GRAPHICAL INTERFACE TO PLOT THE PHASOR DIAGRAM OF A RAILWAY SYNCHRONOUS GENERATOR

Lecturer Eng. **Gabriela-Dana PETROPOL-SERB,** PhD[1], Eng. **Ion PETROPOL-SERB**

[1]University of Craiova, Faculty of Electric Engineering,
[2]S.C. RELOC S.A., Craiova.

REZUMAT. Diagrama fazorială a generatorului sincron de tracțiune este larg folosită pentru a analiza comportarea acestuia în timpul funcționării, dar şi ca instrument de lucru în proiectarea sa. Deoarece atât funcționarea generatoarelor sincrone de tracțiune cât şi activitatea de proiectarea a acestora apelează la calculul numeric, este justificată realizarea unei interfete grafice de trasare a diagramei fazoriale ca subrutină de sine-stătătoare ce poate fi folosită în aplicații multiple. Limbajul grafic folosit este GUIDE Matlab. Prin urmare, putem spune că tematica lucrării este actuală.

Cuvinte cheie: generator sincron de tracțiune, diagramă fazorială, interfață grafică, GUIDE Matlab.

ABSTRACT. Phasor diagram of a synchronous generator used in railway traction is widely used to analyze its behavior during operations, but also as a working tools in its designing. Since both the operation of a railway traction synchronous generator and also its design work , impose to use the numerical calculation, is justified the achievement of a graphical interface to draw a phasor diagram as a stand-alone subroutine, which could be used in multiple applications. The graphical language used is the GUIDE Matlab. So, it could be concluded that, in this context, the paper is actual.

Keywords: railway traction synchronous generator, phasor diagram, graphical interface, GUIDE Matlab.

1. INTRODUCTION

The phasor diagram is an important tool for illustrating and studying the operation of an electrical machine by means of complex phasors (vectors). Phasor diagrams can be used only with AC analysis with linear circuits, or with nonlinear circuits in working point linearization. For the salient pole synchronous machine are expected different variants of the phasor diagram using the principle of two reactions, which was proposed by Blondel.

In this theory we considered the following assumptions:

- The armature has 3 identical, symmetrically placed, lumped windings a, b, c;
- The rotor windings E, D, Q are placed in the direction of the two orthogonal axes: d (direct) and q (quadrate);
- The winding E represents the field winding, the windings D and Q are fictitious windings to account for: damper windings and the effects of currents in the iron parts of the rotor;
- The characteristic magnetic linear;
- Absence of losses iron;
- Torques losses by friction and ventilation proportional. Figure 1 shows us the schema of the synchronous machine in the two axes theory and figure 2 shows the magnetic fields of the machine.

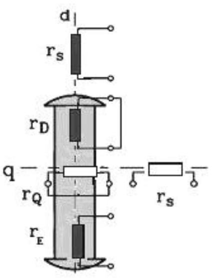


Fig.1. Synchronous machine:
d, q axis; s: stator winding; *D, Q*: damper windings; E: field windings

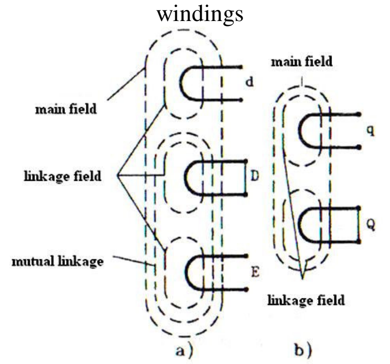


Fig. 2. Magnetic fields of the machine in:
a). d - axis and b). q - axis.

To establish the voltage equations must be considered a phase winding and its load (figure 3).

$$Ri + u = \sum u_e \quad (1)$$

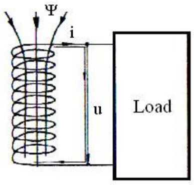


Fig. 3. Circuit of a phase of a synchronous generator

Variants of the phasor diagram for a salient pole synchronous machine are realized using the principle of two reactions. As in [2], [4], [5], the phasor diagram for the salient pole synchronous generator (figure 4) can be constructed from the following equation (2):

$$\underline{U} = \underline{U}_{eE} + \underline{U}_{ead} + \underline{U}_{eaq} - \underline{Z}\underline{I} = \\ = \underline{U}_{eE} - jX_d\underline{I}_d - jX_q\underline{I}_q - \underline{Z}\underline{I} \quad (2)$$

$$\underline{I} = \underline{I}_d + K_q\underline{I}_q + K_E\underline{I}_E \quad (3)$$

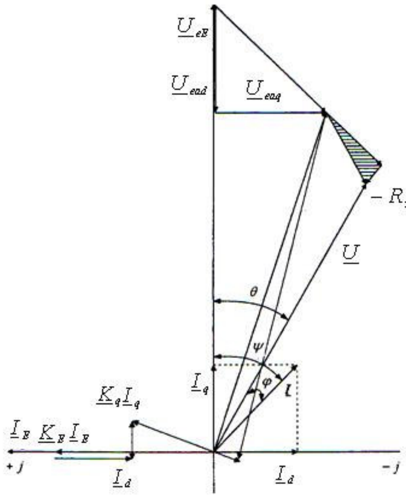


Fig.4. Phasor diagram for a synchronous generator with salient pole

The conventional simplified phasor diagram according to (4) is shown in figure 5.

$$\underline{U} = \underline{U}_{eE} + \underline{U}_{ead} + \underline{U}_{eaq} = \underline{U}_{eE} - jX_d\underline{I}_d - jX_q\underline{I}_q \quad (4)$$

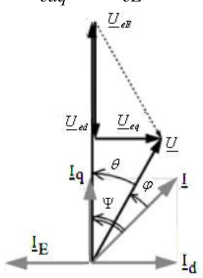


Fig. 5. Simplified phasor diagram for the salient pole synchronous generator

În calculus and in phasor diagrams are distinguished the angles: $\Psi = \varphi + \theta$, which is the phase angle between $\underline{I}, \underline{U}_{eE}$, and θ, which is the phase angle between $\underline{U}, \underline{U}_{eE}$, which is the internal angle of the machine and describe the size and load character, reason for that is very important for the behavior of the machine.

2. GRAPHICAL USER INTERFACE

General considerations:

A GUI is a type of computer human interface which solves the screen appearence in a problem with which are confronted computer users. This tool has appeared as a necessity in the process of give the information to an observer or a customer. As in figure 6 there are two ways we might increase the amount of input information.

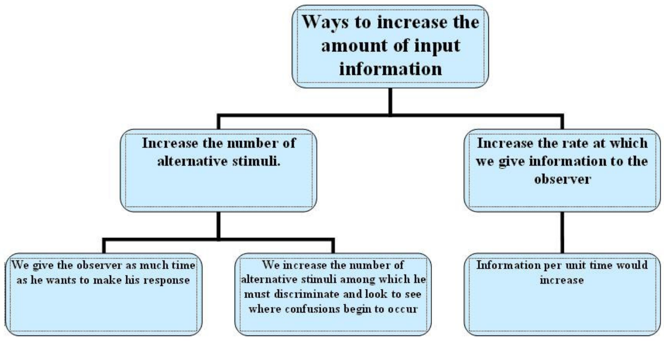


Fig. 6. Ways to increase the amount of input informations

In this paper we are interested in the first alternative. We give the observer as much time as he wants to make his response; we simply increase the number of alternative stimuli among which he must discriminate and look to see where confusions begin to occur. Confusions will appear near the point that we are calling his "channel capacity."For do this we choose GUIDE of MATLAB.

GUIDE, the MATLAB Graphical User Interface development environment, provides a set of tools for creating GUIs. These tools greatly simplify the process of laying out and programming a GUI [4].

GUIDE is a set of layout tools and also generates an M-file that contains code to handle the initialization and launching of the GUI. This M-file provides a framework for the implementation of the callbacks (the functions that execute when users activate components in the GUI).

Creating a GUI involves two basic tasks:

- Laying out the GUI components;
- Programming the GUI components.

A GUI is an arborescent structure like in figure 7.

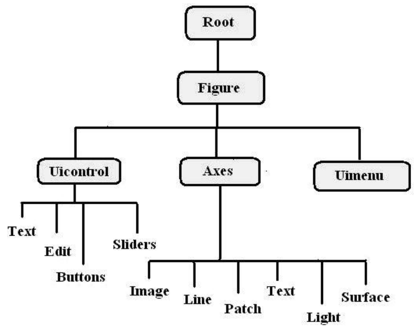


Fig.7. Basic structure of a GUI

3. MODEL RESULTS

The theoretical aspects shown hereinbefore are applied for a synchronous generator used in railway traction with the data given in table 1. The reason to realised this subroutine was to simplified the design process of a synchronous generator and the first use of this subroutine was to find rated solenation and the voltage drop.

TABLE 2.

The proposed design theme

Operation at cruising speed of the locomotive	Apparent power : 320 KVA
	Maximum voltage generated: 600 V
	Minimum current generated : 720 A
	Speed: 1800 rpm
Operation in hard conditions	Apparent power: 330 KVA
	Minimum voltage generated: 320 V
	Maximum current generated: 1360 A
	Speed: 1800 rpm
Maximal values	Maximum voltage generated: 620 V
	Maximum current generated: 2400 A
	Maximum speed : 2240 rpm
Frequency f_1 :	50 Hz
Class of protection / Class of insulation	IP21/H

To create this interface, first, we realised a subscript m-file named ‚diagrama’ to define the entering data and the vectors. We used two syntaxes to define the vectors:

vecarrow([0 0],[0 V],'r'),grid on,hold on

and

compass(V,'r'),hold on.

Second step was to realised a script m-file, named ‚diagrama_fazoriala’, in which we were defined all uicontrol function which create our interface. An uicontrol is a function which create a user interface control object. Most uicontrol object perform a predefined action. MATLAB supports numerous styles of uicontrols, each suited for a different purpose. In our case we used: editable text fields, list boxes, push buttons, static text labels.

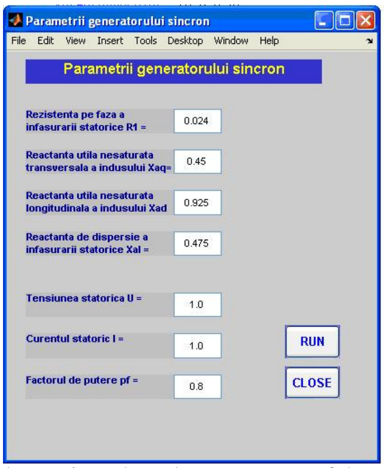
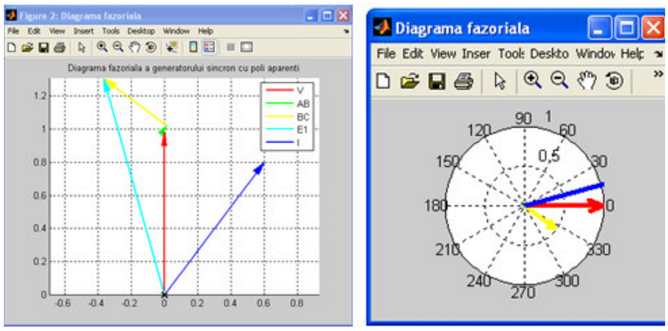


Fig.8. Window to introduce the parameters of the synchronous generator

When we run this script we obtain the following figures.

In figure 8 we can introduce the parameters of the calculated generator. When we want to obtain the diagram we push RUN button. If we want to close the application we push CLOSE button.

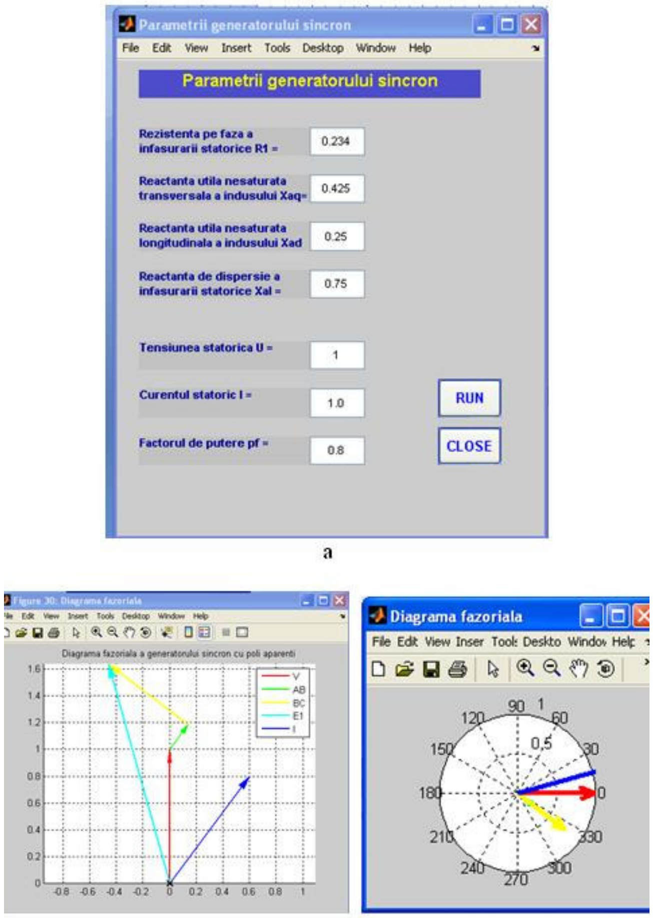
When we push RUN button will appear the two forms of the phasors diagram (figure 9): that with arrow vectors (figure 9a) and those with compass vectors (figure 9b).



a b

Fig. 9. Phasors diagram

We can change the parameters and we obtain the results as in figure 10.



b c

Fig.10. New phasors diagram: a – windows to define the parameters, b – arrows diagram, c-compass diagram.

5. CONCLUSIONS

✓ A GUI allows a computer user to move from application to application.

✓ A good GUI makes an application easy, practical, and efficient to use, and the marketplace success of today's software programs depends on good GUI design.

✓ GUI, usually, has three major components: a windowing system, an imaging model, and an application program interface. The windowing system builds the windows, menus, and dialog boxes that appear on the screen. The imaging model defines the fonts and graphics that appear on the screen. The application program interface is the means in which the user specifies how and what windows and graphics appear on the screen.

BIBLIOGRAPHY

[1] **Campeanu, A.**, *Introducer in the dynamics of the alternative current electric machines* (translate from Romanian), Editura Academiei Române, Bucureşti 1998.

[2] **Popa, G.** - *Railway traction with three phases asynchronous motors* (translate from Romanian), Ed. Matrix Rom, Bucureşti, 2005.

[3] **Mihailescu, *D.*** - *Magnetic sustentation and propulsion in railway transportation* (translate from Romanian),Ed. Academiei Române, 2008, ISBN 978-973-27-1633-5.

[4] **Crow, D., and Jansen, B. J.** - *The Graphical User Interface: An Introduction*, 1998. Seminal works in computer human interaction. SIGCHI Bulletin. 30(3), 24-28.

[5] **Winograd,T**. - *From Programming Environments to Environments for Designing*. Communications of the ACM. Vol. 38. No.6:65-74. 26 27, June 1995.

[6] **Miller, G. A**. - *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information.* - originally published in The Psychological Review, 1956, vol. 63, pp. 81-97.

About the authors

Lecturer. Eng. **Gabriela Dana PETROPOL ŞERB,** PhD
University of Craiova
email:gpetropol@em.ucv.ro

Gabriela Dana Petropol Şerb was born in Craiova, Romania, in 1967. She graduated the Faculty of Electromechanics and obtained the degree in electromechanical engineering from the University of Craiova, Craiova, Romania, in 1991. She received the Master and Ph.D. degrees in electrical engineering from the University of Craiova, Craiova, Romania, in 2001 and 2008, respectively. In 2002, she joined the Department of Electrical Machines, University of Craiova, as an Assistant Professor, and in 2007 became a Lecturer. Her current research interests include electrical machines and drives, numerical method and waters engineering, with over 30 technical publications. Dr. Petropol is a fellow of the General Association of the Engineers in Romania, the Romanian Society of Automatics and Technical Informatics; the Robotics Society of Romania.

Dipl. Eng. **Ion PETROPOL ŞERB**.
S.C. RELOC S.A.
email:ipetropol@yahoo.com

Ion Petropol Şerb was born in Galați, Romania, in 1967. He graduated the Faculty of Electromechanics, specialized in electrical traction, and obtained the degree in electromechanical engineering from the University of Craiova, Craiova, Romania, in 1992. He received the Master degree in electrical engineering from the University of Craiova, Craiova, Romania, in 2001 and now is in the third stage of doctoral research. He has extensive industrial experience in engineering design, manufacturing and management in areas of railway traction (repairing and modernizing of the locomotives at S.C. RELOC S.A. Craiova) and also many years experience in management and in projects implementation. His current research interests include electrical machines and drives, numerical methods and environmental engineering applied in industry, with over 20 technical publications. Mr. Petropol is a fellow of the General Association of the Engineers in Romania, the Robotics Society of Romania and the Romanian Society of Automatics and Technical Informatics.