data rather than a problem with the simplified feeder model. To gain an understanding of the maximum possible error, the maximum error of the real power was found to be 2.22%.

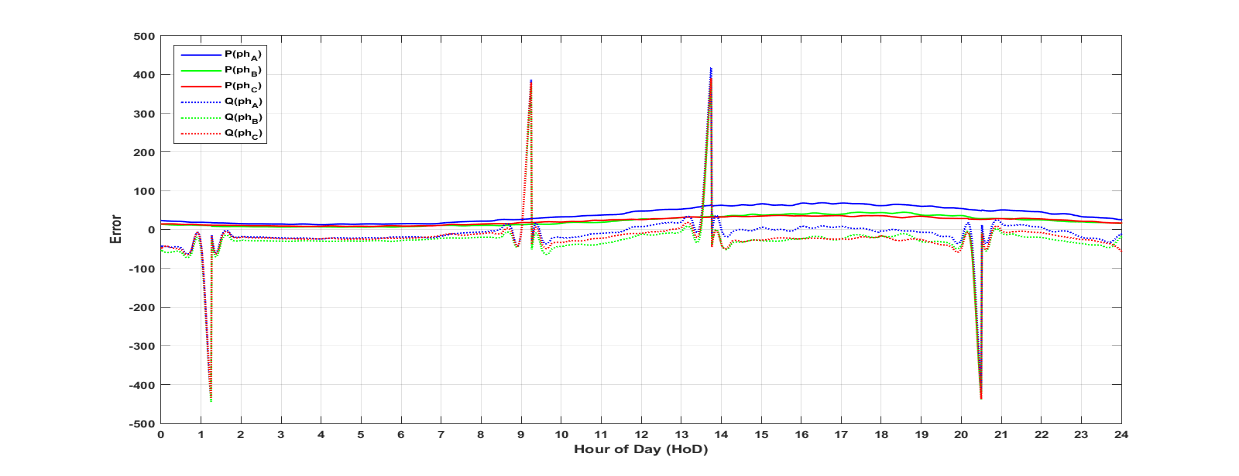
Figure Xa: Reactive Power from 24-hour QSTS of Feeder 4

Figure Xb: Reactive Power Error from 24-hour QSTS of Feeder 4

The results and benefits of the reduction are best summarized by Table X. According the reduction in the number of lines and loads, there was approximately a 95% reduction in the system. The benefit of this reduction was over 99% reduction in runtime for the QSTS analysis.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Lines** | **Loads** | **Run Time** |
| Detailed | 2196 | 1047 | ~ 2 day / year sim |
| Reduced | 67 | 78 | ~ 16 min / year sim |
| *Reduction* | *96.9%* | *92.6%* | *99.5%* |

Table X: Percent Reduction of Lines, Loads, and Run Time

The benefit of this reduction corresponds to an exponential time savings. Most of the DER planning tools require iterative changes to the system with a subsequent powerflow in OpenDSS. For each tool, the number of iterations is reduced by, on average, 95% and the OpenDSS requires only a fraction of the runtime to run the powerflow. As the number and complexity of the tools and feeders increase, these time savings compound. This reduction model provides a way of performing analysis accurately and quickly.