An open-source tool to provide a common interface for simulating different types of power systems

Marko Buha, Amanda Sullivan, Tommy Wei, Giancarlo Dalle Ave

Department of Chemical Engineering, McMaster University, 1280 Main St. W, Hamilton, Ontario, Canada, L8S 4L8

E-mail: buham@mcmaster.ca | Website: https://dalleavelab.com

Introduction

- Power systems transmit voltage as a set of 3 sinusoidal waves separated in phase by 120 degrees. Currently, there are 2 types of 3 phase power systems networks in common use:
  - A **balanced** network where the 3 phases are provided to all customers.
  - An **unbalanced** network where each customer receives 1 of the phases.
- With the increase in distributed energy generation, power systems have become more complex, increasing the need for **easy-to-use** simulation tools [1]. Notably, there is a need to test various algorithms on both balanced and unbalanced networks. This currently requires 2 simulation setups on 2 separate simulation tools as:
  - **Balanced simulation tools** (such as **PandaPower** [1]) are incapable of simulating unbalanced networks, as each phase serves different customers, meaning that the simulation tool must be able to analyse the phases separately.
  - **Unbalanced simulation tools** (such as **OpenDSS** [2]) are not easy to use for balanced networks as they require the user to enter data for the network and its parameters for each phase, when in fact all 3 phases behave identically. They are also inefficient as they simulate all 3 phases.
- Thus, a need is identified for a simulation tool that can easily simulate both balanced and unbalanced networks.

Methods

- The simulation tool (class diagram in Figure 1) is a Python package constructed using object-oriented programming. This allows for:
  - The abstraction of parameters and functionalities common to different types of networks and their elements to a parent class
  - The presentation of a **unified experience** regardless of which type of network is being simulated
- There are functions for the user to create the network, injection sets (which allow for the power generation or demand at an element to be changed over time), and optimal power flow functions (which establish economic constraints on the network).
- The simulation tool can alternatively import:
  - A network in PandaPower's syntax or OpenDSS's syntax.
  - A set of power injections in the same syntaxes.
- With that done, the simulation tool can use the translators to run a simulation in PandaPower or OpenDSS, obtaining results like the voltage at various locations in the network.
- There is a common csv option for importing and exporting networks and injections, allowing for them to be expressed in a common format.

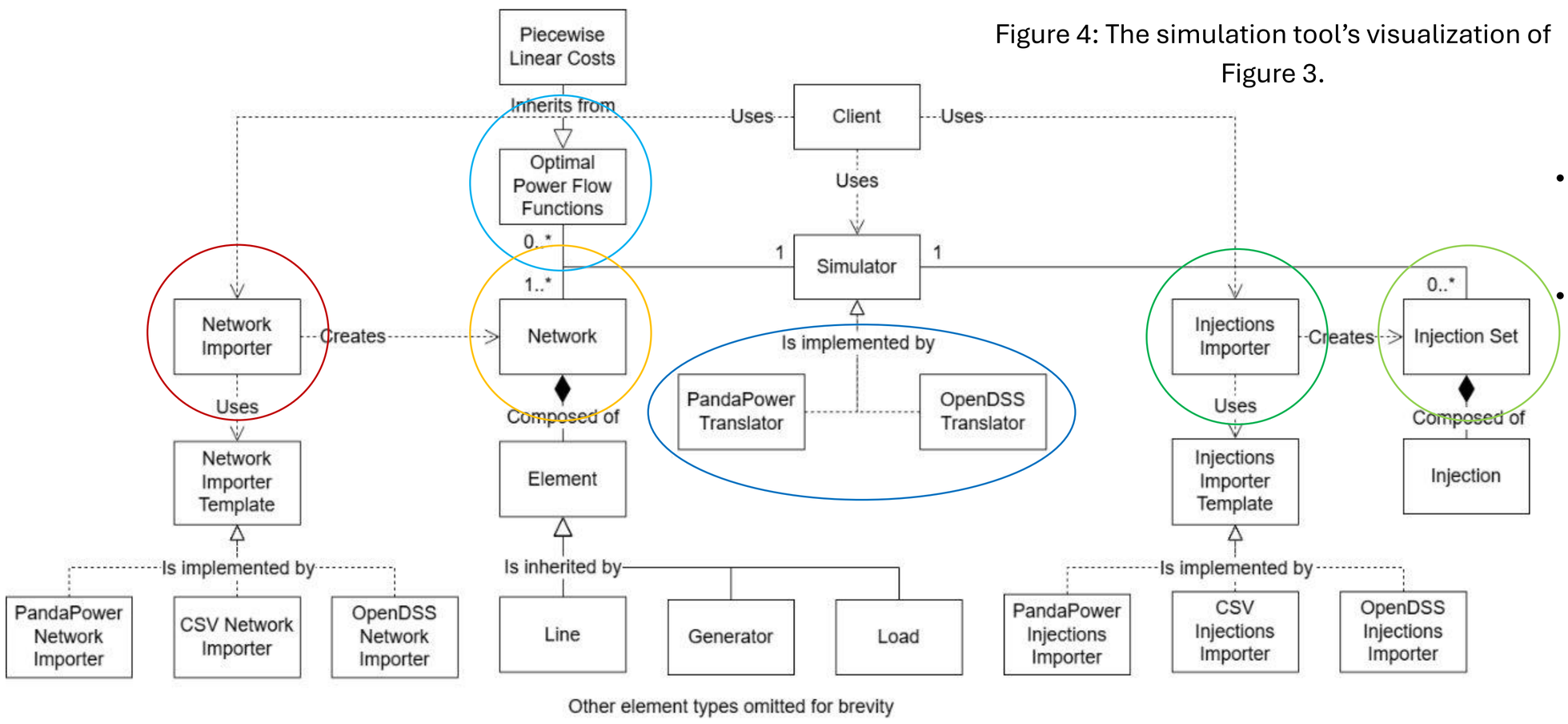

Figure 1: Class diagram of the simulation tool.

Results

- Results can be exported in several formats, which currently include:
  - CSV
  - A plot showing how the results change over time (Figure 2)
  - A visualization of the network structure (Figure 4 is the simulation's version of Figure 3)
- The plots were developed using Plotly, an interactive plotting library for Python

Figure 2: Time series plot of an unbalanced network from the simulation tool.

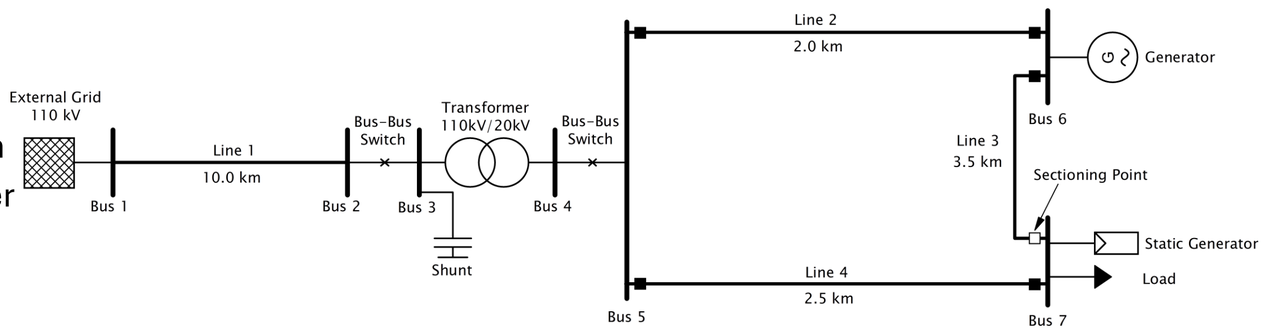

Figure 3: Balanced network [1].

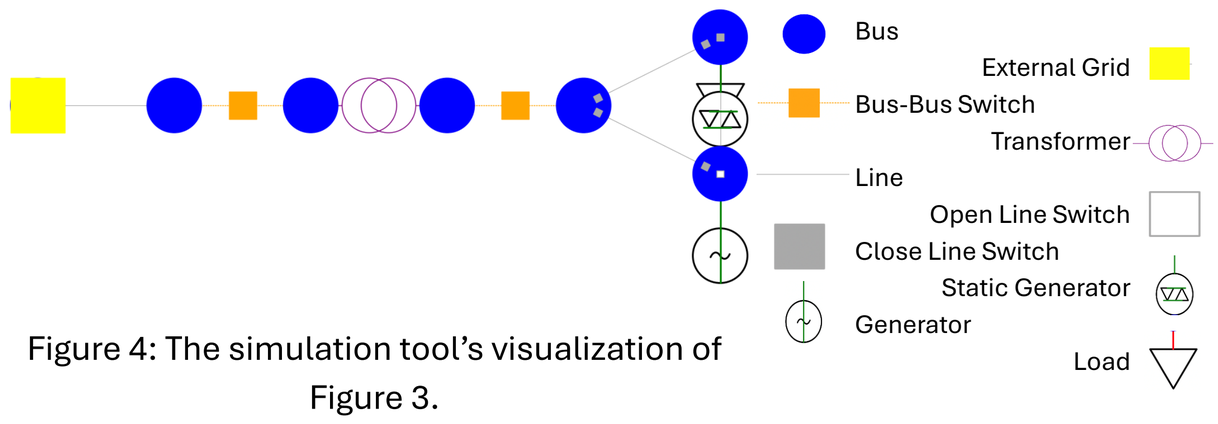

Figure 4: The simulation tool's visualization of Figure 3.

Next Steps

- Addition of further types of simulation such as a short circuit analysis
- Expand the functionality of the network visualization
  - Shortest path searches
  - Visualizing voltages greater than 1 per unit in green and less than 1 per unit in red

References & Acknowledgements

[1] Leon Thurner, Alexander Scheidler, Florian Schäfer, Jan-Hendrik Menke, Julian Dollichon, Friederike Meier, Steffen Meinecke, and Martin Braun. 2018. Pandapower–An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems. IEEE Transactions on Power Systems 33, 6 (2018), 6510–6521. https://doi.org/10.1109/TPWRS.2018.2829021.

[2] Roger C. Dugan, and Thomas E. McDermott. 2011. An open source platform for collaborating on smart grid research. Proc. IEEE Power Energy Soc. Gen. Meeting, 1–7. https://doi.org/10.1109/PES.2011.6039829.

NSERC CRSNG