

Integration of Open Source Tools for Studying Large-Scale Distribution Networks

G. Valverde^{1,*}, A. Arguello¹, R. González¹, J. Quirós-Tortós¹

¹Electric Power and Energy Research Laboratory, University of Costa Rica

*gustavo.valverde@ucr.ac.cr

Abstract: The penetration of Low Carbon Technologies (LCTs) is expected to increase in the near future given the attractive incentives from governments, the cost-effectiveness of the technologies and the appearance of Smart Grid schemes. To understand the corresponding benefits and challenges associated with them, power utilities are expected to run more sophisticated studies supported by more accurate and detailed models of distribution network elements and the network itself. This paper presents the development of two software plugins that allow the integration of a Geographical Information System (GIS) platform with a distribution system simulator. These plugins are intended to provide power utilities free and open source tools to explore the benefits and/or impacts of newly adopted LCTs and to analyze Smart Grid opportunities in large-scale distribution networks. The effectiveness of the plugins is illustrated considering a real DN with 13323 customers in Costa Rica. Results demonstrate that the plugins can successfully support in the creation of detailed network models and studies.

1. Introduction

Planning studies for distribution networks have been fulfilled with the assistance of engineering network models and simulation tools that have helped understanding the network behavior under particular conditions. At present, most distribution system analysis tools are capable of running unbalanced three phase power flows and short circuit analysis, while only a few tools have the capability to perform simulations over a period of time such as day, week, month, or year [1].

Modeling of the distribution network generally ignores the secondary distribution system. Hence the analysis is usually limited to Medium Voltage (MV) level, with distribution transformers replaced by aggregated loads. These simplifications were acceptable because Low Voltage (LV) systems have been considered passive while modeling these secondary systems is not trivial.

The increasing integration of Low Carbon Technologies (LCTs), such as Distributed Energy Resources (DER), and the implementation of Smart Grid schemes at the distribution level brings new modeling requirements in which secondary LV lines and distribution transformers can no longer be neglected, but need to be modeled in detail. As thoroughly discussed in [2], modeling secondary is not easy, and it will take time and effort to accomplish. Having said that, as computer resources grow in power and capability, it is just a matter of time before all successful utilities have made this important step.

Fortunately, most of the information required to model LV elements is readily available in the Geographical Information Systems (GIS) of power utilities [3], hence processing of this GIS data is mandatory. Indeed, GIS provides the needed data management, analysis, and awareness to help the

smart grid really be smart [4]. Reference [2] outlines the data recommendations and requirements when performing analyses on secondary systems. In addition, [5] reports on a database of typical Distribution Network (DN) element parameters to be used as a complement for GIS data.

References [3] and [6] report on the creation of hundreds of LV network models from GIS data in UK. In both cases the files were exported to Matlab for the data processing and creation of the engineering network models in the corresponding power system simulation tool. However, any change in the GIS data would require a new iteration of data (files) exchange among pieces of software which may be time consuming and even tedious when dealing with long DN feeders and hundreds of secondary systems. Hence, the use of power system tools directly fed from GIS would provide great opportunities for easier handling of simulations [7].

At present time, there are many software tools to carry out distribution network studies. Although commercial software are well tested and computationally efficient, they do not allow changing the code for adding new algorithms [8], not to mention the license and update costs. Hence, the Electric Power Research Institute (EPRI) and the US Pacific Northwest National Laboratory developed and released the OpenDSS and the GridLab-D, respectively. Two sophisticated open source software tools for the future smart grid studies [9], [10].

An important drawback of these powerful tools is the fact that they do not make direct use of GIS. This is, although they may use the information from GIS as input parameters, these simulators do not interact with the GIS of the power utility.

This paper presents the integration of OpenDSS with an open source GIS platform, *Quantum GIS (QGIS)*, to carry out easier and more efficient studies of distribution networks. The integration is made possible with the creation of software plugins to a) extract GIS data to automatically build a DN engineering model and b) run OpenDSS as an embedded tool in the GIS platform. This integration is a great opportunity to keep an updated and accurate GIS representation of the network. It is expected that a Graphical User Interface (GUI) for OpenDSS will further enhance the acceptability of OpenDSS, and this opens an opportunity to power utilities and researchers to run their smart grid studies at minimum cost. The plugins will help on planning studies (load growth, network extension, DER integration) but they could also help on operating procedures if some decisions required 'on-line' simulations. In order to provide new opportunities to power utilities, research laboratories and universities to test and simulate novel smart grid solutions, the plugins have been made available in [11].

The remaining of this paper is organized as follows: Section 2 gives a brief description of OpenDSS and the QGIS and how they are integrated. Section 3 presents a detailed explanation of the network model builder using GIS data while Section 4 presents the type of DN studies that can be carried out in the GIS environment. Furthermore, a demonstration of two network studies is presented in Section 5, followed by a discussion in Section 6 of future complementary tools that will take more advantages of GIS from a smart grid perspective. Finally, Section 7 presents the concluding remarks.

2. Integration of Free and Open Source Platforms

In the last decade, open source and free software have become more and more popular in the scientific and engineering community. They have the advantage of flexibility, i.e., users can adjust the code, and customize the tools according to particular needs. In addition, they offer universal accessibility to technological advances and make it possible to accelerate software capabilities as user feedback and contributions are regularly incorporated. Indeed, open source software tools

have demonstrated to foster further developments, ideas and innovation. In the case of power system analysis, reference [12] provides a list of available open source software tool for a variety of transmission and distribution network studies. This paper reports on enhanced simulation tools for smart grid studies thanks to the integration and combination of OpenDSS and QGIS. Both software are briefly explained hereafter.

2.1. OpenDSS

OpenDSS is an open source software package developed by EPRI that can be used to simulate multi-phase AC distribution networks [9] for planning and network analysis. It was developed to support the grid modernization efforts and the integration of DER.

A key aspect of this script-driven, frequency-domain simulation tool is that it allows considering the time dimension (e.g., daily simulations with different time step) - critical to quantifying the impacts of variable sources and loads. According to EPRI, the tool was designed to be indefinitely expandable so that it can be easily modified to meet future needs and smart grid studies.

To create a network model, OpenDSS follows a sequence of definitions [9] for each network element, i.e., generators, lines, transformers and loads. Hence, given the script-written nature of this software, the creation of large-scale networks must be done carefully, and hopefully, in an automated way.

According to EPRI, the software was designed with the recognition that developers would never be able to anticipate what future users will do with it. Hence, a Component Object Model (COM) interface was implemented on the in-process server DLL version of the program. This COM server allows users to integrate OpenDSS with other software packages or programming languages (e.g., Excel with VBA, Matlab, Python) to perform more sophisticated studies. This COM server is the key feature to integrate OpenDSS and QGIS, as explained in Section 4.

2.2. QGIS

GIS are computer based systems used by many service utilities and government institutions to store, read, edit and analyze data referenced by a spatial coordinate system [13].

GIS allow a graphic representation of data. For example, power utilities use GIS to store characteristics and attributes of utility assets such as electric poles, lines, transformers and final customer electric meters, all of them geo-referenced by a coordinate system that allows to find their exact location. The object location is the common attribute among GIS elements. This information is defined according to the appropriate coordinate and projection system.

One of the most popular open source tools for geographical systems is *Quantum GIS* or simply QGIS, under the GNU-GPL license. This free software reads and analyzes *.shp files (compatible with commercial software) of GIS layers which contain information of object classes, e.g., layer of MV lines, layer of MV/LV transformers, etc.

An important feature of QGIS is that it offers the possibility to develop Python written plugins for extra data processing and analysis not available in the existing software [14]. This is the key feature to integrate QGIS and OpenDSS, as reported in Sections 3 and 4.

3. Engineering DN Model Builder

The first tool reported in this paper is called *QGIS2OpenDSS*. It was developed to extract and process GIS data of DN elements to generate the OpenDSS files in an automated way. As presented

in Fig. 1, this plugin makes use of a GUI that allows the user to select the opened GIS layers to be translated into OpenDSS files. Note that users must define a short name that is later used to define all buses and line segments. Additionally, the user must define the path where the loadshapes are located and the desired location of the resulting OpenDSS files.

The tool accepts up to three layers of MV lines, distribution transformers, LV secondary lines, LV services, LV loads and a single substation layer. It is very common to find separate files for overhead lines and underground cables. For underground cables, the user will have to tick the *UG* option shown in Fig. 1.

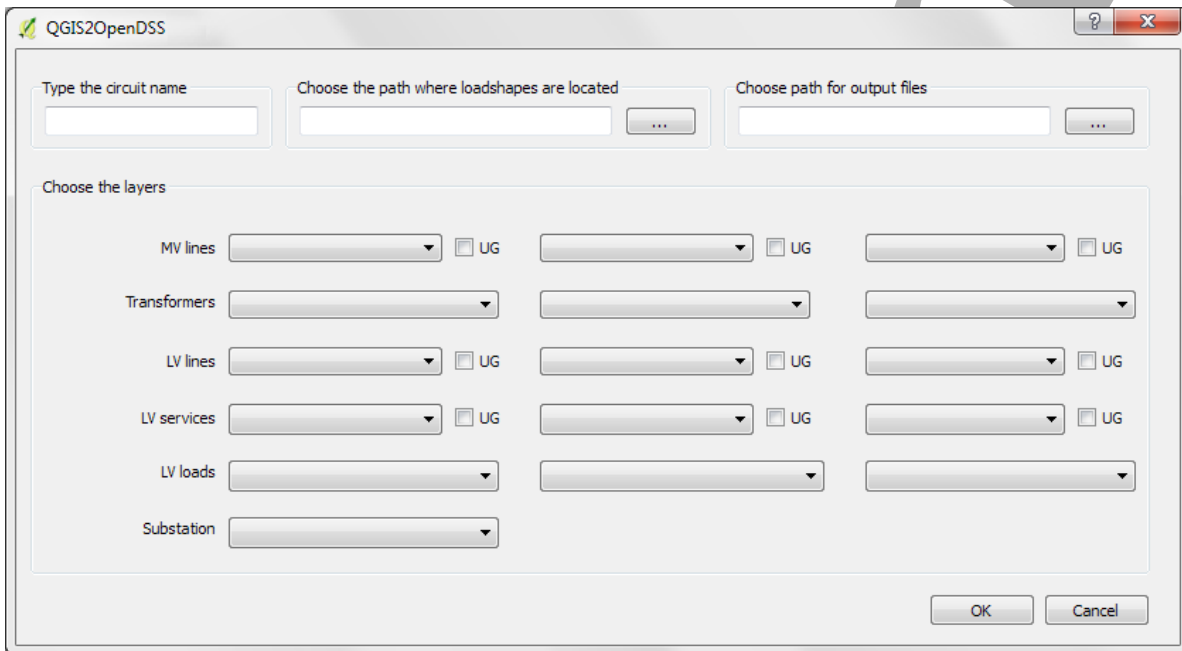


Figure 1. Graphical User Interface for QGIS2OpenDSS

3.1. Data Requirements

This plugin has the following data requirements:

- HV/MV Substation: This layer is optional and should be used when the main power transformer is to be modeled along with the feeder. A single-point layer details the number of windings, impedances, nominal voltages, kVA ratings, number of taps, and the short circuit capacity at HV level.
- MV Lines: The attributes to be found in this layer are the number of phases of the line segment and its identification i.e., *abc* for three phase segments, *ab*, *bc* or *ca* for two phase branches and *a*, *b* or *c* for single phase laterals. Additionally, information of neutral and phase conductors such as length, material and conductor size is later used to get the electrical properties of each conductor in a wire database [5]. The size can be given in Millions of Circular Mills (MCM), American Wire Gauge (AWG) or mm^2 as all attributes are treated as strings.

For overhead lines, the table of attributes must include a line spacing code or letter, e.g., *V* may refer to vertical line spacing and *H* for horizontal spacing, and so on. The plugin

concatenates strings to build a unique OpenDSS line geometry identifier [15]. For example, the identifier *3PMV477AAC3/0AAC-H* stands for a three phase MV line in horizontal spacing whose phase conductors are 477 MCM AAC and neutral conductor #3/0 AWG AAC.

Underground line segments must also include the insulating material, nominal voltage and cable type (concentric neutral or tape shielded cables). This information is used to extract the electrical properties and dimensions of cables from a database, for details see [5].

- **Distribution Transformers:** This layer must include the transformer's nominal voltages and capacity in kVA. For transformer banks, each unit must be specified with a given kVA capacity. Similar to MV lines, it must include the number and identification of phases each transformer is connected to on the MV (primary) side. In the case of three phase transformers it is imperative to include the winding connection (wye, delta, open wye, open delta).

The series impedance and no-load losses of distribution transformers are normally not included in GIS. In order to create an OpenDSS model, the plugin uses a database of typical series impedances based on the nominal voltage and capacity of transformers, see [5]. For single phase three winding transformers, an approximation is made to calculate each winding impedance based on the transformer's nameplate impedance, as explained in [16] and [17].

- **LV lines:** Apart from the length, material and size of neutral and phase conductors, this layer must include the conductor spacing, the number of phases and the operating voltage code to discriminate single phase three wire (split phase 120/240 V) from three phase lines (e.g., 120/208 V or 277/480 V). For underground LV cables, the attributes must include the insulation type and cable spacing code.
- **LV Wire Services:** The LV wire service layer must include the length, type, size and material of conductors. These strings are concatenated to define the linecode identifier in OpenDSS [15]. For example, the linecode identifier *TRPX2AAC* stands for a #2 AWG AAC Triplex cable. This identifier is later used to consult a database of electrical and mechanical parameters of service cables to calculate their corresponding primitive impedance matrices, see [5].
- **LV Loads:** This layer must include the average monthly kWh consumption and customer type: residential, commercial or industrial in order to allocate load profiles, as explained in Section 3.4.

Standardized GIS models are vital to make it possible the translation to engineering models. To the authors' knowledge, most power utilities use very similar procedures to represent network elements and store their information in GIS databases. In case that some information is not found in the list of attributes, an error message will warn the user that some attributes are missing. This message may also appear if the GIS layers use attribute names not recognized by the tool. This is easily overcome by changing the corresponding names in the table of attributes as requested by *QGIS2OpenDSS*.

3.2. Connectivity of Elements

The *QGIS2OpenDSS* connects two or more elements if their coordinates match or when they are closer than a predefined distance. This is accomplished with *Kd-trees*. A *Kd-tree* is a data structure used to organize and manipulate spatial data [18]. This structure allows to quickly find neighbor

points within a radius of a query point. The *spatial.KDTree* sub-package was used to grow the trees in *Python*. Also the following sub-packages were used:

- *tree.query_pairs(r)*: it finds all pairs of points within a distance r in the tree. This was used to connect same class elements, e.g. three phase MV lines.
- *treeA.query_ball_tree(treeB, r)*: it finds all pairs of points from tree A and tree B whose distance is at most r . This was used to connect different class elements, e.g., three phase and single phase MV lines.

For all cases, $r = 0.1$ m. This distance is used to guarantee that small errors in coordinates will not lead to disconnected DN elements.

In order to connect MV lines, the tool builds a tree for each MV line type: single phase, two and three phases. A bus name is created when two line segments ends are found to be connected using the aforementioned packages. In case that one of the segment ends is already assigned with a bus name, the other line segment end will adopt it. For example, the MV bus of transformers adopt the bus name of the MV line segment end they are connected to. In addition, the secondary buses of these transformers are assigned with a new LV bus name.

Similarly, in order to find the connectivity of LV elements, the tool builds trees for LV line segment coordinates (both ends), service coordinates (both ends) and load coordinates. Here all LV elements are compared against each other to define all the LV buses.

3.3. Identification of Errors

The information contained in GIS is very detailed. However, it contains errors that may remain unseen unless a network model needs to be built for engineering analysis. This tool identifies and reports the errors that will affect the simulations in OpenDSS. This report includes the name and coordinates of the problematic elements to facilitate the corresponding correction in the database. After all corrections are carried out, the plugin should be re-run until no errors are reported.

The most common error in GIS is the disconnection of DN elements due to small coordinate mismatches in the order of a few centimeters. If any DN element is identified (with *Kd - tree*) not connected to any other element, the plugin will report it as an isolated element.

The tool also checks for erroneous phase designations of MV lines in GIS. The first check is made when two phase line segments are connected. For example, a *bc* line segment should never be connected to an *ab* or *ca* segment, otherwise an error will be reported.

The next check is made when single phase segments are connected to two phase line segments. For example, an error is reported if a phase *c* line segment is connected to an *ab* line segment. The same procedure is repeated when connecting single phase segments only. The plugin reports errors when connecting $a \rightarrow b$, $b \rightarrow c$ or $c \rightarrow a$ segments.

Similarly, errors are reported when three phase transformers are connected to two phase or single phase lines, when two phase transformers are connected to single phase segment ends, or when the phase identification of a single phase transformer does not match the phase identification of a single phase MV line. Finally, the plugin also reports unknown capacities or unknown nominal voltages of distribution transformers. They are considered unknown values when they are not included in the library of transformer impedances.

3.4. Load Profile Allocation

QGIS2OpenDSS allocates a load profile to each customer based on the customer type and monthly energy consumption. The curves must be loaded by the user as shown in Fig. 1. These curves are scaled up or down to perfectly match the customer's monthly energy consumption. These curves are later fine tuned in the load allocation procedure to match the OpenDSS simulation with the feeder's load profile, as explained in Section 4.2.

The load profiles are created with a resolution of 10 minutes, considering different power utilities, type of day (weekday/weekend) and energy consumption range. These profiles were extracted from a statistical analysis of a nationwide measurement campaign in Costa Rica [19]. These load profiles are constantly improved as more measurements become available. It is expected that load profiles from smart meters will replace the ones obtained from the measurement campaign to assess the benefits and impacts of integrating LCTs into the grid [4].

4. Network Analyzer

The second tool reported in this paper is the *QGIS2runOpenDSS*. This plugin uses the output files of *QGIS2OpenDSS* and runs the power system analyzer OpenDSS as an embedded tool in QGIS. This is made possible thanks to the COM server that links OpenDSS with Python (see Section 4.1).

Fig. 2 presents the GUI in QGIS used to carry out advanced studies for large-scale distribution networks. The user must select the name of the circuit created by *QGIS2OpenDSS* in *Choose the circuit*. The corresponding folder will include a *master.dss* file which calls the complementary files of lines, transformers, and so on. The *substation.dss* file, if created, is used to automatically load the source voltage, angle and frequency, the short circuit capacities, and the name of the first MV bus connected to the main transformer.

The user also has to select the feeder's load shape of active and reactive power along with the respective hour and date. In order to run yearly power flows the file must include at least one year of measurements. This information is registered at substation level and stored in a single *csv* file to carry out the load allocation procedure explained in Section 4.2.

The user must define the type of study to be carried out and the folder where the results will be stored. For snapshot studies, it is required to define the simulation date and hour (available in the feeder's load shape file) while yearly power flows require the simulation resolution (e.g. 1 min, 10 min, 60 min, etc.). The user may also choose to run multiple snapshot or daily simulations if a Monte Carlo approach is to be adopted.

4.1. Integration of QGIS and OpenDSS

QGIS drives OpenDSS via the Python console available in the former. The library *comtypes.py* is used for this purpose, and this is installed using the package management system *pip*. The following steps are required for this integration:

- Open the OSGeo4WSHELL (the command window of QGIS) as administrator
- Change the path to find the file '*get-pip.py*', which should have been previously downloaded. This is done using the '*cd*' command
- Install the package management system *pip* using the following command '*python get-pip.py*'

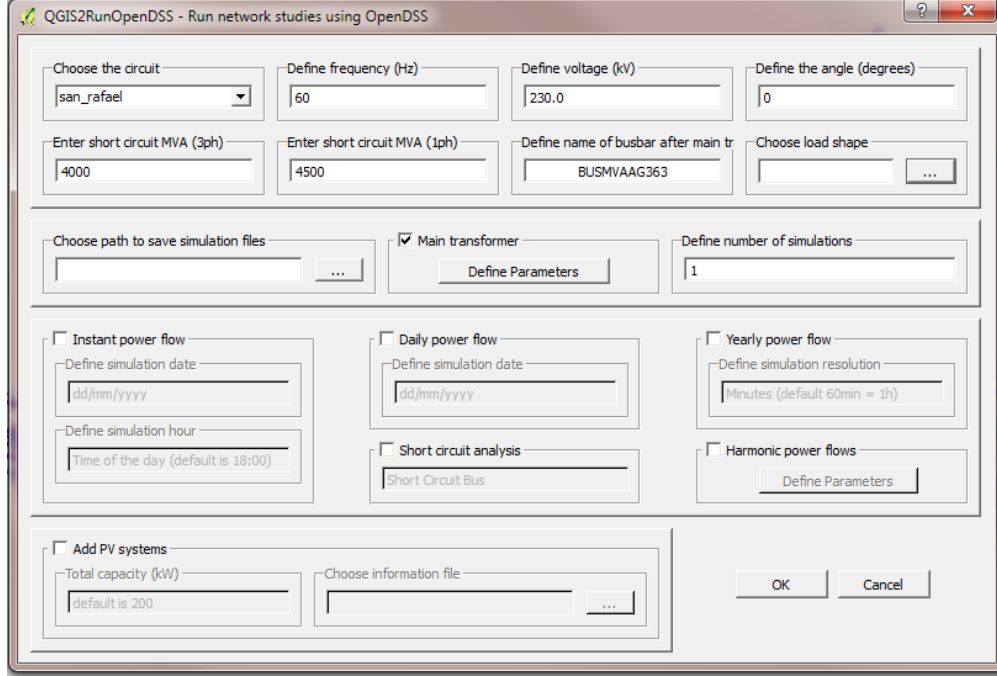


Figure 2. Graphical User Interface for *QGIS2RunOpenDSS*

- Once installed the *pip*, execute the following command to install the *comtypes*: ‘*pip install comtypes*’

It is important to mention that all these steps are automatically executed the very first time the *QGIS2runOpenDSS* is used (provided that QGIS was executed as administrator), and users are notified about the corresponding installation.

4.2. Load Allocation

The load allocation procedure tries to match the simulated aggregated demand of all customers and the network losses with the measured demand of the main feeder. This is achieved by iterating and applying a correction factor to all loads (active and reactive power), for each time instant. This iterative procedure is required as network losses are not known beforehand.

Let $P_{s_i}^{(k)}$ and P_{m_i} be the simulated and measured feeder active demand at time instant i , respectively. In the first iteration, i.e., $k = 1$, the kW correction factor is 1 so that each load is assigned with the original kW value at instant i , taken from the corresponding load profile curve. These demands are used to run an instantaneous (snapshot) power flow simulation to determine the $P_{s_i}^{(1)}$. If the difference between $P_{s_i}^{(1)}$ and P_{m_i} is larger than a predefined tolerance error, the correction factor is updated. At the k^{th} iteration, the kW correction factor for instant i is:

$$kW_{corr}^{(k)} = kW_{corr}^{(k-1)} \frac{P_{m_i}}{P_{s_i}^{(k-1)}}$$

This factor is multiplied to all load values at instant i , and a new power flow simulation is run to obtain the $P_{s_i}^{(k)}$. This iterative procedure is repeated until $P_{s_i}^{(k)} \approx P_{m_i}$. Since there are thousands of loads in the circuits, the experience has shown that only few iterative corrections are required to

fit the feeder's actual demand, as presented in Section 5.2. A similar procedure is carried out for the reactive power curve fitting.

It is expected that information from smart meters will improve the load allocation accuracy to the point that only a small correction will be applied to non-monitored loads and this will also reduce the number of iterations. This load allocation is the first step before running any other network study defined hereafter.

4.3. Types of Network Studies

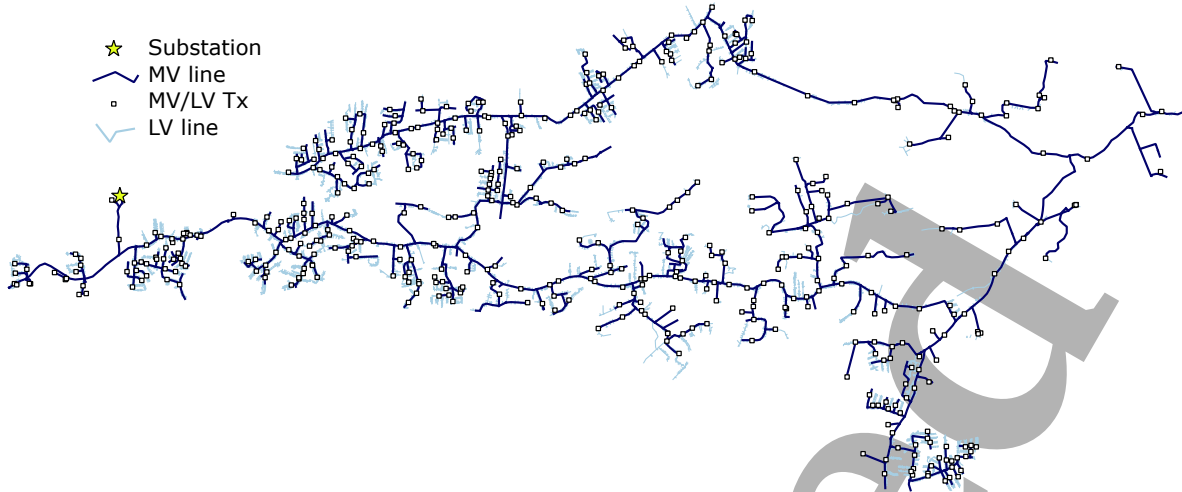
This section provides some details of the network studies currently carried out in QGIS powered by OpenDSS. The results of these studies are stored according to the user's needs in *.csv files.

- **Snapshot Power Flows:** This is the simplest study to analyze the network conditions during a specific instant (e.g., peak demand and minimum load). This approach, however, might result in under or overestimation of network conditions, particularly when considering the variability of renewable energy sources (e.g., photovoltaic systems) and loads (e.g., electric vehicles). In order to run a power flow, the user must select the corresponding snapshot box (see Fig. 2) and define the date (dd/mm/yyyy) and time (hh:mm) of the simulation.

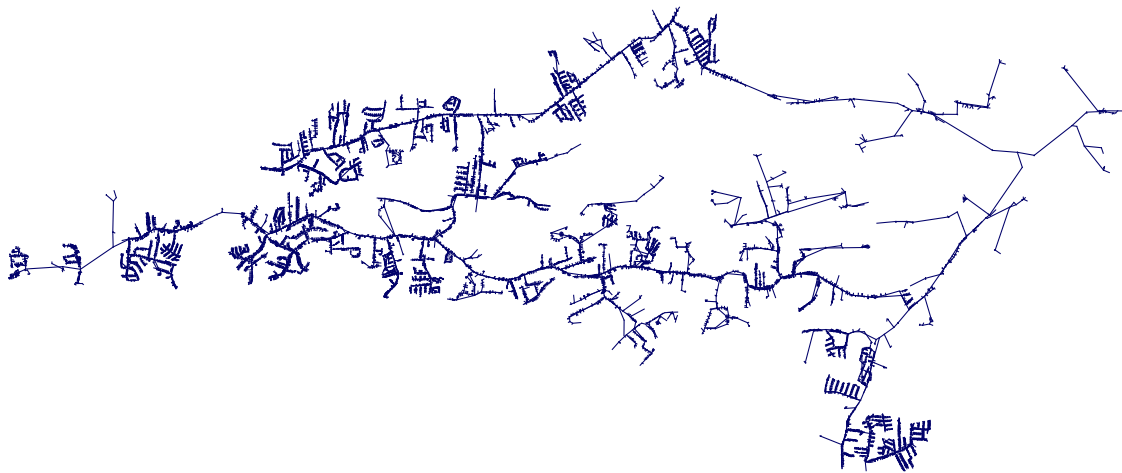
If the user sets a time that is different to the resolution of the simulation, the plugin will find the nearest available simulation instant. For instance, if the user defines 18:05 h, in a 15 minute resolution, the simulation will actually be executed for 18:00 h.

- **Daily Power Flows:** This corresponds to time-series daily power flows. This type of study is very helpful to quantify the impacts of variable sources and loads. The user needs to define the date (dd/mm/yyyy) of the year to be assessed with a given time resolution. For a yearly power flow, the user simply defines the year of interest.
- **Short circuits:** In this study, the user must define the bus name and phases to be short circuited. The user can also define a list of bus names and the plugin will create a fault for each bus in OpenDSS.
- **Harmonic Power Flows:** This study is crucial to assess the quality of the service as well as to understand the impact of harmonic distortion produced by loads and sources on distribution networks. This also allows determining the propagation of current components of frequency other than the fundamental and the resultant distortion of the voltage waveform. If this study is selected, the user must define the number of harmonics to be assessed and the spectrum of each load and source.

These studies can be used to assess the impact of high rooftop PV penetration in the distribution network. For this, the user must enable this option at the bottom of the GUI, as shown in Fig. 2. Here, it is required to define the total capacity to be installed in the circuit and upload a *.csv file with the PV capacity for each customer type and monthly energy consumption level. The list of PV capacities may be the result of a socio-economic study or based on historical data. In the first case, the user knows the most beneficial PV capacity for each customer. In the second case, the user may know the most common PV capacity installed by residential, commercial and industrial customers. This *.csv file is then used to distribute the circuit's total PV capacity among some randomly selected customers. The plugin will search for the customer's monthly energy consumption and allocate the corresponding PV capacity. This allocation stops when the circuit's total capacity is met.



(a) Original DN model in GIS



(b) Resulting DN model in OpenDSS

Figure 3. Translation of QGIS model to OpenDSS

5. Demonstration

5.1. Circuit Builder

This section presents a demonstration of the *QGIS2OpenDSS* tool. It presents the translation of a real 34.5 kV distribution circuit available in the GIS model of CNFL, one of the largest power utilities in Costa Rica. This model includes 2168 MV line sections, 7028 LV (secondary) line sections, 512 MV/LV transformers and 13323 LV loads, all of them stored in different layers. According to the GIS, there are 84 km of MV line sections and 215 km of LV line sections. Fig. 3(a) presents the GIS model of the circuit section provided by CNFL. The GIS files include all attributes required to run the plugin, as explained in Section 3.1.

After selecting the appropriate layers in the GUI, the plugin creates the OpenDSS files in less than 2 minutes on a PC Intel[®] Core i5, 4GB RAM, 32 bits.

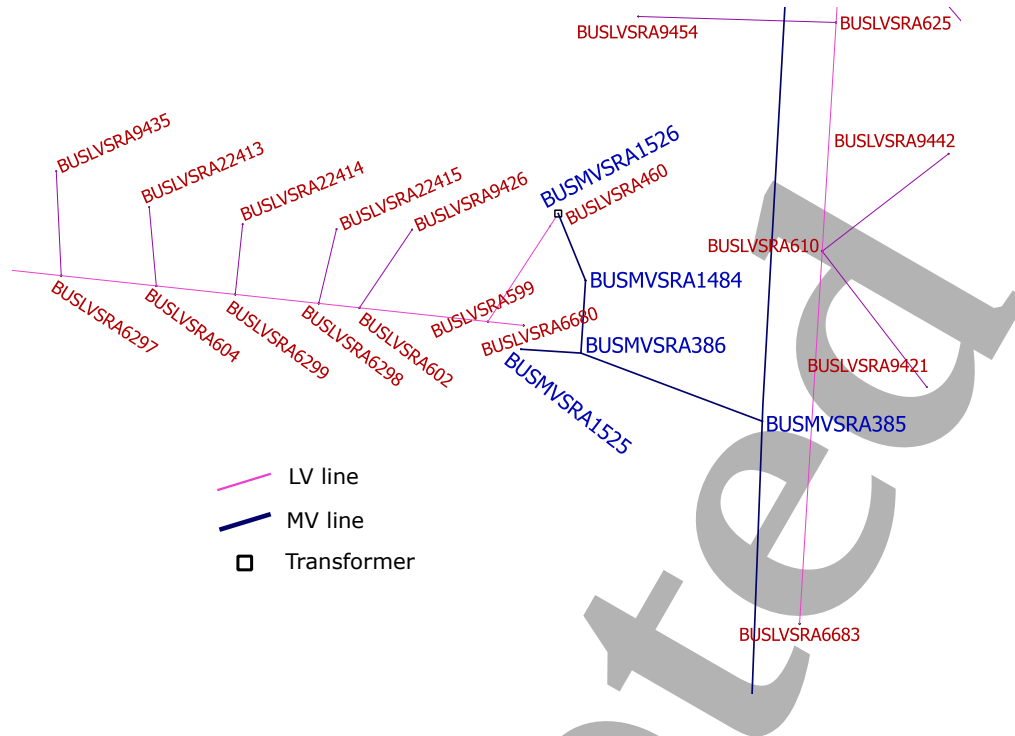


Figure 4. Visualization of bus names in GIS

The first run of *QGIS2OpenDSS* resulted in a model with errors that would result in many problems if used in OpenDSS. The plugin reported 12 disconnected MV line sections, 18 erroneous phase designations of MV lines, two three-phase transformers connected to single-phase MV lines, 12 disconnected LV line sections and 32 disconnected LV loads. Additionally, the plugin reported 2 transformers with a 0 kVA capacity. After these errors were corrected in the GIS database, a second run reported no more errors in the QGIS log message. Fig. 3(b) presents the resulting circuit plot in OpenDSS. It is seen that the engineering model truly represents the GIS model. The small differences are explained by the fact that the resulting OpenDSS model does not use polylines as the GIS model does.

The plugin output includes the *master.dss* file accompanied by the **.dss* files of lines, transformers and loads definitions. In addition, the tool creates a **.csv* file containing the list of the network buses and their coordinates. This file can be used to create a GIS layer to visualize the resulting bus names in QGIS. For illustration purposes, Fig. 4 presents the visualization of some MV and LV bus names defined by the plugin. Note that MV and LV bus names have different bus counters. Having the bus names in the GIS is very helpful when analyzing the OpenDSS simulation results, as will be explained in Section 5.3.

Table 1 presents the time requirements of the plugin to connect the DN elements according to the spatial coordinates. The remaining time is consumed by data reading, data processing and error detections. Note that most of the runtime is consumed by the connectivity of the LV line sections and service cables. The more line sections, the longer it takes to *kdtree* to identify connectivity of DN elements.

Table 1 Connectivity of DN elements

Process	Time [s]
MV line sections	2.81
MV lines with transformers	1.02
Transformers with LV line sections	2.61
LV line sections	9.01
LV lines with service cables	11.20
LV lines with loads	3.68
Service cables	20.79
Service cables with loads	7.62

5.2. Load profile allocation and correction

Based on the monthly energy consumption of customers, the *QGIS2OpenDSS* allocated a typical demand profile to each load. Almost all loads consume 300 kWh or less. Information about customer types were not available for this circuit; however and given that such small energy consumptions are typical for residential customers only, it is considered in this illustration that all customers are residential. In total the plugin allocated 135 load profiles among the 13323 customers.

As explained in Section 4.2 these curves are corrected by an iterative process to match the simulated circuit demand with the actual load curve provided by the user. Here, *QGIS2runOpenDSS* applies the same correction factor to all loads. In this way, the sum of all loads plus the circuit losses equals the measured demand of the feeder at each time instant.

Figure 5 presents the adjustment of one of the 135 load profiles. Note that the adjusted curve maintains the pattern of the original profile and only few corrections are required during the load allocation process. Here, the original load profile is scaled up or down at different time instants depending on the circuit's actual demand. As stated before, the correction is small because a) the original load profile truly characterizes the typical behavior of the corresponding load and b) any correction is distributed among the 13323 customers in the circuit. Certainly, the use of smart meters will further improve the accuracy of simulations as customers will be assigned with their actual load profiles and the allocation will be limited to those customers without smart meters.

The result of the load profile correction is presented in Fig. 6 for an arbitrarily selected day. Note that the simulation equals the measured load profile. In this demonstration all loads were assigned with a constant impedance model but any other load model could have been included instead.

5.3. Illustrative Network Studies

This section illustrates some analysis that can be carried out using *QGIS2RunOpenDSS*. The purpose of this section is to demonstrate different types of detailed studies that can be achieved by the integration of OpenDSS and QGIS. The results are not analyzed in detail as this is out of the scope of the paper. Due to space limitations only three studies are presented: snapshot, daily, and PV impact.

The first illustrative analysis carried out is the snapshot power flow. In this case, it is required to check the LV bus voltages at 19:00 h (peak load condition). For this, the *instant power flow* option was used along with day and hour of interest, as shown in Fig. 2. The plugin alerts the user when the simulation is over. This simulation takes about 2 s due to the massive number of LV buses.

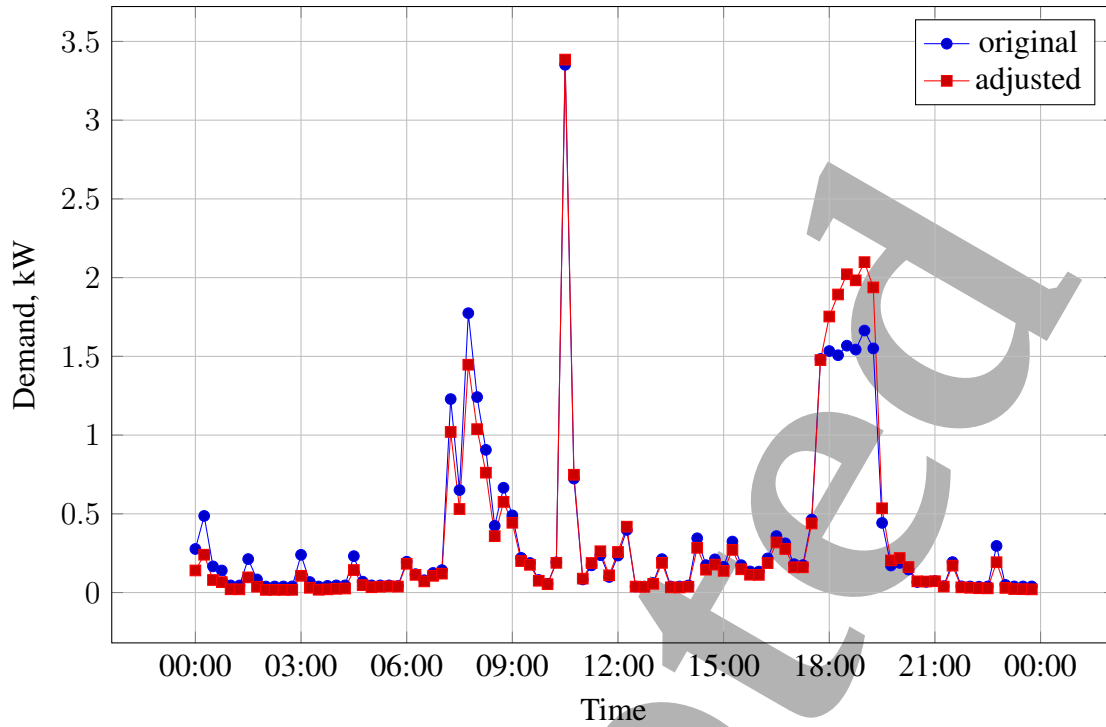


Figure 5. Correction of individual load profile carried out by QGIS2runOpenDSS

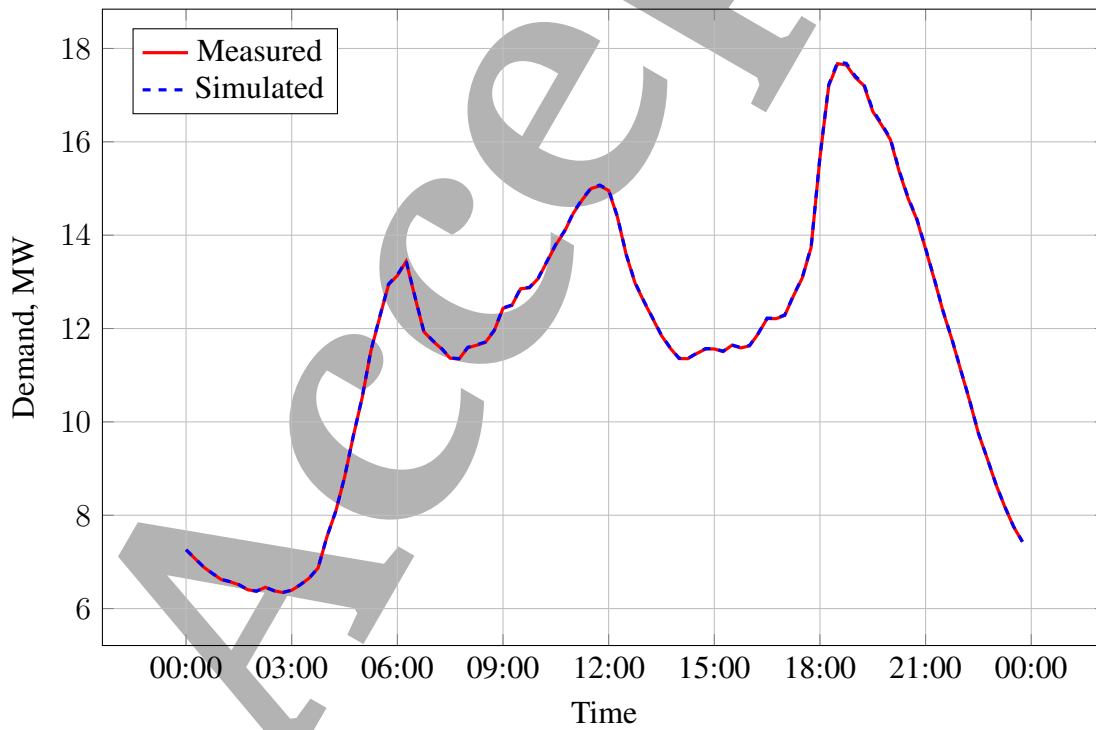


Figure 6. Comparison of actual versus simulated feeder's load profile

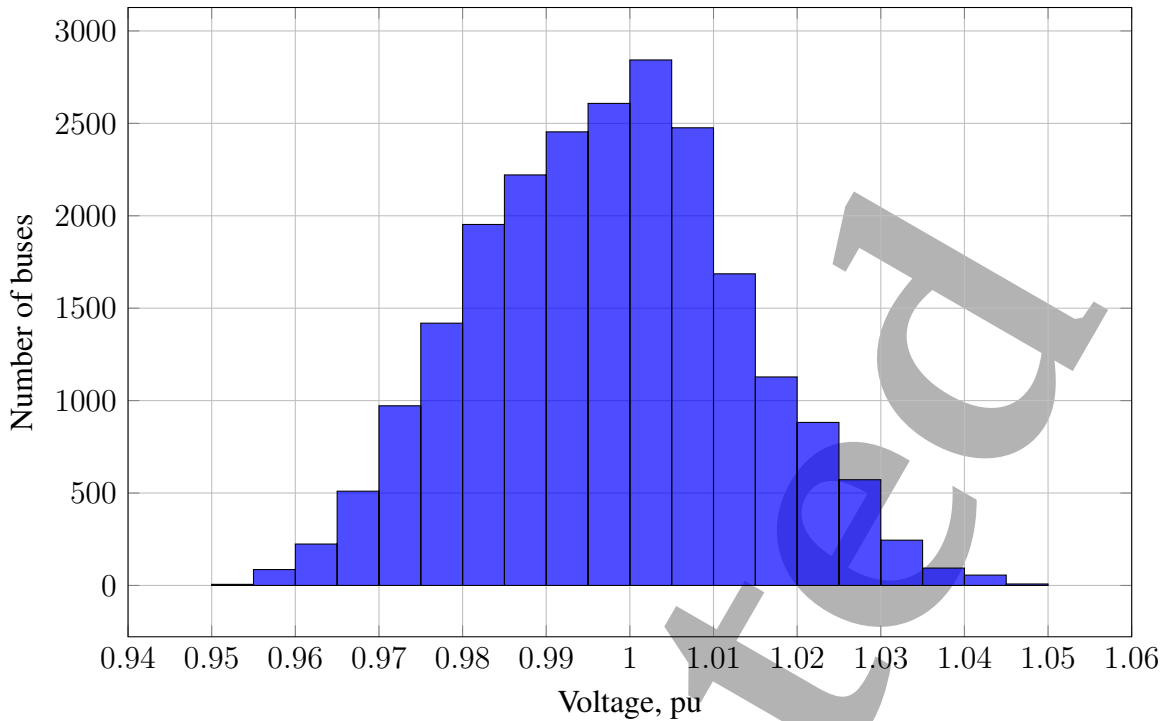


Figure 7. LV bus voltages at peak load (snapshot power flow)

This includes the time needed to correct the original load profiles.

Figure 7 shows the histogram of LV bus voltages as a result of the snapshot power flow simulation. Note that none of the buses are beyond the $\pm 5\%$ bandwidth. In fact, most LV buses are within $\pm 3\%$ of nominal voltage. *QGIS2RunOpenDSS* can also display the current and voltages through monitored transformers and lines. However, due to space limitations they have not been included.

To run a daily power flow, the user must select this option in the GUI of *QGIS2RunOpenDSS*. The daily simulation (in a 15-min resolution) takes on average 50 s, including the runtime of the load allocation algorithm. *QGIS2RunOpenDSS* allows users to export the simulation results of any element (transformer, line and/or cable) in the circuit. For this, the user only needs to select the corresponding monitor from the pull-down list of monitors. Fig. 8 presents the variation of all LV buses throughout the day. All results are presented in box plots which include median value (horizontal line), first and third quartile, and the outliers. As can be seen, all bus voltages remain within the acceptable band of operation even at peak load conditions. The buses with the highest voltages are those close to the MV/LV transformers while the lower voltages are found at the tails of the secondary circuits. The lower voltages occur at peak loading conditions. Note also that the bus voltages presented in Fig. 7 are in agreement with the box plot at 19:00 h.

5.4. PV allocation and impact study

Table 2 presents the PV capacities chosen for residential customers based on their monthly energy consumption. This information was included in a *.csv file to be read by *QGIS2runOpenDSS*, as presented in Fig. 2. According to a socio-economic study carried out for the power utility, residential customers with a monthly energy consumption equal or less than 300 kWh will not install a PV system. Based on this, the maximum expected installed capacity of rooftop PV systems

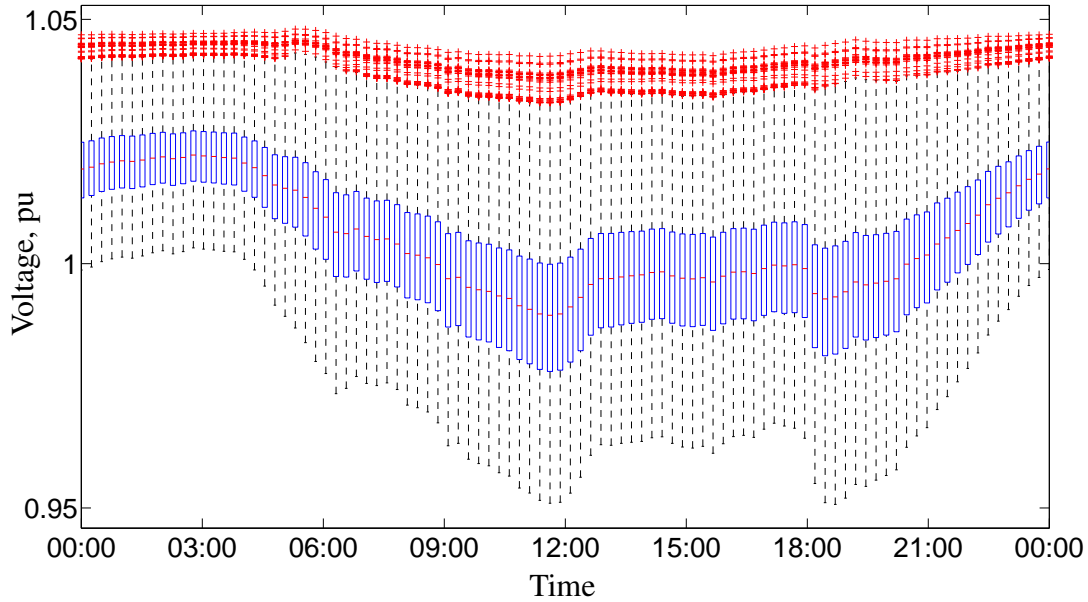


Figure 8. LV bus voltages through out the day (daily power flow)

Table 2 Information required by plugin to assign the PV systems

Monthly consumption (kWh)	Capacity (kW)
> 300	0.00
300 - 320	0.75
330 - 360	1.00
370 - 380	1.25
390 - 680	3.25
690 - 950	5.25

is 1700 kW throughout the circuit.

The plugin randomly allocates these PV systems among the customers until the requested penetration level is met. Figure 9 presents the simulation result of the active power demand when the 1700 kW are installed in the circuit. When compared to the base case, it is seen that the circuit demanded less power from 6:00 to 18:00 h. This small penetration level is not expected to largely affect the operation of the network.

Although not shown here, the voltages increased during the day thanks to the active power injection of rooftop PV systems, when compared to the base case.

6. Discussion

As demonstrated before, detailed studies of future DNs with LCTs will be easily carried out thanks to the integration of GIS and advanced simulation tools such as OpenDSS. In addition, results of socio-economic studies can help to simulate more realistic scenarios. These tools could also be extended to assess the network condition in the presence of PV and energy storage systems along with irradiation layers to better represent the solar potential at a given geographic area. In addition,

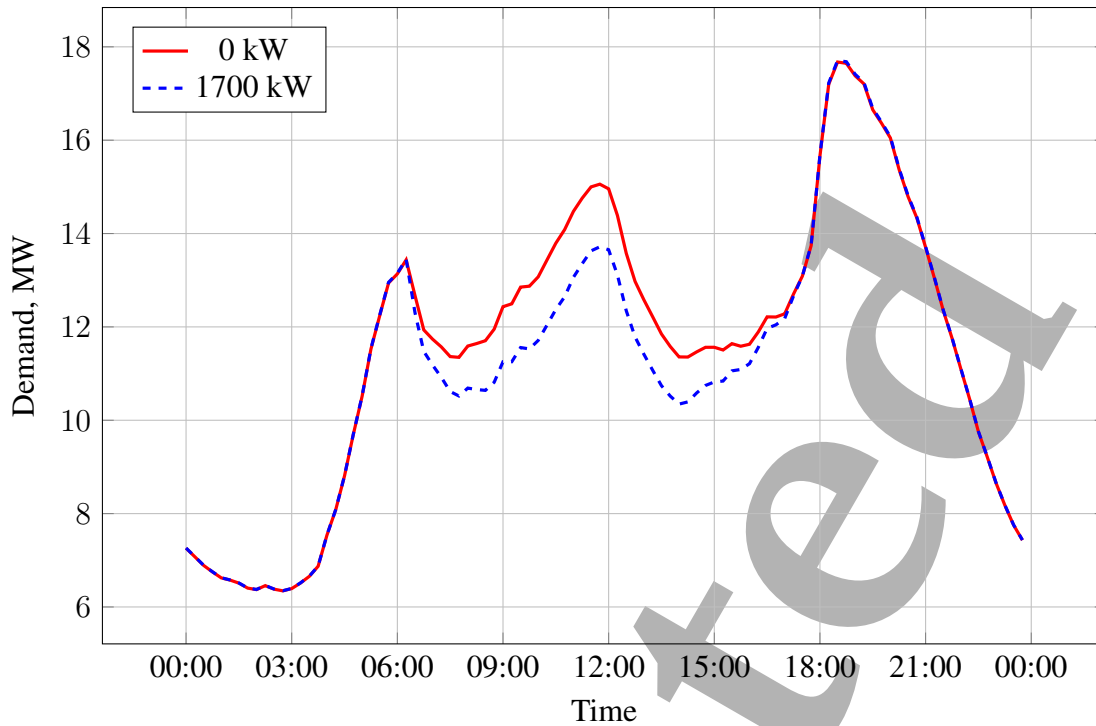


Figure 9. Feeder's load profile with PVs

the integration of transportation layers, could also be used when studying the integration of Electric Vehicles (EVs) into the grid and the optimal location of fast charging stations.

The key to carry out these studies is high quality GIS data. This information needs to be cleaned from spurious data that could result in unrealistic engineering network models. Indeed, the correction of errors is not straightforward and requires knowledge of how GIS network models are built and how they have to be translated for power engineering uses.

From authors experience, most errors are found in secondary systems due to the presence of many short line segments and also because LV systems were not included in traditional studies run by power utilities. To overcome this, the aforementioned tools should be complemented with new plugins able to automatically correct frequently found errors such as disconnected LV elements or nonexistent connections among different secondary circuits. For example, *Kd-tree* could be used to automatically identify service cables inside a circle centered at the disconnected load. If a service cable was found inside the circle, the coordinate of the disconnected load should be corrected to match the coordinate of the service cable end. Similar procedure could be carried out to connect line segments. Unfortunately, not all disconnection problems will be automatically corrected as some cases require human intervention to decide where to connect a disconnected element.

The identification of connected elements is likely to be the most important step when building the engineering model in *OpenDSS*. As reported in Section 3.2, this task was successfully achieved thanks to the *Kd-tree* algorithm. However, the authors believe that the same task could be accomplished by more efficient computational tools based on Graph theory. In this regard, future work will focus on reducing the time requirements for the connectivity of network elements.

To take full advantages of GIS software capabilities, future work will also be focused on visualization of simulation results. This is, the simulation results currently shown in plots will be

visualized in the GIS network model. For instance, overvoltages due to massive integration of PVs or under voltages caused by high EV penetrations at LV level could be represented in the GIS environment by heatmaps or different scale colors. In addition, some GIS properties of DN elements could be also modified (e.g., line thickness or point size) to represent highly loaded or even overloaded lines and transformers.

Finally, detailed simulations of smart distribution networks will be further improved when load profiles from smart meters replace the typical profiles currently used by the developed tools. This will also require the integration of databases fed from smart meters with the QGIS software.

7. Conclusion

This paper has presented the integration of OpenDSS with an open source GIS platform to carry out easier and more efficient studies of distribution networks. Two software plugins were created to extract GIS data to automatically build a distribution network engineering model and run OpenDSS as an embedded tool in the GIS platform. These plugins aim at facilitating the studies that power engineers have to run in the short term by integrating free and open source GIS software with existing free and open source power engineering software. In addition, they have great potential to be used for educational purposes at university level, and for training of junior engineers.

The performance of the simulation tool has been demonstrated on a real circuit in Costa Rica with more than 10000 customers. The illustrative examples highlight the efficiency and convenience of the tool to perform detailed network studies.

The release of these tools is expected to bring new opportunities to power utilities, research laboratories and universities to test and simulate novel smart grid solutions with more realistic network models. In addition, future collaborations from other users will enhance the robustness and generality of the tools.

8. Acknowledgment

The authors would like to thank CNFL for providing the GIS database of the analyzed circuit.

9. References

- [1] R. Arritt and R. Dugan, "Distribution System Analysis and the Future Smart Grid," *IEEE Transactions on Industry Applications*, vol. 47, no. 6, pp. 2343–2350, Nov. 2011.
- [2] G. Shirek, B. A. Lassiter, W. Carr, and W. H. Kersting, "Modeling Secondary Services in Engineering and Mapping," *IEEE Transactions on Industry Applications*, vol. 48, no. 1, pp. 254–262, Feb. 2012.
- [3] A. Navarro-Espinosa, L. Ochoa, R. Shaw, and D. Randles, "Reconstruction of Low Voltage Distribution Networks: From GIS Data to Power Flow Models," in *23rd International Conference on Electricity Distribution*, no. 1273. CIRED, 2015.
- [4] B. Meehan, *GIS for Enhanced Electric Utility Performance*. Artech House, 2013.
- [5] R. Brenes, A. Arguello, J. Quiros-Tortos, and G. Valverde, "Distribution Network Element Model Parameters: Creation of Database," in *IEEE CONCAPAN*, Nov. 2016.

- [6] A. Crossland, N. Wade, and D. Jones, “Extraction of 9,163 Real LV Network Models from DNO GIS Database to Assess Overvoltage from PV and Consequent Mitigation Measures,” in *23rd International Conference on Electricity Distribution*, no. 0915. CIRED, 2015.
- [7] S. Lazarou, V. Vita, P. Karampelas, and L. Ekonomou, “A Power System Simulation Platform for Planning and Evaluating Distributed Generation Systems based on GIS,” *Energy Systems*, vol. 4, no. 4, pp. 379–391, Apr. 2013.
- [8] F. Milano, *Power System Modelling and Scripting*. Springer, 2010.
- [9] R. Dugan and T. E. McDermott, “An Open Source Platform for Collaborating on Smart Grid Research,” in *IEEE PES General Meeting*, 2011.
- [10] D. P. Chassin, K. Schneider, and C. Gerkenmeyer, “GridLAB-D: An Open-Source Power Systems Modeling and Simulation Environment,” in *2008 IEEE/PES Transmission and Distribution Conference and Exposition*, April 2008, pp. 1–5.
- [11] EPERLab, “Release of plugins: Integration of Open Source Tools for Studying Large-Scale Distribution Networks,” 2017. [Online]. Available: <http://eperlab.eie.ucr.ac.cr/documents> [Available from Jul. 2017]
- [12] Task Force on Open Source Software for Power Systems. Open Source Software. IEEE PSACE Committee. [Online]. Available: http://ewh.ieee.org/cmte/psace/CAMS_taskforce/software.htm
- [13] P. Bolstad, *GIS Fundamentals, a First Text on Geographical Information Systems*, 4th ed. Minnesota EEUU: Eider Press, 2002.
- [14] QGIS, *PyQGIS Developer Cookbook*, February 2016. [Online]. Available: <http://docs.qgis.org>
- [15] R. Dugan, *Reference Guide: The Open Distribution System Simulator*, EPRI, June 2013.
- [16] W. H. Kersting, *Distribution System Modeling and Analysis*, 3rd ed. CRC Press, 2012.
- [17] T. Short, *Electric Power Distribution Handbook*. CRC, 2004.
- [18] J. L. Bentley, “Multidimensional Binary Search Trees Used for Associative Searching,” *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, Sept. 1975.
- [19] J. Quiros-Tortos, A. Arguello, and G. Valverde, “Statistical Analysis of Residential Demand Behavior in Costa Rica: Creation of Load Profiles,” in *IEEE CONCAPAN*, Nov. 2016.