Software application implement in Java for electrical lines dimensioning

CRISTIAN ABRUDEAN, MANUELA PANIOIU
Electrical Engineering and Industrial Informatics Department
Polytechnic University of Timisoara, Engineering Faculty of Hunedoara
Revolutiei str. no 5, code 331128, Hunedoara
ROMANIA

Abstract: - In this paper it was present a software package implemented in java useful for dimensioning of low voltage lines mono phase and three phases (AC and DC). The modeled networks are tree type. This software with graphical user interface allows the estimation and calculation of parameters of power transmission lines of electric power. The user has the possibility to input of data and other characteristics of the electric line, passing to the calculation stage when the data are correctly and completely entered. The output of the results is shown on the screen.

Key-Words: -computer software, electrical lines, optimal dimensioning, Java

1 Introduction
The dimensioning of power transmission lines parameters of electric power has an essential importance in the design, construction and simulation of electric power systems. The results obtained are used for the studies of power flow, short circuit and stability. The design of transmission lines is often the key factor for the existence or absence of failures caused by lightning. The monitoring system of the high-voltage transmission tower is a new feature of the state maintenance for transmission lines [1]. It is designed to prevent the materials of transmission tower from being destroyed or stolen. Many engineering studies are usually performed by electric power utilities for the design of new transmission lines [1], [3], [6]. In this paper it was present a software package implemented in java useful for dimensioning of low voltage lines mono phase and three phases (ac and dc). The package is loaded with an easy-to-use graphical user interface that makes it suitable for the interactive input of the geometry and output of the parameters.

2. The electric networks modeling
It was use tree-type networks for modeling. There is a single node, root node, conventional numbering with 0. In this root node there is a voltage source assume ideal. There are not loop links. It is possible to connect 0 or many other nodes to a node and also 0 or many users.

The nodes are characterized by: connecting length between nodes, the conductor material, the isolation and the maximum temperature. It is possible to impose that the connections for a node to be made using a conductor with the same section like the one to be connected.

The users are characterized by active power, efficiency, admissible drop voltage in normal regime of functioning and in starting regime, the starting coefficient and the power factor (only in AC.).

The electrical networks are represented using trees and the dimensioning calculus is optimized for a minimum material consume.

The conditions that must be accomplished by the connections of nodes are:
1. The node connection accomplishes the thermal criteria. This are the first applied criteria because by using this criteria a minimum section value are impose.
2. The drop voltage in nominal functioning regime must be less than admissible value of whatever cup plying users.
3. The start drop voltage must be less than admissible value for each user (It assume that one by one user are started and the other ones remain in nominal functioning regime).

Because an electrical network is in generally the same values for some characteristic (i.e. temperature or material) it was provided the possibility to use defaults values – global for entire grid – for these.

The drop voltage for a section in mono phase ac or dc is:
\[ \Delta U \approx 2 \cdot \rho \frac{l}{s} \cdot I_a + 2 \cdot x\cdot l \cdot I_r, \quad (1) \]

Where \( \Delta U \) is drop voltage (loss voltage), \( \rho \) is material resistivity, \( l \) = the conductor length, \( s \) is the conductor section, \( x \) is specific reactance, \( I_a \) is the active current and \( I_r \) is the reactive current (zero in DC).

The drop voltage for ac three phase (in case of all users being symmetrical)

\[ \Delta U \approx \rho \frac{l}{s} \cdot I_a + x\cdot l \cdot I_r, \quad (2) \]

where \( \Delta U \) is the voltage loss on a phase.

The current for a user are:

- in DC:
  \[ I = \frac{P}{\eta \cdot U}, \quad (3) \]

- in AC single phase:
  \[ I_a = \frac{P}{\eta \cdot U} \cdot \cos \varphi; \quad (4) \]
  \[ I_r = \frac{P}{\eta \cdot U} \cdot \sin \varphi \quad (5) \]

- in AC three phase:
  \[ I_a = \frac{P}{3 \cdot \eta \cdot U} \cdot \cos \varphi; \quad (6) \]
  \[ I_r = \frac{P}{3 \cdot \eta \cdot U} \cdot \sin \varphi \quad (7) \]

Using line voltages was avoided in order to obtain a model close to the mono phase ac model and for avoiding the calculus with \( \sqrt{3} \).

### 3. The software package. The classes

3.1 ErrorReport

This class is use for exceptions report when using Java virtual machine. This is the usual situation. The class constructor use a parameter a Throwable object or a subclass and also a boolean variable. The boolean variable is use for decide if the application are terminated.

The Throwable.printStackTrace() method print the error message in a PrintStream object.

3.2 Stas

The class name suggests their destination. The class contains constants, arrays of constants, searching methods or interpolation methods. All methods and variables are static. This class are use for electrical network modeling. The class contains inner classes Grid.Node for nodes and Grid.Node.Consumer for users. The inner classes can use references to the class that encapsulate them (i.e. in Grid.Node can be use Grid.this) They cannot be instantiate out of the Grid class. The Grid class contains a series of set and get methods use for private member access and default values for entire network. These values will be used for current values or for initialize new nodes or new users depending of some boolean variables. So, by example if the variable etaImplValue = 0,8 and isEtaImpl = true then the efficiency of all consumers will be individually establish. If a new consumer will be created, the efficiency for this consumer will be \( \eta = 80\% \).

Grid.Node.precalc() method calculate the total active and reactive currents and the admissible loss voltage based on the users (consumers) characteristics. Also, it determines a minimum section based on thermal criteria and based on a simplified admissible drop voltage criteria (it calculate the minimum section for obtaining the admissible drop voltage only for this node). The purpose of this method is to accelerate the calculus by limit the possible section set. There is a recursive method.

The Grid.Node.calculate(double u, double r, double x) method performed all calculus in order to obtain a minimum volume. The \( u \) parameter is the voltage node to which is made the connection. The parameters \( r \) and \( x \) are the cumulate resistance and reactance between node and source and are used for start checking’s. The return value is the total volume for this section and for all sub-nodes, or the value 0 if the dimensioning criteria cannot be accomplished. The method search for a section that verified the dimensioning criteria and then increase the section until obtain 3 values in increasing order. Then the method return to the section for which was obtained the minimum volume. Grid and Grid inner classes implement java.io.Serializable interface used for load/save files.

3.3MainFrame

This class are derived from javax.swing.JFrame and is the graphical user interface for the application. The constructor for this class use an argument an Grid object. This object will be data source for GUI. The GUI are depicted in fig. 1. By using the “Calculate” button the dimensioning calculus will be made. For the electrical network from fig. 2, the calculus are performed in approx. 0.005 s. If the network is deep with many nodes and branches, the time need for computing may be longer (seconds or even minutes).
The inner class `VolumeLabel` is derived from `javax.swing.JLabel` and used for total volume displaying using a mark out format.

The inner class `SerializationFileFilter` is derived from `javax.swing.filechooser.FileFilter` and used for file filters in Open/Save file dialogs.

The inner class `ColorCellRender` is derived from `DefaultTableCellRenderer` (from package `javax.swing.table`). This class is used for modifying the swing component used for displaying some cell tables. The cells corresponding to default values (cannot modifying from the main window) are display on a grey-blue background. The cells corresponding to the computed results are displayed on a yellow background. The cells displayed on a dark gray background are irrelevant for the modeled network (i.e. \( \cos \varphi \) in DC). If a cell value are identify to be a floating point number, will be displayed with maximum 3 decimals.

`OptionsFrame` are a separate window and can be close/open using “global values” button. This window is show in fig. 3. The second column contains checkbox controls for implicit values.

The tables (`JTable`) are associate with data models being a handler between table and dates that can be displayed or modify. This class (`OptionsFrame`) contains two inner classes (`OptionTable` and `OptionTableModel`).

`NodeTable` implement the table of nodes. The row 3 from fig. 1. can be interpret like: “the node 3 are connected with the node 2 using a cable with length 40 m, the cable material are copper, isolation PVC, ...
the functioning temperature 25°C. The results are: section 35 mm² and a nominal voltage 4,163 %".

The model for this table is an inner class. The model communicate with the instance for class Grid transmitted by MainFrame constructor. ConsTable implement the users (consumers) table. Row 6 from fig 1 can be read like this: "The users 6 are connected to node 6, this util power is 8000W, the efficiency 90 %, the admissible loss voltage 5%, the admissible loss voltage on start is 12 %, the start coefficient 4 , $\cos \varphi = 1$ (irrelevant in DC). Results a loss voltage on start 11,805%.

Both NodeTable and ConsTable contain a inner class PopupListener, used for open popup menus. Such a menu are show in fig. 4.

---

The table ConsTable offer a single option – for deleting the selected consumer. The described tree from the previous example is depicted in fig. 2. The nodes numbering cannot be controlled through the interface. The assigned nodes numbers (also consumer’s numbers) describe the introducing order only. For avoiding errors, the numbering must respect the following rule: when a number is assigned to a new node it must be connection between the new node ant the tree root. In the tree from fig. the node 3 cannot be introduce before the nodes 1 and 2.
4 Conclusion

This software package is very useful for electrical grid dimensioning for single phase and three phase electrical grids and also for DC and AC. The software can be also useful as educational software for students in electrical engineering.

References: