DEVELOPING DYNAMIC SIMULATIONS AND ANIMATIONS BY USING LINEAR TRANSFORMATION MATRIXES AND POSITION VECTORS FOR ELECTRICAL EDUCATION

Ahmet ALTINTAŞ

Dumlupınar University, Technical Education Faculty, Department of Electrical Education, Kütahya.
a_altintas@dumlupinar.edu.tr

ABSTRACT

Teaching and learning techniques using computer-based resources greatly improve the effectiveness and efficiency of learning process. Today, there are a lot of simulation and animation packages in use; some of them are developed for professional purposes, and some are developed for educational purposes. The education-purposed packages can not be flexible sufficiently in different science branch. Therefore, some educators prefer developing his/her own simulation and animation packages. This paper reports on the developing of dynamic simulation/animation to enhance learning and interest by using computer graphics. Thus, a new point of view is introduced to educators in order to realize personal dynamic simulations/animations. In the method computer graphics, playing an important role in simulations and animations, created with the linear transformation matrixes and position vectors. The paper, also, shows how linear transformation can be successfully applied to computer graphics. Moreover, the paper briefly introduces several examples, which will be given as animation problems coming from electrical education.

Key words - Electrical Education, Simulation, Animation, Linear transformations

LİNEER DÖNÜŞÜM MATRİSLERİ VE POZİSYON VEKTÖRLERİNİ KULLANARAK ELEKTRİK EĞİTİMİ İÇİN DİNAMİK SİMÜLASYON VE ANİMASYON GELİŞTİRME

ÖZET

Bilgisayar tabanlı kaynakları kullanan öğretme ve öğrenme teknikleri, öğrenme işleminin verimliliğini ve etkinliğini büyük oranda arttırmaktadır. Günümüzde, bir çok simülasyon ve animasyon paket programı mevcuttur; bunlardan bazıları profesyonel amaçlı ve bir kısmı da eğitim amaçlı geliştirilmiştir. Eğitim amaçlı geliştirilen paket programlar, farklı bilim dalları için yeterince esnek değildir. Bu yüzden bazı eğitimciler kendilerine özgün simülasyon ve animasyon paket programlarını geliştirmeyi tercih etmektedir. Bu çalışma, ilgi ve öğrenmeyi arttırmak amacıyla, bilgisayar grafiklerini kullanarak farklı dinamik animasyonlar gerçekleştirmek için yeni bir yöntem sunmaktadır. Bu sayede eğitimcilere, özgün dinamik animasyonlar yapmak için yeni bir bakış açısı sunulmuştur. Yöntemde, dinamik animasyonlarda önemli bir rol oynayan bilgisayar grafikleri, lineer dönüşüm matrisleri ve pozisyon vektörleri ile oluşturulmuştur. Çalışma, lineer dönüşümlerin bilgisayar grafiklerine nasıl başarı ile uygulandığını da göstermektedir. Ek olarak çalışmada, elektrik eğitimi ile ilgili bir kaç animasyon örneği verilmiştir.

Anahtar kelimeler - Elektrik Eğitimi, Simülasyon, Animasyon, Lineer dönüşümler

1. INTRODUCTION

In the sense of teaching pedagogy, the traditional treatment of all fields of engineering tends to be highly theoretical and mathematical with heavy emphasis on equation derivation and algorithmic development. Such an approach is convenient from the instructor's point of view but may not be beneficial to the students. The term animation covers a broad range of software applications such as kinematics and dynamics. For kinematic animations, use of keyframes and motion capture constitutes the primary mean of driving the animation sequence (e.g. Poser, 3D Studio Max, Jack, etc.). Dynamic animation is driven by the outputs of a simulation program to provide a 2D or 3D display of the physical characteristics of the application. The output may contain simplified geometric objects such as line, rectangles circles, etc. A number of researchers have reported their effort in using animation to enhance the learning process [1-3]. In [1], an engineering animation tool was introduced. In [2], a real time simulation/animation tool was developed to facilitate the evaluation of active suspension system. In [3], animation of flexible manufacturing system was carried out. Also, a Visual C++, Direct-3D, Matlab, VisSim, and Lab VIEW based softwares were generated for interactive modeling, simulation and animation [4-8].

Simulations and animations often enrich modern education in all areas. Today, there are a lot of simulation and animation packages in use, and each of them has some strong and some weak features. Important demands for education are visualization of the simulation results and the interactivity of the simulation. Animations based on interactive simulations are an effective way to go deeper inside a problem. The user/student has to have the ability to influence parameters and/or conditions during the simulation/animation and thereby see the effect of these variations immediately in his simulation/animation.

The dynamic simulations-animations allow us to see physical movement of the different pieces. Simulations and animations are based on the computer graphics. Many of the most important programs for computer graphics have been written in traditional programming languages (Fortran, Pascal, C, etc.) [9]. However, in the last recent years, the general-purpose numerical computation programs (NCP) are gaining more and more popularity. Today, they are well established as a powerful alternative to traditional programming languages in many different areas, as electrical engineering, mechanical engineering, signal processing, power systems, etc. Nevertheless, not all the NCPs offer the same advantages and features. After a careful analysis, in this paper Matlab package program has been decided to use in order to realize educational-purposed animations and simulations.

Matlab is a matrix-based software for scientific and engineering numeric computation and visualization. Matlab is chosen as the programming tool primarily because of interactive mode of work, immediate graphics facilities, built-in functions, the possibility of adding user-written functions, simple programming and its wide availability on computing platforms [10]. These factors make Matlab an excellent language for teaching and a powerful tool for research and practical problem solving. Matlab also allows creating movies either saving a number of different pictures and then playing them back or by continually erasing and redrawing the objects on the screen. The first method is more advisable in situations in which each frame is fairly complex and cannot be redraw rapidly.

The simulation-animation packages provide a convenient tool so that many scenarios can be tried with ease. This is very helpful for better understanding. However, some educators are recommending and using his/her own products for simulation and animation, because the package programs can not be flexible sufficiently in different science branch, and may not produce satisfactory solutions under given conditions. In this paper, it is aimed to give a different point of view of dynamic simulation/animation; thus, an educator would realize his/her own package program easily, and improve the effectiveness of utilizing animations and simulations in every course. In this method, in order to realize the dynamic simulation-animation, linear transformation matrixes and computer graphics are used. To realize a dynamic simulation-animation, the dynamic simulation of model of the system should be firstly performed by means of a proper method (differential equations, transfer functions, state-space model and etc.). After getting the necessary data from dynamic simulation, the stage of dynamic animation can be carried out. Every part of the system is formed graphically with the position vectors, and the moving parts are treated successively with the proper linear transformation (translating, rotating, scaling and etc.) by using the simulation's results. Then, the successive figure pages are combined with the 'movie' function of Matlab, and then the dynamic animation is realized successfully.

2. THE METHOD: LINEAR TRANSFORMATIONS AND COMPUTER GRAPHICS

Computer graphics is a complex and diversified technology. Presently this technology is used for a large variety of purposes; e.g., computer aided design and manufacturing, architectural rendering, advertising illustrations, animated movies, and etc. Computer graphics are based on the fundamental linear transformations. Computer graphics is usually formed by rotating, translating, scaling and performing various projections on the data. These basic orientations or viewing preparations are generally performed using 3×3 or 4×4 transformation matrix operating on the data represented in homogeneous coordinates. When a sequence of transformations is required, each individual transformation matrix can be sequentially applied to the points to achieve the desired result [11,12].

A point is represented in two dimensions by its coordinates. These two values are specified as the elements of a 1-row, 2-column matrix: $[x\ y]$. Alternately, a point is represented by a 2-row, 1-column matrix:

$[x\ y]^T$. These two representation matrixes are frequently called position vectors. In this study, a row matrix formulation of the position vector is used. A series of points, each of which is a position vector relative to some coordinate systems, is stored in a computer as a matrix. By drawing lines between the position vectors, lines, curves or figures are generated. After defining required transformation matrix and multiplying it by the position vectors, the transformed shape can be attainable.

A number of transformations such as rotation, reflection, scaling, shearing and etc., can be realized by the general 2×2 transformation matrix. The general transformation matrix is given in Eq.1. The effect of the terms a, b, c and d in the 2×2 matrix can be identified separately. The terms b and c cause a shearing in the y and x directions respectively. The terms a and d act as scale factors. Thus, the general 2×2 matrix produces a combination of shearing and scaling. In the same manner, different combinations of the terms a, b, c and d are, also, perform rotation and reflection.

$$[T]=\begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{1}$$

The origin of the coordinate system is invariant with respect to all of these transformations. However, it is necessary to be able to modify the position of the origin. This difficulty can be overcome by using the homogeneous coordinate systems. The homogeneous coordinates of a non-homogeneous position vector $[x\ y]$ are $[x\ y\ 1]$. Thus, the general transformation matrix will be 3×3 (Eq.2).

$$[T]=\begin{bmatrix} a & b & 0 \\ c & d & 0 \\ m & n & 1 \end{bmatrix} \tag{2}$$

where the elements a, b, c and d of the upper left 2×2 sub-matrix have exactly the same effects with the general 2×2 transformation matrix. m and n are the translation elements.

2.1. Translation

The pure two-dimensional translation matrix is given in Eq.3. As noted previously; m, n are the translation factors in the x and y directions, respectively;

$$\begin{bmatrix} x^* & y^* & 1 \end{bmatrix}=\begin{bmatrix} x & y & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ m & n & 1 \end{bmatrix} \tag{3}$$

where x^* and y^* are coordinates of the translated position vector [11,12].

2.2. Scaling

Scaling is controlled by the magnitude of the two terms on the primary diagonal of the matrix: a, d (Eq.4). If the magnitudes are equal, uniform scaling occurs about the origin; if the magnitudes are not equal, a distortion occurs;

$$\begin{bmatrix} x^* & y^* & 1 \end{bmatrix}=\begin{bmatrix} x & y & 1 \end{bmatrix}\begin{bmatrix} a & 0 & 0 \\ 0 & d & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

where x^* and y^* are coordinates of the scaled position vector.

2.3. Rotation

In general, a rotation about an arbitrary point can be accomplished by first translating the point to the origin, performing the required rotation, and then translating the result back to the original center of rotation. Thus, rotation of the position vector $[x\ y\ 1]$ about the point (m,n) through an arbitrary angle can be accomplished by Eq.5;

$$\begin{bmatrix} x^* & y^* & 1 \end{bmatrix}=\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}^T\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ \begin{matrix} -m(\cos\theta-1) \\ +n\sin\theta \end{matrix} & \begin{matrix} -n(\cos\theta-1) \\ -m\sin\theta \end{matrix} & 1 \end{bmatrix} \tag{5}$$

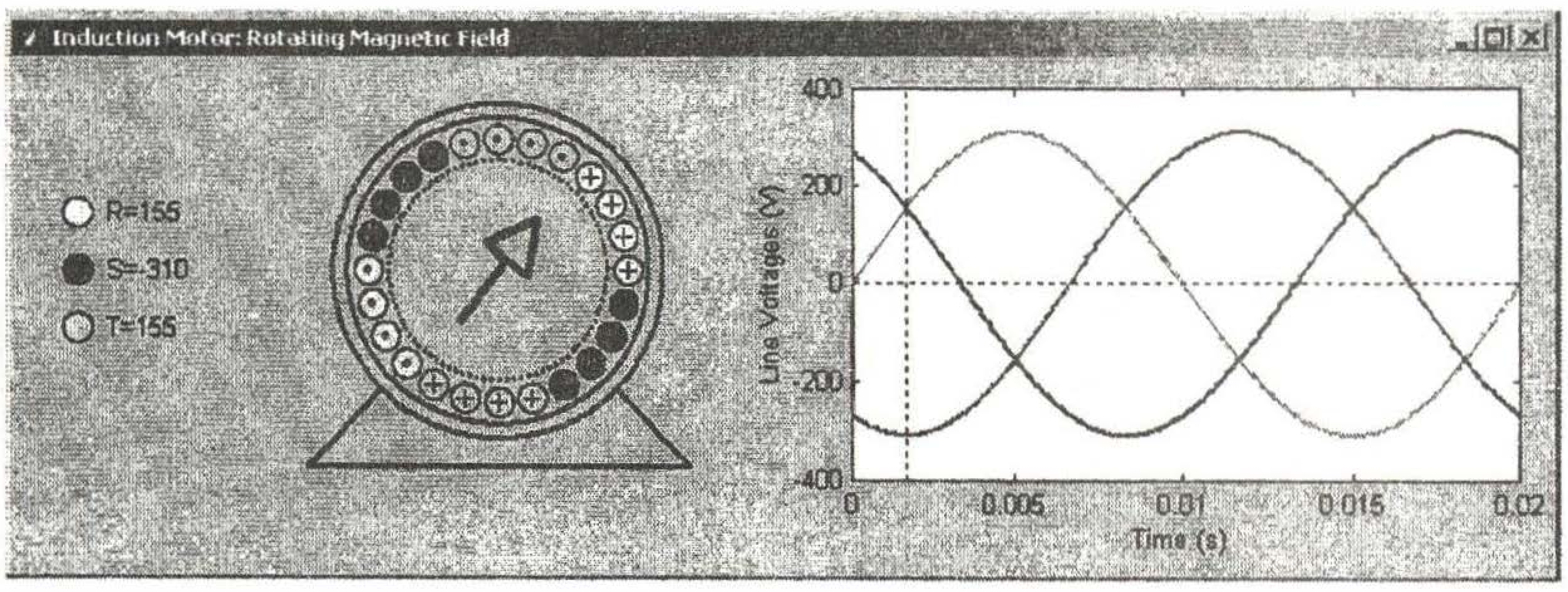
where x^* and y^* are coordinates of the rotated position vector.

3. EXEMPLARY APPLICATIONS

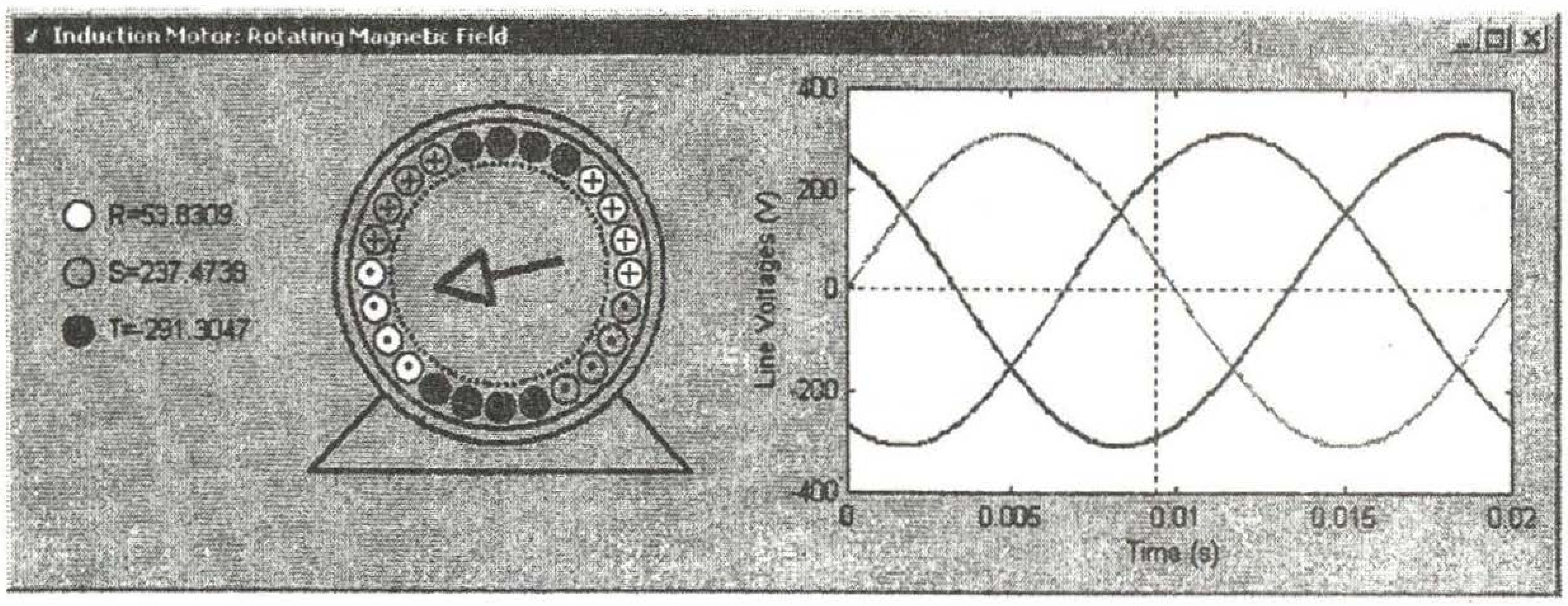
3.1. Rotating Magnetic Field in Induction Motors

Induction motor (IM) is used for many applications because of its low purchase, rugged construction and operating characteristics. The basic structure of IM consists of a stator, a rotor, and two end covers. The stator is a three-phase winding placed in the slots of a laminated steel core. The winding itself is made of formed coils which are connected to give three single-phase windings spaced 120° electrical degrees apart. When three-phase current, also 120° electrical degrees apart, are then passed through the windings, a rotating magnetic field is set up and travels around the inside of the stator core. The speed of this field depends upon the number of the stator poles and the frequency of the power source. This rotating magnetic field causes the rotor to rotate [13,14].

In the program, after getting the system parameters (slot and pole numbers), required calculations are done. Then, the windings distributed in the slots across the periphery of the stator are denoted as colored circles; 3-phase (R, S, T) windings are colored with green, red and blue, respectively. In addition, the colored windings indicate the current flow and direction, where the tone of color is related to the instantaneous value of the winding current. It is assumed that the concentrated and full-chorded coils are used in IM (the short-chorded coils are not taken into consideration). During the animation, the instantaneous values of the phase voltages are displayed in time numerically and graphically; thus, the user can correlate between sinusoidal voltages/currents and rotating magnetic field (phase difference between line voltage and line current is not taken into consideration). Rotating magnetic field of the IM is animated by accumulating successive positions of the rotating magnetic field for one-cycle. The resultant outputs of the program displaying the position of the rotating magnetic field are given in Fig.1. In Fig.1.b, at 165 electrical degrees, the voltages in phase windings R and S are the same in direction; the voltage in winding T equals to sum of the R and S, but opposite direction. Also, simplified flowchart of the developed Matlab program, performing dynamic animation of IM, is given in Fig.2. Because of saving space, the flowcharts related to the rest of exemplary applications are not given any longer.



(a)



(b)

Figure 1. Instantaneous positions of the rotating magnetic field at, a) 30 electrical degree, b) 165 electrical degree.

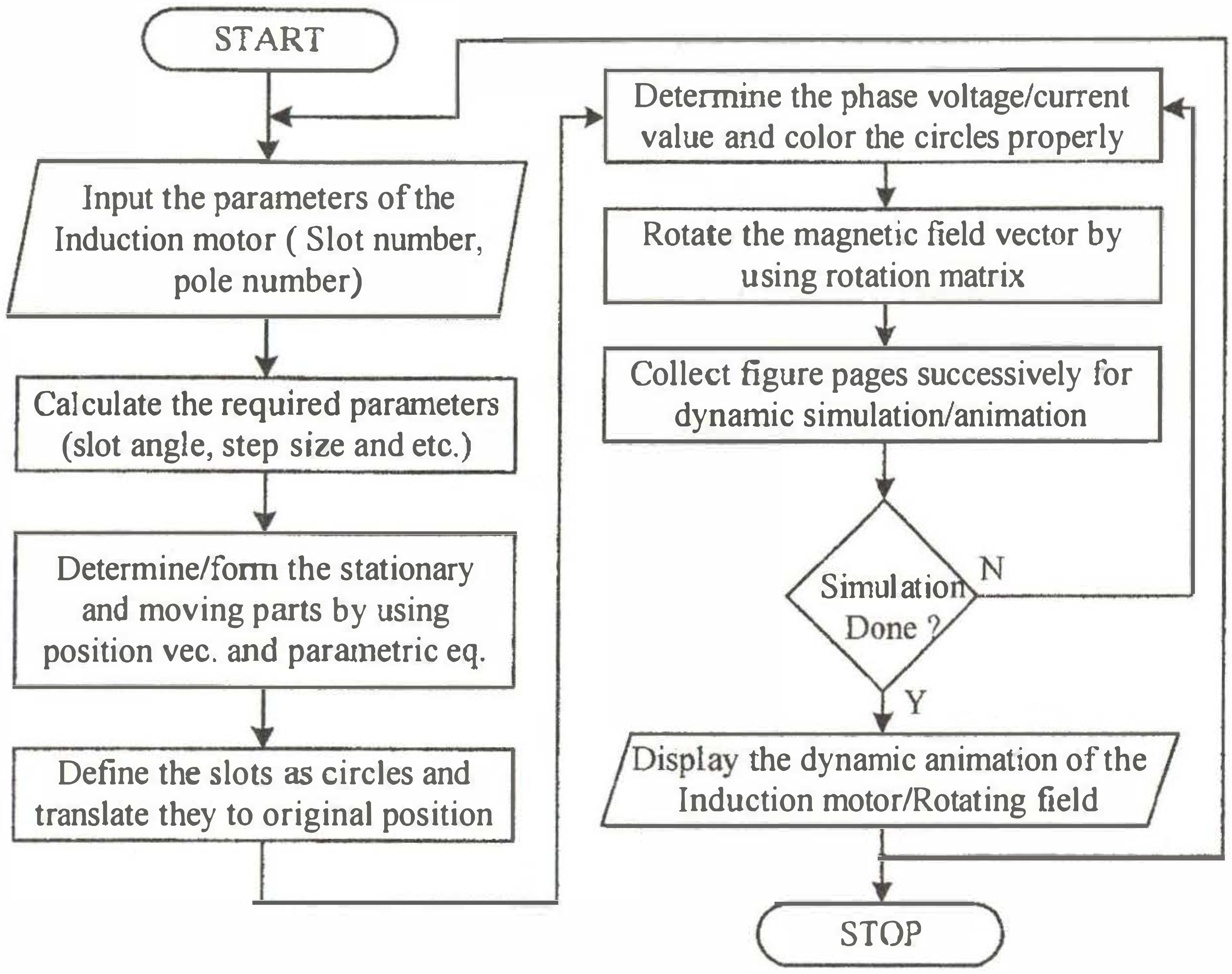


Figure 2. Simplified flowchart of the developed Matlab program performing dynamic animation of IM.

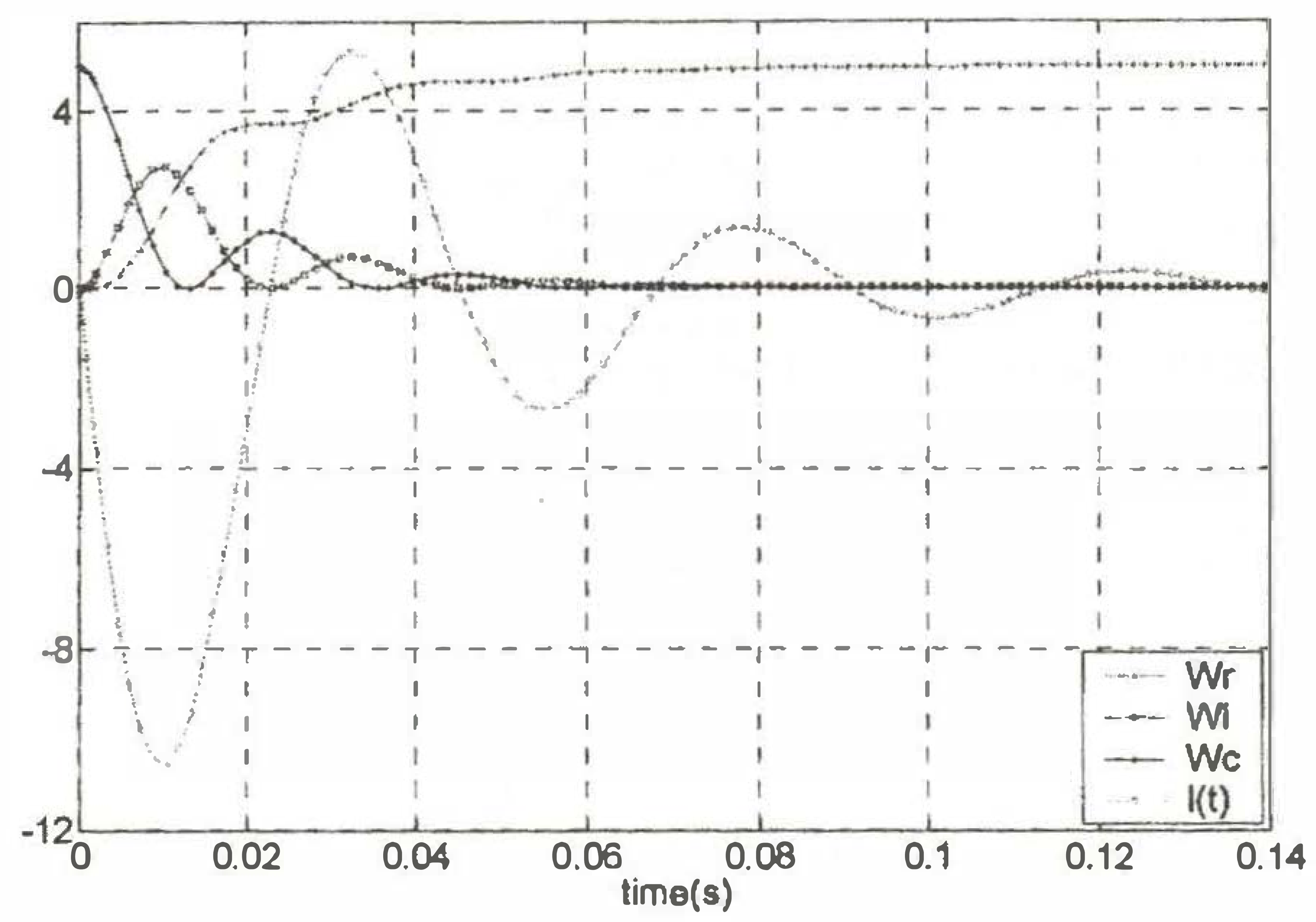
3.2. Electric Circuits

In electric circuits, there are generally three types of elements as loads, resistance (*R*), capacitance (*C*) and inductance (*L*). Resistance is just one of the characteristics of the electric circuits, but only the resistance dissipates power [15]. Current causes an electric field to be set up within a capacitor. The electric field is a form of energy that normally does not leave the circuit. Capacitive energy is stored until conditions cause the electric field to collapse. Just as with a capacitor, energy in the form of a magnetic field can be stored in an inductance. Storing energy in a capacitor and an inductor takes place in antiphase with respect to each other. In this exemplary application, dynamic simulation of a circuitry combined from RLC elements connected in series was realized. The impressed voltage and RLC values, which will be entered by the program user, are selected as simulation parameters. In the simulation, fluctuations of the energy and current are demonstrated in graphically and numerically (Fig.3). For educators and students this approach is certainly an effective way in order to clarify the effects of RLC elements in circuitry; because the variables of electricity (current, voltage, power, energy, etc.) are not visible with naked eye.

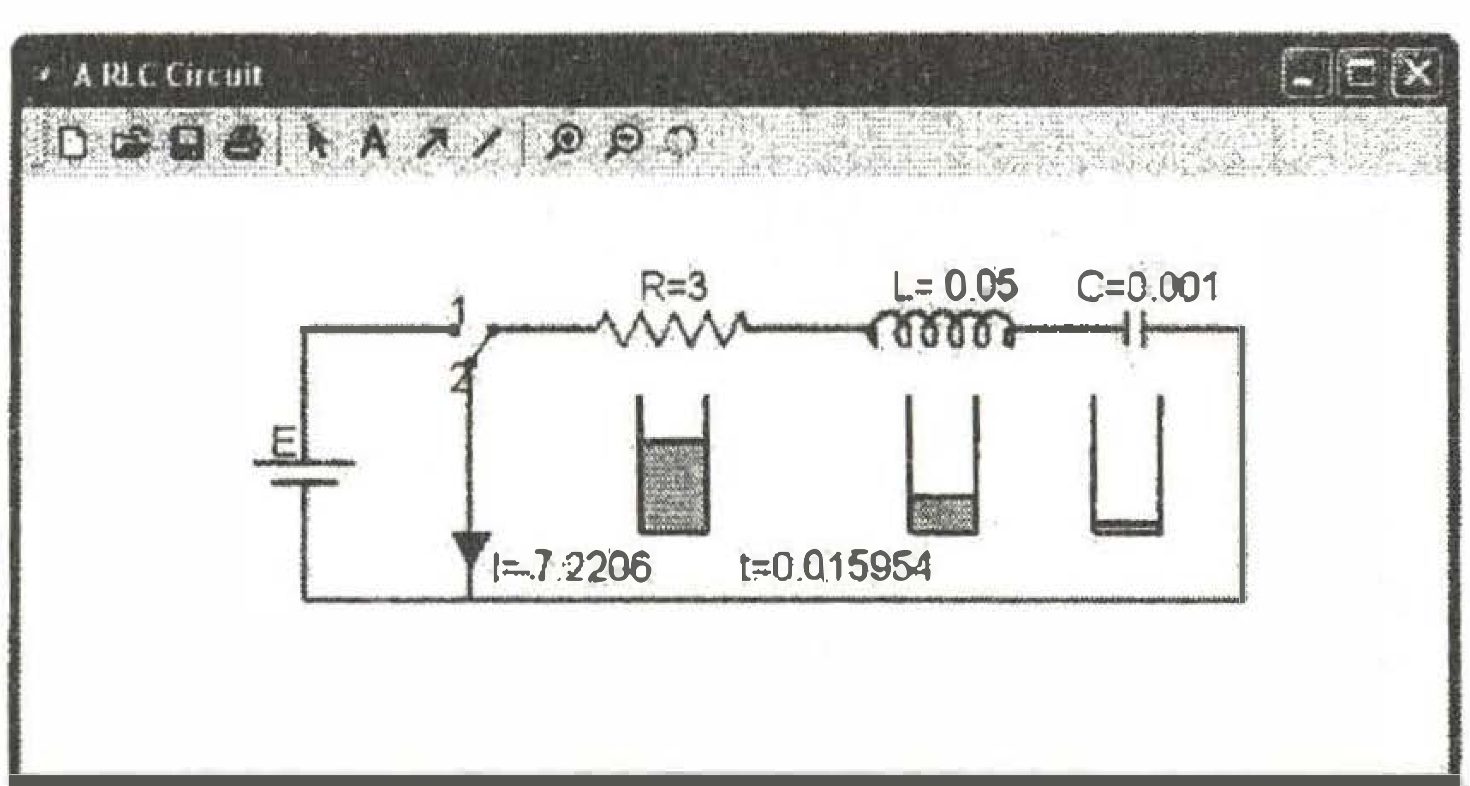
Such a simple series RLC circuit can be represented with linear second order differential equations with constant coefficient (Eq.6); where Q denotes electric charge [16]. ODE45 solver, a built-in function of Matlab, is used to solve the second order linear ordinary differential equation. In the dynamic simulation it was supposed that the switch in the Fig.3b was first positioned at position 1; it was held there as far as the capacitor was fully charged; and at the instant *t*=0 the switch is brought from the position 1 to the position 2. Thus, initial conditions of the differential equations will be both the initial charge of the capacitor $Q(0) = C.E$ and the initial current in the circuit $Q'(0) = 0$.

$$L\frac{d^2Q}{dt^2} + R\frac{dQ}{dt} + \frac{1}{C}Q = 0 \tag{6}$$

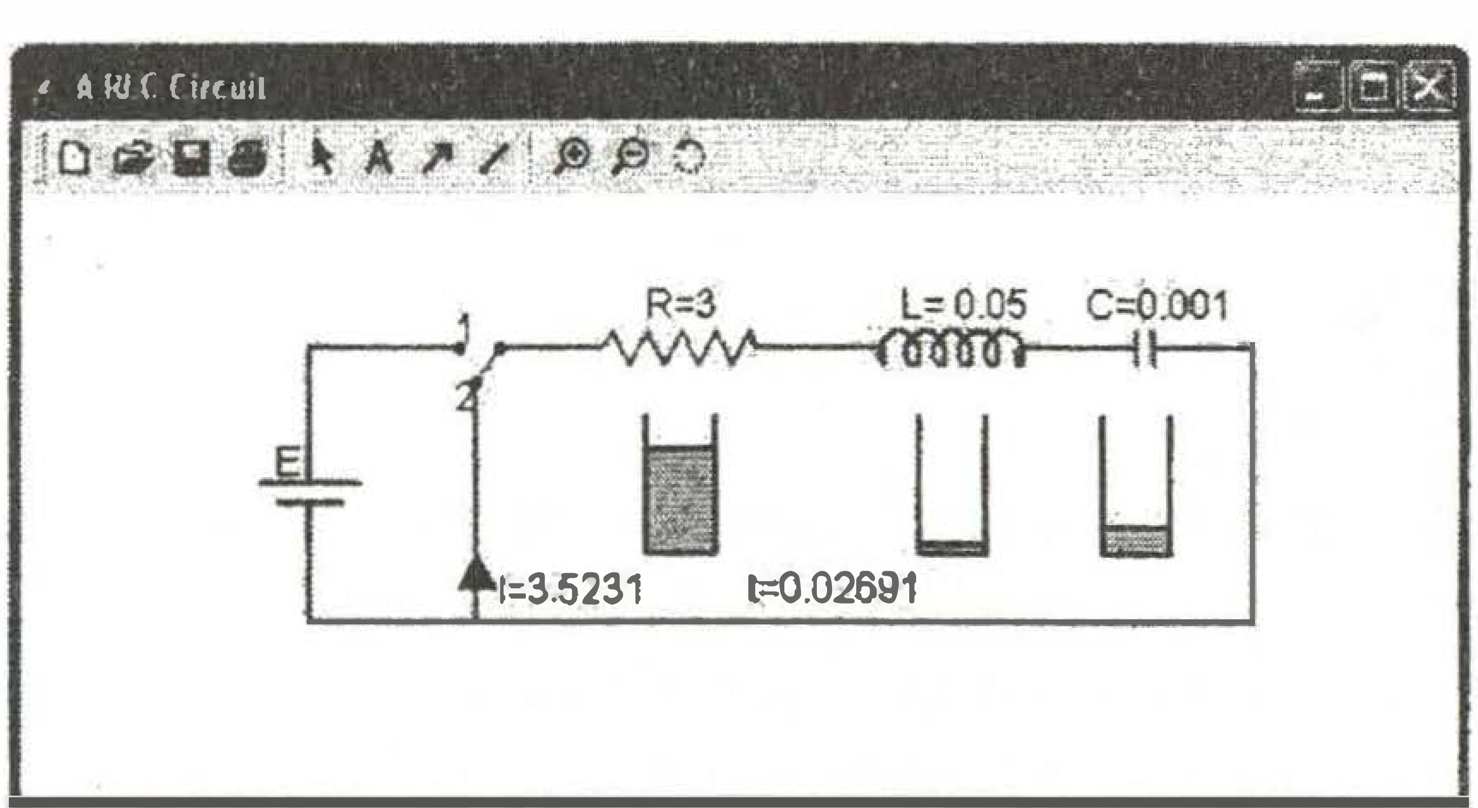
An exemplary dynamic animation of the circuit with the user parameters $E = 100V$, $L = 0.05H$, $R = 3\Omega$, $C = 0.001F$ is given in Fig.3. Graphical representations of the selected parameters are given in Fig.3.a. The selected parameters are the fluctuating of energy in LC elements (Wl, Wc), the energy dissipated in R (Wr) and the current (I(t)). Fig.3.b,c show instantaneous schematics of the dynamic animation of the circuitry at the instant *t*=0.015954*s* and *t*=0.02691*s*.; where two-dimensional reservoirs indicate amount of energy related to the *RLC* elements; direction of the arrow shows direction of the circuit current.



(a)



(b)



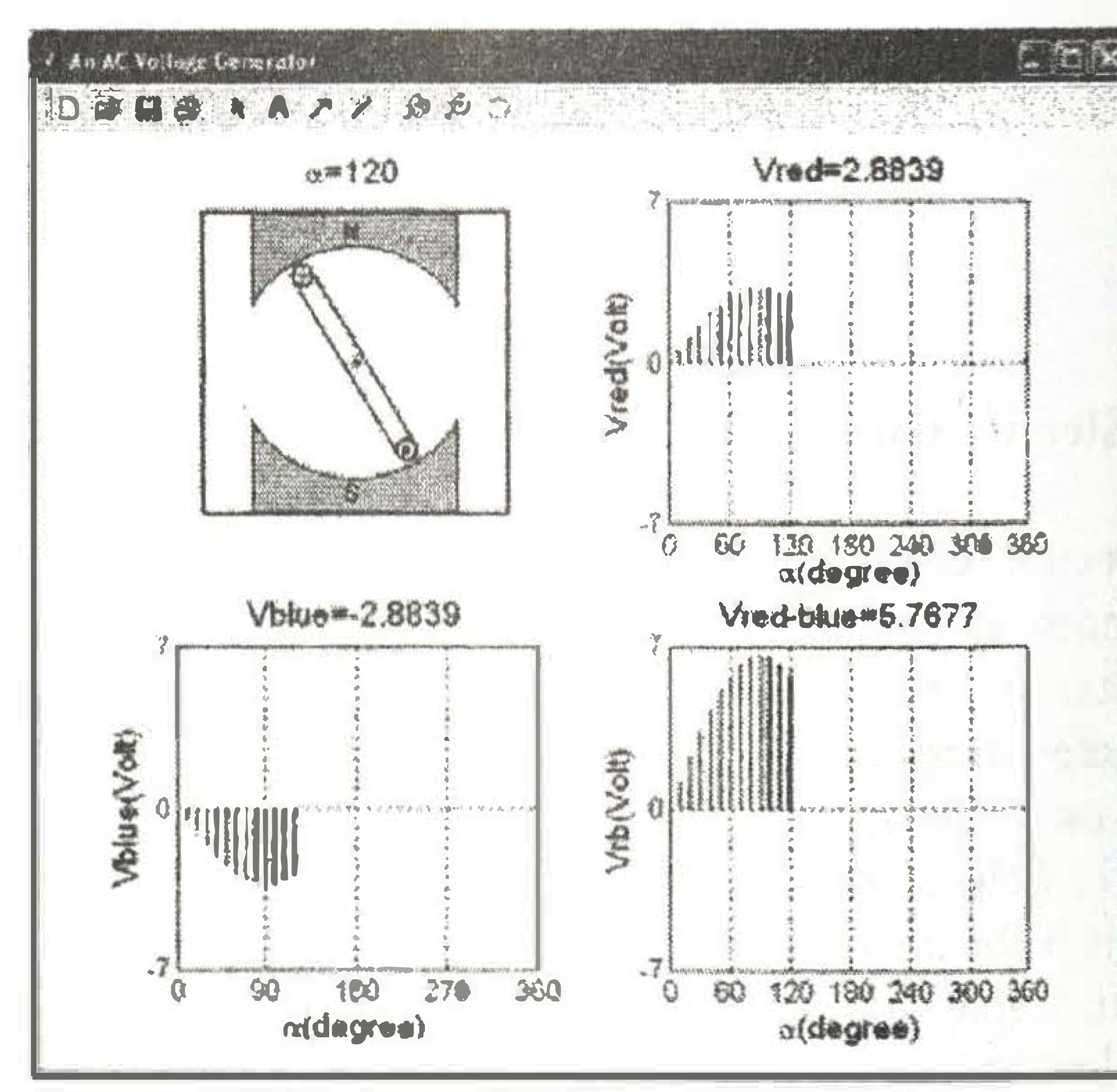
(c)

Figure 3.a) Graphical representations of the selected circuit parameters, b,c) Instantaneous schematics of the dynamic animation of the circuitry at the instant t=0.015954s and t=0.02691s.

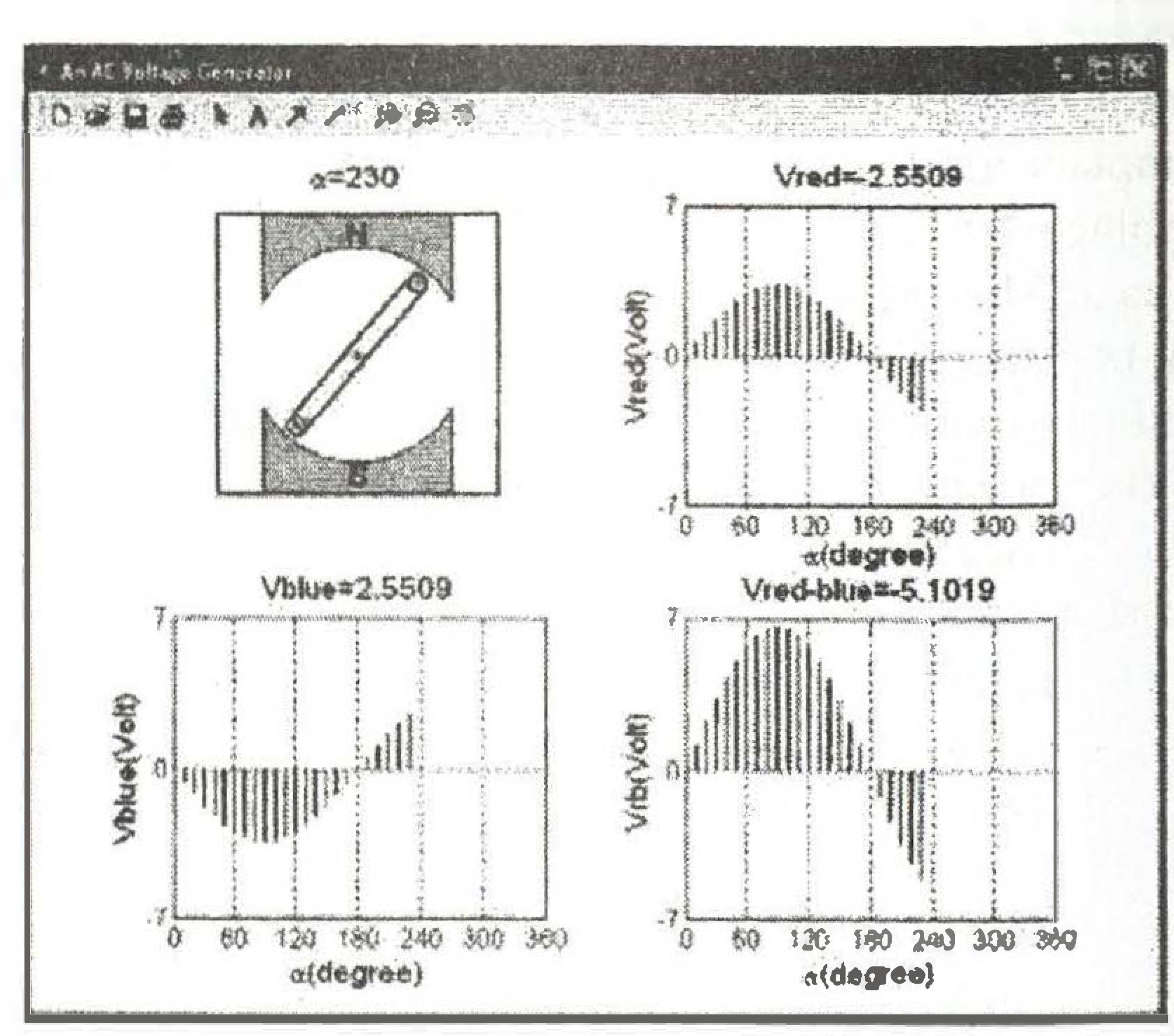
3.3. An Alternating-Voltage Generator

The third example is an alternating-voltage generator. The dynamic animation of a simple alternating-voltage generator consisting of a single coil rotating a uniform magnetic field is shown in Fig.4. In such a system if the ends of the coil are connected to two sliprings, the alternating voltage can be observed on an oscilloscope. This voltage pattern is a typical sine wave. The generated voltage in an armature conductor is expressed by the formula $e = Blv10^{-8}$; where e is the generated voltage in the armature conductor in volts, B is the magnetic flux density of the field, l is the length of the armature conductor, v is the velocity of the rotation of the coil. The instantaneous voltage in each position is determined by $e_{\max}\sin(\alpha)$.

In this exemplary application B,l and v are selected as parameters of the dynamic simulation. It was assumed that the conductors of the coil have been moved counterclockwise with an interval of 10° (rotating-angle interval can be simply changeable by the user). During the animation, the directions ⦿⊕ and the instantaneous values of the induced voltage in each side of the coil and the overall voltage are displayed numerically and graphically; thus, the user can simply correlate between the induced sinusoidal voltage and the simulation parameters. In order to determine sides of the coil, they are colored with red and blue. Instantaneous positions of the dynamic animation of the ac voltage generator at the position of α =120° and α =230° given in Fig.4.



(a)



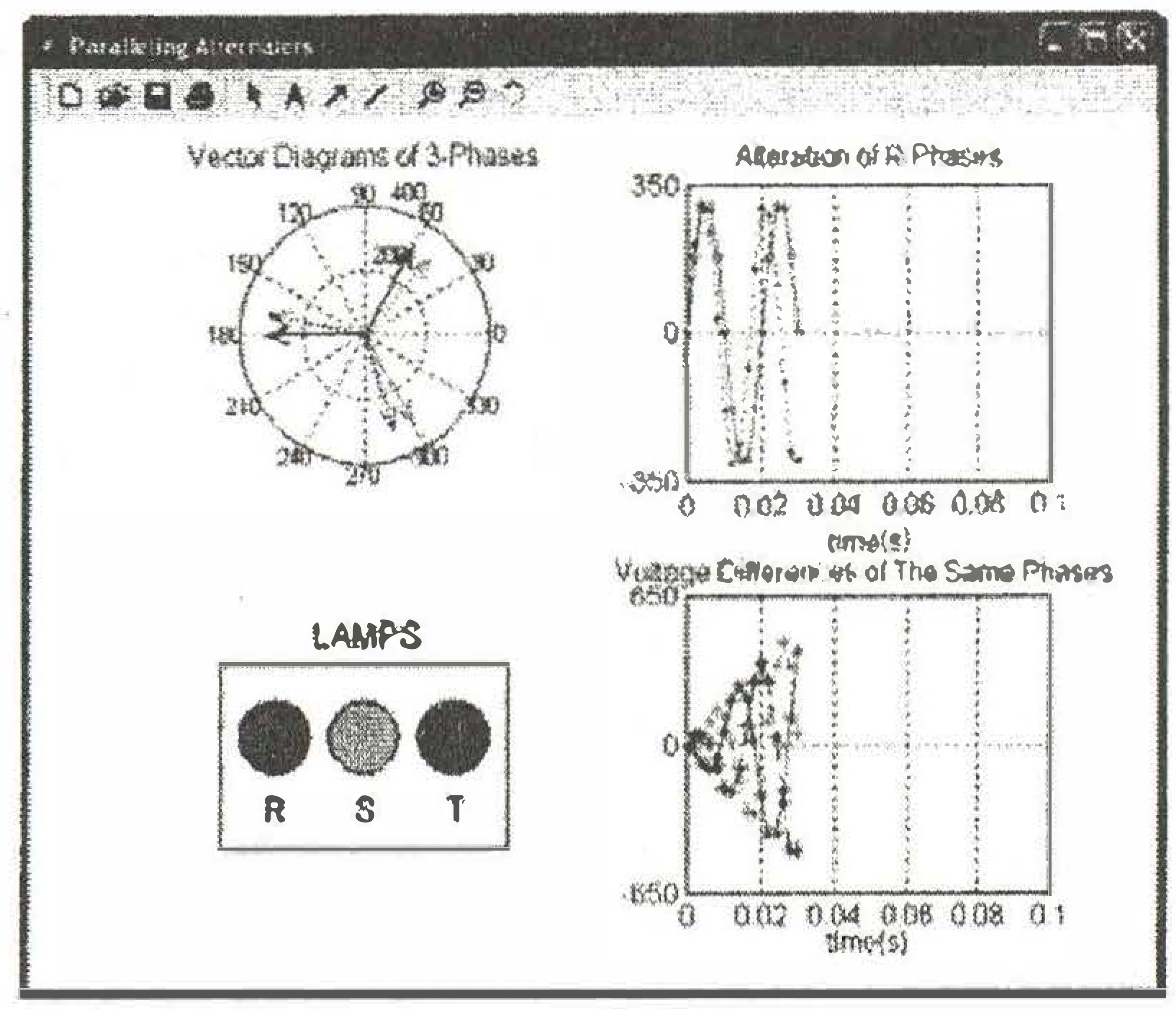
(b)

Figure 4. Instantaneous positions of the dynamic animation of the ac voltage generator at the position of, a) α =120°, b) α =230°.

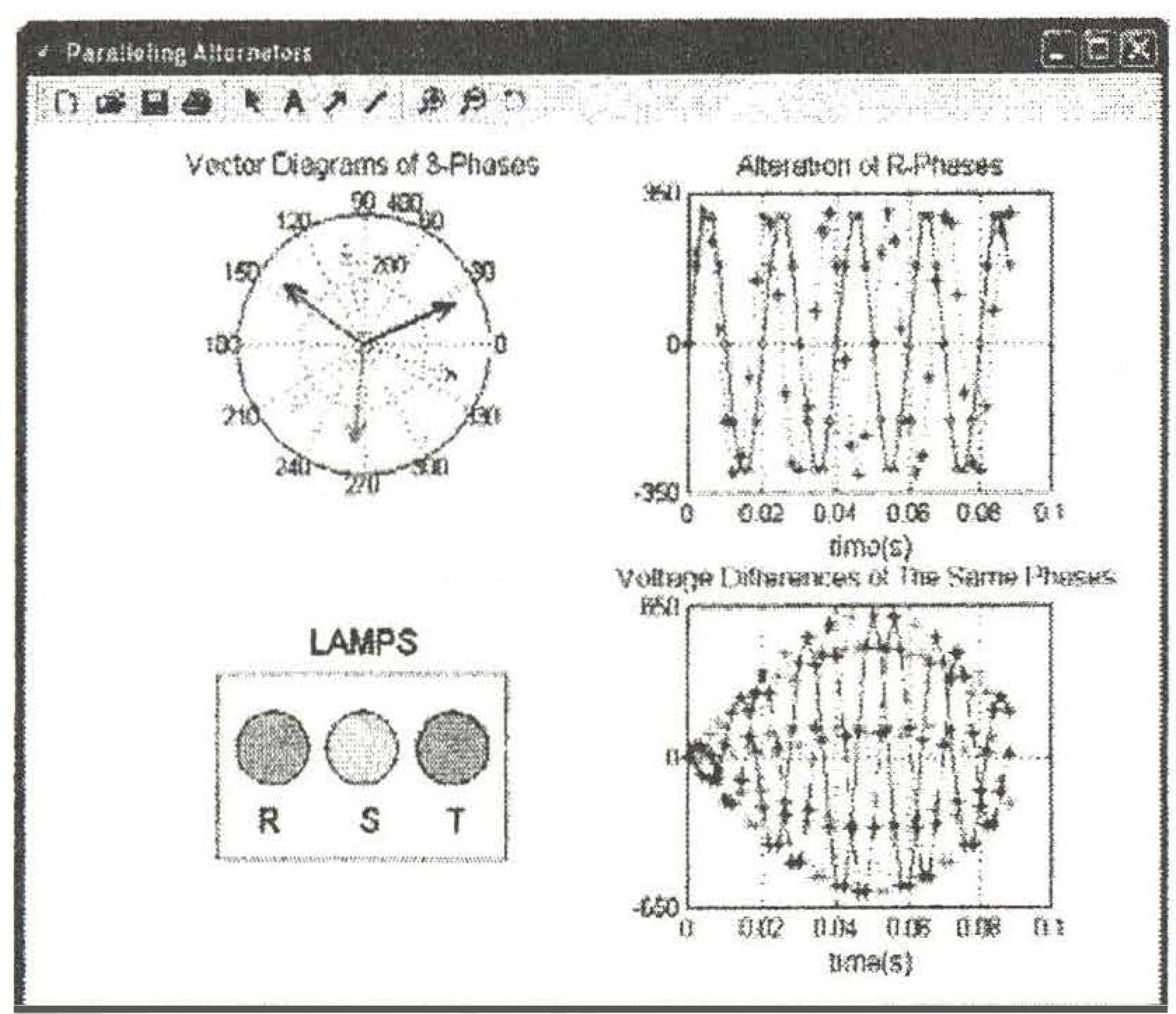
3.4. Paralleling Alternators

Power generating stations operate several alternators in parallel. This practice is preferred to the use of a single large generator due to some reasons. To parallel alternators, the following conditions must be observed: the output voltage of the alternators must be equal; the frequencies of the alternators must be the same; the output voltage of the alternators must be in phase. When these conditions are met, the alternators are said to be in synchronism. The use of synchronizing lamps is a simple way to check these relationships. One way of connecting the synchronizing lamps is called the three lamps dark method. The three lamps will go on and off in unison. The rate at which the lamps go on and off depends upon the frequency difference between two alternators.

In this exemplary application, the dynamic animation of two alternators which will be operated in parallel has been performed. The input parameters of the dynamic animation are the terminal voltage and the frequencies of the alternators. In this case, the phase rotation or phase sequence is supposed to be correct. Instantaneous two positions of the dynamic animation of alternators which will be operated in parallel are given in Fig.5. In here, the terminal voltage was chosen as 310V, and the frequencies were chosen as $f1=50Hz$ (the network) and $f2=60Hz$ (the incoming alternator). Because of clarifying the events graphically, $f2$ was purposely chosen much more bigger than that should be. At the top-left figure in Fig.5, the terminal voltages are expressed with rotating vectors; the solid-lined vectors denote the network; and the dashed-line vectors denote the incoming alternator; the three phases (RST) are colored with red, green and blue, respectively. At the top-right figure in Fig.5, alterations of the R-phases in time are given. The colored lamps, whose tone are proportional to the voltage difference between the same-named phases, are shown at the bottom-left figure in Fig.5. As a function of time, the alterations of the voltage differences between the same-named phases are displayed at the bottom-right in Fig.5. As a result of that animation, the user will see the alterations and effects of the parameters mentioned before.



(a)



(b)

Figure 5. Instantaneous positions of the dynamic animation of paralleling alternators, a) t=0.03s, b) t=0.088s

4. CONCLUSION

Computer-based teaching and learning techniques, using linear transformation and computer graphics, are greatly enhancing the learning process. Computer graphics used for educational purposes are requiring more features (flexibility, reliability, visuality, user-friendly). In this study, a different approach to the dynamic animation and simulation was introduced. This approach is recommended to some educators who want to develop his/her own animation package program in order to use improve the effectiveness of utilizing animations and simulations in his courses. In this approach, Matlab package program is the main equipment to deal with linear transformation matrix because of being matrix-based software. An equipment of secondary importance is linear transformation matrixes and position vectors. While linear transformation matrixes performs transformations such as scaling, translating, rotating and etc.; position vectors constitute the physical part of the system in 2D or 3D-space. The animations are realized with carrying the moving part to required position at the right time. Meanwhile, the stationary parts are not taken into consideration. In this paper, how to make dynamic simulations-animations in electrical education was explained with exemplary applications clearly. Thus, one can constitute special-purposed animations by using the method introduced. Linear transformation matrixes can also be used for three-dimensional simulation and animations; for this purpose, new algorithms and package programs may be developed on further studies.

REFERENCES

[1] Neelamkavil F., O'Tuathail E., "Systems for engineering animation with automatic motion control", Computer-Aided Engineering Journal, vol:8/3, p:103-115, 1991.

[2] Huang N., Cheok T.G., Settle T., "Real-time simulation and animation of suspension control system using TI TMS320C30 DSP", Simulation, vol:61/6, p:405-416, 1993.

[3] Dimirovski G.M., Radojicic P.C., and etc., "Modeling, control and animated simulation of complex processes in robotised FMS", Proc. of the 20th Int. Conference on Industrial Electronics, Control and Instrumentation, part:2/3 p:1141-1146, 1994.

[4] Watkins J., Piper G., and etc, "Computer animation: A visualization tool for dynamic system simulations", Proc. of the 1997 ASEE Annual Conference, 1997.

[5] Villareal S., Wynn C., Eastwood D., Zoghi B., "Design, development and evolution of Web-based materials featuring computer-animated simulations", Proc. of 1998 28th Annual Frontiers in Education Conference, FIE, part:2/3, p:588-593, 1998.

[6] Tilbury D., Messner W., "Control education on the WWW: Using Matlab for Control design, simulation and visualization", Proc. of the 1998 ASME Int. Mechanical Eng. Congress and Exposition, p:65-70, 1998.

[7] Ravn O., Larsen T., Andersen N., "Simulation and animation in Simulink and VRML", Proc. of the 1999 IEEE Int. Symposium on Computer Aided Control System Design, p:120-125, 1999.

[8] Rios C., Lim C.I, and etc, "Multivariable analysis and control of a cart-pendulum-seesaw system using an animation tool", Proc. of the IEEE Conference on Decision and Control, p:825-860, 1999.

[9] Rojiani K.B., "Programming in C with Numerical Methods for Engineers", Prentice–Hall International, Inc., 1996.

[10] Biran A., Breiner M., "Matlab for Engineers", Addison-Wesley Pub. Comp., 1995.

[11] Rogers D. F., "Procedural elements for computer Graphics", McGraw-Hill Pub., New York, 1985.

[12] Rogers D. F., Adams J.A., "Mathematical elements for computer graphics, International 2nd Edition", McGraw-Hill Pub. Comp., 1990.

[13] Duff J.R., Kaufman M., "Alternating Current Fundamentals", Delmar Pub., By Litton Edu. Pub. Inc., 1980.

[14] Krishnan R., "Electric Motor Drives: Modeling, Analysis, and Control", Prentice Hall, Inc., 2001.

[15] Ridsdale R.E., "Electrics Circuits, 2nd ed.", McGraw-Hill Book Comp., 1984.

[16] William E.B., Richard C. DiPrima, "Elementary Differential Equations and Boundary Value Problems, 2nd ed.", John Wiley & Sons, Inc., NY, 1969.