XXI IMEKO World Congress “Measurement in Research and Industry”
August 30 – September 4, 2015, Prague, Czech Republic

SELECTED ASPECTS OF NUMERICAL SOLUTION OF DAMPED OSCILLATOR IN PTC PRIME 3.0 ENVIRONMENT

Piotr Jankowski [1]

[1]Department of Marine Electrical Engineering , Gdynia Maritime University, Gdynia, Poland,
keopiotr@am.gdynia.pl

Abstract – The paper presents the solution of damped nonlinear oscillator with the use of different numerical procedures in Prime 3.0 environment. As a technical example the case of ferroresonant circuit was used. The presented results demonstrate both possibilities of the environment Prime 3.0 and difficulties in achieving convergence of solutions for considered extreme nonlinear example.

Keywords: numerical procedures, chaos, ferroresonance

1. INTRODUCTION

Irregular and unpredictable evolution of many nonlinear systems is called chaos. A characteristic feature of this state is the fact that the system does not repeat their behaviour even approximately. Chaotic state can occur in many technical systems that can be described by means of non-linear system of equations, which usually does not have an analytical solution. In addition, these systems can be sensitive hence a small change of any parameter can affect the final result. Currently, for numerical solutions of these nonlinear issues, a popular environment as Matlab, Mathematica, or Mathcad is applied. However, for certain strongly nonlinear cases a disadvantage in using these environments may be limited representation of a number to 17 digits.

This paper presents both possibilities and difficulties, that may occur in obtaining the numerical solution of nonlinear equations in the PTC Prime 3.0 environment. This program is the latest 64-bit version of Mathcad. As an electrical engineering example, power system model for ferroresonance state has been used [1].The case of chaotic behavior for the ferroresonance state was presented in detail in [2] where the authors discussed the energy system applying unloaded transformer for a break in one phase.

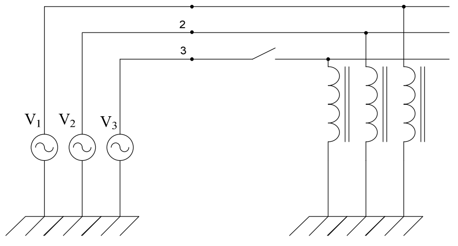
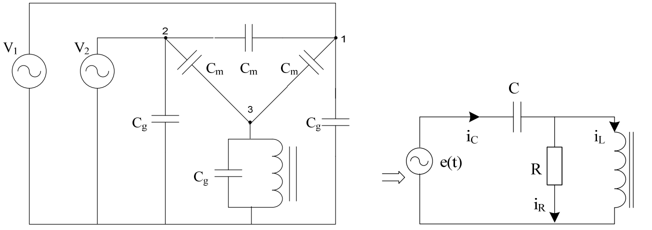


Fig. 1. Power system configuration [2] .

The model for the system of Fig.1 taking into account capacitance between phases and to ground, and reduced circuit in relation to the non-linear inductance is shown in Fig.2.



a) b)

Fig. 2. Model of system a) and its reduced ferroresonant circuit b).

In the previous considerations, the authors of [2] approximated transformer's magnetization characteristics by a 3rd order polynomial which led to the description of the circuit of Fig.2 by Duffing equation. However, further studies showed that the more adequate approximation is a polynomial (1) for n = 11

$$i = a\Psi + b\Psi^{n} \tag{1}$$

The parameters involved in the circuit (Fig.2) and in (1) depend on the type of transformer and power supply line and its length [1]: Parameters that were adopted for testing simulation in Prime 3.0 environment are presented in tab1.

2. NUMERICAL PROCEDURES IN PTC PRIME

Ferroresonant equivalent circuit (Fig.2b) can be described by the following equations (2):

$$\begin{aligned} &\frac{d\Psi}{dt} = -u_c + e(t) \\ &\frac{1}{R}\frac{d\Psi}{dt} - C\frac{du_c}{dt} = -a\Psi - b\Psi^{11} \end{aligned} \tag{2}$$

When appointing voltage u_c from the first equation of system (2) after substituting into the second we get 2rd order differential equation called nonlinear damped oscillator:

$$\frac{d^2\Psi}{dt^2} + \frac{1}{RC}\frac{d\Psi}{dt} + \frac{a}{C}\Psi + \frac{b}{C}\Psi^{11} = \omega E_m \cos(\omega t) \tag{3}$$

Most numerical procedures in *Prime* environment requires the preparation of a system of equations in normal form. This system of equations can be written as matrix equation:

$$\mathbf{A}\cdot\overset{\square}{\mathbf{x}}=\mathbf{b} \qquad (4)$$

where: $\left[\overset{\square}{\mathbf{x}}\right]^T=\left[\dfrac{d\Psi}{dt},\dfrac{du_c}{dt}\right]$

Then the system of equations in normal form is obtained as:

$$\overset{\square}{\mathbf{x}}=\mathbf{D}=\mathbf{A}^{-1}\mathbf{b} \qquad (5)$$

Such a method of obtaining a normal form is convenient for a large number of equations. But we must remember that for the non-stationary case, that is when the elements of the matrix **A** are not constant, matrix inversion takes place in every step of the numerical procedure which considerably increases the computation time. Therefore, if it is possible, for simple cases, one should prepare a ready system of equations in normal form. For the considered system vector **D**, which was used in the applied procedures has the form:

$$\mathbf{D}(t,\mathbf{x})=\begin{bmatrix} -x_1+E_m\cdot\sin(\omega\cdot t) \\ \dfrac{1}{RC}\cdot(-x_1+E_m\cdot\sin(\omega\cdot t))+\dfrac{a}{C}\cdot x_0+\dfrac{b}{C}\cdot x_0^{11}\end{bmatrix} \qquad (6)$$

Table 1. Type size for manuscript (in points).

$e(t)=E_m\sin(\omega t)$	a[A/Wb]	b[A/Wb^{11}]	C[nF]	R[Ω]
E_m=13.5kV	2.8x10^{-3}	7.2x10^{-3}	1250.77	10^{20}

Below there are the results of solution of differential equations (2) for the parameters shown in Table 1 using various procedures in *Prime* 3.0.

2.1. Rkfixed and Odesolve procedure

Rkfixed procedure uses a fourth-order Runge-Kutta method to solve system of first order differential equations. In order to use this procedure in the *Prime* environment enter:

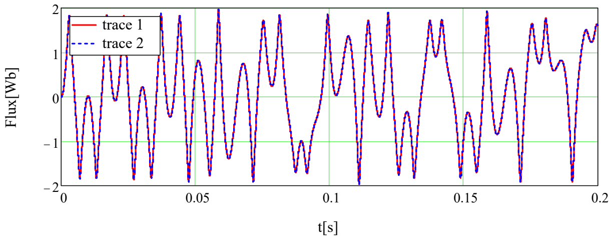
$$X := rkfixed\,(x, t_{initial}, t_{end}, n, D) \qquad (7)$$

where: *X* is a matrix in which the first column is the time and the remaining columns are the solutions for corresponding time, *x* is a vector of initial values, *n* is the number of points, and *D* is vector (6).

Odesolve procedure is very convenient since it allows the direct solution of non-linear equation of k-th order. To use this procedure one should initiate the so called *solve block* by typing the word *given* and then entering differential equation (in the considered case this is (3)) , the initial conditions for time $t_{initial}$ and finally:

$$\Psi := Odesolve(t, t_{end}, n) \qquad (8)$$

The result of *Odesolve* procedure is Ψ(t) function. An additional advantage of this procedure is that the computation time using *Odesolve* is many times shorter compared to other procedures that use vector *D* (6). Fig.3 shows the excellent convergence of flux linkage waveforms obtained from both *rkfixed* and *Odesolve* procedures for a small voltage E_m = 1500V. However, the results for the higher voltage were already diverging.



F

ig. 3. Flux for *rkfixed* and *Odesolve* procedure for E_m=1500V.

In the further examination one tried to achieve convergence of results for different procedures for the voltage as in Table 1, where similar parameters as in [1] were used. For this purpose, simulations were performed for the integral step h = 10^{-5} and the half-step. As seen in Fig.4 for the voltage E_m = 13500V, flux waveforms were converged to a time t = 0.11s

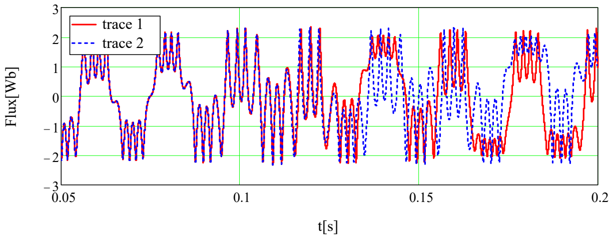


Fig. 4. Flux for *rkfixed* (for h step -trace 1, for 0.5h step -trace 2).

Unfortunately, further reducing the integral step in the *rkfixed* procedure did not improve the convergence.

2.2. Bulstoer procedure

The next procedure which has been used to improve the convergence of solutions was *Bulstoer* procedure which uses Bulirsch-Stoer method for smooth solutions. Its introduction in *Prime* is as follow:

$$X := Bulstoer(x, t_{initial}, t_{end}, n, D) \qquad (9)$$

The argument list and the matrix returned by *Bulstoer* are identical to that for *rkfixed*.

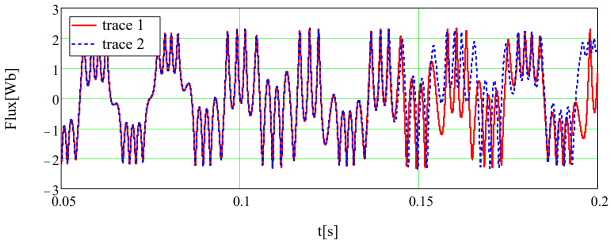


Fig. 5. Flux for *Bulstoer* (for h step -trace 1, for 0.5h step -trace 2).

In this case, as shown in Fig.5 waveforms of solutions for different integrating steps began to diverge from t=0.14s.

2.3. Stiffb and Stiffr procedure

Thanks to a matrix system of equations (4), we can calculate the condition number of matrix **A**. *Prime* has procedures that calculate the condition number based on L1 norm (*cond1*), L2 norm (*cond2*) or Euclidean norm (*conde*):

$$A := \begin{pmatrix} 1 & 0 \\ \frac{1}{R} & -C \end{pmatrix} \qquad \mathrm{cond1}(A) = 7.995 \times 10^5$$

$$\mathrm{cond2}(A) = 7.995 \times 10^5 \qquad \mathrm{conde}(A) = 7.995 \times 10^5$$

The calculated condition numbers indicate that our system is not the best conditioned. Therefore procedures *Stiffb* and *Stiffr* designed for stiff systems were tested:

$$X := Stiffb(x,t_{initial},t_{end},n,D,J) \qquad (10)$$

Both procedures require an additional argument, which is the matrix ***J***. The rest of the argument is the same as for *rkfixed*. ***J*** matrix consists of a column $\partial \mathbf{D}/\partial t$ and Jakobian matrix $\partial \mathbf{D}/\partial x_k$ where **D** is vector (6). For the considered system vector ***J***, has the following form:

$$\boldsymbol{J}(t,x) = \begin{bmatrix} \omega \cdot E_m \cos(\omega \cdot t) & 0 & -1 \\ \frac{1}{RC} \cdot \omega \cdot E_m \cos(\omega \cdot t) & \frac{a}{C} + 11\frac{b}{C} \cdot (x_0)^{10} & \frac{-1}{RC} \end{bmatrix} \qquad (11)$$

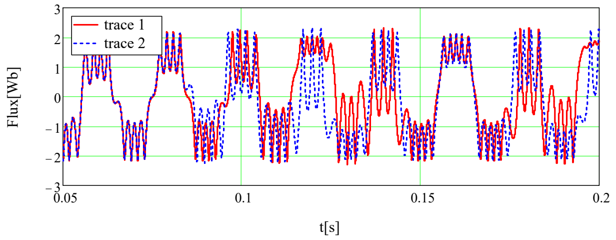


Fig. 6. Flux for *Stiffb* (for h step -trace 1, for 0.5h step -trace 2).

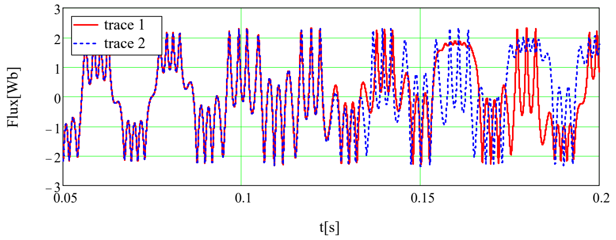


Fig. 7. Flux for *Stiffr* (for h step -trace 1, for 0.5h step -trace 2).

The results of using both procedures are shown in Fig.6 and Fig.7. *Stiffr* procedure is more efficient than *Stiffb* procedure, but the convergence is still unsatisfactory. The resulting effect appears to be worse than for the procedure *Bulstoer*. Therefore, another attempt was made to get a better convergence of solutions of the (3) by applying the procedure *rkadapt.*

2.4. Rkadapt procedure

Rkadapt procedure analyzes how fast the solution is changing and adapts its integration step respectively. The argument list and the matrix returned by *Rkadapt* are identical to that for *rkfixed.*

$$X := Rkadapt(x,t_{initial},t_{end},n,D) \qquad (12)$$

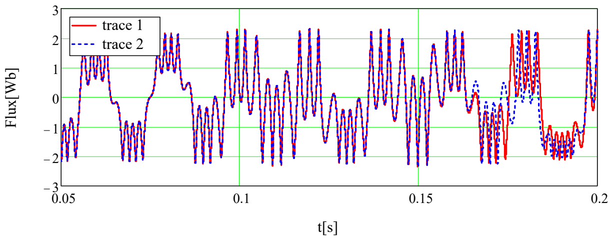


Fig. 8. Flux for Rkadapt (for h step -trace 1, for 0.5h step -trace 2).

Observing Fig.8 one can see that the convergence of solutions for *Rkadapt* procedure is satisfactory for the time t=0.165s. Unfortunately, similarly as for all the above procedures, reducing the integration step did not improve the convergence of the solution.

3.CONVERGENCE AND PHASE PORTRAITS

To avoid numerical problems associated with widely different coefficients of differential equation conversion approach to dimensionless form is applied. The dimensionless form of (9) is following:

$$\frac{d^2x(\tau)}{d\tau^2} + k_2 \cdot \frac{dx(\tau)}{d\tau} + k_3 \cdot x(\tau) + k_4 \cdot x(\tau)^{11} = \cos(\tau) \quad (13)$$

where: $k_2 = \frac{1}{\omega_s RC} \quad k_3 = \frac{a}{\omega_s^2 C} \quad k_4 = \frac{b}{\omega_s E_m \cdot C} \cdot \left(\frac{E_m}{\omega_s}\right)^{11}$

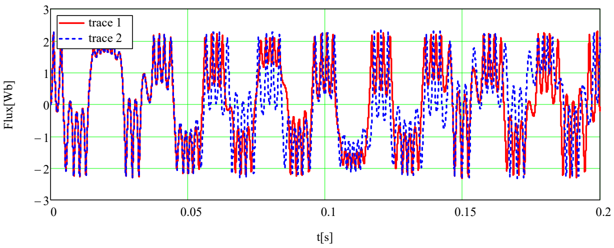


Fig. 9. Flux for *Odesolve* for dimensional-(trace1) and dimensionless form -(trace2).

In Fig. 9 the solution results of both (3),(13) equations (after scaled results from dimensionless equations) are compared. As it can be seen, from a certain point bifurcation of waveforms begins, although these waveforms for the case b=0 are of course identical.

As with the previous procedures, reducing integration step that is increasing the number of steps to a fixed range does not improve convergence.

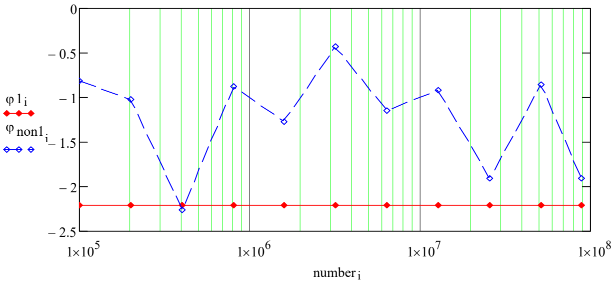


Fig. 10. Flux for *Odesolve* for dimensional and nondimensional form against number of steps (t=0.05s).

Fig. 10 shows the value of fluxes for t=0.05 obtained from the dimensionless and dimensional equation with the increasing number of steps. In turn, Fig. 11 shows the relative error defined between the i-th and i + 1 iteration (it means $h_{i+1}=0.5h_i$)

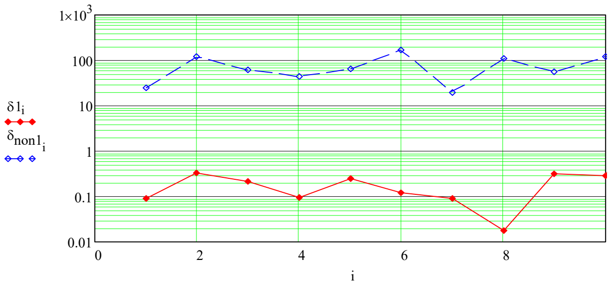


Fig. 11. Flux relative error for *Odesolve* for dimensional and nondimensional form.

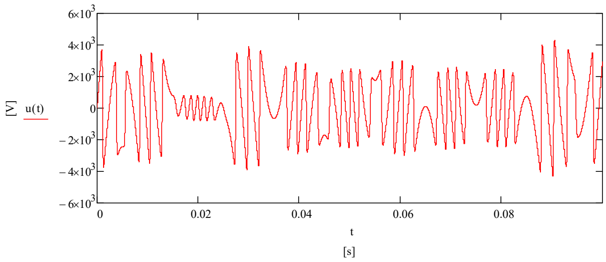


Fig. 12. Voltage across coil.

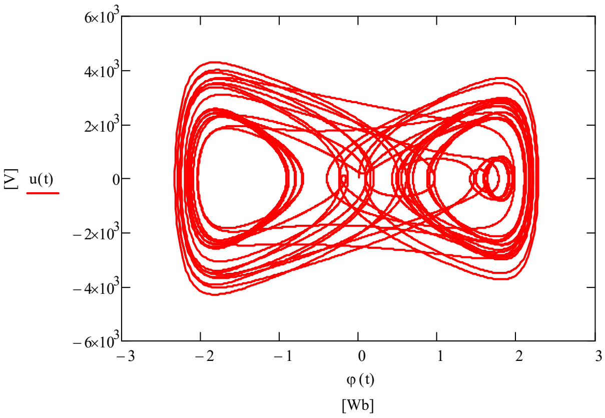


Fig. 13. Phase portrait for (0.1s) time range and E_m=13.5kV

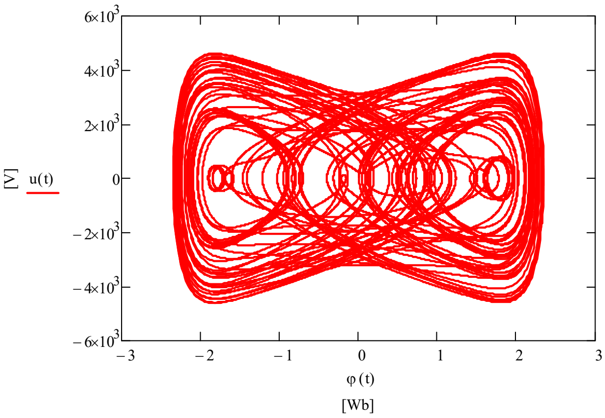


Fig. 14. Phase portrait for (0.2s) time range and E_m=13.5kV

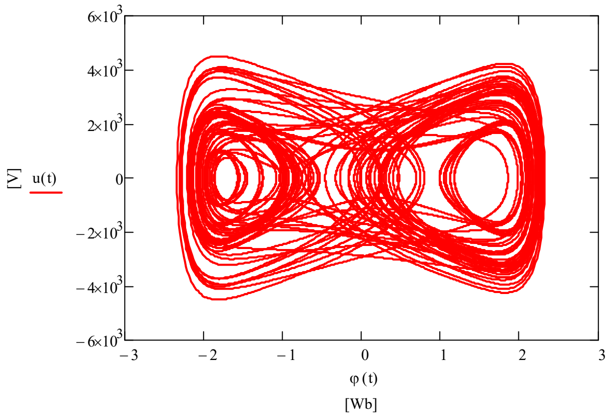


Fig. 15. Phase portrait for (49s+10T) time range and E_m=13.5kV

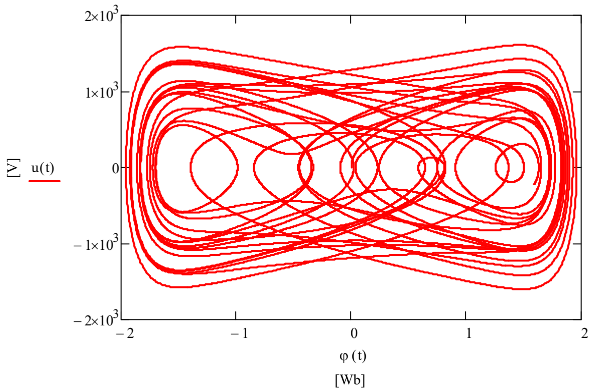


Fig.16. Phase portrait for (0.2s) time range and E_m=1500V

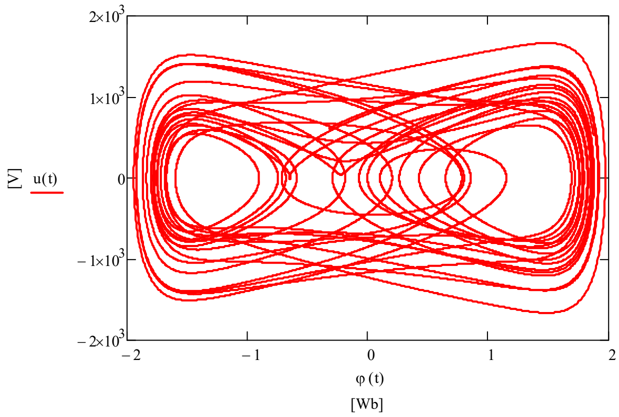


Fig.17. Phase portrait for (49s+10T) time range and E_m=1500V

It should be stressed that the *Odesolve* procedure apart from being very fast has another advantage over the other procedures. *Odesolve* returns the result interpreted by *Prime* as a function, which allows direct calculation of the derivative with an existing differentiation procedure in *Prime*. In the present example, the voltage across coil can be obtained by calculating the derivative of flux function (Fig.12). In turn, with the flux function and its derivative, we can determine the phase portrait. Fig.13 shows the phase portrait of the (2) for the range to t_{end}= 0.1s where convergence of solutions was met, and Fig.14 shows similar phase portrait but for bigger range (t_{end}=0.2) for which the convergence is not achieved.

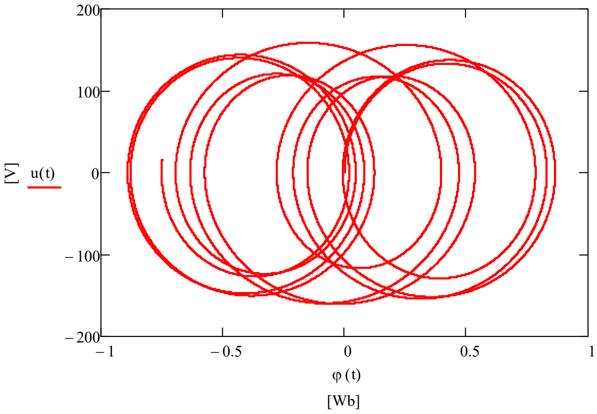


Fig.18. Phase portrait for (0.2s) time range and voltage E_m=135V

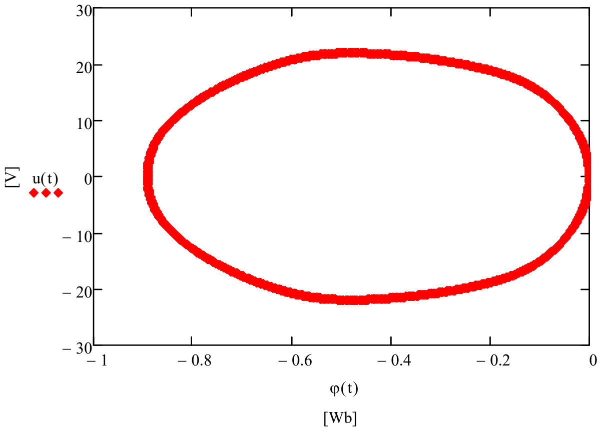


Fig.19. Poincare map for E_m=135V

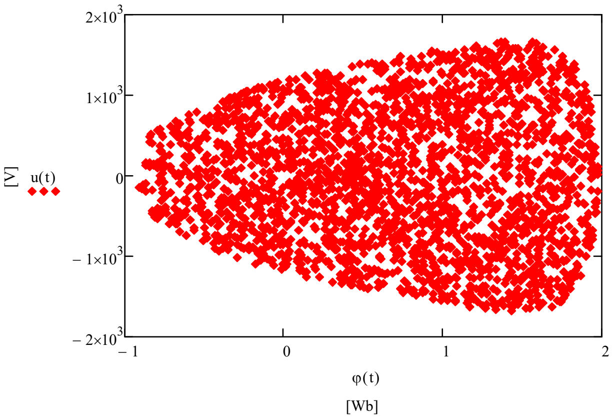


Fig.20. Poincare map for E_m=1500V

Because the waveforms obtained at E_m = 13.5kV and for range (0;0.2s) have not reached periodicity hence additional simulations were performed for a wide range (0;50s). Fig.15 shows the phase portrait for this simulation, but in the interval (49;49.2s), because programs like *Prime* generate plots limited to 50000 points. An interesting result is the phase portrait for E_m = 1500V in the interval (0;0.2s) (Fig.16), for which the course of the flux was converging. The phase portrait for the interval (49; 49.2s) has a similar character (Fig.17). However, it turned out that for the range much greater than 0.2s the charts of flux (for steps h and 0.5h) , began to diverge.

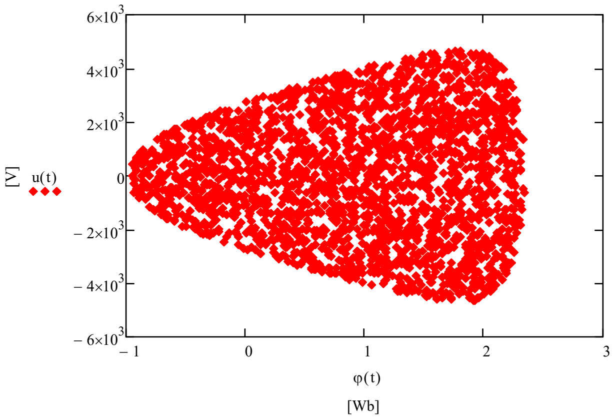


Fig.21. Poincare map for E_m=13.5 kV

Further simulations have shown that at E_m = 135V convergence was maintained in arbitrary range. Phase portrait for this case is shown in Fig.18 and it kept this form for arbitrary range. In chaos theory, a better representation of the chaotic behavior is the Poincare map (Fig.20,21,22.) As one can see, only for E_m = 135V the Poincare map has the form a closed curve (Fig.19). For the remaining cases, the maps in Fig.20 and Fig.21 are typical of chaotic behavior.

4. CONCLUSIONS

The presented examples of solutions of strongly nonlinear equation like damped oscillator showed great difficulties in obtaining convergence in the *Prime* environment. Their evidence is the difference of the results obtained from the dimensional and dimensionless equations. *Prime* procedures in case of problems with obtaining convergence indicate the following message: *not converging*. In all presented examples this message does not appear. Therefore, in such cases, the results should be viewed very critically because, as it was shown by above simulations, for an exceeded range it was not possible to achieve a satisfactory convergence even for the maximum number of $89x10^6$ steps in *Prime*. Perhaps a greater range for convergent solutions would be obtained if the representation of a number was not limited to 17 digits[4]. Nowadays, the properties of such cases are analyzed on the basis of the chaos theory. An interesting result seems to be the similarity of phase portraits despite different time ranges, where the solution is convergent in one case, and divergent in the second case. For this reason, the presented cases indeed showed the advantages of Poincare maps in the identification of chaotic states.

ACKNOWLEDGMENTS

The project was financed by the National Science Centre Poland under contract no. DEC-2012/07/E/ST8/01688

REFERENCES

[1] J. R. Marti, and A.C Soudack, “Ferrresonace in power system: fundamental solution”, *IEE Proc.C,* pp 321-329, July 1991.

[2] A.E.. Araujo, J.R. Marti and A.C Soudack, “Ferroresonance in power systems: chaotic behaviour”, *IEE Proc.C*, Vol.140, No3. May 1993

[3] B. Pałczyńska “Time -frequency analysis of non-stationary magnetic fields” *XVI IMEKO*, pp. 31-35, Florence Italy 2008

[4] IEEE 754-2008 - Standard for Floating-Point Arithmetic. DOI 10.1109/IEEESTD.2008.4610935, 2008.