

Power System Demos: A Graphical Aid for Lecturing and Training Purposes

Pieter H. Schavemaker, *Member, IEEE*, Robert Reijntjes, and Lou van der Sluis, *Senior Member, IEEE*

Abstract—A software program is described for demonstration purposes during the lectures in the undergraduate course “Power System Analysis I” at the Delft University of Technology. The software visualizes certain modeling and computational aspects of the power system analysis and gives the student insight into the effects of certain actions without making elaborate computations. The students can make copies of the software to study and practice with it.

Index Terms—Courseware, education, power systems.

I. INTRODUCTION

AT THE Delft University of Technology, the undergraduate course “Power System Analysis I” is based on the first 9 chapters of the book “Power System Analysis” by Grainger and Stevenson [1]. The graduate course “Power System Analysis II” is based on the later chapters of this book.

One of the difficulties in teaching power system analysis, is that there are a lot of items involved that are rather difficult to imagine such as: phasors, reactive (imaginary) power, apparent power, three phase systems, travelling waves and so on. Furthermore, often rather tedious computations have to be made to show the effects of certain modeling and control actions. In order to visualize certain modeling and computational aspects of the power system analysis, and in order to give the student insight into the effects of certain actions without making elaborate computations, software has been developed to support the text materials. Furthermore, the classes get more attractive and diverse when the software is used to support the verbal lecture. The software follows the line of the book by Grainger and Stevenson. Therefore, several demonstrations have been developed to support the material of the text book.

The software is developed in Delphi 3/4/5 under the Windows 95/98/NT operating system. The software includes the power systems demonstration program—which is the main program—and the demonstrations per chapter. The demonstrations are compiled to dlls (dynamic link libraries) which can be loaded by the main program. Therefore, several people can develop demonstration programs and compile them to dll-files, without the necessity to recompile the whole program. Another advantage is that the main program can be used by several groups, writing their demonstration software in the pre-defined dll-structure. An advantage for the student is that the software modules can be gained from the internet. Therefore, only the

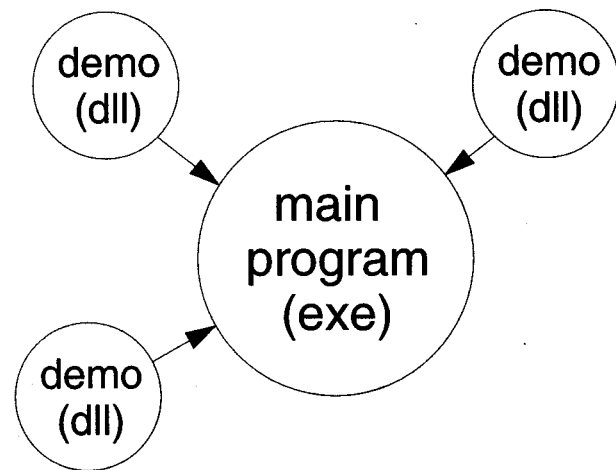


Fig. 1. Power systems demos: program structure.

demonstration software needed for a particular course has to be downloaded.

II. SOFTWARE STRUCTURE

The demonstration software has been written in Delphi 3/4/5 under the Windows 95/98/NT operating system. The Delphi programming language enables the programmer to create a very user-friendly interface to the software.

The software has been built up of one main program (the executable) and the demonstration programs which are stored in dlls (dynamic link libraries). The main program and demonstration dlls can be obtained from the internet. There are several advantages to this approach:

- several people can develop demonstration programs and compile them to dll-files, without the necessity to recompile the whole program
- the main program can be used by several groups, writing their demonstration software in the dll-structure in whatever programming language they prefer
- the students can obtain the software module needed for one particular course (or part of it) without downloading a huge program (i.e., a program that contains all of the demonstrations).

The program structure is shown in Fig. 1. The layout of the main program is shown in Fig. 2. The interface for adding dlls to and removing dlls from the main program is shown in Fig. 3. The addition of demonstration programs has to be done only once: after that time (or after closing the main program), the list with loaded dlls is remembered. The Help-menu changes dynamically depending on the demonstration program that is

Manuscript received June 19, 2000; revised February 12, 2001.

The authors are with the Electrical Power Systems Group, Delft University of Technology, Delft, The Netherlands.

Publisher Item Identifier S 0885-8950(01)09422-6.

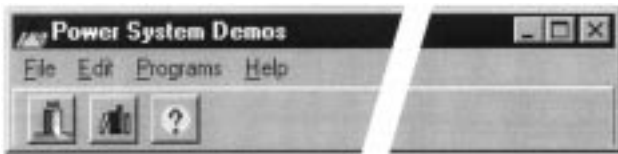


Fig. 2. Power systems demos: main program.

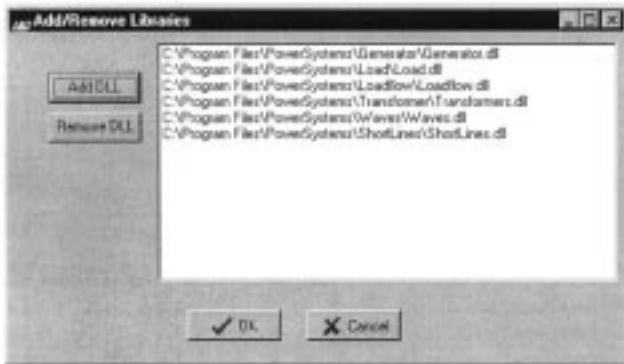


Fig. 3. Power systems demos: add/remove demonstration programs.

run. The help text and accompanying graphics are displayed in a web-browser.

III. DEVELOPED DEMONSTRATION DLLS

In order to visualize certain modeling and computational aspects of the power system analysis, and in order to give the student insight into the effects of certain actions without making elaborate computations, software has been developed to support the theory in the text book [1]. The developed demonstrations are described hereunder.

A. Basic Concepts

In order to make the relation between time-varying quantities and phasors more clear, a program has been developed that allows the user to vary the R and X -values of a series load that is connected to a fixed voltage. Furthermore, the concepts of instantaneous, active and reactive power and the power triangle become transparent. The user interface of the Load-demo is shown in Fig. 4.

B. Transformers

The demonstration program Transformers is used to make the concept of phase shifting, as it occurs with the various $Yy/Yd/Dy/Dd$ -connections that are possible with three phase transformers, visible. The user can specify the type of three phase transformer and the phase shift (in hours) from drop-down boxes. The connection of the primary and secondary windings are shown with the corresponding voltages in a clock-like diagram as shown in Fig. 5. Every secondary voltage of the transformer is in phase with a corresponding primary voltage (in other words: a coil on the secondary side is always magnetically coupled with one of the coils on the primary side). Of course, the phase difference between the voltage phasors of a three phase system is $4 \times 30 = 120$ degrees. Therefore, the phase-difference between primary and secondary

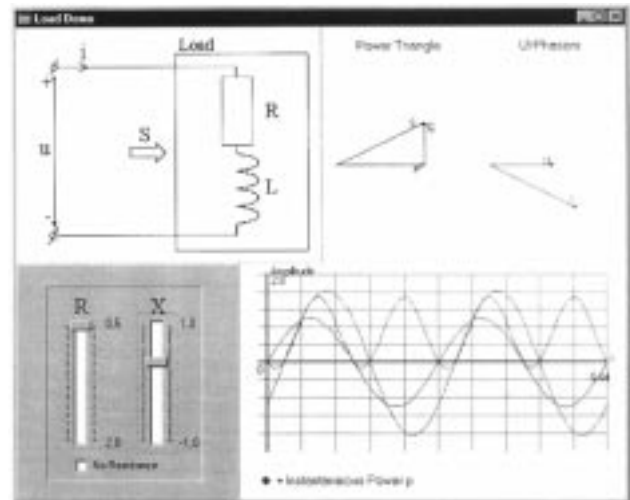


Fig. 4. Power systems demos: load demonstration.

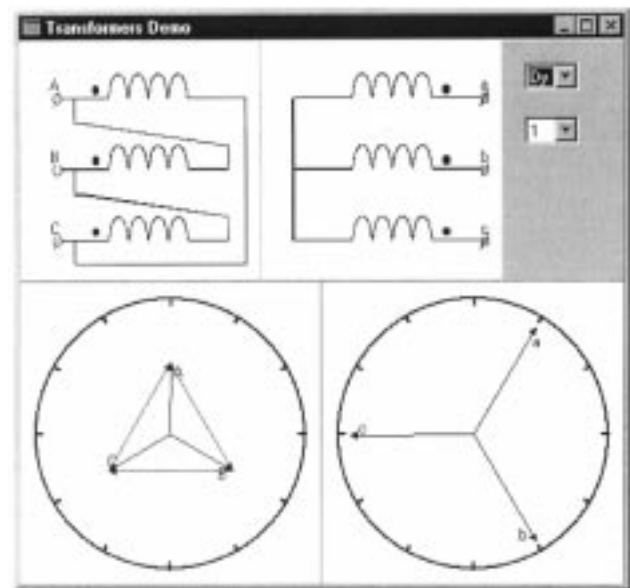


Fig. 5. Power systems demos: transformer demonstration.

voltages is always equal to a multiplicity of 30 degrees. As the hours on a clock are distributed around a circle with angles of 30 degrees, it is possible to express the phase difference between the primary and secondary voltages in hours, as is common practice in the Netherlands.

C. Synchronous Machine

The Generator demo is shown in Fig. 6. The generator is connected to an infinite bus. Therefore, the terminal voltage and the speed (of rotation) are fixed and unalterable. The generator can be governed by two controls:

- the excitation (by means of the "Voltage Trackbar")
- the mechanical torque on the shaft (by means of the "Power Trackbar").

The effect of these controls on the amount of generated active and reactive power can be monitored. Furthermore, the user can see what trajectories the phasors follow in the depicted phasor diagram.

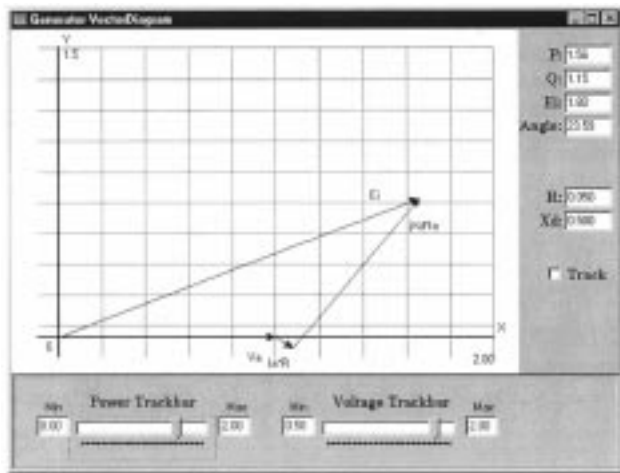


Fig. 6. Power systems demos: generator demonstration.

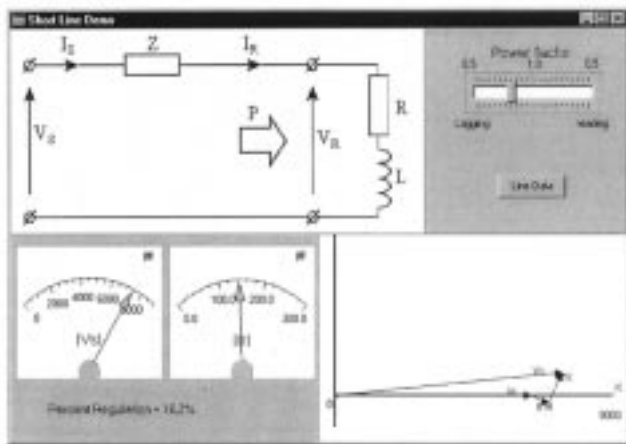


Fig. 7. Power systems demos: shortlines demonstration.

D. Current and Voltage Relations on a Transmission Line

For this topic, three demonstration programs have been developed. The first program shows the effect of different loads on the sending end voltage of a short line. The user can vary the power factor of the load that consumes a fixed active power at a constant receiving end voltage. The change in sending end voltage and the current are shown by means of two meters. Furthermore, the voltage regulation (i.e., the rise in voltage at the receiving end, expressed in percent of full-load voltage, when full load at a specified power factor is removed while the sending end voltage is kept constant) is calculated and displayed. The Shortlines demonstration program is shown in Fig. 7.

The second program shows the effect of the line length on the application of the three different line models: short line (series impedance only), medium line (nominal- π circuit) and long line (equivalent- π circuit). The user can vary the line length by using the dragbar or typing a value in the edit box under the dragbar. The line parameters are adjusted to the new line length and the new sending end voltage is computed (the receiving end voltage is kept constant while a constant active power is consumed). These values are shown in the graphical representations of the line models. The user is also able to change the series-impedance per mile and the shunt admittance per mile of

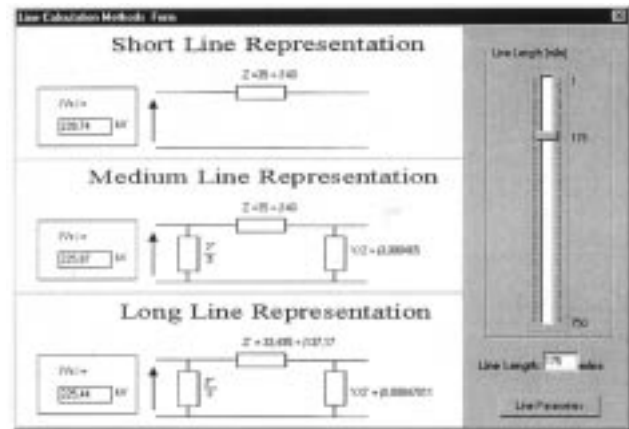


Fig. 8. Power systems demos: lines demonstration.

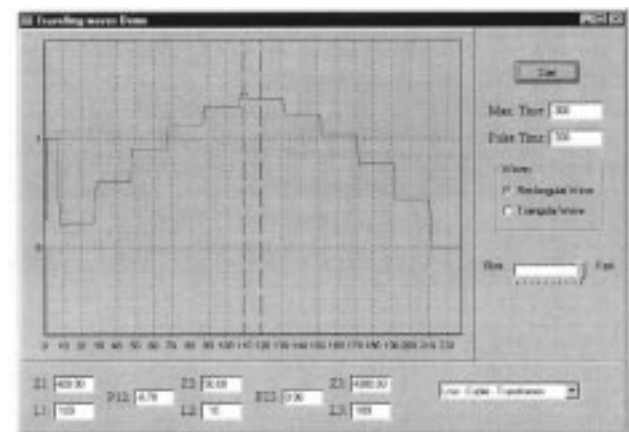


Fig. 9. Power systems demos: waves demonstration.

the line by clicking the “Line Parameters”-button. By using this demonstration the student “sees” why the three models are used for the different line lengths. The Lines demonstration is shown in Fig. 8.

The third program shows the effect of discontinuities on traveling waves. The rather complex wave shapes that result from the reflections and refractions, are computed and displayed; so the user can see how the wave shapes are being composed. The user can select a rectangular or triangular wave type and specify the pulse time of the wave. The configuration, on which the wave travels, is as follows: three line pieces in series that can be assigned different line lengths and different surge impedances. Selection of a configuration, e.g., “line with open end” or “line-cable-transformer” and so on, can be selected by using a drop-down box. When a certain configuration has been chosen, the corresponding surge impedances Z_i and lengths L_i of the line pieces are shown at the bottom of the window. These values can be modified by the user. The reflection coefficients are computed and displayed. The Waves demonstration is shown in Fig. 9.

E. The Impedance Model and Network Calculations

In order to visualize the systematic method for building up the Z_{bus} -matrix, the following demonstration has been developed. The Z_{bus} -matrix is created starting from the reference node by

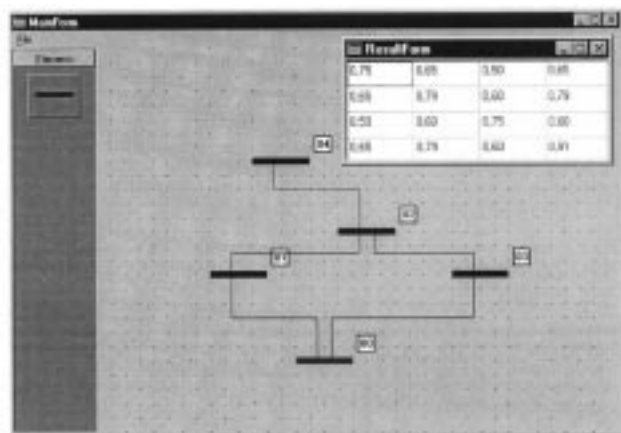


Fig. 10. Power systems demos: Z matrix demonstration.

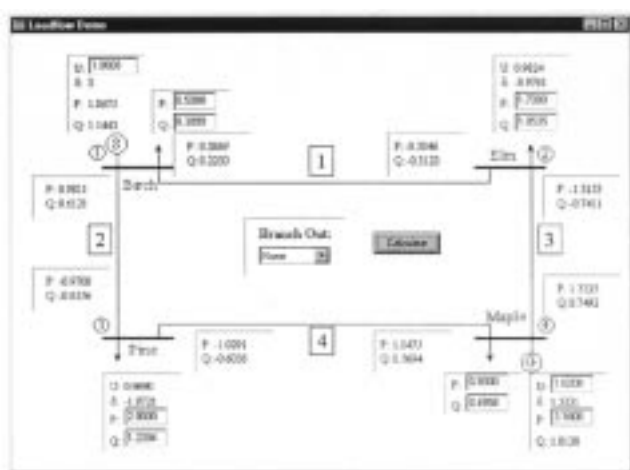


Fig. 11. Power systems demos: Loadflow demonstration.

adding new nodes and connections between them. Therefore, the Z_{bus} -matrix is build up step by step: after each connection (between nodes) added, the Z_{bus} -matrix is updated and shown. The student can create his own network by the graphical interface. The Z_{bus} -demonstration is shown in Fig. 10.

F. Power-Flow Solutions

The Loadflow demonstration has been made to support the examples used in the book by Grainger and Stevenson [1], where power-flow computations are made on a 4-node network. The student can perform power-flow computations with the 4-node network and is able to change P , Q -values of the loads and the P , V -values of the generator to get some feeling with it. Furthermore, the student can see what happens with the power flows in the network, when one of the branches is out of service. The power-flow computation is performed and displayed step by step; so the student can check every step in his own hand-made computations. The Loadflow demonstration of the 4-node network is shown in Fig. 11. The step by step computational results are shown in Fig. 12.

IV. HOW TO OBTAIN THE SOFTWARE

The software can be obtained from the homepage of the Electrical Power Systems group of the Delft University of Technology: eps.et.tudelft.nl

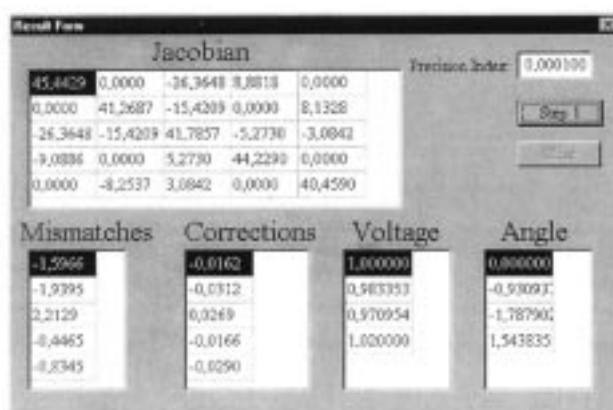


Fig. 12. Power systems demos: step by step loadflow results.

We make no warranties, explicit or implicit, that the program is free of error. The program can be copied freely. For remarks and suggestions concerning this program, please contact the Electrical Power Systems group of the Delft University of Technology e.g., by email:

P.H.Schavemaker@ITS.TUdelft.NL or

R.Reijntjes@ITS.TUdelft.NL.

Because of the positive and enthusiastic comments of the students, it is planned to create demonstrations for other courses too. Therefore, it is worth to check the homepage regularly for new demonstration programs.

V. CONCLUSION

A software program is developed for demonstration purposes during the lectures in the undergraduate course "Power System Analysis I" at the Delft University of Technology. The software has been built up of one main program (the executable) and the demonstration programs which are stored in dlls (dynamic link libraries). The software visualizes certain modeling and computational aspects of the power system analysis and gives the student insight into the effects of certain actions without making elaborate computations. The students can make copies of the software to study and practice with it. The program is available on the internet. Therefore, only the demonstration software needed for a particular course has to be downloaded.

REFERENCES

- [1] J. J. Grainger and W. D. Stevenson, Jr., *Power System Analysis*: McGraw-Hill, 1994.

Pieter H. Schavemaker was born in Velsen, the Netherlands on November 30, 1970. He received the M.Sc. degree in electrical engineering from the Delft University of Technology in 1994. After graduation, he performed research on power system state estimation with the Electrical Power Systems Group for one year. In 1995, he started as an application engineer programming substation control systems with ABB in The Netherlands. Since 1996, he has been with the Electrical Power Systems Group where he is currently Assistant Professor. He is working on a Ph.D. research on "digital testing of high-voltage circuit breakers" within the framework of a European project. His main research interests include power system transients and power system calculations.

Robert Reijntjes was born in Ulft, the Netherlands on January 12, 1969. He received the B.Sc. degree in electrical engineering from the Hogeschool Arnhem. In 1993, he joined the Electrical Power Systems Group where he is developing software tools for support of teaching activities. Furthermore, he is involved in the development of a digital dynamic power system simulator for students to practice with.

Lou van der Sluis was born in Geervliet, the Netherlands on July 10, 1950. He received the M.Sc. degree in electrical engineering from the Delft University of Technology in 1974. He joined the KEMA High Power Laboratory in 1977 as a test engineer and was involved in the development of a data acquisition system for the High Power Laboratory, computer calculations of test circuits and the analysis of test data by digital computer. In 1990, he became a part-time professor and since 1992, he has been employed as a full-time professor at the Delft University of Technology in the Electrical Power Systems Department. Prof. Van der Sluis is a Senior Member of IEEE and convener of CC-03 of Cigre and Cired to study the transient recovery voltages in medium and high voltage networks.