

Distribution Network Model Builder for OpenDSS in Open Source GIS Software

P. Quesada, *Graduate Student Member, IEEE*, A. Arguello, *Graduate Student Member, IEEE*,
J. Quirós-Tortós, *Member, IEEE*, and G. Valverde, *Member, IEEE*

Electrical Power and Energy Research Laboratory
School of Electrical Engineering, University of Costa Rica
11501-2060 UCR, San José, Costa Rica
gvalverde@eie.ucr.ac.cr, jquiros@eie.ucr.ac.cr

Abstract—This paper presents a tool to automatically generate distribution network models for the power engineering software OpenDSS. The tool was developed in a free and open source Geographical Information System (GIS) software that allows a direct translation of GIS to OpenDSS files. The paper shows the potential of the developed piece of software and how it can help power engineers to carry out detailed studies of medium and low voltage networks. It also explains data requirements in the GIS database and all the steps followed to create the OpenDSS files. In order to demonstrate the capabilities of the model builder, the paper presents the translation of a real distribution circuit GIS model to OpenDSS.

Index Terms—Data extraction, distribution network modelling, geographical information system, kd-tree, OpenDSS, open source software, Python, QGIS, smart grids.

I. INTRODUCTION

In order to assess and study the impact of disruptive technologies such as photovoltaic systems, storage systems and electric vehicles on Medium Voltage (MV) and Low Voltage (LV) networks, it is necessary to have advanced simulation tools and detailed models of the Distribution Network (DN) and its components.

Software packages such as OpenDSS [1] and GridLabD [2] are open source tools with advanced modeling techniques and high performance computing capabilities. They can easily handle hundreds and thousands of MV and LV line segments, including single, two phase and three phase connections and they have the ability to run time series power flow studies.

Simulation results of these pieces of software highly depend on the quality and availability of network data (type, size and length of conductors or location and capacity of distribution transformers) which is typically stored in the GIS of power utilities. Hence, the interrogation and extraction of GIS data is a key step required to build DN models [3].

Reference [4] presents a network model builder for more than 9000 LV networks, in the north west of the UK. The GIS files were exported to Matlab where the extraction of information and creation of network models takes place. However, computational time requirements were not reported. These models were later used to assess the voltage rise in LV networks due to rooftop Photovoltaic (PV) installations. In [5], the authors report the creation of more than 500 UK

LV network models from GIS data exported to Matlab. The Breadth First Search (BFS) algorithm was used to connect line segments.

The work in [6] reports on the creation of MV feeder models in Matlab for OpenDSS and CYMDIST (a commercial software package) extracted from GIS data to assess the impact of PV systems. The authors used the Minimum Spanning Tree algorithm to reduce the network model to facilitate dynamic analysis of distribution feeders with high PV penetrations.

The aforementioned references required the use of third software package to translate the GIS information to the respective power flow 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. Hence, the use of power system tools directly fed from GIS environment provides great opportunities for easier handling of simulations [7].

This paper reports on a software plugin that creates the OpenDSS network model directly from an open source GIS software environment. The plugin not only avoids the exportation of files but allows to easily update the network model in case that a change or correction is made in the GIS database. The plugin is able to report disconnected elements and wrong or missing data. In addition, it automatically allocates a demand profile to each load based on the type and energy consumption of customers. This plugin is part of a series of tools under development that are helping power engineers to analyze Smart Grid technologies [8].

The remaining of this paper is as follows: Section II includes a brief summary of GIS, the open source software QGIS and the *QGIS2OpenDSS* plugin. Section III details the required list of attributes for each DN element in order to build its model in OpenDSS. In addition, Section IV explains how DN elements are connected based on their coordinates while Section V lists the type of errors the plugin can report. Moreover, an explanation of the load profile allocation is presented in Section VI while the printing of the OpenDSS files is presented in Section VII. Finally, a demonstration of the plugin capabilities is shown in Section VIII and conclusions are drawn in Section IX.

II. GEOGRAPHICAL INFORMATION SYSTEMS

Geographical information systems 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 [9].

GIS allows 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.

One of the most popular open source tools for geographical systems is *Quantum GIS (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 [10]. The *QGIS2OpenDSS* is one of those plugins developed at the University of Costa Rica to speed up the modeling of DN circuits. This plugin extracts and processes GIS data to automatically generate the OpenDSS files required to run the simulator. To make it easier for the user, the plugin contains a Graphical User Interface (GUI) that facilitates the translation of data.

Fig. 1 presents the GUI of *QGIS2OpenDSS*. So far, the plugin is available in Spanish and English versions. The language selection is made automatically depending on the detected operating system language.

The user must define a code or short name (without spaces) of the feeder to be modeled. It is recommended to use, at most, three letters as they will be later used to name all buses and line segments. In addition, the user will need to define the directory path in which the newly created OpenDSS files will be stored. Failing to do so will not allow the user to build the DN model.

The GUI allows the user to select the layer of DN elements to be modeled in OpenDSS. The layers will appear in the GUI if those layers were previously opened in *QGIS*. To differentiate underground cables from overhead lines, the user must indicate so in the check box next to the selected layer.

If the user decided to create LV load models, the plugin would ask for the path where the *.csv files of load profile curves are located. This allows the plugin to run the load profile allocation, as explained in Section VI.

III. DATA REQUIREMENTS IN QGIS2OPENDSS

The object location is the common attribute among DN elements. This information is defined according to the appropriate coordinate and projection system. So far, *QGIS2OpenDSS* works with the CRTM05-WGS84 projection. However, other projections can be used as long as their Cartesian coordinates are given in meters.

This section summarizes the additional list of attributes for each DN element required to build the DN model in OpenDSS. In case that one or more of the following attributes are not

found in the list of attributes, an error message will pop up to warn the user that some attributes are missing. Nonrecognition may occur if the attribute name in the table does not match the attribute name in the plugin. This is easily overcome by changing the corresponding names in the table of attributes.

A. MV Lines

The table of attributes must include the number of phases of the line segment and its identification e.g. *abc* for three phase systems, *ab*, *bc* or *ca* for two phase systems and *a*, *b* or *c* for single phase systems. In addition, it is required to include the length, material and size (in American Wire Gauge (AWG) or mm²) of neutral and phase conductors of each line section.

The coordinates of all receiving and sending ends are also used to perform the connectivity of MV lines, as presented in Section IV.

Overhead lines and underground cables are read from separate GIS layers. In case of overhead lines, the table must include a line spacing code or letter, see Fig. 2: *V* may refer to vertical line spacing and *H* for horizontal spacing, and so on. This information along with size and material of conductors will be used to create the corresponding line geometries in OpenDSS. For underground line segments, the attributes should include a line spacing code, the insulation type (EPR, XLPE, PVC) and nominal voltage of cables. Finally, the cable type may be required in case that both concentric neutral and tape shielded cables are used.

B. MV/LV Transformers

This layer must include the number and identification of phases, winding connections for three phase transformers (wye, delta, open wye, open delta), nominal voltages and capacity in kVA. When modeling multiple transformer in a bank configuration, each unit must be specified with a given kVA capacity. Similar as for MV lines, the transformer coordinates are used to connect transformers with MV and LV line segments, as explained in Section IV.

The series impedance and no-load losses of distribution transformers are normally not included in GIS, specially for old transformers [3]. In order to create an OpenDSS model, the plugin uses a look up table of typical series impedances based on the nominal voltage and capacity of transformers, as listed in Appendix A of [11].

C. LV Lines

The table of attributes for LV lines must include 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). In addition, it is required to include the length, material and size of neutral and phase conductors.

Similar to MV lines, overhead and underground lines need to be stored in separate layers. In the case of single phase overhead LV lines, they have a unique vertical spacing of 20 cm between conductors, as presented in Fig. 2. For underground LV cables, the attributes must include the insulation type and cable spacing code.

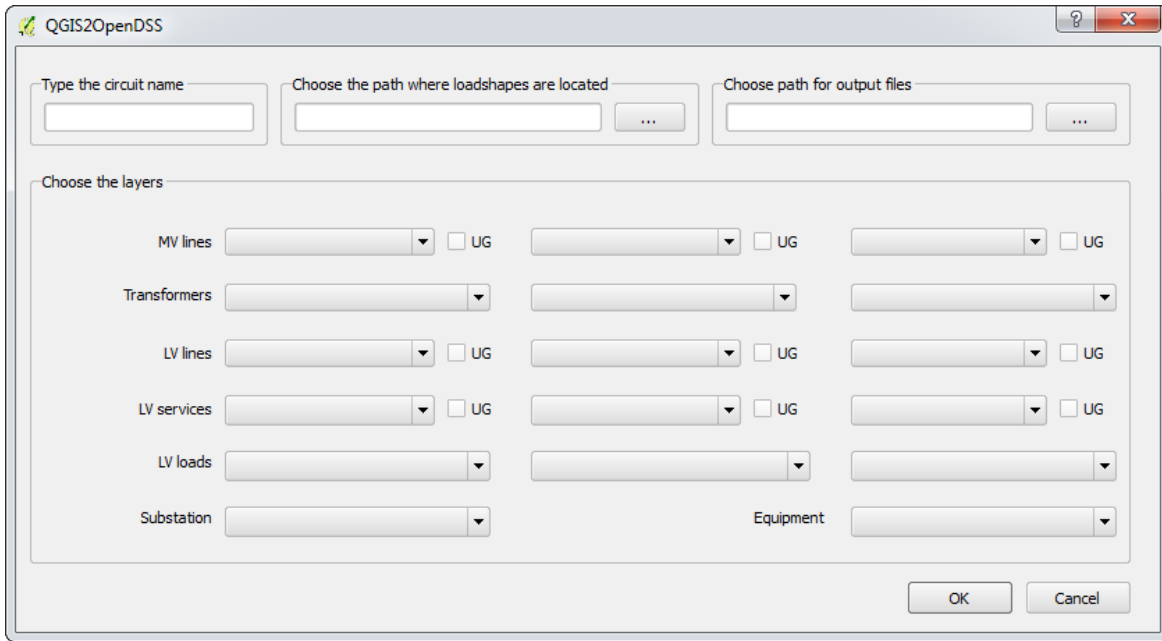


Figure 1. Graphical User Interface

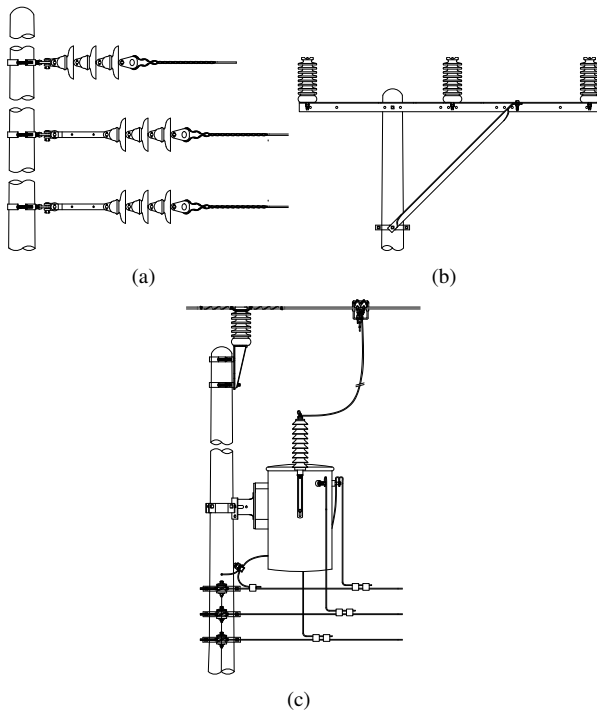


Figure 2. Line spacing: (a) Vertical MV line, (b) Horizontal MV line, (c) single phase MV line and single phase LV line

Note that the coordinates of all receiving and sending ends are also used to perform the connectivity of LV lines with MV/LV transformers, LV services and loads.

D. LV Wire Services

The data required to model the LV services includes configuration, size, length and material of conductors. The con-

figuration consists of a representation of the way the wires are constructed (Duplex, Triplex, or Quadriplex). The coordinates are used to connect LV services with primary LV lines and loads.

E. LV Loads

The representation of consumers is an important component when modeling secondary networks [3]. *QGIS2OpenDSS* asks for the monthly kWh consumption and customer type: residential, commercial or industrial. This information is vital to allocate a load profile for each consumer, see Section VI. Additionally, the nominal voltage and number of phases are taken from the LV service to which the load is connected to.

F. HV/MV Substation

A single-point layer details the number of windings, nominal voltages, kVA ratings, number of taps and winding impedances. Other pieces of equipment such as capacitors or regulators will be included in *Equipment*.

IV. CONNECTIVITY OF ELEMENTS

Two elements are considered electrically connected if their coordinates match or when they are closer than a predefined distance. In this regard, *QGIS2OpenDSS* uses *Kd-trees* to speed up the identification of connected elements.

The *spatial.KDTree* sub-package was used to build the trees in *Python*. Also the following sub-packages were used:

- *tree.query_pairs(r)*: 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)*: 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.

A. Connectivity of MV Lines

The plugin internally makes a distinction between single, two phase and three phase lines to sequentially connect MV line segments. At first, the plugin uses the list of coordinates (at both ends) of three phase line segments and creates a tree. Then, a list of connected three phase line segment ends are found with $tree.query_pairs(r)$.

A bus is created when two line segments ends are identified. For this, the algorithm assigns a new bus number to line segments ends that have not been connected before. In case that one of the segment ends is already assigned with a bus number, the other segment end will adopt it.

Coordinates of two phase MV line segments are also used to grow a tree. Then, two phase line segments are found connected with three phase segments using $treeA.query_ball_tree(treeB, r)$. All two phase segment ends identified in this step are assigned with the bus number of the corresponding three phase segment ends they are connected to.

Similar to three phase MV lines, connections of two phase segments are found with $tree.query_pairs(r)$. And again, a new bus number is created if those segment ends were not previously connected. Otherwise, they will be assigned with the bus number of the segment ends they are connected to. This procedure is repeated for single phase line segments which have to be compared with two and three phase segments.

B. Connectivity of Transformers

MV/LV transformers are classified into single phase units, two phase banks, three phase units and three phase banks whose coordinates are used to create the corresponding trees. Each tree is compared against MV line trees to identify the primary bus each transformer is connected to.

The MV bus of transformers will adopt the bus number of the MV line segment end they are connected with. In addition, the secondary buses of these transformers are assigned with a new LV bus number, independent of the MV bus counter.

C. Connectivity of LV Lines, Services and Loads

In order to build LV networks it is required to grow trees for LV line segment coordinates (both ends), service coordinates (both ends) and load coordinates. At first, LV segment ends are associated with transformer by means of $treeA.query_ball_tree(treeB, r)$. After this step is completed, LV segment ends connected to transformers are assigned with the LV bus number of the transformers they are connected to.

Later, a list of connected LV segment ends are found with $tree.query_pairs(r)$. The algorithm assigns a new LV bus number to line segments ends that have not been connected

before. In case that one of the segment ends is already assigned with a bus number, the other segment end will adopt it.

The LV service coordinates are also compared with transformer and LV segment coordinates. This comparison allows to assign a bus number to sending ends of LV services. In addition, $tree.query_pairs(r)$ is used to connect and assign a bus number to LV service segments connected among them.

Finally, the load coordinates are used to grow the loads tree. The latter is compared against LV line, LV service and transformer trees to assign an LV bus number to all loads.

V. DETECTION OF ERRONEOUS GIS DATA

GIS usually contains erroneous data which may remain unseen until a model is built for engineering software use. This plugin identifies and reports these errors along with the coordinates of the problematic elements, so that users can modify the GIS database. The plugin should be re-run until no errors are reported.

A. Disconnected Elements

The most notorious error in GIS database is the disconnection of DN elements due to small coordinates displacements which may be in the order of a few centimeters. If both ends of the same MV or LV segment are found disconnected, the plugin will store and report it as an isolated line segment.

In case that a transformer or LV load is found not connected to any line segment end, the plugin will also report it as an isolated element.

B. Wrong Phase Designations

As explained in Section III, attribute tables of MV line segments and transformers have to include the phase identification. The plugin uses this information to check if the elements to be connected have the correct phase designation, otherwise an error is reported.

The first check is carried out when two phase line segments are connected. Here, the phase identification of both line segments must be the same. This is, a line segment with phases ab should never be connected to a bc or ca segment. This type of error in database should be corrected before the OpenDSS files are printed.

The second check is performed when single phase line segments are connected with two phase segments. For example, an error is reported if a phase a line segment is being connected to a bc 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.

The classification of transformers based on number of phases allows to detect wrong phase designations when they are connected to MV segment ends. For this, the plugin reports when three phase transformers (unit or bank) are connected to single or two phase segment ends, 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 segment end.

C. Missing Transformer Models

The plugin reports unknown capacities or unknown nominal voltages of MV/LV transformers. They are considered unknown values when they are not included in the look up table of distribution transformer impedances. The typical reported error is the existence of 0 kVA transformers or rounding of standard values, e.g. 35 or 40 kVA instead of 37.5 kVA. These errors need to be corrected in the GIS database in order to successfully complete the transformer model in OpenDSS.

VI. LOAD PROFILE ALLOCATION

For each load type, the plugin finds the largest kWh consumption to create an array of quantized energy values, starting from 25 kWh up to the largest monthly kWh value in fixed steps, e.g. in steps of 25 kWh for residential loads.

Each customer is associated with a quantized energy value that will be later used to assign one of the available load profile curves. For example, suppose a fictitious case of five residential customers whose monthly energy consumptions are 824.21, 29.22, 221.74, 117.41 and 185.51 kWh. The plugin would associate them a quantized energy value of 825, 25, 225, 125 and 175 kWh, respectively.

The plugin also loads a set of available daily load profiles for each load type and calculates, by integration, their quantized energy value. This allows to easily associate customers with the available load profiles. If a customer has a quantized energy value without an associated curve, the program will create a new load profile curve by scaling up or down the closest available load profile.

As a matter of example, Fig. 3 presents a sample of available residential load profiles. The associated quantized energy values are Curve1 (red): 150 kWh, Curve2 (blue): 125 kWh, Curve3 (green): 825 kWh and Curve4 (yellow): 175 kWh. Recalling again the previous example of five residential customers, the red curve would be left unused and two customers (25 kWh and 225 kWh) would require an estimated load profile. To do so, the algorithm would take the blue and yellow curves to scale them down and up. Note that a large variety of available load profiles will increase the possibilities to build a more accurate allocation. This allocation is used as input for the OpenDSS file creator, explained below.

VII. OPENDSS FILE CREATOR

Once all elements are connected and assigned with a bus number, the plugin uses the list of attributes presented in Section III to create the OpenDSS files. The output files are *MVlines.dss*, *Transformers.dss*, *LVlines.dss*, *LVservices.dss*, *LVloads.dss* and *Loadshapes.dss*. Moreover, the plugin creates a list of buses with the corresponding coordinates which are printed in a *.csv file. This file is used to plot the circuit in OpenDSS as will be shown in Section VIII.

The bus number of DN elements is taken from the connectivity algorithm explained in Section IV. This number is added to the name of the circuit and voltage level identifier (i.e. MV or LV). In addition, the phase identification of DN elements is converted to OpenDSS format: phase *a*, *b* and *c*

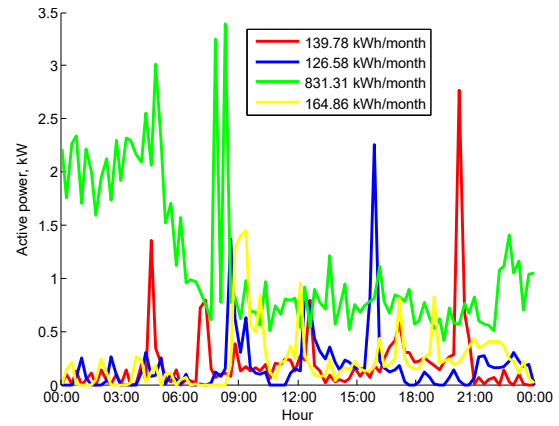


Figure 3. Daily load profiles

become .1 .2 and .3, respectively. This identification is used to define the number of nodes in *bus1* and *bus2* of lines and transformers in OpenDSS, for details see [1].

The geometry names of MV and LV lines include the number of phases, the voltage level identifier, material and size of phases and neutral conductor as well as the line spacing code. For example, *3PMV477AAC3/0AAC_H* stands for a three phase medium voltage line with 477 MCM AAC phase conductors and 3/0 AWG AAC neutral conductor in horizontal line spacing. Note that an additional *.dss file of line configurations needs to be added to specify the particular geometry, spacing and wiredata of lines and cables used by each power utility. In this file, the user also has to define the LV services (Triplex and Quadriplex cables) characterized by a linecode.

The *Transformers.dss* file also includes the typical impedances (taken from the look up table) and the winding connections. Single phase split phase transformers and three phase banks are connected in this step depending on the winding connections specified in the table of attributes.

Finally, the *LVloads.dss* includes the bus name, phases, nominal voltage, connection type (wye or delta) and demand curve. It is important to say that the kW value and power factor of all loads are set to 1. These values could be changed externally to fit the sum of all loads plus losses with the main feeder load profile.

VIII. DEMONSTRATION

To date, this plugin has been used to translate real distribution network GIS models of two power utilities to analyze in OpenDSS the impact of large penetration of rooftop PV systems. The analyzed circuits ranges from 5000 to 26000 customers and the plugin execution time averaged 60 s approximately, on a PC Intel® Core i5, 4GB RAM, 32 bits.

This paper presents the translation of a small portion of a large distribution circuit available in the GIS database of CNFL, one of the largest power utilities in Costa Rica. This 34.5 kV circuit section includes overhead lines and underground cables serving residential and few commercial customers in an extension of 2.5 kilometers.

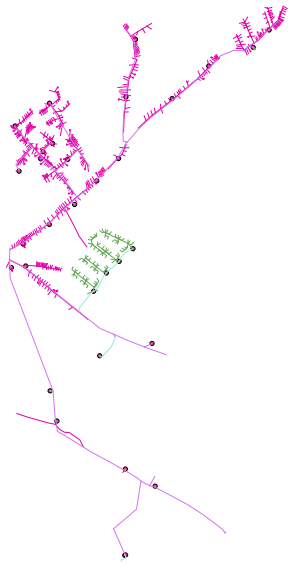


Figure 4. GIS model provided by the power utility

Fig. 4 presents the GIS model of the circuit section provided by CNFL. The small circles represent MV/LV transformers and the lines represent MV and LV line sections. This model is made of 126 MV buses and 1400 LV buses. There are 71 3ϕ line sections, 55 1ϕ MV line sections, 32 MV/LV transformers, 1373 LV line sections and wire services, and 654 LV loads, all of them stored in different layers.

After selecting the appropriate layers in the GUI, the plugin creates the OpenDSS files in less than 5 s. Most of the runtime is consumed by the connectivity of the 1373 LV line sections. The more line sections, the longer it takes to *kdtree* to identify connectivity of DN elements. Fig. 5 presents the OpenDSS circuit plot. It is seen that the OpenDSS model truly represents the GIS model.

IX. CONCLUSION

In order to assess the potential benefits of smart grid technologies and to better understand the impacts of new agents such as PVs, storage and electric vehicles at MV and LV level, it is imperative to have more sophisticated tools to model distribution networks as best as possible.

This paper reports on a software plugin that creates the OpenDSS network model directly from an open source GIS software environment. The plugin not only avoids the exportation of files but allows to easily update the network model in case that a change or correction is made in the GIS database. In addition, this plugin is able to report disconnected elements and wrong or missing data, and automatically allocates a demand profile to each load based on the type and energy consumption of customers.

This plugin is the first of a list of applications under development aiming at facilitating the studies that power engineers have to run in the short term by integrating free

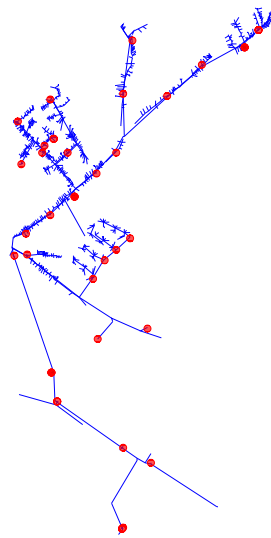


Figure 5. OpenDSS model created by QGIS2OpenDSS

and open source GIS software with the existing free and open source power engineering software.

X. ACKNOWLEDGEMENTS

The authors would like to thank Mr. Raúl Fernández and Mr. Guido Godínez, affiliated to CNFL, for providing the circuit's GIS database.

REFERENCES

- [1] R. Dugan, *Reference Guide: The Open Distribution System Simulator*, EPRI, June 2013.
- [2] K. Schneider, D. Chassin, Y. Chen, and J. Fuller, "Distribution power flow for smart grid technologies," in *IEEE/PES Power Systems Conference and Exposition*, March 2009, pp. 1–7.
- [3] 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.
- [4] 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. CIREN, 2015.
- [5] 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. CIREN, 2015.
- [6] R. Ayyanar and A. Nagarajan, "Distribution system analysis tools for studying high penetration of pv with grid support features," Arizona State University, Tech. Rep., September 2013.
- [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] R. Gonzáles, A. Arguello, G. Valverde, and J. Quirós-Tortós, "Openss-based distribution network analyzer in open source gis environment," in *IEEE PES Transmission and Distribution Latin America*, Morelia, Mexico, 2016.
- [9] P. Bolstad, *GIS Fundamentals, a First Text on Geographical Information Systems*, 4th ed. Minnesota EEUU: Eider Press, 2002.
- [10] QGIS, *PyQGIS developer cookbook*, February 2016. [Online]. Available: <http://docs.qgis.org>
- [11] T. Gonen, *Electric Power Distribution System Engineering*. Mc Graw Hill, 1986.