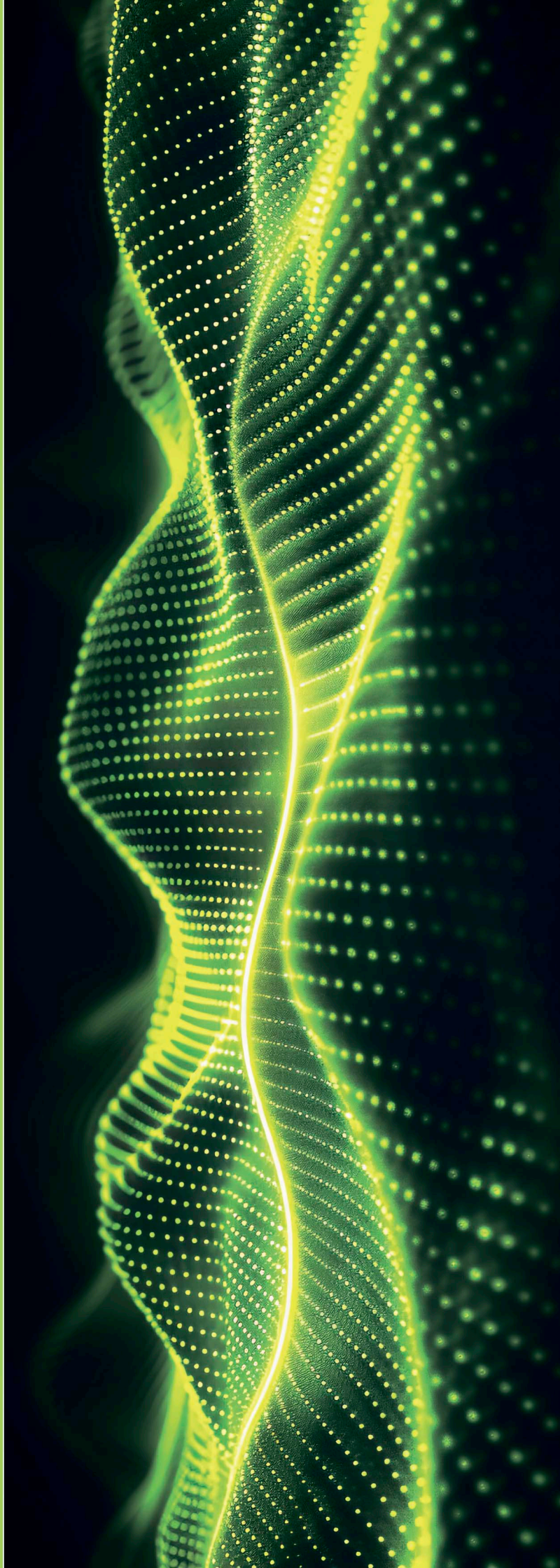
L. Glukhivskyi

Nonlinear oscillations: numerical polyharmonic simulating



L. Glukhivskyi

Nonlinear oscillations: numerical polyharmonic simulating

Kyiv 2024

DOI: http://doi.org/10.35668/978-966-479-146-2

ISBN 978-966-479-146-2 (online)
UDC 534 : 621.3.021:681.3.068.004.4

Reviewers: I. Kuzio, doct. of engin., prof.
V. Maliar, doct. of engin., prof.

Glukhivskyi L. Yo. Nonlinear oscillations: numerical polyharmonic simulating [Electronic resource] / L. Glukhivskyi ; Translation from Ukrainian and editors L. Yo. Glukhivskyi. – Kyiv : UkrISTEI, 2024. – 198 p.
Author's translation from the book *Л. Й. Глухівський «Нелінійні коливання: чисельне полігармонічне моделювання», Київ : «Альфа ПіК», 2008.*

The book describes the basics of the differential harmonic method, which belongs to the class of methods for numerical modeling of nonlinear oscillations, that is, periodic processes, in systems of one or another physical nature, the variables of which are connected by nonlinear connections.The method is based on finding periodic solutions of systems of nonlinear differential equations with their approximation by Fourier series and taking into account the required number of higher harmonics.

The basic software of the differential harmonic method, developed in the FORTRAN-90 language, is described. It consists of 24 software modules. When modeling a specific nonlinear oscillation, it is necessary to develop three more modules, the development principle of which is given. Ten examples of computer modeling of various types of nonlinear oscillations are given, in particular: forced, parametric and free.

The book is intended for specialists in the field of computer modeling and analysis of periodic processes in nonlinear systems of mechanics, electrical engineering, automatic control, radio physics, acoustics, etc., and can also be used as a study guide for graduate students and doctoral students in the relevant sciences.

ISBN 978-966-479-146-2 (online)

Preface to the translation of the book

The monograph [75], the English translation of which is offered to your attention, was published in 2008 in Ukrainian. The print run of 500 copies sold out a long time ago. An electronic copy of this book was exhibited and is exhibited in a number of electronic libraries: at National University "Kyiv Polytechnic Institute named after I. Sikorskyi", National University “Lviv Polytechnic”, Zaporizhzhia National University, in the Vernadsky Library, in Z-library, in the reference database "Ukrainika Naukova", etc.

Analysis of references in scientific articles and dissertations to this book and visits to its electronic copies in electronic libraries of Ukraine showed that the material presented in the book is still relevant. The publication of its English translation and its location in electronic libraries and repositories will make the book, we hope, available to a much wider circle of scientists, graduate students and students, not only in Ukraine, but also abroad.

Today, Ukraine is on the path of integration into the European space, in particular in the scientific and technical sphere. The author's desire to participate in this process prompted him to undertake the work of translating his book into English. The completed translation was edited by the author.

During the translation, typographical errors and inaccuracies found in the original have been corrected, and some additions suggested by the author have also been taken into account.

The foreword

One of the sections of the theory of oscillations, which studies the general laws of oscillatory processes in various systems of different physical nature, is the theory of nonlinear oscillations - oscillations in systems with nonlinear connections. When modeling processes in such systems, it is usually necessary to take their nonlinearities into account as fully as possible.

The development of various theories, including and separate sections of the theory of nonlinear oscillations, was always stimulated by the need to solve specific applied problems in one or another scientific and technical field. The author of this book worked for a long enough time on solving the problems of calculating periodic processes in nonlinear electrical engineering and electromechanics. To calculate nonlinear oscillations in systems and devices of this area, he developed a special numerical method called ***differential harmonic***. It was based on the well-known in the theory of oscillations method of harmonic balance and some numerical methods for solving nonlinear systems of finite (algebraic, transcendental) equations using the apparatus of differential calculus. The author's two monographs [17, 20] can be considered a certain summary of this work.

This book is attempt of generalization of the received results and representations of the developed method together with its software as sufficient universal tool for calculation of nonlinear oscillations in systems of the different physical nature, not only in area of electrical engineering.

In terms of its style, this book is an attempt to present the material as a study guide for senior students and graduate students of technical universities in a wide range of specialties: electrical engineering, radio physics, electronics, automation, mechanics, acoustics, etc., where the problems of modeling nonlinear oscillations arise. The author hopes that the book will be used by specialists in computer calculation and design.

The book also can serve as the manual for development of the special software for scientific and engineering calculations, in particular - on algorithmic language FORTRAN.

The manuscript of the book was carefully reviewed by the reviewers - employees of the Lviv Polytechnic National University, Doctor of Technical Sciences, Professor I.V. Kuzyo and Doctor of Technical Sciences, Professor Malyar V.S. They made a number of suggestions for improving the presentation of the material, and the author is very grateful to them for this.

The author will be grateful to everyone who will send feedback about this book and, perhaps, send suggestions and wishes for its improvement to the author's email: gl.lev42@gmail.com

INTRODUCTION

The generalized coordinates of elements of technical systems of any physical nature (mechanical, acoustic, hydraulic, electric, radio-electronic, automatic control etc.) can be considered as variable functions of time. Among those coordinates there are: a) mechanical linear or angular distance, speed and acceleration; b) electric charge, current and voltage; c) magnetic intensity, induction, flux, etc. If change of coordinates of system in time is nonmonotonic they say there is an oscillatory process in this system .

Oscillatory processes in physical systems are either steady-state (stationary) or transitive processes from some steady-state oscillatory processes to other steady-state processes. A separate case of stationary oscillatory process is periodic oscillatory process (periodic oscillation). Periodic oscillation is such oscillation in which dependences of coordinates of the system of time are periodic functions.

The simplest kind of periodic oscillation is a simple harmonic oscillation (or simply - a harmonic oscillation)

$$a[t] = A_c \cos \omega t + A_s \sin \omega t \qquad \text{(B.1a)}$$

or

$$a[t] = A \cos(\omega t + \varphi), \qquad \text{(B.1b)}$$

here a - variable coordinate; A_c, A_s, A - amplitudes of harmonic oscillation of this coordinate; t - time; ω - circular frequency; φ - initial phase.

Note that in the expressions (B.1), the variable t is written in square brackets before the equal sign and after the variable a. In these expressions (and in other expressions later in this book), this way of writing means that the variable a is the dependent variable of another variable t that serves as its argument, that is, it is the independent variable. If a variable depends on two or more independent variables, they must be separated by commas when placed in square brackets.

Amplitude and initial phase in expression (B.1b), amplitudes in expression (B.1a) are connected by the following dependences

$$A = \sqrt{A_c^2 + A_s^2}; \quad \varphi = arctg(A_s / A_c). \qquad \text{(B.2)}$$

Dependence of the kind (B.1) is known as a common solution (primitive integral) of the differential equation

$$\frac{d^2 a}{d t^2} + \omega^2 a = 0, \qquad \text{(B.3)}$$

which describes free oscillations (they also say - natural oscillations) in linear conservative systems, i.e. without a energy dissipation. For example, it can be: a mechanical vibratory system - a horizontally located spring fixed at one end and connected with a ball on the other end; a mechanical mathematical pendulum in the form of a ball hanging on a non-elastic string; an electric oscillatory circuit made up by an inductance and an electric capacitor, which are connected in series etc.

A ball, an element of the system in the first example, is subjected to action of two forces: the force of inertia

$$F_1 = m\frac{d^2x}{dt^2}\ , \qquad \text{(B.4)}$$

which is product of the weight of the ball m and its acceleration (of the second time derivative of the deviation x of the ball from equilibrium position), and the elastic potential force

$$F_2 = s\,x, \qquad \text{(B.5)}$$

which is product of rigidity of the spring s and the deviation x. Under the Newton's second law the sum of these two forces is equal to zero:

$$F_1 + F_2 = m\frac{d^2x}{dt^2} + s\,x = 0. \qquad \text{(B.6)}$$

Having designated

$$x = a \quad ; \qquad s/m = \omega^2, \qquad \text{(B.7)}$$

we come to the equation of the kind (B.3).

In the second example, the inertia force which acts on a ball (the ball has mass m, the string has length l, the angle of deviation of the string from vertical position - θ, this force is tangent to the circle described by the ball) is equal to

$$F_1 = m\,l\frac{d^2\theta}{dt^2}. \qquad \text{(B.8)}$$

The potential restoring force, which is tending to return the ball to the state of balance, is equal to

$$F_2 = m\,g\sin\theta, \qquad \text{(B.9)}$$

here g is the acceleration of the gravitation.

As in the previous case, if considering that the sum of these two forces according to the Newton's second law is equal to zero, and also that at small angles of deviation of the string from vertical it is possible to accept $\sin\theta \approx \theta$, and if to designate

$$\theta = a; \quad g/l = \omega^2, \qquad \text{(B.10)}$$

again we come to the equatio of the kind (B.3).

In the third example (the electric oscillatory circuit created by a capacitor which has the charge q and by the inductive coil connected with the flux ϕ) current i of the circuit and voltage u_c of the caoacitor are connected by means of two equations: the equation written under the second Kirchgoff law

$$\frac{d\phi}{dt} + u_c = 0 \qquad \text{(B.11)}$$

and the equation connecting the values of the current and the charge of the caapacitor

$$\frac{dq}{dt} = i. \qquad \text{(B.12)}$$

Suppose the capacitor and the inductive coil of this circuit are electromagnetically linear

$$\phi = Li; \quad q = Cu_c, \qquad \text{(B.13 a,b)}$$

where L - the inductance of the coil and C - the capacity of the condenser (constants). After differentiation of the left and the right parts (B.12) with respect to independent variable t we shall receive

$$\frac{di}{dt} = \frac{d^2q}{dt^2} \quad . \qquad \text{(B.14)}$$

In view of (B.13a) and (B.14)

$$\frac{d\phi}{dt} = L\frac{di}{dt} = L\frac{d^2q}{dt^2} \quad . \qquad \text{(B.15)}$$

After insertion in the formula (B.11) the expressions for the derivative $d\phi/dt$ from the formula (B.15) and the expressions for voltage u_c from the formula (B.13b) and accepting the designations

$$q = a \quad ; \quad \frac{1}{LC} = \omega^2 \qquad \text{(B.16)}$$

we come again to the equation of the kind (B.3).

It is expedient to note, that in respect of electric circuits the terms "oscillation of stream", "oscillation of voltage", etc. as a rule are not used in special literature, and there the terms "alternating current", "alternating voltage" are used.

Examples of oscillations which we have considered above, are oscillations in conservative systems (in systems without dissipation of energy, which are under influence only of potential forces). Oscillations in such systems are not being attenuated, their amplitudes depend only on the initial conditions.

In a dissipative system, in which processes are accompanied by dissipation of energy, the steady oscillations of its coordinates of the kind (B.1) can exist only in the presence of a periodic external driving force whose work compensates this dissipation of energy.

If dissipation of energy in a system is caused by viscous damping it is necessary to consider the force of viscous damping which is proportional to the speed, that is the first derivative of coordinate a. An example of such amortization in mechanical systems is friction in a hydraulic shock-absorber. The equation of movement in such systems looks like [27]

$$\frac{d^2a}{dt^2} + 2h\frac{da}{dt} + \omega^2 a = p[t], \qquad \text{(B.17)}$$

here $h-$ factor of amortization.

By means of this equation it is possible to describe also change in time of the electric charge of a condenser in an oscillatory circuit which besides inductance and electric capacity has also the active resistance r connected in series with them and which dissipates energy. It becomes obvious, if besides designations (B.16) also the designation

$$r / L = 2h \qquad \text{(B.18)}$$

is accepted and $p[t]$ considered as alternating electromotive force of an external source.

If in expression (B.17) external driving force $p[t]$ is a simple harmonious oscillation, in this case the expression (B.1) is the periodic solution of this differential equation.

Writing down the equation of movement, in the further we shall transform higher order differential equations to a system of first order differential equations. For example we shall transform the second order differential equation (B.17) to a system of two first order differential equations

$$\frac{da}{dt} - b = 0; \quad \frac{db}{dt} + 2hb + \omega^2 a = p \qquad \text{(B.19)}$$

also we shall write down this system in a matrix-vector form

$$\frac{d\vec{x}}{dt} + \vec{z} = \vec{e}, \qquad \text{(B.20)}$$

here

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\| = \left\| \begin{matrix} a \\ b \end{matrix} \right\|; \qquad \text{(B.21)}$$

$$\vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -b \\ 2hb + \omega^2 a \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ 2hx_2 + \omega^2 x_1 \end{matrix} \right\|; \qquad \text{(B.22)}$$

$$\vec{e} = \left\| \begin{matrix} e_1 \\ e_2 \end{matrix} \right\| = \left\| \begin{matrix} 0 \\ p[t] \end{matrix} \right\| . \qquad \text{(B.23)}$$

In the vector differential equation (B.20) the vector $\vec{z}$ is a linear function of vector $\vec{x}$ because the differential equation (B.17) is linear :

$$\vec{z} = \left\| \begin{matrix} 0 & -1 \\ \omega^2 & 2h \end{matrix} \right\| \cdot \vec{x} \quad . \qquad \text{(B.24)}$$

A more complicated case of periodic oscillations is polyharmonic oscillation, that is such oscillation which can be described presented by the sum of two or more simple harmonic

oscillations of different multiples frequencies. In a common case such oscillation is described by Fourier series in the form of

$$a[t] = A_0 + \sum_{\nu=1}^{n} (A_{c\nu} \cos \nu\omega t + A_{s\nu} \sin \nu\omega t) \qquad \text{(B.25a)}$$

or

$$a[t] = A_0 + \sum_{\nu=1}^{n} A_\nu \cos(\nu\omega t + \varphi_\nu). \qquad \text{(B.25b)}$$

In linear systems (in systems in which all connections are expressed by linear dependences) polyharmonic oscillations are possible only when external driving forces which cause these oscillations, also are polyharmonic. In this case, the external driving force in equation (B.19) should be as follows

$$p[t] = P_0 + \sum_{\nu=1}^{n} (P_{c\nu} \cos \nu\omega t + P_{s\nu} \sin \nu\omega t), \qquad \text{(B.26)}$$

and then dependences of a kind (B.25) are the periodic decision of the differential equation (B.20).

It is known that the principle of superposition extends to linear oscillatory systems. According to this principle the harmonics of oscillatory processes of various orders (with different values of variable ν) are mutually independent , and in expression (B.25), which is a solution of the equation (B.20), only the amplitudes of those harmonics which are present in driving force (B.26) differ from zero.

In nonlinear systems, that is systems, what have communications with nonlinear parameters (or even one communication is nonlinear), periodic oscillations of coordinates are polyharmonic even then when external driving forces are simple harmonic oscillations as nonlinearity of system generate the high harmonics. In this case the formula of a kind (B.25) precisely describe oscillatory process only if $n = \infty$ (infinity). In practical calculations of oscillations in nonlinear systems (oscillations in nonlinear systems also are named *nonlinear oscillations* among of specialists) value of n they aspire to take as possible smaller, but not smaller from that value at which necessary accuracy of approximation of oscillations is satisfied.

If in the equation of the kind (B.20) dependence of the vector $\vec{z}$ from the vector $\vec{x}$ is nonlinear we shall count nonlinear such equation . A subject of consideration in this book in the further it will be the nonlinear vector differential equations and algorithms of search of their periodic solutions.

As the nonlinear equation of the kind (B.20) - (B.23) contains driving force $p[t]$ which is function only of time, it is non-autonomous equation. Nonlinear oscillations in systems which are described by such equations, are the *forced oscillations.*

The equation of the kind (B.20), describing oscillation in some systems, can not contain driving forces which are functions only time, but time is available among arguments of vector function $\vec{z}$. Then the equation of movement of system gets the kind

$$\frac{d\vec{x}}{dt} + \vec{z}[\vec{x}, t] = 0. \qquad \text{(B.27)}$$

Nonlinear oscillations in systems which are described by the equation of the kind (B.27), can arise only when there is periodic change of parameters of rigidity, inertia or dissipation of energy and when there is their certain parity. Such oscillations have name *parametrical oscillations* .

If in the equation of the kind (B.27) time among arguments of function $\vec{z}$ is absent

$$\frac{d\vec{x}}{dt} + \vec{z}[\vec{x}] = 0 \ , \qquad \text{(B.28)}$$

this equation is autonomous, and oscillations, as the periodic solution of this equation, are having name - *self-oscillations*. Dissipative systems which can have such oscillations, should have mechanisms of replenishment of energy which is being dissipated during each period of oscillations.

Not always the differential equations of movement of systems write down so that their coordinates, dependences of which are determining, are being contained under derivative signs. In such cases under derivative signs other variables which are functionally connected with these coordinates write down.

As an example we shall consider the electric nonlinear one-planimetric circuit containing consistently connected inductance, which linkage ϕ is nonlinear function (owing to saturation) the current i of a circuit

$$\phi = \phi[i] \ ; \qquad \text{(B.29)}$$

active resistance, whose voltage drop u_r is nonlinear function of the circuital current

$$u_r = u_r[i] \ ; \qquad \text{(B.30)}$$

capacitance, whose charge q is nonlinear function of the voltage drop u_c of capacitance

$$q = q[u_c] \qquad \text{(B.31)}$$

and the electromotive force e , and they are periodic functions of time. The differential equation of the kind (B.20) for this circuit is:

$$\frac{d}{dt}\vec{y}[\vec{x}] + \vec{z}[\vec{x}] - \vec{e}[t] = 0 \, , \qquad \text{(B.32)}$$

here

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\| = \left\| \begin{matrix} i \\ u_c \end{matrix} \right\| ; \quad \vec{y} = \left\| \begin{matrix} y_1 \\ y_2 \end{matrix} \right\| = \left\| \begin{matrix} \phi \\ q \end{matrix} \right\| ; \qquad \text{(B.33 a, b)}$$

$$\vec{z} = \begin{Vmatrix} z_1 \\ z_2 \end{Vmatrix} = \begin{Vmatrix} u_r[i] + u_c \\ -i \end{Vmatrix} = \begin{Vmatrix} u_r[x_1] + x_2 \\ -x_1 \end{Vmatrix}. \qquad \text{(B.33 c)}$$

Here the vector $\vec{z}$ is nonlinear function of a vector $\vec{x}$ (nonlinear function $u_r[x_1]$ is a component of a variable z_1). For more general case the vector $\vec{z}$ can be function also of vector $\vec{y}$:

$$\vec{z} = \vec{z}[\vec{x}, \vec{y}]. \qquad \text{(B.34)}$$

The equations of kinds (B.20) and (B.32) are written down in normal Cauchy form - they are solved concerning derivatives. More general form of image of the vector differential equation is the following form

$$B\frac{d}{dt}\vec{y} + \vec{z} - \vec{e} = 0\ , \qquad \text{(B.35)}$$

here B - some constant matrix. If this matrix is identity matrix, image of the equations is equivalent as image in normal form. On occasion can be $\vec{y} = \vec{x}$, then $\vec{z} = \vec{z}[\vec{x}]$, and the equation becomes as (B.20), (B.27) or (B.28).

Separate rows of a matrix B can be zero, then equation of a kind (B.35 represents system of the algebraic and differential equations.

Definition of nonlinear oscillations in any system of any physical nature from the point of view of mathematics is search of the periodic solution (or of periodic solutions) of nonlinear systems of the differential equations, which describe this system.

In development of methods of calculation and the analysis of nonlinear oscillations (search of periodic solutions of nonlinear systems of the differential equations describing these fluctuations) during long time , since the end of 18-th century, were engaged many known scientists: A. Puankare, A. Lyapunov, Van der Pol, J. Strett, A.Krilov, A.Andronov, L.Mandelshtam, M. Bogolyubov, J. Mitropolsky, j. Hale, T. Hayasy and others. They have developed set of the analytical methods, which have enabled to comprehend essence of nonlinear oscillations.

The beginning of a computer epoch has given a strong push to development of numerical methods of search of periodic solutions of nonlinear systems of the differential equations. In this direction, in particular - in the field of nonlinear electrical engineering, it is necessary to note T.Aprill's and T.Trik's investigations and a significant contribution of the Ukrainian scientists G.Pukhov, A.Samoylenko, V. Bondarenko, A. Petrenko, L.Sinitskiy, R.Filts, etc.

Consideration of methods of calculation of nonlinear oscillations can be found among set of monographies and manuals, in particular in [2, 8, 10, 27, 29, 32, 37, 38, 40, 42, 45, 46, 51, 53, 55, 56, 60, 62, 65, 66].

The methods which are based on Fourier series are take a separate place among methods of calculation of nonlinear oscillations. The most known among them are: [4-7, 27, 32, 46, 51]: a method of harmonic linearization, a method of small parameter, asymptotic methods of Krylov and Bogolyubov, a method of harmonic balance. Further we shall briefly characterize their essence.

In the method of harmonic linearization we accept an assumption: input variables and output variables for each nonlinear system are monoharmonious (the high harmonics are neglected). Amplitudes of harmonics of output variables are defined on amplitudes of input variables by means of formulas for definition of coefficients of Fourier series, thus real nonlinear relations in system are replaced by linear relations between the first harmonics of input variables and output variables.

According to a method of small parameter at the solved differential equations allocate separately nonlinear part from a multiplier, a being small parameter. So, if in the independent equation of a kind (B.17) the adder instead of constant multiplier $2h$ contains nonlinearity $F[a, \frac{da}{dt}]$ the equation is led to a kind

$$\frac{d^2a}{dt^2} + \varepsilon F \frac{da}{dt} + \omega^2 a = 0 \ . \qquad \text{(B.36)}$$

When the parameter ε has zero value the oscillatory system corresponding this equation, becomes linear conservative. The periodic solution for last is accepted as approximation to which at some nonzero value ε search for corrections in the form of series on degrees of this small parameter. The method can be applied only to quasilinear systems for which the parameter ε is small.

By means of asymptotic methods of Krylov - Bogolyubov, in particular - of averaging method (other name - a method of slowly varying amplitudes), periodic solution for quasi linear systems with small attenuation is searching as monoharmonious fluctuation of a kind (B.1), in which amplitudes (or amplitudes and phases) are functions slowly varying . The nonlinear equations which are being solved are being reduced to more simple equations by means of averaging on the period of values of amplitudes . These equations is called a bridged.

Among above considered methods the method of harmonic balance differs that it can be applied not only to quasi linear systems, and to essentially nonlinear, and allows to use for search of dependences of movement of coordinates not only one harmonic, and to search for the solution in the form of polyharmonious approximation. The essence of a method is replacement of periodic solutions in the differential equations describing oscillation by Fourier series. Thus the problem of search of the periodic solution of the differential equations is reduced to the solution of system of the nonlinear finite equations (i.e. not differential – algebraic, transcendental), whose unknown variables are amplitudes of harmonics. The methods which

are based on such approximation, in [27] refer to as frequency methods of obtaining of periodic solutions.

From the beginning the method of harmonic balance was considered by researchers as analytical: with its help at the analysis of nonlinear oscillatory systems searched for the analytical dependences connecting amplitudes of harmonics of variable coordinates with parameters of system and amplitudes of harmonics of driving forces. It opened opportunities to the analysis of the found dependences and definition of conditions of existence of fluctuations. However at reception of such dependences when except for the account of the basic harmonic the task of the account and the high harmonics is put, it leads to so difficult analytical transformations and formulas (especially at presence of nonlinearity in the functional dependences, connecting two and more variable), that they lose "transparency" and practical suitability to the analysis. Therefore only in rare instances at the analysis of real nonlinear oscillatory systems, except for the most simple this method is possible to consider even one of the high harmonic together with the basic harmonic [7].

If the method of harmonic balance is considered numerical, that is, when it is applied in each specific case, the goal is not to obtain analytical dependences of of amplitudes harmonics of variable coordinates, but only numerical values of amplitudes, then this avoids complications when accounting for each next higher harmonic. Such application of a method of harmonic balance in a combination with some numerical methods of the solution of systems of the nonlinear algebraic equations was offered by the author of this book to the solution of some important applied problems in area of nonlinear electrical engineering [9, 11 - 26, 68]. The offered updating of a method of harmonic balance as of universal numerical method of definition of periodic solutions of nonlinear systems of the differential equations has been named by author the ***differential harmonic method***. One of the major tasks at its development was to achieve high levels of formalization and minimization of volume of a spadework at its application to calculation of nonlinear fluctuations in each concrete case.

The regular statement of this method and examples of its application will are made in following chapters of this book.

Chapter 1

MATHEMATICAL BASES
OF DIFFERENTIAL HARMONIC METHOD

1.1 Harmonic algebraization of the differential equations

Consideration of this question we shall begin with assertion, that the differential harmonic method begins with a method of harmonic balance.

The first action, which is necessary for use of method of harmonic balance, is transformation of the differential equations, whose periodic solution is sought, to the algebraic equations, whose unknown variables are amplitudes of Fourier series approaching the periodic solution. Such transformation of the differential equations in algebraic we shall name their harmonic algebraization .

For simplicity of a description of harmonic algebraization and algorithm of search of the periodic decision we shall consider at first an example of one nonlinear differential equation of the first order. It is obvious, that it can be only the non-autonomous equation because only the forced oscillation may be described by differential equation of the first order. We will consider harmonious algebraization of nonlinear systems of the differential equations, which may be autonomous or non-autonomous, and also algorithms of search of their periodic decisions only after that.

Thus, the differential equation is considered

$$\frac{dy}{dt} + z = e, \tag{1.1}$$

here

$$e = e\left[t\right] = e\left[t + T\right] \tag{1.2}$$

– the set external compelling force (indignation), being T-periodic function of time (here T - the period);

$$y = y\left[x\right]; \tag{1.3}$$

$$z = z\left[x, y\right] \tag{1.4}$$

- some nonlinear functional dependences.

As it has been already noted in introduction, square brackets in this book are applied in formulas only for record inside of them argument (arguments) of functional dependences (not at formulas square brackets are traditionally applied to links to references).

The periodic decision of the nonlinear scalar differential equation (1.1) is oscillation with the period T of coordinate x

$$x = x[t] = x[t+T], \tag{1.5}$$

which is argument of nonlinear dependences (1.3), (1.4). Variables y i z in equation (1.1) are periodic dependencies of time with the same period :

$$y = y[t] = y[t+T]; \tag{1.6}$$

$$z = z[t] = z[t+T] \ . \tag{1.7}$$

Let's carry out approximation of dependences (1.2), (1.5) - (1.7) by trigonometrical series

$$\begin{gathered} a[t] = A_0 + \sum_{\nu=1}^{n} (A_{c\nu} \cos \nu\omega t + A_{s\nu} \sin \nu\omega t); \\ a = x,\ y,\ z,\ e; \quad A = X, Y, Z, E, \end{gathered} \tag{1.8}$$

here

$$\omega = 2\pi / T \tag{1.9}$$

– the basic circular frequency (circular frequency of the first harmonic).

After substitution in (1.1) approximations of dependences of variables y, z and e as functions of time by series of a kind (1.8) and performance of operation of differentiation is received transcedental equation

$$\begin{gathered} -\omega Y_{c1} \sin \omega t + \omega Y_{s1} \cos \omega t - ... - \nu\omega Y_{c\nu} \sin \nu\omega t + \nu\omega Y_{s\nu} \cos \nu\omega t - ... \\ ... - n\omega Y_{cn} \sin n\omega t + n\omega Y_{sn} \cos n\omega t + Z_0 + Z_{c1} \cos \omega t + Z_{s1} \sin \omega t + ... \\ ... + Z_{c\nu} \cos \nu\omega t + Z_{s\nu} \sin \nu\omega t + ... + Z_{cn} \cos n\omega t + Z_{sn} \sin n\omega t = \\ E_0 + E_{c1} \cos \omega t + E_{s1} \sin \omega t + ... + E_{c\nu} \cos \nu\omega t + E_{s\nu} \sin \nu\omega t + ... \\ ... + E_{cn} \cos n\omega t + E_{sn} \sin n\omega t. \end{gathered} \tag{1.10}$$

The equation (1.10) is satisfied for all values t only in case if

$$
\begin{aligned}
&Z_0 = E_0;\\
&(\omega Y_{s1} + Z_{c1})\cos\omega t = E_{c1}\cos\omega t;\\
&(-\omega Y_{c1} + Z_{s1})\sin\omega t = E_{s1}\sin\omega t;\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&(\nu\omega Y_{s\nu} + Z_{c\nu})\cos\nu\omega t = E_{c\nu}\cos\nu\omega t;\\
&(-\nu\omega Y_{c\nu} + Z_{s\nu})\sin\nu\omega t = E_{s\nu}\sin\nu\omega t;\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&(n\omega Y_{sn} + Z_{cn})\cos n\omega t = E_{cn}\cos n\omega t;\\
&(-n\omega Y_{cn} + Z_{sn})\sin n\omega t = E_{sn}\sin n\omega t.
\end{aligned}
\qquad (1.11)
$$

From (1.11) we receive system of the nonlinear algebraic equations

$$
\begin{aligned}
&Z_0 = E_0;\\
&\omega Y_{s1} + Z_{c1} = E_{c1};\\
&-\omega Y_{c1} + Z_{s1} = E_{s1};\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&\nu\omega Y_{s\nu} + Z_{c\nu} = E_{c\nu};\\
&-\nu\omega Y_{c\nu} + Z_{s\nu} = E_{s\nu};\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&n\omega Y_{sn} + Z_{cn} = E_{cn};\\
&-n\omega Y_{cn} + Z_{sn} = E_{sn}
\end{aligned}
\qquad (1.12)
$$

With the purpose of compactness we shall write down system of the equations (1.12) in the matrix form. For this purpose using coefficients of trigonometrical series (1.8) (constant components and cosines amplitudes with sines amplitudes), we shall form matrixes-vectors of kind

$$
\vec{X}_\Gamma = \begin{Vmatrix} X_0 \\ X_{c1} \\ X_{s1} \\ \vdots \\ X_{cn} \\ X_{sn} \end{Vmatrix};\quad \vec{Y}_\Gamma = \begin{Vmatrix} Y_0 \\ Y_{c1} \\ Y_{s1} \\ \vdots \\ Y_{cn} \\ Y_{sn} \end{Vmatrix};\quad \vec{Z}_\Gamma = \begin{Vmatrix} Z_0 \\ Z_{c1} \\ Z_{s1} \\ \vdots \\ Z_{cn} \\ Z_{sn} \end{Vmatrix};\quad \vec{E}_\Gamma = \begin{Vmatrix} E_0 \\ E_{c1} \\ E_{s1} \\ \vdots \\ E_{cn} \\ E_{sn} \end{Vmatrix}. \qquad (1.13a, b, c, d)
$$

Let's name these matrixes-vectors as vectors of amplitudes. Their size (quantity of elements) is equal $N_g = 1 + 2n$.

Further matrixes-vectors can be designated and so:

$$\vec{A}_\Gamma = colon(A_0, A_{c1}, A_{s1}, \dots, A_{cn}, A_{sn}). \qquad (1.13e)$$

Often, even before the calculation of nonlinear oscillations, it can be concluded that in the coordinates of the analyzed system the amplitudes of harmonics of some orders are always zero. For example, if in electric circuits nonlinear elements have characteristics of odd type, then at sinusoidal supply of these circuits currents and voltages of branches do not have constant components and can have harmonics only of odd orders [41]. In three-phase electrical circuits connected in a "star" without a neutral conductor, the phase currents cannot have harmonics whose order is a multiple of 3, and so on. In such cases, at the already selected value n , the size of the amplitude vectors (the number of their elements) decreases due to the removal from consideration of those amplitudes that are known to be zero.

Amplitude vectors that do not have among their components harmonic amplitudes of some orders of magnitude smaller than , we will call *abbreviated amplitude vectors*.

If the coordinates of periodic processes contain only odd harmonics, then for this case the reduced vectors of amplitudes have the form

$$\vec{A}_\Gamma = colon(A_{c1}, A_{s1}, A_{c3}, A_{s3}, ..., A_{cn}, A_{sn}), \qquad (1.14)$$

here n is an odd number. The size of the amplitude vector (1.14) $N_g = 1 + n$.

Taking into account the notation (1.13) or (1.14), the system of algebraic equations (1.12) can be written as:

$$\omega D \vec{Y}_\Gamma + \vec{Z}_\Gamma = \vec{E}_\Gamma \ , \qquad (1.15)$$

here

$$D = \begin{Vmatrix} 0 & & & & & & & \\ & 0 & 1 & & & & & \\ & -1 & 0 & & & & & \\ & & & 0 & 2 & & & \\ & & & -2 & 0 & & & \\ & & & & & \ddots & & \\ & & & & & & 0 & n \\ & & & & & & -n & 0 \end{Vmatrix} \qquad (1.16)$$

- constant square matrix of order N_g . Hereinafter we will call it *the differentiation matrix*.

If in the equation of the form (1.15) the vectors of amplitudes are reduced, then the matrix D is also an abbreviated differentiation matrix. It is obtained from the matrix of the form (1.16) after removing from it those rows and columns that correspond to the removed components of the vectors of amplitudes.

If the time dependences of the system coordinates contain only odd harmonics, then for this case the abbreviated differentiation matrix has the form

$$D = \begin{Vmatrix} 0 & 1 & & & & & \\ -1 & 0 & & & & & \\ & & 0 & 3 & & & \\ & & -3 & 0 & & & \\ & & & & \ddots & & \\ & & & & & 0 & n \\ & & & & & -n & 0 \end{Vmatrix} \qquad (1.17)$$

The order of this matrix is $N_g = 1 + n$.

The algebraic vector equation (1.15) can be considered as a harmonic representation of the differential equation (1.1). The transition from (1.1) to (1.15) is a harmonic algebraization of the differential equation (1.1).

Note that when performing harmonic algebraization of differential equations with periodic solutions, it is not necessary to perform operations (1.10) - (1.12) each time. This transformation in order to obtain the already completed form of the algebraic vector equation of the form (1.14) can be formalized:

1) the variables of the differential equation (in equation (1.1) these are variables $y,\ z,\ e$) must be replaced by the corresponding amplitude vectors (here are the amplitude vectors $\vec{Y}_\Gamma$, $\vec{Z}_\Gamma$ and $\vec{E}_\Gamma$);

2) replace the differentiation operation by multiplying on the left by the circular frequency ω and matrix D.

Equation (1.15) explicitly includes amplitude vectors $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$ and implicitly the amplitude vector $\vec{X}_\Gamma$, which is the root of this equation. Amplitude vectors $\vec{Y}_\Gamma$, $\vec{Z}_\Gamma$ and $\vec{X}_\Gamma$ are interrelated. Let's follow this connection.

Assume that the value of the vector $\vec{X}_\Gamma$ is known. Then, taking into account (1.8), at $a = x$, is also known (approximately) function (1.5). Using it as an argument of nonlinear functions (1.3) and (1.4), we can obtain the approximation of T --periodic dependences (1.6) and (1.7). The latter can be decomposed into Fourier series of the form (1.8) at $a = y$ and $a = z$ with the determination of the coefficients of the series by known formulas

$$A_0 = \frac{1}{2\pi}\int_0^{2\pi} a_\eta d\eta; \qquad A_{cv} = \frac{1}{\pi}\int_0^{2\pi} a_\eta \cos v\eta \, d\eta; \qquad (1.18)$$
$$A_{sv} = \frac{1}{\pi}\int_0^{2\pi} a_\eta \sin v\eta \, d\eta,$$

here a_η - value of function $a = a[\eta]$ when the value of the angular coordinate $\eta = \omega t$.

Since the vectors of amplitudes $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$ are formed from these coefficients by formulas (1.13) or (1.14), if the values of these coefficients have become known, then the values of the vectors of amplitudes $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$ have also become known.

As you can see, for each given value of the vector $\vec{X}_\Gamma$ you can find the corresponding values of the vectors $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$. Therefore, there are such dependencies

$$\vec{Y}_\Gamma = \vec{Y}_\Gamma[\vec{X}_\Gamma]; \quad \vec{Z}_\Gamma = \vec{Z}_\Gamma[\vec{X}_\Gamma]. \qquad (1.19a, b)$$

Let's call these dependences *harmonic characteristics*. They can be considered as harmonic mappings of functions (1.3) and (1.4).

Harmonic characteristics (1.19) are nonlinear due to the nonlinearity of dependences (1.3) and (1.4).

The algorithm for calculating the harmonic characteristics (1.19) is described here schematically, it is considered in more detail below.

The procedure of harmonic algebraization is considered above on the example of the differential equation of the form (1.1), in which the sign of the derivative is not a coordinate x, the periodic dependence of time (1.5) of which is the solution of this equation, but a variable y, that is a function of the variable x. This notation of the differential equation is more general, it occurs, for example, in the analysis of electric circuits, when in the differential equations that describe them, there are derivatives of time not currents, but flux couplings, which are functions of these currents. If in the differential equation the intermediate variable is absent and under the sign of the derivative there is a direct variable, the periodic dependence of which is the sought solution, the differential equation has the form

$$\frac{dx}{dt} + z[x] = e[t], \qquad (1.20)$$

then its harmonic reflection is a finite equation of the form

$$\omega D \vec{X}_\Gamma + \vec{Z}_\Gamma = \vec{E}_\Gamma \ . \qquad (1.21)$$

1.2 Determining the periodic solution

After performing harmonic algebraization of the differential equation (1.1) or (1.20), the determination of their periodic solutions is reduced to the solution of nonlinear vector finite equations (1.15) or (1.21). Consider first the solution of equation (1.15).

Although the amplitude vector in equation (1.15) does not explicitly appear, it is the root of this equation. To find it, we use one of the most effective numerical methods for solving nonlinear algebraic (finite) equations - Newton's iterative method [44, 48]. It is characterized by a fairly high - quadratic - rate of convergence. However, to ensure the very convergence in its application, it is necessary to set, as is known, a "good" zero approximation, which is located within the so-called area of attraction of the root.

To obtain such a "good" approximation, we will use the method proposed in [64], called the h-characteristic method. According to the h-characteristic method, multiply in equation (1.15) the vector of forcing forces by a scalar parameter h and obtain a new equation

$$\omega D \vec{Y}_\Gamma + \vec{Z}_\Gamma = h \vec{E}_\Gamma . \qquad (1.22)$$

At h = 0 in (1.22) there is no forcing force, and then this vector equation has a trivial (ie - zero) solution $\vec{X}_\Gamma = 0$. At h = 1, equations (1.22) and (1.15) are identical. Dependence of the amplitude vector $\vec{X}_\Gamma$ on the parameter h

$$\vec{X}_\Gamma = \vec{X}_\Gamma [h] \qquad (1.23)$$

is a h-characteristic of equation (1.22). At h = 0 it passes through the zero solution, and at h = 1 - through the root of equation (1.15).

Dependence (1.23) can be obtained by integrating some vector differential equation for which this function is a solution. To obtain the following differential equation, we differentiate equation (1.22) by the parameter h :

$$\omega D \frac{d\vec{Y}_\Gamma}{d h} + \frac{d\vec{Z}_\Gamma}{d h} = \vec{E}_{\Gamma.} \qquad (1.24)$$

Here the derivatives of the amplitude vectors $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$ with the parameter h must be disclosed according to the rule of differentiation of complex functions, taking into account the existence of harmonic characteristics (1.19):

$$\omega D \frac{d\vec{Y}_\Gamma}{d\vec{X}_\Gamma}\frac{d\vec{X}_\Gamma}{d h} + \frac{d\vec{Z}_\Gamma}{d\vec{X}_\Gamma}\frac{d\vec{X}_\Gamma}{d h} = \vec{E}_{\Gamma.} \qquad (1.25)$$

Then, taking the notation

$$\frac{d\vec{Y}_\Gamma}{d\vec{X}_\Gamma} = S_{Y\Gamma}; \quad \frac{d\vec{Z}_\Gamma}{d\vec{X}_\Gamma} = S_{Z\Gamma} \ , \qquad (1.26a, b)$$

differential equation (1.25) is reduced to the form

$$(\omega D S_{Y\Gamma} + S_{Z\Gamma}) \frac{d\vec{X}_\Gamma}{d h} = \vec{E}_\Gamma \ . \qquad (1.27)$$

Here $S_{Y\Gamma}$ and $S_{Z\Gamma}$ are square matrices of the order N_g of differential parameters of harmonic characteristics (1.19). Let us call these matrices *matrices of differential harmonic parameters* (MDHP).

MDHP (1.26) relates the values of infinitesimal increments of the components of vectors $\vec{Y}_\Gamma$, $\vec{Z}_\Gamma$ and $\vec{X}_\Gamma$ ie, infinitesimal increments of amplitudes of all considered harmonics of

dependences (1.5) - (1.7). If functions (1.3) and (1.4) are nonlinear, then matrices (1.26) are variables and are functions of the vector $\vec{X}_\Gamma$ .. The MDHP calculation algorithm is outlined below.

To obtain the dependence (1.23) in tabular form, the differential equation (1.27) must be integrated by one of the numerical methods for the parameter h from h = 0 at zero initial conditions $\vec{X}_\Gamma = 0$ to h = 1. The value of the vector $\vec{X}_\Gamma$ obtained at h = 1 can be considered as an approximate solution of the finite equation (1.15) and as a "good" zero approximation to refine the solution by Newton's method.

The formula for refining the solution of equation (1.15) by Newton's method has the form [44, 48]

$$\vec{X}_{\Gamma(l+1)} = \vec{X}_{\Gamma(l)} - W_{(l)}^{-1}\,\vec{H}_{(l)}\,, \qquad (1.28)$$

here l - iteration number;

$$W_{(l)} = \omega\, D\, S_{Y\Gamma(l)} + S_{Z\Gamma(l)} \qquad (1.29)$$

- the value of the Jacobi matrix of the left part of equation (1.15) at $\vec{X}_\Gamma = \vec{X}_{\Gamma(l)}$;

$$\vec{H}_{(l)} = \omega\, D\, \vec{Y}_{\Gamma(l)} + \vec{Z}_{\Gamma(l)} - \vec{E}_\Gamma \qquad (1.30)$$

- the value of the residual vector of equation (1.15) at $\vec{X}_\Gamma = \vec{X}_{\Gamma(l)}$.

In order to refine the solution of equation (1.15) according to the iterative scheme (1.28) not to rotate the martix, but to solve a system of linear equations, this scheme can be written as:

$$\begin{aligned} &W_{(l)} \cdot \Delta\vec{X}_{\Gamma(l)} = \vec{H}_{(l)}\,; \\ &\vec{X}_{\Gamma(l+1)} = \vec{X}_{\Gamma(l)} - \Delta\vec{X}_{\Gamma(l)}. \end{aligned} \qquad (1.31)$$

In (1.31) first line is a system of linear equations with respect to the vector unknown $\Delta\vec{X}_{\Gamma(l)}$ - it is vector of amendments.

Iterations according to schemes (1.28) or (1.31) must be performed until the required accuracy of solution of equation (1.15) is reached.

If the differential equatiom whose periodic solution we are looking for has the form (1.20) and its harmonic reflection has the form (1.21), than the differential equation required to obtain h -characteristic has the form

$$(\omega\, D + S_{Z\Gamma})\,\frac{d\vec{X}_\Gamma}{d\,h} = \vec{E}_\Gamma \qquad (1.32)$$

And necessary for refining the solution by Newton's mrthod the residual vector and the Jacobi's matrix in l-th iteration have the form

$$\vec{H}_{(l)} = \omega D \vec{X}_{\Gamma(l)} + \vec{Z}_{\Gamma(l)} - \vec{E}_{\Gamma} \ ; \qquad (1.33)$$

$$W_{(l)} = \omega D + S_{Z\Gamma(l)} \ . \qquad (1.34)$$

As you can see, to obtain the root of equation (1.15) or (1.21) in the described way, it is necessary to calculate the values of the amplitude vectors $\vec{Y}_{\Gamma}$, $\vec{Z}_{\Gamma}$ and matrices $S_{Y\Gamma}$, $S_{Z\Gamma}$ for the value of the amplitude vector $\vec{X}_{\Gamma}$ when obtaining the zero approximation by calculating the h-characteristic and at each iteration when refining the solution by Newton's method.

1.3 Algorithm for calculating harmonic characteristics

In the previous section, the calculation of the values of the amplitude vectors $\vec{Y}_{\Gamma}$ and $\vec{Z}_{\Gamma}$ for the given value of the amplitude vector $\vec{X}_{\Gamma}$, ie the calculation of the harmonic characteristics (1.19), is described schematically. Let's look at it in more detail in order to obtain an algorithm suitable for computer implementation.

If the value of the amplitude vector $\vec{X}_{\Gamma}$ is given, then the value of the variable x for any value of the angular coordinate $\eta = \omega t$ is determine, taking into account (1.8), by the formula

$$x = X_0 + \sum_{\nu=1}^{n} (X_{c\nu} \cos \nu\eta + X_{s\nu} \sin \nu\eta). \qquad (1.35)$$

Using the nonlinear dependence (1.3), which connects the variables x and y, and the set of values of the variable x calculated by formula (1.35) for the set of values of the angle η from zero to 2π, we can define in the form of a table 2π-periodic dependence $y = y[\eta]$ as a function of angle η. Then, similarly, using the dependence (1.4), which connects the variables z, x and y, and the dependence just obtained in the form of a table dependence $y = y[\eta]$ we can determine the dependence $z = z[\eta]$, in tabular form, which is also a 2π-periodic function of the angle η. To find the values of the components of the vectors of amplitudes $\vec{Y}_{\Gamma}$ and $\vec{Z}_{\Gamma}$, it is necessary to obtain the numerical dependences $y = y[\eta]$ and $z = z[\eta]$ and decompose them into Fourier series by the formulas of the form (1.18):

$$Y_0 = \frac{1}{2\pi}\int_0^{2\pi} y[\eta]\,d\eta; \qquad Y_{c\nu} = \frac{1}{\pi}\int_0^{2\pi} y[\eta]\cos\nu\eta\,d\eta;$$
$$Y_{s\nu} = \frac{1}{\pi}\int_0^{2\pi} y[\eta]\sin\nu\eta\,d\eta; \qquad \text{(1.36а)}$$

$$Z_0 = \frac{1}{2\pi}\int_0^{2\pi} z[\eta]\,d\eta; \qquad Z_{c\nu} = \frac{1}{\pi}\int_0^{2\pi} z[\eta]\cos\nu\eta\,d\eta;$$
$$Z_{s\nu} = \frac{1}{\pi}\int_0^{2\pi} z[\eta]\sin\nu\eta\,d\eta; \quad \nu = 1, \ldots n. \qquad \text{(1.36б)}$$

The integrals in formulas (1.36) will be calculated by one of the known numerical methods. To do this, we apply a one-dimensional grid of equidistant m nodes for the period $\omega T = 2\pi$ (see Fig. 1.1), the number of which must be sufficient to ensure the required accuracy of calculating the values of the integrals.

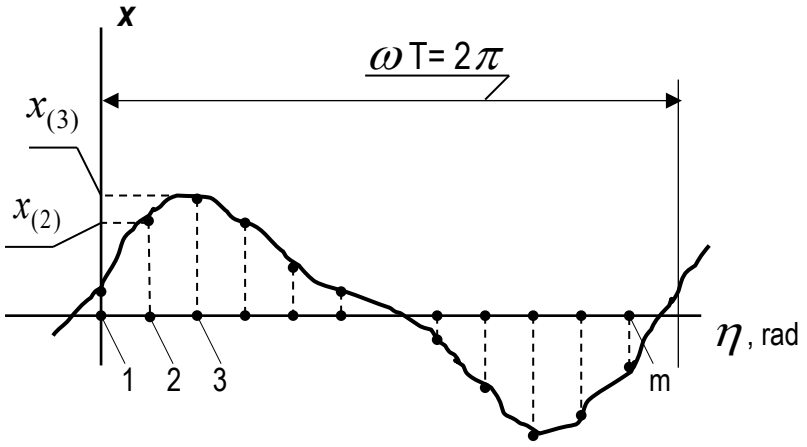


Fig. 1.1. Dependence of the variable x on the angular coordinate η on the period

If the dependences $x = x[\eta]$, $y = y[\eta]$ and $z = z[\eta]$ contain only odd harmonics and the amplitude vectors have the form (1.14), then in integrals (1.36) the upper limits of integration must be changed from 2π to π, before the integrals put a coefficient 2 and impose a one-dimensional grid of m nodes on the half-period.

From the values of the functions $x = x[\eta]$, $y = y[\eta]$ and $z = z[\eta]$ in equidistant m nodes of the grid in the period (half-period, if the searched functions contain only odd harmonics) we form column vectors

$$\vec{x}_в = colon(x_{(1)},\ x_{(2)},\ \ldots, x_{(m)}); \qquad \text{(1.37а)}$$
$$\vec{y}_в = colon(y_{(1)},\ y_{(2)},\ \ldots, y_{(m)}); \qquad \text{(1.37б)}$$
$$\vec{z}_в = colon(z_{(1)},\ z_{(2)},\ \ldots, z_{(m)}), \qquad \text{(1.37в)}$$

which we call *nodal* vectors.

The value of the node vector $\vec{x}_в$ (ie - the set of values of the variable x in the nodes of the period) can be obtained by the value of the amplitude vector $\vec{X}_Г$ by performing a matrix operation

$$\vec{x}_в = F\,\vec{X}_Г\,, \qquad (1.38)$$

Here

$$F = \begin{Vmatrix} 1 & \cos\eta_{(1)} & \sin\eta_{(1)} & \cos 2\eta_{(1)} & \sin 2\eta_{(1)} & \cdots & \cos n\eta_{(1)} & \sin n\eta_{(1)} \\ 1 & \cos\eta_{(2)} & \sin\eta_{(2)} & \cos 2\eta_{(2)} & \sin 2\eta_{(2)} & \cdots & \cos n\eta_{(2)} & \sin n\eta_{(2)} \\ \vdots & & & & & & & \\ 1 & \cos\eta_{(j)} & \sin\eta_{(j)} & \cos 2\eta_{(j)} & \sin 2\eta_{(j)} & \cdots & \cos n\eta_{(j)} & \sin n\eta_{(j)} \\ \vdots & & & & & \cdots & & \\ 1 & \cos\eta_{(m)} & \sin\eta_{(m)} & \cos 2\eta_{(m)} & \sin 2\eta_{(m)} & \cdots & \cos n\eta_{(m)} & \sin n\eta_{(m)} \end{Vmatrix} \qquad (1.39)$$

- a matrix with dimensions $m \times N_g$, the elements of which are units in the first column and in the other columns - numerical values of trigonometric functions $\cos\nu\eta$, $\sin\nu\eta$ $(\nu = 1,...,n)$ in the grid m nodes for the period.

Having calculated by formula (1.38) the value of the node vector $\vec{x}_в$, the structure of which is given by formula (1.37a), determine the values of the node vectors $\vec{y}_в$ and $\vec{z}_в$ whose structures are given by formulas (1.37b) and (1.37c), in the following order:

1) by the values of the n1st, 2nd,…, m-th component of the vector $\vec{x}_в$ and by the dependence (1.3) we determine the values of the 1st, 2nd,…, m-th component of the vector $\vec{y}_в$;

2) by the values of the 1st, 2nd,…, m-th component of the vectors $\vec{x}_в$ and $\vec{y}_в$ and depending on (1.4) we determine the values of the 1st, 2nd,…, m-th component of the vector $\vec{z}_в$.

Calculating the values of vectors $\vec{y}_в$ and $\vec{z}_в$ the value of the vector $\vec{x}_в$ is the implementation of the so-called "instantaneous" model of the process in the period (half): the instantaneous values of the independent variable x in the nodes of the period (half) determine the instantaneous values in the same nodes of dependent variables y and z .

According to the values of the nodal vectors $\vec{y}_в$ and $\vec{z}_в$ found by the described method, it is possible to calculate the values of the corresponding vectors of amplitudes $\vec{Y}_Г$ and $\vec{Z}_Г$. Each of the components of these vectors should be determined by formulas (1.36), using one of the known numerical methods for calculating the values of definite integrals. So, for the components of the vector $\vec{Y}_Г$ we write

$$Y_0 = \frac{1}{2\pi}\int_0^{2\pi} y\,d\eta \approx \frac{1}{2\pi}\frac{2\pi}{m}(\xi_{(1)}y_{(1)} + \xi_{(2)}y_{(2)} + \dots$$

$$\dots + \xi_{(m)}y_{(m)}) == \frac{1}{m}(\xi_{(1)}y_{(1)} + \xi_{(2)}y_{(2)} + \dots + \xi_{(m)}y_{(m)});$$

$$Y_{c\nu} = \frac{1}{\pi}\int_0^{2\pi} y\cos\nu\eta\,d\eta \approx \frac{1}{\pi}\frac{2\pi}{m}(\xi_{(1)}y_{(1)}\cos\nu\eta_{(1)} +$$

$$+ \xi_{(2)}y_{(2)}\cos\nu\eta_{(2)} + \dots + \xi_{(m)}y_{(m)}\cos\nu\eta_{(m)}) =$$

$$= \frac{2}{m}(\xi_1 y_{(1)}\cos\nu\eta_{(1)} + \xi_{(2)}y_{(2)}\cos\nu\eta_{(2)} + \dots$$

$$\dots + \xi_{(m)}y_{(m)}\cos\nu\eta_{(m)});$$

$$Y_{s\nu} = \frac{1}{\pi}\int_0^{2\pi} y\sin\nu\eta\,d\eta \approx \frac{1}{\pi}\frac{2\pi}{m}(\xi_{(1)}y_{(1)}\sin\nu\eta_{(1)} +$$

$$+ \xi_{(2)}y_{(2)}\sin\nu\eta_{(2)} + \dots + \xi_{(m)}y_{(m)}\sin\nu\eta_{(m)}) =$$

$$= \frac{2}{m}(\xi_{(1)}y_{(1)}\sin\nu\eta_{(1)} + \xi_{(2)}y_{(2)}\sin\nu\eta_{(2)} + \dots \qquad (1.40)$$

$$\dots + \xi_{(m)}y_{(m)}\sin\nu\eta_{(m)}),$$

here $\eta_{(1)}, \eta_{(2)}, \dots, \eta_{(m)}$ - the value of the angle η in the $\eta_{(1)}, \eta_{(2)}, \dots, \eta_{(m)}$ nodes to pass the period (southern period); $\xi_{(1)}, \xi_{(2)}, \dots, \xi_{(m)}$ - weights of the selected quadrature formula (formula for calculating the values of a definite integral). When using the quadrature formula of rectangles or trapezoids $\xi_{(1)} = \xi_{(2)} = \dots = \xi_{(m)} = 1$; if we apply Simpson's quadratic formula, then the number of nodes m for the periods (southern period) must be even and $\xi_{(1)} = 2/3,\ \xi_{(2)} = 4/3,\ \dots,\ \xi_{(m-1)} = 2/3,\ \xi_{(m)} = 4/3$.

Calculation by formulas of the form (1.40) values of all components of vectors $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$ can be carried out by performing the following matrix operations:

$$\vec{Y}_\Gamma = G\,\vec{y}_в;\quad \vec{Z}_\Gamma = G\,\vec{z}_в, \qquad (1.41\ а, б)$$

here

$$G = \frac{2}{m}\,\Theta\,F^T\xi_\eta \qquad (1.42)$$

– matrix with dimensions $N_g \times m$;

$$\Theta = diag\,(\frac{1}{2},\ 1,\ 1, \dots, 1) \qquad (1.43)$$

– diagonal matrix with size N_g ;

F^T – a matrix transposed with respect to the matrix (1.39);

ξ_η – diagonal order m matrix, the elements of the diagonal of which are the weights of the selected quadrature formula.

The matrices F and G we will call *matrices of harmonic transformations*, in particular G - the matrix of direct harmonic transformation (from values in nodes of the period or half-period to the values of harmonic amplitudes) and F - the matrix of inverse harmonic transformation (from values of harmonic amplitudes to values in nodes). When solving each specific problem (project) for the given values n and m values of these matrices should be calculated only once and kept unchanged until the end of the calculations at these values n and m. Expressions (1.39) and (1.42) for matrices of harmonic transformations correspond to random ones, when the amplitudes of harmonics have constant components and all harmonics up to n-th including. If abbreviated amplitude vectors are used in the calculations, then instead of matrices F and G in formulas (1.39) and (1.42) should be the reduced matrices of harmonic transformations $F_{c\kappa}$ and $G_{c\kappa}$.

The matrix $F_{c\kappa}$ can be obtained from the matrix F by removing from it those columns that correspond to the components of the amplitude vector, which are removed in the formation of a shortened amplitude vector. Thus, if the abbreviated amplitude vector has the form (1.14), ie has in its composition harmonics of only odd orders, then the matrix $F_{c\kappa}$ takes the form

$$F_{c\kappa} = \begin{Vmatrix} \cos\eta_{(1)} & \sin\eta_{(1)} & \cos 3\eta_{(1)} & \sin 3\eta_{(1)} & \cdots & \cos n\eta_{(1)} & \sin n\eta_{(1)} \\ \cos\eta_{(2)} & \sin\eta_{(2)} & \cos 3\eta_{(2)} & \sin 3\eta_{(2)} & \cdots & \cos n\eta_{(2)} & \sin n\eta_{(2)} \\ \vdots & & & & & & \\ \cos\eta_{(j)} & \sin\eta_{(j)} & \cos 3\eta_{(j)} & \sin 3\eta_{(j)} & \cdots & \cos n\eta_{(j)} & \sin n\eta_{(j)} \\ \vdots & & & & & & \\ \cos\eta_{(m)} & \sin\eta_{(m)} & \cos 3\eta_{(m)} & \sin 3\eta_{(m)} & \cdots & \cos n\eta_{(m)} & \sin n\eta_{(m)} \end{Vmatrix}. \quad (1.44)$$

When calculating the value of this matrix, keep in mind that the angular coordinate of the m - th node $\eta_{(m)} = \pi(m-1)/m$, ie the grid of m nodes is plotted in half, and that the number n is odd.

A matrix $G_{c\kappa}$ can be obtained by formula (1.42) if instead of a matrix F^T we substitute a matrix $F_{c\kappa}^T$ and instead of a matrix Θ - a matrix $\Theta_{c\kappa}$ formed from a matrix Θ by removing rows and columns with element numbers that are removed from the amplitude vector to form a shortened amplitude vector. In the case where the reduced amplitude vectors have harmonic amplitudes of only odd orders, the matrix Θ is a unit matrix.

The algorithm described above for calculating the values of the amplitude vectors $\vec{Y}_\Gamma$ and $\vec{Z}_\Gamma$ for a given value of the amplitude vector $\vec{X}_\Gamma$, ie the calculation of one "point" of the

harmonic characteristic (1.19), is called algorithm 1.1. Its operations can be described by the following sequence of directives:

a) for a given value of the vector of amplitudes $\vec{X}_{\Gamma}$ by formula (1.38) to determine the value of the nodal vector $\vec{x}_{в}$;

b) by the values of the components of the vector $\vec{x}_{в}$ and formulas (1.3) and (1.4), which can be specified both analytically and in tabular form, calculate the values of all components of the node vectors $\vec{y}_{в}$ and $\vec{z}_{в}$ (calculation of the values of the nodal vectors and the value of the nodal vector $\vec{y}_{в}$ and $\vec{z}_{в}$ - is the implementation of the instantaneous period (or half-period) model of the process);

c) according to formulas (1.41) and the values of the vectors calculated $\vec{y}_{в}$ and $\vec{z}_{в}$ according to item “b” of this algorithm and calculate the values of the vectors of amplitudes $\vec{Y}_{\Gamma}$ andi $\vec{Z}_{\Gamma}$.

1.4 Algorithm for calculating matrix values of differential harmonic parameters

First, we derive formulas for calculating the values of the differential parameters of harmonic characteristics (1.19) - matrices $S_{Y\Gamma}$ and $S_{Z\Gamma}$. According to formulas (1.26), (1.38) and (1.41) we have

$$S_{Y\Gamma}=\frac{d\vec{Y}_{\Gamma}}{d\vec{X}_{\Gamma}}=\frac{d\vec{Y}_{\Gamma}}{d\vec{y}_{в}}\frac{d\vec{y}_{в}}{d\vec{x}_{в}}\frac{d\vec{x}_{в}}{d\vec{X}_{\Gamma}}=G\,S_{ув}\,F;$$
$$S_{Z\Gamma}=\frac{d\vec{Z}_{\Gamma}}{d\vec{X}_{\Gamma}}=\frac{d\vec{Z}_{\Gamma}}{d\vec{z}_{в}}\frac{d\vec{z}_{в}}{d\vec{x}_{в}}\frac{d\vec{x}_{в}}{d\vec{X}_{\Gamma}}=G\,S_{zв}\,F, \qquad (1.45\ а,б)$$

here

$$S_{ув}=\frac{d\,\vec{y}_{в}}{d\vec{x}_{в}};\quad S_{zв}=\frac{d\,\vec{z}_{в}}{d\vec{x}_{в}} \qquad (1.46\ а,б)$$

- square order m atrices. They are diagonal because in nodal vectors $\vec{y}_{в}$, $\vec{z}_{в}$ and $\vec{x}_{в}$ interconnected are only their components of the same name (ie components with the same indices, for the same values of the angular coordinate in the period or half-period). Diagonal elements of matrices (1.46) are the values of complete derivatives of functions (1.3) and (1.4) on the variable x in the nodes of the grid on the period (half-period).

In the general case, when a variable z is a function of two arguments - x and y, we have

$$S_{zв}=\frac{d\vec{z}_{в}}{d\vec{x}_{в}}=\frac{\partial\vec{z}_{в}}{\partial\vec{y}_{в}}\frac{d\vec{y}_{в}}{d\vec{x}_{в}}+\frac{\partial\vec{z}_{в}}{\partial\vec{x}_{в}}=S_{zув}\,S_{ув}+S_{zхв}, \qquad (1.47)$$

here

$$S_{zyв} = \frac{\partial \vec{z}_в}{\partial \vec{y}_в}; \quad S_{zxв} = \frac{\partial \vec{z}_в}{\partial \vec{x}_в} \qquad (1.48\ а,б)$$

- diagonal matrices of order m, their elements are equal to the values in the nodes of the grid of partial derivatives of the function z according to its arguments y and x.

Therefore, to calculate by formula (1.45) the values of MDGP $S_{YГ}$ and $S_{ZГ}$ you must first calculate the values of the matrices

$$S_{yв} = diag\left(\frac{dy}{dx}\Big|_{(1)}, \frac{dy}{dx}\Big|_{(2)}, ..., \frac{dy}{dx}\Big|_{(m)}\right); \qquad (1.49a)$$

$$S_{zyв} = diag\left(\frac{\partial z}{\partial y}\Big|_{(1)}, \frac{\partial z}{\partial y}\Big|_{(2)}, ..., \frac{\partial z}{\partial y}\Big|_{(m)}\right); \qquad (1.49b)$$

$$S_{zxв} = diag\left(\frac{\partial z}{\partial x}\Big|_{(1)}, \frac{\partial z}{\partial x}\Big|_{(2)}, ..., \frac{\partial z}{\partial x}\Big|_{(m)}\right). \qquad (1.49c)$$

The elements of these matrices are calculated using dependencies

$$\frac{dy}{dx} = \frac{dy}{dx}[x]; \quad \frac{\partial z}{\partial y} = \frac{\partial z}{\partial y}[x, y]; \quad \frac{\partial z}{\partial x} = \frac{\partial z}{\partial x}[x, y], \qquad (1.50)$$

which are obtained by differentiating (analytical or numerical) dependences (1.3) and (1.4).

Calculation of values of diagonal elements of matrices $S_{yв}$, $S_{zyв}$ and $S_{zxв}$ on values of elements of a vector $\vec{x}_в$, together with calculation of values of nodal vectors and, is the implementation of the instantaneous period (half-period) model of the process.

Matrices $S_{yв}$, $S_{zв}$, $S_{zyв}$ and $S_{zxв}$ are called matrices of differential parameters of characteristics (1.3) and (1.4) in grid nodes or matrices of node differential parameters (MNDP).

The algorithm for calculating the value of MNDP by the method described above (call it algorithm 1.2) can be expressed by the following sequence of directives:

a) for a given value of the amplitude vector $\vec{X}_Г$ and formula (1.38) to calculate the value of the nodal vector $\vec{x}_в$;

b) by the value of the vector $\vec{x}_в$ and the dependence (1.3) to calculate the value of the vector $\vec{y}_в$ (instantaneous on the period (half-period) model of the process);

c) according to the values of vectors $\vec{x}_в$, $\vec{y}_в$ and formulas or algorithms that approximate the dependences (1.50), calculate the value of MNDP $S_{yв}$, $S_{zyв}$, and $S_{zxв}$ (the instantaneous (half-period) process model);

d) according to formulas (1.47) to calculate the value of MNDP S_{z6};

e) by formulas (1.45) calculate the value of MNDP S_{YT} and S_{ZT}.

Consider the second method of calculating the values of MNDP, more economical in its numerical implementation - it requires fewer arithmetic operations.

The matrix S_{YT}, which is a derivative of the vector function $\vec{Y}_T$ by the vector argument $\vec{X}_T$, in the expanded form of the record has the form

$$S_{YT} = \left\| \begin{array}{ccccccc} \frac{\partial Y_0}{\partial X_0} & \frac{\partial Y_0}{\partial X_{c1}} & \frac{\partial Y_0}{\partial X_{s1}} & \cdots & \frac{\partial Y_0}{\partial X_{c\mu}} & \cdots & \frac{\partial Y_0}{\partial X_{sn}} \\ \frac{\partial Y_{c1}}{\partial X_0} & \frac{\partial Y_{c1}}{\partial X_{c1}} & \frac{\partial Y_{c1}}{\partial X_{s1}} & \cdots & \frac{\partial Y_{c1}}{\partial X_{c\mu}} & \cdots & \frac{\partial Y_{c1}}{\partial X_{sn}} \\ \frac{\partial Y_{s1}}{\partial X_0} & \frac{\partial Y_{s1}}{\partial X_{c1}} & \frac{\partial Y_{s1}}{\partial X_{s1}} & \cdots & \frac{\partial Y_{s1}}{\partial X_{c\mu}} & \cdots & \frac{\partial Y_{s1}}{\partial X_{sn}} \\ \vdots & \vdots & \vdots & & \vdots & & \vdots \\ \frac{\partial Y_{s\nu}}{\partial X_0} & \frac{\partial Y_{s\nu}}{\partial X_{c1}} & \frac{\partial Y_{s\nu}}{\partial X_{s1}} & \cdots & \frac{\partial Y_{s\nu}}{\partial X_{c\mu}} & \cdots & \frac{\partial Y_{s\nu}}{\partial X_{sn}} \\ \vdots & \vdots & \vdots & & \vdots & & \vdots \\ \frac{\partial Y_{sn}}{\partial X_0} & \frac{\partial Y_{sn}}{\partial X_{c1}} & \frac{\partial Y_{sn}}{\partial X_{s1}} & \cdots & \frac{\partial Y_{sn}}{\partial X_{c\mu}} & \cdots & \frac{\partial Y_{sn}}{\partial X_{sn}} \end{array} \right\|. \quad (1.51)$$

Let's derive an expression for one of the elements of the matrix (1.51) - a partial derivative $\partial Y_{s\nu} / \partial X_{c\mu}$. Taking into account (1.36a) and (1.35) we obtain

$$\begin{aligned} \frac{\partial \mathrm{Y}_{\mathrm{s}\nu}}{\partial \mathrm{X}_{\mathrm{c}\mu}} &= \frac{1}{\pi}\int_0^{2\pi} \frac{\partial y}{\partial X_{c\mu}} \sin\nu\eta d\eta = \frac{1}{\pi}\int_0^{2\pi} \frac{dy}{dx}\frac{\partial x}{\partial X_{c\mu}} \sin\nu\eta d\eta = \\ &= \frac{1}{\pi}\int_0^{2\pi} \frac{dy}{dx} \cos\mu\eta \sin\nu\eta\, d\eta = \frac{1}{\pi}\int_0^{2\pi} \chi \cos\mu\eta \sin\nu\eta\, d\eta, \end{aligned} \quad (1.52)$$

here

$$\chi = \frac{dy}{dx} = \chi[\eta] \quad (1.53)$$

- dependence on the angular coordinate η of the derivative dy/dx.

Expressions for other elements are displayed similarly. Here are them without output (distrustful reader is recommended to display them yourself):

$$\frac{\partial Y_0}{\partial X_0}=\frac{1}{2}\int_0^{2\pi}\chi\, d\eta; \qquad \frac{\partial Y_0}{\partial X_{c1}}=\frac{1}{2}\int_0^{2\pi}\chi\cos\eta\, d\eta;$$

$$\frac{\partial Y_0}{\partial X_{s1}}=\frac{1}{2}\int_0^{2\pi}\chi\,\sin\eta\, d\eta; \qquad \frac{\partial Y_0}{\partial X_{sn}}=\int_0^{2\pi}\chi\sin n\eta\, d\eta;$$

$$\frac{\partial Y_{c1}}{\partial X_0}=\int_0^{2\pi}\chi\cos\eta\, d\eta; \qquad \frac{\partial Y_{c1}}{\partial X_{c1}}=\int_0^{2\pi}\chi\cos^2\eta\, d\eta;$$

$$\frac{\partial Y_{c1}}{\partial X_{s1}}=\int_0^{2\pi}\chi\cos\eta\sin\eta\, d\eta; \quad \frac{\partial Y_{c1}}{\partial X_{sn}}=\int_0^{2\pi}\chi\cos\eta\sin n\eta\, d\eta;$$

$$\frac{\partial Y_{s1}}{\partial X_0}=\int_0^{2\pi}\chi\sin\eta\, d\eta; \qquad \frac{\partial Y_{s1}}{\partial X_{c1}}=\int_0^{2\pi}\chi\sin\eta\cos\eta\, d\eta;$$

$$\frac{\partial Y_{s1}}{\partial X_{s1}}=\int_0^{2\pi}\chi\sin^2\eta\, d\eta; \qquad \frac{\partial Y_{s1}}{\partial X_{sn}}=\int_0^{2\pi}\chi\sin\eta\sin n\eta\, d\eta;$$

$$\frac{\partial Y_{c\nu}}{\partial X_0}=\int_0^{2\pi}\chi\cos\nu\eta\, d\eta; \qquad \frac{\partial Y_{c\nu}}{\partial X_{c1}}=\int_0^{2\pi}\chi\cos\nu\eta\cos\eta\, d\eta;$$

$$\frac{\partial Y_{c\nu}}{\partial X_{s1}}=\int_0^{2\pi}\chi\cos\nu\eta\sin\eta\, d\eta; \quad \frac{\partial Y_{c\nu}}{\partial X_{sn}}=\int_0^{2\pi}\chi\cos\nu\eta\sin n\eta;$$

$$\frac{\partial Y_{sn}}{\partial X_0}=\int_0^{2\pi}\chi\sin n\eta\, d\eta; \qquad \frac{\partial Y_{sn}}{\partial X_{c1}}=\int_0^{2\pi}\chi\sin n\cos\eta\, d\eta;$$

$$\frac{\partial Y_{sn}}{\partial X_{s1}}=\int_0^{2\pi}\chi\sin n\eta\sin\eta\, d\eta; \quad \frac{\partial Y_{sn}}{\partial X_{sn}}=\int_0^{2\pi}\chi\sin^2 n\eta\, d\eta. \qquad (1.54)$$

Let's approximate the dependence $\chi=\chi[\eta]$ by the Fourier series:

$$\chi[\eta]=R+\sum_{\nu=1}^{2n}(P_\nu\cos\nu\eta+Q_\nu\sin\nu\eta). \qquad (1.55)$$

After substituting (1.55) into formulas (1.54), performing trigonometric transformations and integrating using tabular integrals, expressions (1.54) are determined by the coefficients of the series (1.55), and the matrix (1.51) takes the form:

$$S_{YI}=\frac{1}{2}\times \qquad (1.56)$$

$2R$	P_1	Q_1	..	P_ν	Q_ν	..	P_n	Q_n
$2P_1$	$2R+P_2$	Q_2	..	$P_{\nu-1}+P_{\nu+1}$	$Q_{\nu-1}+Q_{\nu+1}$	..	$P_{n-1}+P_{n+1}$	$Q_{n-1}+Q_{n+1}$
$2Q_1$	Q_2	$2R-P_2$	..	$-Q_{\nu-1}+Q_{\nu+1}$	$P_{\nu-1}-P_{\nu+1}$	..	$-Q_{n-1}+Q_{n+1}$	$P_{n-1}-P_{n+1}$
...	...	...	..	...	...	..	...	...
$2P_\nu$	$P_{\nu-1}+P_{\nu+1}$	$-Q_{\nu-1}+Q_{\nu+1}$	..	$2R+P_{2\nu}$	$Q_{2\nu}$	..	$P_{n-\nu}+P_{n+\nu}$	$Q_{n-\nu}+Q_{n+\nu}$
$2Q_\nu$	$Q_{\nu-1}+Q_{\nu+1}$	$P_{\nu-1}-P_{\nu+1}$	..	$Q_{2\nu}$	$2R-P_{2\nu}$	..	$-Q_{n-\nu}+Q_{n+\nu}$	$P_{n-\nu}-P_{n+\nu}$
...	...	...	..	...	...	..	...	...
$2P_n$	$P_{n-1}+P_{n+1}$	$-Q_{n-1}+Q_{n+1}$	..	$P_{n-\nu}+P_{n+\nu}$	$-Q_{n-\nu}+Q_{n+\nu}$	..	$2R+P_{2n}$	Q_{2n}
$2Q_n$	$Q_{n-1}+Q_{n+1}$	$P_{n-1}-P_{n+1}$	..	$Q_{n-\nu}+Q_{n+\nu}$	$P_{n-\nu}-P_{n+\nu}$	..	Q_{2n}	$2R-P_{2n}$

All coefficients of the series (1.55) to the $2n$-th harmonic can be calculated, by analogy with formulas (1.41), by performing such a matrix operation

$$\vec{V}_{YT} = G_{2n}\,\vec{v}_{ув}\,, \tag{1.57}$$

here

$$\vec{V}_{YT} = colon(R, P_1, Q_1, ..., P_\nu, Q_\nu, ..., P_{2n}, Q_{2n}) \tag{1.58}$$

- vector of amplitudes of dimension $1+4n$, formed from the coefficients of the series (1.55);

$$\vec{v}_{ув} = colon\left(\frac{dy}{dx}\Big|_{(1)}, \frac{dy}{dx}\Big|_{(2)}, ..., \frac{dy}{dx}\Big|_{(m)}\right) \tag{1.59}$$

- vector-column, the components of which are diagonal elements of the matrix (1.49 a);

G_{2n} - a matrix of direct harmonic transformation of the form (1.42), but which has $N_g = 1+2n$ not but $N_{g1} = 1+4n$ lines.

Formulas for calculating the values of the matrix S_{ZT} are obtained similarly, only when calculating the values of the vector $\vec{V}_{ZT}$ of the form (1.58) by a formula similar to formula (1.57), the vector $\vec{v}_{zв}$ is formed by the rule

$$\vec{v}_{zв} = colon\left(\frac{\partial z}{\partial y}\Big|_{(1)} \cdot \frac{dy}{dx}\Big|_{(1)} + \frac{\partial z}{\partial x}\Big|_{(1)}, \; ... \; , \frac{\partial z}{\partial y}\Big|_{(m)} \cdot \frac{dy}{dx}\Big|_{(m)} + \frac{\partial z}{\partial x}\Big|_{(m)}\right). \tag{1.60}$$

The algorithm for calculating the value of MNDP in the manner described above (algorithm 1.3) can be expressed by the following sequence of directives:

a) according to the given value of the amplitude vector $\vec{X}_T$ and by the formula (1.38) calculate the value of the nodal vector $\vec{x}_в$;

b) by the value of the vector $\vec{x}_в$ and using the dependence (1.3) calculate the value of the nodal vector $\vec{y}_в$ (instantaneous model of the process);

c) the values of vectors (1.59) and (1.60) are calculated from the values of vectors $\vec{x}_в$, $\vec{y}_в$ and expressions for derivatives (1.50) (instant process model);

d) by the formula of the form (1.57) calculate the values of the vectors $\vec{V}_{YT}$ and $\vec{V}_{ZT}$;

e) by the components of the vectors $\vec{V}_{YT}$ and $\vec{V}_{ZT}$ and formula (1.56) calculate the value of all elements of the matrices S_{YT} and S_{ZT}.

Algorithm 1.3 calculates the value of MNDP compared to algorithm 1.2 is implemented by a larger computer program, but it is more economical in terms of machine time. Indeed, the number of multiplications when performing matrix operations by formula (1.45) is proportional to $(1+2n)^2$, while when determining by formula (1.57) the coefficients of the series (1.55) in the form of vectors (1.58) the number of multiplications is proportional to $1+4n$.

If only abbreviated amplitude vectors appear in the problem, then the MNDP that correspond to them are also abbreviated. The abbreviated MNDP can be obtained from (1.51) or (1.56) by extracting in it those rows and columns that correspond to the extracted amplitudes of harmonics in the formation of abbreviated amplitude vectors.

For the case when only odd harmonics are present in the periodic process and the shortened amplitude vectors have the form (1.14), in (1.56) it is necessary to remove rows and columns corresponding to the constant component and harmonics of even orders. Then we see that the MNDP elements are formed from the amplitudes of only pair harmonics of the Fourier series, which represents the dependence (1.53) of the derivative on the angular coordinate. In particular, in the case when only harmonics with numbers 1, 3 and 5 are taken into account, the MNDP matrix takes the form

$$S_{YT} = \frac{d\vec{Y}_{Tc}}{d\vec{X}_{Tc}} = \frac{1}{2} \text{ x}$$

$2R + P_2$	Q_2	P_2+P_4	Q_2+Q_4	P_4+P_6	Q_4+Q_6
Q_2	$2R - P_2$	$-Q_2+Q_4$	P_2-P_4	$-Q_4+Q_6$	P_4-P_6
P_2+P_4	$-Q_2+Q_4$	$2R + P_6$	Q_6	P_2+P_8	Q_2+Q_8
Q_2+Q_4	P_2-P_4	Q_6	$2R - P_6$	$-Q_2+Q_8$	P_2-P_8
P_4+P_6	$-Q_4+Q_6$	P_2+P_8	$-Q_2+Q_8$	$2R + P_{10}$	Q_{10}
Q_4+Q_6	P_4-P_6	Q_2+Q_8	P_2-P_8	Q_{10}	$2R - P_{10}$

(1.56a)

Note some properties of MDGP.

An important property of MDGP is its symmetry. Yes, if the matrix is divided into four blocks (in formula (1.56) it is done with double lines)

$$S_{Y\Gamma} = \frac{1}{2} \cdot \left\| \begin{matrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{matrix} \right\| , \qquad (1.61)$$

that S_{22} is, a square symmetric matrix and $S_{21} = S_{12}^T$. This property of symmetry should be taken into account when calculating the matrix $S_{Y\Gamma}$: calculate only the elements of the matrix located on its diagonal and above (or below) the diagonal, and other elements - by simply assigning the value of the corresponding element from the other half of the matrix.

The symmetry of the matrix S_{22} is a consequence of the symmetry of the interrelationships of infinitesimal increments of harmonics of different orders of dependence (1.7) and (1.5). The fact that the elements $\partial Y_{s\mu} / \partial X_{c\nu}$ and $\partial Y_{c\nu} / \partial X_{s\mu}$ of matrices (1.51) are the same means that with the same small increments of amplitudes $X_{c\nu}$ and $X_{s\mu}$ of dependence $x = x[\eta]$ small increments of amplitudes $Y_{s\mu}$ and $Y_{c\nu}$ of dependence $y = y[\eta]$ will also be the same.

In the general case, all elements of MDGP are nonzero, and then this indicates a complete relationship between harmonics of all orders of function and argument. In a special case, when the dependences (1.3) and (1.4) are linear functions, then in the matrices $S_{Y\Gamma}$ andi $S_{Z\Gamma}$ nonzero are only diagonal elements, and they are all the same: the linear dependence of the variables y and z from the variable x determines the same - relationship of all harmonics of function and argument of the same order.

Analyzing the obtained formulas for calculating the values of MDGP elements, it should also be noted a very interesting pattern: the elements of the MDGP matrix are determined by the coefficients of the series (1.55) only up to the $2n$-th harmonic, and each element is determined by one of these coefficients or the sum or difference of only two.

In the first approximation, MDGP $S_{Y\Gamma}$ connects vectors of small increments $\Delta\vec{X}_\Gamma$ and $\Delta\vec{Y}_\Gamma$ vectors of amplitudes $\vec{X}_\Gamma$ and $\vec{Y}_\Gamma$:

$$\Delta\vec{Y}_\Gamma = S_{Y\Gamma}\, \Delta\vec{X}_\Gamma . \qquad (1.62)$$

Thus, a small increase in the amplitude of any harmonic of a variable y (as a function) consists of terms, each of which is a contribution of a small increase in the amplitude of the corresponding harmonic of the variable x (as a argument). Thus, a small increment ΔY_{c1} (increase in the cosine amplitude of the first harmonic of the variable y) is determined by the formula (let the harmonics be taken into account only for the 3rd inclusive):

$$\begin{aligned}\Delta Y_{c1} &= \Delta Y_{c1,0} + \Delta Y_{c1,c1} + \Delta Y_{c1,s1} + \Delta Y_{c1,c2} + \Delta Y_{c1,s2} + \Delta Y_{c1,c3} + \Delta Y_{c1,s3} = \\ &= S_{YГ(c1,0)}\Delta X_0 + S_{YГ(c1,c1)}\Delta X_{c1} + S_{YГ(c1,s1)}\Delta X_{s1} + S_{YГ(c1,c2)}\Delta X_{c2} + \\ &\quad + S_{YГ(c1,s2)}\Delta X_{s2} + S_{YГ(c1,c3)}\Delta X_{c3} + S_{YГ(c1,s3)}\Delta X_{s3} = \\ &= \quad 2P_1\Delta X_0 + (2R + P_2)\Delta X_{c1} + Q_2\Delta X_{s1} + (P_1 + P_3)\Delta X_{c2} + \\ &\quad + (Q_1 + Q_3)\Delta X_{s2} + (P_2 + P_4)\Delta X_{c3} + (Q_2 + Q_4)\Delta X_{s3},\end{aligned} \tag{1.63}$$

here

$S_{YГ(c1,j)}\ (j = 0, c1, s1, c2, s2, c3, s3) -$ the corresponding element of the second row of the matrix $S_{YГ}$; $\Delta X_j\ (j = 0, c1, s1, c2, s2, c3, s3) -$ increments of the constant component and the cosine and sine amplitudes of the harmonics.of the dependence $x = x[\eta]$.

By formula (1.63), the first term of the increment ΔY_{c1},formed by the increment of the constant component of the dependence $x = x[\eta]$ is determined only by the cosine amplitude of the first harmonic of the dependence (1.55). The second term, formed by the increase in the cosine amplitude of the first harmonic of dependence $x = x[\eta]$, is determined only by the constant component and the cosine amplitude of the second harmonic of dependence (1.55). The third term, formed by the increase of the sinusoidal amplitude of the first harmonic of dependence $x = x[\eta]$, is determined only by the sinusoidal amplitude of the second harmonic of dependence (1.55). The fourth term, formed by the increase in the cosine amplitude of the second harmonic of the dependence $x = x[\eta]$, is determined only by the cosine amplitudes of the first and third harmonics of the dependence (1.55). The fifth term, which is formed by the increase of the sinusoidal amplitude of the second harmonic of dependence $x = x[\eta]$, is determined only by the sinusoidal amplitudes of the first and third harmonics of dependence (1.55), etc.

Therefore, in the terms of formula (1.63) small increments of cosine amplitudes ofof the harmonics dependence $x = x[\eta]$ are multiplied only by the cosine amplitude of the harmonic of the dependence (1.55) with the closest order on the right or by the sum of cosine amplitudes of dependence harmonics (1.55) with the nearest orders of magnitude harmonics of dependence $x = x[\eta]$. The small increments of the sine amplitudes of the harmonics of the dependence $x = x[\eta]$ are multiplied only by the sine amplitude of the dependence harmonic (1.55) with the closest order on the right or by the sum of the sine amplitudes of the harmonics of the dependence $x = x[\eta]$ with the nearest orders on the left and right.

This observation of the relationship between the increments of the amplitudes of the harmonics of variables x and y (a similar relationship is between the increments of the amplitudes of the harmonics of variables x and z) is new, it is worth noting experts in the design of nonlinear devices the relationship between the desired harmonics of the variable-argument and the variable-function.

1.5. Periodic solutions of nonlinear systems of differential equations

In the previous sections, we considered the essence of numerical polyharmonic modeling of forced oscillations on a simple example - consideration of the algorithm for finding the periodic solution of a nonlinear scalar differential equation. Now we can proceed to a more complex task - to consider algorithms for finding periodic solutions of nonlinear systems of differential equations.

1.5.1. The notation form of a nonlinear system of differential equations

As noted in the Introduction, the oscillations described by nonlinear systems of differential equations can be forced, parametric or auto-oscillations. To ensure the same approach to the numerical simulation of all these types of oscillations using unified software, the system of differential equations describing the oscillations will be reduced to the form

$$\frac{d\,\vec{y}}{dt} + \vec{z} - \vec{e} = 0, \qquad (1.64)$$

or

$$\frac{d\,\vec{x}}{dt} + \vec{z} - \vec{e} = 0, \qquad (1.65)$$

or

$$B\frac{d\,\vec{y}}{dt} + \vec{z} - \vec{e} = 0, \qquad (1.66)$$

or

$$B\frac{d\,\vec{x}}{dt} + \vec{z} - \vec{e} = 0\ , \qquad (1.67)$$

here $\vec{x} = \begin{Vmatrix} x_1 \\ \vdots \\ x_j \\ \vdots \\ x_k \end{Vmatrix}$; $\vec{y} = \begin{Vmatrix} y_1 \\ \vdots \\ y_j \\ \vdots \\ y_k \end{Vmatrix}$; $\vec{z} = \begin{Vmatrix} z_1 \\ \vdots \\ z_j \\ \vdots \\ z_k \end{Vmatrix}$; $\vec{e} = \begin{Vmatrix} e_1 \\ \vdots \\ e_j \\ \vdots \\ e_k \end{Vmatrix}$ (1.68 a, b, c, d)

- matrix-columns (vectors-columns), each of which has k elements, and the vector $\vec{e}$ is a vector of external forcing forces;

$$B = \begin{Vmatrix} b_{11} & \cdots & b_{1j} & \cdots & b_{1k} \\ \vdots & & & & \\ b_{i1} & \cdots & b_{ij} & \cdots & b_{ik} \\ \vdots & & & & \\ b_{k1} & \cdots & b_{kj} & \cdots & b_{kk} \end{Vmatrix} \qquad (1.69)$$

- constant square matrix of the k-th order.

Each of the vector differential equations in the notation forms (1.64) - (1.67) is a system k scalar differential equations of the first order.

If there is a periodic dependence on time

$$\vec{x} = \vec{x}[t] = \vec{x}[t+T] \ , \qquad (1.70)$$

that satisfies the differential equation in one of the notation forms (1.64) - (1.67), it is its desired periodic solution.

In the notation forms (1.64) and (1.66) under the sign of the derivative there is not a vector $\vec{x}$ whose periodic dependence (1.70) is the desired periodic solution of the vector differential equation, but some vector $\vec{y}$ which is a nonlinear function of the vector $\vec{x}$

$$\vec{y} = \vec{y}[\vec{x}] \qquad (1.71)$$

or, in expanded form,

$$\begin{aligned} y_1 &= y_1[x_1,...,x_j,...,x_k]; \\ &\vdots \\ y_k &= y_k[x_1,...,x_j,...,x_k]. \end{aligned} \qquad (1.71a)$$

If in the equation in one of the notation forms (1.64) - (1.67) the vector of external forcing forces $\vec{e}$ is nonzero and is a $T-$periodic function of time

$$\vec{e} = \vec{e}[t] = \vec{e}[t+T], \qquad (1.72)$$

and the vector $\vec{z}$ is some nonlinear function of the vector $\vec{x}$

$$\vec{z} = \vec{z}[\vec{x}] \qquad (1.73)$$

or ot the vectors $\vec{x}$ and $\vec{y}$

$$\vec{z} = \vec{z}[\vec{x}, \vec{y}] \ , \qquad (1.74)$$

then the equation describes *the forced oscillations*.

The vector functional dependence (1.73), if written in expanded form, has the form

$$\begin{aligned} z_1 &= z_1[x_1,...,x_j,...,x_k]; \\ &\vdots \\ z_k &= z_k[x_1,...,x_j,...,x_k]. \end{aligned} \qquad (1.73a)$$

and dependence (1.74) – the form

$$\begin{aligned} z_1 &= z_1[x_1,...,x_j,...,x_k,y_1,...,y_j,...,y_k]; \\ &\vdots \\ z_k &= z_k[x_1,...,x_j,...,x_k,y_1,...,y_j,...,y_k]. \end{aligned} \qquad (1.74a)$$

If in the equation of one of the notation forms (1.64) - (1.67) the vector $\vec{e}$ is zero and among the arguments of the vector $\vec{z}$ is time t

$$\vec{z} = \vec{z}[\vec{x}, t] \qquad (1.75)$$

or

$$\vec{z} = \vec{z}[\vec{x}, \vec{y}, t], \qquad (1.76)$$

then such an equation can describe *parametric oscillations*.

The expanded form of functional dependence (1.75) is as follows

$$\begin{aligned} z_1 &= z_1[x_1,...,x_j,...,x_k,t]; \\ &\vdots \\ z_k &= z_k[x_1,...,x_j,...,x_k,t] \end{aligned} \qquad (1.75a)$$

and the expanded form of dependence (1.76) is as follows

$$\begin{aligned} z_1 &= z_1[x_1,...,x_j,...,x_k,y_1,...,y_j,...,y_k,t]; \\ &\vdots \\ z_k &= z_k[x_1,...,x_j,...,x_k,y_1,...,y_j,...,y_k,t]. \end{aligned} \qquad (1.76a)$$

Finally, if in the equation of one of the notation forms (1.64) - (1.67) the vector $\vec{e}$ is zero, and the vector $\vec{z}$ among its arguments does not contain time, ie is a function of only the vector $\vec{x}$ as specified in (1.73), or vectors $\vec{x}$ and $\vec{y}$ as specified in (1.74), such an equation can describe *self-oscillation*.

1.5.2. Harmonic algebraization of a vector differential equation

The first step in determining the periodic solution of a vector differential equation in one of the forms of notation (1.64) - (1.67) is its harmonic algebraization - the notation of an algebraic nonlinear vector equation, which is a harmonic reflection of the vector differential equation.

Performing harmonic algebraization of a scalar differential equation is described in Section 1.1. Harmonic algebraization of a vector differential equation, or a system of first-order differential equations, is performed similarly. The periodic dependences on the time of the variables $x_1,...,x_k$ (vector $\vec{x}$), $y_1,...,y_k$ (vector $\vec{y}$), $z_1,...,z_k$ (vector $\vec{z}$), $e_1,...,e_k$ (vector) are approximated by trigonometric series of the form (1.8) and we substitute these approximations into a system of differential equations. After performing the differentiation operation, this system is transformed into a system of algebraic equations, the amplitudes of the harmonics of the trigonometric series of which are unknown.

Such algebraization, by analogy with the formal rule set forth in Section 1.1, can be performed as follows: each vector of variables in the vector differential equation must be replaced by an amplitude vector and the differentiation operation must be replaced by multiplication on the left by the circular frequency of the first harmonic and the differentiation matrix.

As a result of harmonic algebraization of the vector differential equation (1.64) we obtain the vector algebraic equation

$$\omega D^* \vec{Y}_\Gamma^* + \vec{Z}_\Gamma^* - \vec{E}_\Gamma^* = 0, \qquad (1.77)$$

here

$$D^* = diag(D,...,D,...,D); \qquad (1.78)$$

$$\vec{Y}_\Gamma^* = colon(\vec{Y}_{1\Gamma}, ..., \vec{Y}_{j\Gamma},..., \vec{Y}_{k\Gamma}); \qquad (1.79)$$

$$\vec{Z}_\Gamma^* = colon(\vec{Z}_{1\Gamma}, ..., \vec{Z}_{j\Gamma},..., \vec{Z}_{k\Gamma}); \qquad (1.80)$$

$$\vec{E}_\Gamma^* = colon(\vec{E}_{1\Gamma}, ..., \vec{E}_{j\Gamma},..., \vec{E}_{k\Gamma}). \qquad (1.81)$$

The root of the vector algebraic equation (1.77) is a vector

$$\vec{X}_\Gamma^* = colon(\vec{X}_{1\Gamma}, ..., \vec{X}_{j\Gamma},..., \vec{X}_{k\Gamma}). \qquad (1.82)$$

Since in the differential equations of the form (1.64) - (1.67) the variables are -component vectors of the form (1.68), each of the vectors (1.79) - (1.82) consists of components, each of which is a vector of the amplitudes of the form. (1.13). Let us call the vectors (1.79) - (1.82) *composite vectors of amplitudes*.

The formation of composite amplitude vectors from simple amplitude vectors of the form (1.13) by the method of their sequential recording one after another is called a sequential method of forming a composite amplitude vector. With this method of forming composite amplitude vectors, the matrix (1.78) is block-diagonal, it has the same blocks diagonally - matrices of form differentiation (1.16) or (1.17). The matrix (1.78) is called a composite differentiation matrix.

The result of harmonic algebraization of the differential equation (1.65) is an algebraic equation

$$\omega D^* \vec{X}^*_\Gamma + \vec{Z}^*_\Gamma - \vec{E}^*_\Gamma = 0 \ . \qquad (1.83)$$

The result of harmonic algebraization of differential equations (1.66) and (1.67) is equation

$$\omega B^*_\Gamma D^* \vec{Y}^*_\Gamma + \vec{Z}^*_\Gamma - \vec{E}^*_\Gamma = 0 \qquad (1.84)$$

and

$$\omega B^*_\Gamma D^* \vec{X}^*_\Gamma + \vec{Z}^*_\Gamma - \vec{E}^*_\Gamma = 0, \qquad (1.85)$$

here

$$B^*_\Gamma = \begin{Vmatrix} B_{11\Gamma} & \cdots & B_{1j\Gamma} & \cdots & B_{1k\Gamma} \\ \vdots & & & & \\ B_{i1\Gamma} & \cdots & B_{ij\Gamma} & \cdots & B_{ik\Gamma} \\ \vdots & & & & \\ B_{k1\Gamma} & \cdots & B_{kj\Gamma} & \cdots & B_{kk\Gamma} \end{Vmatrix} \qquad (1.86)$$

- constant square block matrix of order kN_g. Its components are blocks - diagonal matrices of order N_g and form

$$B_{ij\Gamma} = diag(b_{ij}, b_{ij}, ..., b_{ij}), \qquad (1.87)$$

having their elements corresponding to the elements of the matrix (1.69).

Records of nonlinear vector algebraic equations (1.77), (1.83) - (1.85) are generalized by one record

$$\vec{U}^*_\Gamma[\vec{X}^*_\Gamma] = 0. \qquad (1.88)$$

In this section and in all subsequent sections, the presence in the names of the amplitude vectors or the corresponding matrices of the upper right index "*" (asterisk) carries the information that these amplitude vectors or matrices belong to the vector algebraic equation obtained by harmonic algebraization of the system of differential equations.

The harmonic algebraization of systems of differential equations described in this section does not have to be performed by the user of the method when modeling oscillations, if, as will be shown below:

1) the software described in Chapter 2 is used;
2) the system of differential equations is reduced to one of the forms (1.64) - (1.67).

In other cases and if a detailed analysis of the software components of the software presented in Chapter 2 is necessary, the user may need to perform the procedure of harmonic algebraization of the considered nonlinear system of differential equations.

1.5.3. Determination of the periodic solution of a nonlinear system of differential equations

If the vector differential equation describes forced or parametric oscillations, then the circular frequency of the fundamental harmonic of its periodic solution is known in advance. In the case of self-oscillations, it is unknown and is determined together with the amplitudes of the harmonics. In order to unify the solution search algorithm so that it is suitable for both forced or parametric oscillations and for self-oscillations, we will consider the value of the variable and introduce a vector

$$\vec{X}^* = \left\| \begin{matrix} \vec{X}^*_{\Gamma} \\ \omega \end{matrix} \right\| . \qquad (1.89)$$

We consider it as the root of some nonlinear vector finite equation

$$\vec{U}^*_{\omega}[\vec{X}^*] = 0 . \qquad (1.90)$$

Since the number of unknowns is one more than the number of unknowns of equation (1.88), another scalar equation must be added to equation (1.88) when it is formed.

For the case when the last component of the vector (1.89) is known in advance and equal to the given value $\omega_з$ (forced or parametric oscillations), this additional equation can be written as

$$\omega - \omega_з = 0 \qquad (1.91)$$

or, in matrix form,

$$C_{в,n}\,\vec{X}^* - \omega_з = 0 , \qquad (1.91a)$$

here

$$C_{в,n} = \| 0, 0, \ldots, 0, 1 \| \qquad (1.92)$$

- matrix-row size $kN_g + 1$, formed of zeros and one unit, which occupies the last position.

If the value ω is wanted (self-oscillation), then this additional equation can be an equation

$$C_a\,\vec{X}^* = 0 , \qquad (1.93)$$

here

$$C_a = \| 0, 1, 0, 0, \ldots, 0 \| \qquad (1.94)$$

- matrix-row $kN_g + 1$ size, formed of zeros and one unit, which occupies the second position. The appearance of this matrix is due to the following considerations.

Nonlinear autonomous systems of differential equations describing self-oscillations are nonisochronous, ie the frequency of self-oscillations is not given, but is determined by the internal parameters of the system, which depend on the amplitudes of the harmonics of the regime quantities of the oscillatory process. Since such oscillations are not tied to any external

forcing forces, the time coordinate of the beginning of the period of oscillations can be chosen arbitrarily. Thus, for the beginning of the period it is possible to take such value of time at which one of harmonics of one of variables of process passes through zero, and for this purpose it is necessary to accept cosine amplitude of this harmonic equal to zero. The fact that in the matrix-row (1.92) the unit occupies the second position means that equation (1.93) is set equal to zero cosine component of the amplitude of the 1st harmonic X_{1c1} of dependence $x_1 = x_1[t]$, in the case when constant components and harmonics of all orders are taken into account, and sinusoidal component of amplitude 1st harmonic X_{1s1}, if only harmonics of odd orders are taken into account.

Thus, the left-hand side of equation (1.90) in the simulation of forced or parametric nonlinear oscillations has the form

$$\vec{U}^*_\omega[\vec{X}^*] = \left\| \begin{matrix} \vec{U}^*_\Gamma[\vec{X}^*_\Gamma] \\ C_{в.п}\,\vec{X}^* - \omega_з \end{matrix} \right\| \qquad (1.95)$$

and in the simulation of self-oscillations - view

$$\vec{U}^*_\omega[\vec{X}^*] = \left\| \begin{matrix} \vec{U}^*_\Gamma[\vec{X}^*_\Gamma] \\ C_a\,\vec{X}^* \end{matrix} \right\|. \qquad (1.96)$$

The search for the periodic solution of a nonlinear vector differential equation in one of the notation forms (1.64) - (1.67) is reduced to determining the root of the nonlinear vector finite equation (1.90). First, we will look for an approximate value of this root and then refine it.

To obtain the approximate value of the root of equation (1.90), we use the method of continuation by parameter in the following modification [48]. We give an arbitrary initial value $\vec{X}^* = \vec{X}^*_0$ and calculate the value of the left part of equation (1.90), ie - the discrepancy (residual) vector

$$\vec{H}^*_0 = \vec{U}^*_\omega[\vec{X}^* = \vec{X}^*_0]. \qquad (1.97)$$

This discrepancy vector will have numerical zero value only if $\vec{X}^* = \vec{X}^*_0$ is the root of the equation (1.90).

We assume that the algorithm for calculating the values of the composite amplitude vectors $\vec{Y}^*_\Gamma$ and $\vec{Z}^*_\Gamma$ required for the calculation $\vec{U}^*_\Gamma[\vec{X}^*_\Gamma]$, ie the left side of the equation in one of the forms of notation (1.77), (1.83) - (1.85), the value of the composite amplitude vector $\vec{X}^*_\Gamma$, ie the calculation of harmonic characteristics

$$\vec{Y}^*_\Gamma = \vec{Y}^*_\Gamma[\vec{X}^*_\Gamma]; \quad \vec{Z}^*_\Gamma = \vec{Z}^*_\Gamma[\vec{X}^*_\Gamma], \qquad (1.98)$$

is known (it will be discussed below). Using the discrepancy vector $\vec{H}^*_0$, we create a new equation

$$\vec{U}^*_\omega[\vec{X}^*] - (1-h)\vec{H}^*_0 = 0, \qquad (1.99)$$

in which h - a scalar parameter. When $h = 0$ the solution of equation (1.99) is known, this is a given value $\vec{X}^* = \vec{X}_0^*$, and when $h = 1$ equation (1.99) becomes identical to equation (1.90) and, therefore, their solutions coincide. At continuous change of parameter h there is a dependence

$$\vec{X}^* = \vec{X}^*[h] \,, \qquad (1.100)$$

which when $h = 0$ passes through $\vec{X}^* = \vec{X}_0^*$ and when $h = 1$ passes through the desired root of equation (1.90). The dependence (1,100) is integral with respect to some vector differential equation, which can be obtained by differentiation by the parameter h of equation (1.99). Let us perform this differentiation, taking into account that in this equation the vector $\vec{H}_0^*$ is a constant:

$$\frac{d\vec{U}_\omega^*}{dh} + \vec{H}_0^* = \frac{d\vec{U}_\omega^*}{d\vec{X}^*}\frac{d\vec{X}^*}{dh} + \vec{H}_0^* = 0 \qquad (1.101)$$

or

$$W^* \frac{d\vec{X}^*}{dh} = -\vec{H}_0^*, \qquad (1.102)$$

here W^* - Jacobi matrix of equation (1.90). This matrix has the following structure –

$$W^* = \begin{array}{|c|c|} \hline W_\Gamma^* & W_\omega^* \\ \hline \multicolumn{2}{|c|}{C} \\ \hline \end{array} \,, \qquad (1.103)$$

here C - matrix-string of size $kN_g + 1$, in the case of modeling of forced or parametric quantities it is a matrix $C_{в.п}$ of the form (1.92) and in the case of modeling of self-generation - a matrix C_a of the form (1.94);

$$W_\Gamma^* = \frac{\partial \vec{U}_\Gamma^*}{\partial \vec{X}_\Gamma^*} \qquad (1.104)$$

- square size $kN_g \times kN_g$ matrix;

$$W_\omega^* = \frac{\partial \vec{U}_\Gamma^*}{\partial \omega} \qquad (1.105)$$

- matrix-column size kN_g.

Now we will reveal the content of matrices W_{Γ}^{*} and W_{ω}^{*}.

If equation (1.88) has the form (1.77), then, given that the vector $\vec{E}_{\Gamma}^{*}$ is invariant, we obtain

$$W_{\Gamma}^{*}=\frac{d}{d\,\vec{X}_{\Gamma}^{*}}(\omega D^{*}\vec{Y}_{\Gamma}^{*}+\vec{Z}_{\Gamma}^{*}-\vec{E}_{\Gamma}^{*})=\omega D^{*}S_{Y\Gamma}^{*}+S_{Z\Gamma}^{*}\,, \qquad (1.106)$$

here

$$S_{Y\Gamma}^{*}=\frac{d\vec{Y}_{\Gamma}^{*}}{d\vec{X}_{\Gamma}^{*}};\quad S_{Z\Gamma}^{*}=\frac{d\vec{Z}_{\Gamma}^{*}}{d\vec{X}_{\Gamma}^{*}} \qquad (1.107)$$

- matrices, which are the differential parameters of the harmonic characteristics (1.98);

$$W_{\omega}^{*}=\frac{d}{d\,\omega}(\omega D^{*}\vec{Y}_{\Gamma}^{*}+\vec{Z}_{\Gamma}^{*}-\vec{E}_{\Gamma}^{*})=D^{*}\vec{Y}_{\Gamma}^{*}\,. \qquad (1.108)$$

If equation (1.88) has the form (1.83), then

$$W_{\Gamma}^{*}=\omega D^{*}+S_{Z\Gamma}^{*};\quad W_{\omega}^{*}=D^{*}\vec{X}_{\Gamma}^{*}. \qquad (1.109)$$

If equation (1.88) has the form (1.84), then

$$W_{\Gamma}^{*}=\omega\,B_{\Gamma}^{*}D^{*}S_{Y\Gamma}^{*}+S_{Z\Gamma}^{*};\quad W_{\omega}^{*}=B_{\Gamma}^{*}D^{*}\vec{Y}_{\Gamma}^{*}\,. \qquad (1.110)$$

If equation (1.88) has the form (1.85), then

$$.W_{\Gamma}^{*}=\omega\,B_{\Gamma}^{*}D^{*}+S_{Z\Gamma}^{*};\quad W_{\omega}^{*}=B_{\Gamma}^{*}D^{*}\vec{X}_{\Gamma}^{*}. \qquad (1.111)$$

We integrate the differential equation (1.102) by one of the numerical methods from the value $h=0$ and initial conditions $\vec{X}^{*}=\vec{X}_{0}^{*}$ to $h=1$. Obtained at $h=1$ the value $\vec{X}^{*}$ we accept as a “good” initial approximation to refine the solution of equation (1.90) by the iterative Newton method according to the formulas

$$\begin{aligned} W_{(l)}^{*}\;\Delta\vec{X}_{(l)}^{*}&=\vec{H}_{(l)}^{*}\,;\\ \vec{X}_{(l+1)}^{*}&=\vec{X}_{(l)}^{*}-\Delta\vec{X}_{(l)}^{*}\,, \end{aligned} \qquad (1.112\ \text{a,b})$$

here l - iteration number; $\vec{H}_{(l)}^{*}$ - discrepancy of equation (1.90) at $\vec{X}^{*}=\vec{X}_{(l)}^{*}$.

Depending on the form of equation (1.88): (1.77), (1.83), (1.84) or (1.85), the elements W_Γ^* and W_ω^* of Jacobi matrices W^* are calculated by the formulas, respectively (1.106), (1.108), (1.109), (1.110) and (1.111), and the residual vector - respectively by formulas

$$\vec{H}_{(l)}^* = \left\| \begin{matrix} \omega D^* \vec{Y}_{\Gamma(l)}^* + \vec{Z}_{\Gamma(l)}^* - \vec{E}_\Gamma^* \\ C\vec{X}^* \end{matrix} \right\| ; \qquad (1.113a)$$

$$\vec{H}_{(l)}^* = \left\| \begin{matrix} \omega D^* \vec{X}_{\Gamma(l)}^* + \vec{Z}_{\Gamma(l)}^* - \vec{E}_\Gamma^* \\ C\vec{X}^* \end{matrix} \right\| ; \qquad (1.113b)$$

$$\vec{H}_{(l)}^* = \left\| \begin{matrix} \omega B_\Gamma^* D^* \vec{Y}_{\Gamma(l)}^* + \vec{Z}_{\Gamma(l)}^* - \vec{E}_\Gamma^* \\ C\vec{X}^* \end{matrix} \right\| ; \qquad (1.113c)$$

$$\vec{H}_{(l)}^* = \left\| \begin{matrix} \omega B_\Gamma^* D^* \vec{X}_{\Gamma(l)}^* + \vec{Z}_{\Gamma(l)}^* - \vec{E}_\Gamma^* \\ C\vec{X}^* \end{matrix} \right\| . \qquad (1.113d)$$

Note that in the case of calculating the forced oscillations and if the solution of equation (1.90) set the initial values $\vec{X}_\Gamma^* = 0$ and $\omega = \omega_з$ (while the amplitude vectors $\vec{Y}_\Gamma^*$ and $\vec{Z}_\Gamma^*$ will be zero and the vector $\vec{X}^*$ will have a value $\vec{X}^* = colon(\vec{0}^*, \omega_з)$, where $\vec{0}^*$ - zero compound vector of amplitudes), the discrepancy vector will take value $\vec{H}_0^* = -colon(\vec{E}_\Gamma^*, 0)$ and equation (1.102) - form

$$W^* \frac{d\vec{X}^*}{dh} = \left\| \begin{matrix} \vec{E}_\Gamma^* \\ 0 \end{matrix} \right\| . \qquad (1.102a)$$

In this case, the dependence (1.100) will be a h-characteristic (see Section 1.2), which at $h = 0$ passes through the value of the vector $\vec{X}^* = colon(\vec{0}^*, \omega_з)$ and when $h = 1$ through the root of equation (1.90).

1.5.4. Harmonic characteristics and their differential parameters

To calculate by formulas (1.113) the values of the residual vectors when refining the solutions of equation (1.88) by Newton's method, as well as when calculating the value of the residual vector (1.97) by numerically integrating the differential equation (1.102) requires the value of the vector $\vec{X}^*_\Gamma$ calculate the values of vectors $\vec{Y}^*_\Gamma$ and $\vec{Z}^*_\Gamma$. Vectors $\vec{X}^*_\Gamma$, $\vec{Y}^*_\Gamma$ and $\vec{Z}^*_\Gamma$ and connect harmonic characteristics (1.98). Their calculation is performed in the sequence specified by the following formulas:

$$\vec{x}_в^{\,*} = F^* \vec{X}^*_\Gamma; \quad \vec{y}_в^{\,*} = \vec{y}_в^{\,*}[\vec{x}_в^{\,*}]; \quad \vec{z}_в^{\,*} = \vec{z}_в^{\,*}[\vec{x}_в^{\,*}, \vec{y}_в^{\,*}]; \qquad \text{(1.114 а, б, в)}$$

$$\vec{Y}^*_\Gamma = G^* \vec{y}_в^{\,*}; \quad \vec{Z}^*_\Gamma = G^* \vec{z}_в^{\,*}, \qquad \text{(1.115 а, б)}$$

here G^*, F^* - composite matrices of direct and inverse harmonic transformations;

$\vec{x}_в^{\,*}, \vec{y}_в^{\,*}, \vec{z}_в^{\,*}$ - vectors formed from the values of the components of the functions $\vec{x} = \vec{x}[\eta], \ \vec{y} = \vec{y}[\eta], \ \vec{z} = \vec{z}[\eta]$ in the grid nodes in the period (and if only odd harmonics are taken into account - then in the half-period), ie from the values of the components of the vectors (1.68).

Let's call vectors $\vec{x}^*_в, \vec{y}^*_в, \vec{z}^*_в$ composite nodal vectors. They can be formed from the values of the components of vectors (1.68) in different ways. A possible method of forming composite nodal vectors is a sequential method:

$$\begin{gathered} \vec{a}_в^{\,*} = colon(\vec{a}_{1в}, ..., \vec{a}_{jв}, ..., \vec{a}_{kв}); \\ \vec{a}_{jв} = colon(a_{j(1)}, a_{j(2)}, ..., a_{j(m)}); \\ a = x, y, z. \end{gathered} \qquad \text{(1.116)}$$

In this way, the composite nodal vector is formed from simple nodal vectors of the form (1.37), which belong to the 1st, 2nd, ..., k -th component of the vector $\vec{a}$, respectively. In this formation of composite nodal vectors, the matrices of harmonic transformations F^* and G^* used in formulas (1.114) and (1.115) have the form

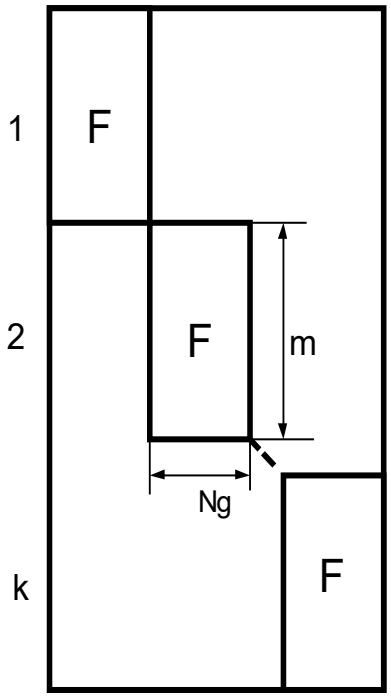


Fig. 1.2 The structure of the matrix of the form (1.117a)

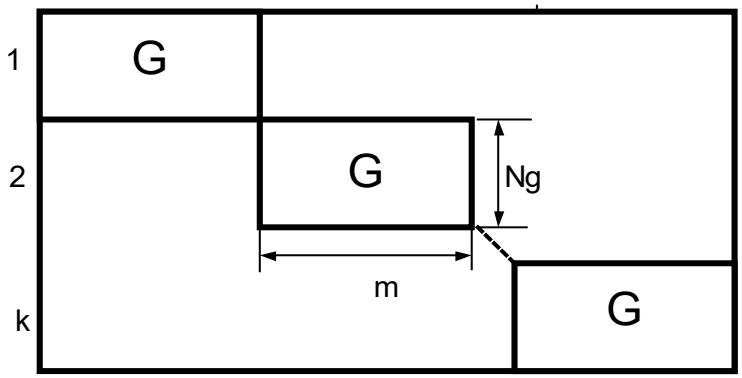


Fig. 1.3 The structure of the matrix of the form (1.117b)

$$F^* = diag(F,...,F,...,F); \qquad (1.117a)$$

$$G^* = diag(G,...,G,...,G). \qquad (1.117b)$$

The matrix F^* has k the same blocks as the form matrix F (1.39) or (1.44), and the matrix G^* has k the same blocks as the matrix G of the form (1.42), as shown in Figures 1.2 and 1.3.

The second possible way of forming composite nodal vectors is a parallel method:

$$\vec{a}_в^* = colon(a_{1(1)}, a_{2(1)},..., a_{k(1)},\ a_{1(2)}, a_{2(2)},..., a_{k(2)},... \\ ..., a_{1(m)}, a_{2(m)},..., a_{k(m)}), \quad a = x, y, z. \qquad (1.118)$$

In this formation of the composite nodal vector as its components, the values of the 1st, 2nd, ..., k-th component of the vector $\vec{a}$ in the first node of the period (or half-period, if only odd harmonics are taken into account) follow, then in the same order the values these components in the second, third and all other nodes up to the last - m-th node. As practice has shown, in the software implementation of algorithms of the differential harmonic method, a parallel method is a more expedient way of forming composite nodal vectors. When using it, it is a bit easier to programmatically implement instant (or half-period) process models.

For a parallel method of forming composite nodal vectors, the matrix has the form (see Fig. 1.4)

$$F^* = colon(F_1,..., F_\chi,..., F_m), \qquad (1.119)$$

here F_χ – block-diagonal matrix of dimensions $k \times kN_g$ that has k the same blocks, each of which is a matrix-string of the form

$$\left\| 1 \cos\eta_{(\chi)} \sin\eta_{(\chi)} \cos 2\eta_{(\chi)} \sin 2\eta_{(\chi)} \ldots \cos n\eta_{(\chi)} \sin n\eta_{(\chi)} \right\|,$$

if in each variable the constant component and harmonics of all orders, or a kind are considered

$$\left\| \cos\eta_{(\chi)} \sin\eta_{(\chi)} \cos 3\eta_{(\chi)} \sin 3\eta_{(\chi)} \ldots \cos n\eta_{(\chi)} \sin n\eta_{(\chi)} \right\|,$$

if only harmonics of odd orders are taken into account.

The matrix G^* for the case when the composite node vectors have the form (1.118), is calculated by the formula

$$G^* = \frac{2}{m} \Theta^* F^{*T} \xi_\eta^*, \qquad (1.120)$$

where Θ^* – block-diagonal matrix composed of k identical blocks, each of which is a matrix (1.43); F^{*T} - matrix transposed on the tray to the matrix (1.119a);

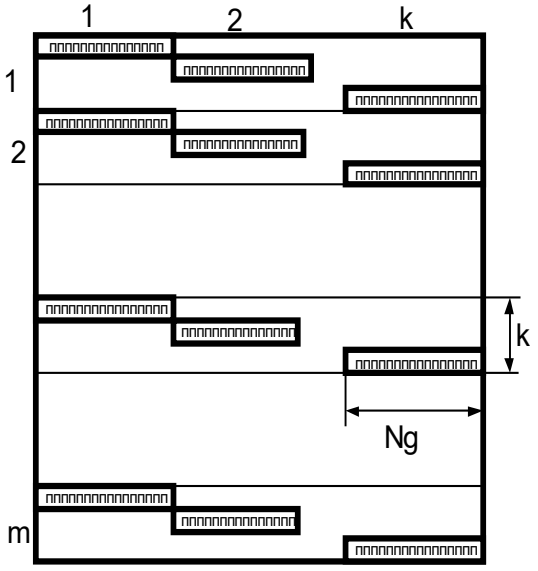


Fig. 1.4. Structure of the matrix of the form (1.119)

ξ_η^* - diagonal matrix of dimension $k\,m$, which is single, if numerical methods based on quadrature formulas of rectangles or trapezoids are used to calculate certain integrals, and which when using Simpson's formula has the form

$$\xi_\eta^* = diag\left(\frac{2}{3}, \frac{2}{3}, \ldots, \frac{2}{3}, \frac{4}{3}, \frac{4}{3}, \ldots, \frac{4}{3}, \frac{2}{3}, \frac{2}{3}, \ldots, \frac{2}{3}, \frac{4}{3}, \frac{4}{3}, \ldots, \frac{4}{3}, \ldots\right),$$

here diagonally groups with the same numbers have N_g elements.

The algorithm for calculating harmonic characteristics (1.98) can be expressed by the following sequence of directives (algorithm 1.4):

a) for a given value $\vec{X}_\Gamma^*$ according to the formula (1.114a) to calculate the value of the composite nodal vector $\vec{x}_в^*$;

b) by the values of the components of the vector $\vec{x}_в^*$ using the dependences

(1.71) or (1.74) calculate the values of the components of the composite nodal vectors $\vec{y}_в^{\,*}$ and $\vec{z}_в^{\,*}$ (instantaneous period or half-period process model);

c) by formulas (1.115a, b) calculate the values of vectors $\vec{Y}_Г^*$ and $\vec{Z}_Г^*$.

When calculating the values of the main block $W_Г^*$ of the Jacobi matrix by formulas (1.106), (1.109), (1.110) or (1.111) it is necessary to calculate the values of matrices (1.107) according to the given value of the vector $\vec{X}_Г^*$. These matrices consist of k^2 blocks, each of which is a square matrix of dimension N_g, and have the form:

$$S_{YГ}^* = \left\| \begin{matrix} \dfrac{\partial \vec{Y}_{1Г}}{\partial \vec{X}_{1Г}} & \cdots & \dfrac{\partial \vec{Y}_{1Г}}{\partial \vec{X}_{jГ}} & \cdots & \dfrac{\partial \vec{Y}_{1Г}}{\partial \vec{X}_{kГ}} \\ \vdots & & \vdots & & \vdots \\ \dfrac{\partial \vec{Y}_{jГ}}{\partial \vec{X}_{1Г}} & \cdots & \dfrac{\partial \vec{Y}_{jГ}}{\partial \vec{X}_{jГ}} & \cdots & \dfrac{\partial \vec{Y}_{jГ}}{\partial \vec{X}_{kГ}} \\ \vdots & & \vdots & & \vdots \\ \dfrac{\partial \vec{Y}_{kГ}}{\partial \vec{X}_{1Г}} & \cdots & \dfrac{\partial \vec{Y}_{kГ}}{\partial \vec{X}_{jГ}} & \cdots & \dfrac{\partial \vec{Y}_{kГ}}{\partial \vec{X}_{kГ}} \end{matrix} \right\|; \qquad (1.121)$$

$$S_{ZГ}^* = \left\| \begin{matrix} \dfrac{\partial \vec{Z}_{1Г}}{\partial \vec{X}_{1Г}} & \cdots & \dfrac{\partial \vec{Z}_{1Г}}{\partial \vec{X}_{jГ}} & \cdots & \dfrac{\partial \vec{Z}_{1Г}}{\partial \vec{X}_{kГ}} \\ \vdots & & \vdots & & \vdots \\ \dfrac{\partial \vec{Z}_{jГ}}{\partial \vec{X}_{1Г}} & \cdots & \dfrac{\partial \vec{Z}_{jГ}}{\partial \vec{X}_{jГ}} & \cdots & \dfrac{\partial \vec{Z}_{jГ}}{\partial \vec{X}_{kГ}} \\ \vdots & & \vdots & & \vdots \\ \dfrac{\partial \vec{Z}_{kГ}}{\partial \vec{X}_{1Г}} & \cdots & \dfrac{\partial \vec{Z}_{kГ}}{\partial \vec{X}_{jГ}} & \cdots & \dfrac{\partial \vec{Z}_{kГ}}{\partial \vec{X}_{kГ}} \end{matrix} \right\|. \qquad (1.122)$$

By definition, they are complete derivatives of composite amplitude vectors $\vec{Y}_Г^*$ and $\vec{Z}_Г^*$ by the composite amplitude vector $\vec{X}_Г^*$, ie, differential parameters of harmonic characteristics (1.98), so we call them *composite matrices of differential harmonic parameters*.

By analogy with formula (1.45), the values of matrices (1.121) and (1.122) can be calculated by formulas

$$S^*_{YГ} = G^* S^*_{yв} F^*; \quad S^*_{ZГ} = G^* S^*_{zв} F^*, \tag{1.123}$$

here $S^*_{yв}, S^*_{zв}$ – composite matrices of differential harmonic parameters.

If the composite nodal vectors are formed by rule (1.116), then in (1.123) the matrices F^* and G^* have the form (1.117), and the composite matrices of nodal differential parameters $S^*_{yв}, S^*_{zв}$ consist of k^2 blocks and have the form

$$S^*_{yв} = \begin{Vmatrix} s_{yв\,1,1} & \cdots & s_{yв\,1,j} & \cdots & s_{yв\,1,k} \\ \vdots & & \vdots & & \vdots \\ s_{yв\,i,1} & \cdots & s_{yв\,i,j} & \cdots & s_{yв\,i,k} \\ \vdots & & \vdots & & \vdots \\ s_{yв\,k,1} & \cdots & s_{yв\,k,j} & \cdots & s_{yв\,k,k} \end{Vmatrix}; \tag{1.124a}$$

$$S^*_{zв} = \begin{Vmatrix} s_{zв\,1,1} & \cdots & s_{zв\,1,j} & \cdots & s_{zв\,1,k} \\ \vdots & & \vdots & & \vdots \\ s_{zв\,i,1} & \cdots & s_{zв\,i,j} & \cdots & s_{zв\,i,k} \\ \vdots & & \vdots & & \vdots \\ s_{zв\,k,1} & \cdots & s_{zв\,k,j} & \cdots & s_{zв\,k,k} \end{Vmatrix}, \tag{1.124b}$$

and their internal blocks are diagonal matrices of the following form:

$$\begin{aligned} s_{yв\,i,j} &= diag\left(\frac{dy_i}{dx_j}\Big|_{(1)}, \frac{dy_i}{dx_j}\Big|_{(2)}, \ldots, \frac{dy_i}{dx_j}\Big|_{(m)}\right); \\ s_{zв\,i,j} &= diag\left(\frac{dz_i}{dx_j}\Big|_{(1)}, \frac{dz_i}{dx_j}\Big|_{(2)}, \ldots, \frac{dz_i}{dx_j}\Big|_{(m)}\right). \end{aligned} \tag{1.125 а,б}$$

It should be borne in mind that in the general case, the element $\frac{dz_i}{dx_j}$ of the matrix (1.125b) is determined by the formula

$$\frac{dz_i}{dx_j} = \frac{\partial z_i}{\partial y_1}\frac{dy_1}{dx_j} + \ldots \frac{\partial z_i}{\partial y_j}\frac{dy_j}{dx_j} + \ldots + \frac{\partial z_i}{\partial y_k}\frac{dy_k}{dx_j} + \frac{\partial z_i}{\partial x_j}\Big|_{\vec{y}=const}. \qquad (1.126)$$

If the composite nodal vectors are formed by the rule (1.118), then in (1.123) the matrices F^* and G^* have the form (1.119) and (1.120), and the matrices $S^*_{yв}$, $S^*_{zв}$ are block-diagonal and contain m diagonal blocks - dimensional k matrices (see Fig. 1.5). :

$$S^*_{yв} = diag\left(\frac{d\vec{y}}{d\vec{x}}\Big|_{(1)}, \frac{d\vec{y}}{d\vec{x}}\Big|_{(2)}, \ldots, \frac{d\vec{y}}{d\vec{x}}\Big|_{(m)}\right); \qquad (1.127a)$$

$$S^*_{zв} = diag\left(\frac{d\vec{z}}{d\vec{x}}\Big|_{(1)}, \frac{d\vec{z}}{d\vec{x}}\Big|_{(2)}, \ldots, \frac{d\vec{z}}{d\vec{x}}\Big|_{(m)}\right). \qquad (1.127b)$$

In (1.127b) the diagonal block - the matrix $d\vec{z}/d\vec{x}$ - is determined by the formula

$$\frac{d\vec{z}}{d\vec{x}} = \frac{\partial\vec{z}}{\partial\vec{y}}\frac{d\vec{y}}{d\vec{x}} + \frac{\partial\vec{z}}{\partial\vec{x}}. \qquad (1.128)$$

A view of a composite matrix of node parameters (1.127), in which the relationships between variables in each of the nodes in the period (or half-period) are compactly reflected by its diagonal block (in matrices of the form (1.124), the structure of which corresponds to a sequential method of forming composite node vectors , this connection is “smeared” on all its blocks) and makes a parallel method of forming composite nodal vectors of the form (1.118) more attractive. This method is chosen when developing the appropriate procedures for DGM software outlined in Chapter 2.

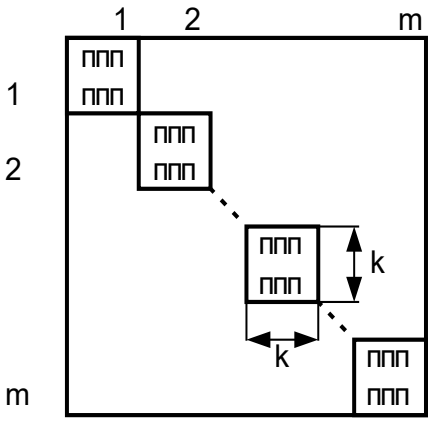


Fig. 1.5. Structure of the matrix of the form (1.127)

The values of the blocks of matrices (1.121) and (1.122) can be calculated by algorithms 1.2 or 1.3. If algorithm 1.2 is used, then the calculation of the value of the ij-th block of the matrix $S^*_{YГ}$ is performed by the formula

$$S_{YГ\,i,j} = \frac{\partial\vec{Y}_{iГ}}{\partial\vec{X}_{jГ}} = G\,S_{Yв\,i,j}\,F. \qquad (1.129)$$

If we use algorithm 1.3, then the calculation of the values of the blocks of matrices (1.121) and (1.122) is carried out by formulas (1.56). In this case, to calculate the values of the blocks of these matrices, it is necessary to find by formula (1.57) the coefficients of decompositions in the Fourier series to the $2n$-th harmonic of dependences, the numerical values of which in the m grid nodes are given by diagonal elements of matrices (1.125).

Algorithm for calculating the value of the composite vector of amplitudes $\vec{X}^*_{\Gamma}$ of the value of the composite matrix of differential harmonic parameters (1.121) or (1.122) for the case when the values of its internal blocks are calculated by formula (1.56), expressed by the sequence of such directives (*algorithm 1.5*):

a) for a given value of the vector $\vec{X}^*_{\Gamma}$ according to the directives a) and b) of the algorithm 1.4 calculate the values of compound node vectors $\vec{x}_в^{\,*}$ and $\vec{y}_в^{\,*}$;

b) from dependencies

$$\frac{dy_i}{dx_j}=\frac{dy_i}{dx_j}[x_1,...,x_j,...,x_k];$$

$$\frac{dz_i}{dx_j}=\frac{dz_i}{dx_j}[x_1,...,x_j,...,x_k,y_1,...,y_j,...,y_k]; \qquad (1.130)$$

$$i,\ j=1,\ ...,k,$$

obtained by analytical or numerical differentiation of functions (1.71) and (1.73) or (1.74) and which are an integral part of the instantaneous period (half-period) model process, calculate the value of the composite matrix of nodal differential parameters $S^*_{yв}$ (or $S^*_{zв}$) of the form (1.124) or (1.127);

c) from the elements of the inner blocks of the matrix $S^*_{yв}$ (or $S^*_{zв}$) form (1.124), which are diagonal matrices, or from the elements of the diagonal blocks of this matrix, if it has the form (1.127), to form vectors of forms (1.59) and (1.60);

d) using the directives d) and e) of algorithm 1.3 by formula (1.57) to calculate the values of vectors $\vec{V}_{Y\Gamma}$ and $\vec{V}_{Z\Gamma}$ the form (1.58) for all values of indices i and j;

e) by the values of vectors $\vec{V}_{Y\Gamma}$ and $\vec{V}_{Z\Gamma}$ form (1.58) for all values of indices and by formula (1.56) to form the values of all elements of all internal blocks of the composite matrix of differential harmonic parameters $S^*_{Y\Gamma}$ (or $S^*_{Z\Gamma}$).

1.5.5. Algorithms of search of periodic solution

Consider the algorithm for finding the solution of the equation of the form (1.90). Its first component is the algorithm for numerical integration of the vector differential equation (1.102) under the initial conditions $\vec{X}^*=\vec{X}^*_0$ from $h=0$ to $h=1$.

The sequence of operations at each step (substep) in the numerical integration of the differential equation (1.102) to obtain an approximate solution of equation (1.90) is called *algorithm 1.6*. We describe it with a sequence of the following directives:

a) for the value of the vector $\vec{X}^*$ known from the previous step (a, therefore, and $\vec{X}^*_\Gamma$) according to the directives a) and b) of algorithm 1.4 calculate the values of the vectors $\vec{x}_в^*$ and $\vec{y}_в^*$;

b) according to algorithm 1.5 calculate the values of the composite matrices of differential harmonic parameters $S^*_{Y\Gamma}$ and $S^*_{Z\Gamma}$;

c) by formula (1.103) calculate the value of the matrix W^*;

d) by solving (1.102) as a system of linear equations relative to derivatives determine the value of the vector $d\vec{X}^* / dh$;

e) according to the formula corresponding to the selected method of numerical integration of the vector differential equation, calculate the value of the increment $\Delta\vec{X}^*$ and the new value (at the end of the step) of the vector $\vec{X}^*$.

We refine the approximate value of the solution of equation (1.90) obtained by algorithm 1.6 by the iterative method of Newton according to scheme (1.112). The sequence of operations on one iteration is as follows (*algorithm 1.7*):

a) for the value of the vector $\vec{X}^*$ known from the previous iteration (and, therefore, $\vec{X}^*_\Gamma$) by algorithm 1.4 to calculate the values of the vectors $\vec{Y}^*_\Gamma$ and $\vec{Z}^*_\Gamma$;

b) by formula (1.113) calculate the value of the discrepancy vector $\vec{H}^*$;

c) according to the directives a) - d) of algorithm 1.6 to calculate the value of the matrix W^*;

d) by solving (1.112a) as a system of linear equations to calculate the value of the correction vector $\Delta\vec{X}^*$ and by formula (1.112b) - the improved value of the vector $\vec{X}^*$.

The algorithm for calculating the instantaneous (for given time points in nodes in the period or half-period) values of variables $\vec{y}$, $\vec{z}$ and matrices $\frac{d\vec{y}}{d\vec{x}}$, $\frac{d\vec{z}}{d\vec{x}}$, according to the instantaneous values of the vector $\vec{x}$ in these nodes implements an instantaneous mathematical model in the period or half-period of the object - *instantaneous model*. Directive b) of algorithm 1.4 and directive b) of algorithm 1.5 (section 1.5.4) implement the appeal to this model to calculate the values of components of nodal vectors $\vec{y}^*_в$, $\vec{z}^*_в$ and matrices of nodal differential parameters $S^*_{yв}$, $S^*_{zв}$ in time nodes of the grid on the period (or half-period) by the value of the nodal vector $\vec{x}^*_в$.

The algorithm for calculating the values of the vectors of amplitudes $\vec{Y}^*_\Gamma$ and $\vec{Z}^*_\Gamma$ and and matrices $S^*_{Y\Gamma}$ and $S^*_{Z\Gamma}$ by the value of the amplitude vector $\vec{X}^*_\Gamma$ is a mathematical model

of the object under study for the amplitudes of the harmonics of its mode coordinates or, in short, its *harmonic model*. It is accessed by the algorithms 1.6 and 1.7 described above.

In the computer implementation of the instantaneous period (or half-period) model of the process, which calculates the values of the components of vectors $\vec{y}, \vec{z}$ and matrices $d\vec{y}/d\vec{x}$, $d\vec{z}/d\vec{x}$ by the value of the vector $\vec{x}$ in all nodes of the period (half-period), the following should be kept in mind. For most types of nonlinearities that an oscillatory system can have, the order of bypassing the nodes does not matter when performing these calculations, and the nodes of the period (half-period) are bypassed in the order from the first to the last. However, there are types of nonlinearities when it is necessary to start traversing nodes from some internal node to the last, and then - from the first to this internal. Examples from the field of nonlinear electrical engineering can be an instantaneous model in the period of a controlled valve (thyristor) or nonlinear inductance with the characteristic of magnetization of the hysteresis form. Problems with such nonlinearities are discussed later in Chapters 3 and 4.

1.5.6. Investigation of the stability of nonlinear oscillations

The oscillation of a physical system is considered stable if, as a result of any single accidental perturbation, it recovers, ie returns to its previous state. A study of the stability of a nonlinear oscillation from a mathematical point of view is a study of the stability of the periodic solution of a nonlinear vector differential equation (ie, a system of first-order nonlinear differential equations) that describes this oscillation. To do this, the nonlinear vector differential equation is linearized in the region of the periodic solution, and for the thus obtained linear differential equation, the corresponding characteristic equation is written. The roots of the latter contain information about the studied stability [2, 7, 8, 30, 37, 48, 52, 56, 60, 64, 66].

By calculating nonlinear oscillations by the differential harmonic method (as well as by the harmonic balance method), the periodic solutions of the differential equations are approximated by Fourier series - the sums of harmonics of different frequencies multiples of the fundamental harmonic frequency. Due to the random perturbation of the oscillatory process, the amplitudes of the harmonics of these approximations become variable over time. In the case of stability of the periodic process after a certain time after perturbation, the values of the amplitudes of the harmonics should be established and become as they were before the perturbation.

The operation of harmonic algebraization of differential equations (see sections 1.1 and 1.5.2) is performed with the assumption that the coefficients of the Fourier series (harmonic amplitudes), which approximate the time dependences of the variables, are constant. Therefore, after differentiation of these approximations during algebraization, the derivatives do not appear in the obtained expressions. If we consider the coefficients of the Fourier series to be variable, then due to the differentiation of expression (1.8) we obtain

$$\frac{da}{dt} = \frac{dA_0}{dt} + \sum_{\nu=1}^{n}\left(\frac{dA_{c\nu}}{dt}\cos \nu\omega t - A_{c\nu}\nu\omega \sin \nu\omega t + \frac{dA_{s\nu}}{dt}\sin \nu\omega t + A_{s\nu}\nu\omega \cos \nu\omega t\right). \quad (1.131)$$

Then the differential equation (1.65) is transformed not into a finite equation (1.83), but into a new differential equation

$$\frac{d\vec{X}^*_\Gamma}{dt} + \omega D \vec{X}^*_\Gamma + \vec{Z}^*_\Gamma = \vec{E}^*_\Gamma \ . \qquad (1.132)$$

In the steady-state periodic regime, the amplitudes of the harmonics are constant, then $d\vec{X}^*_\Gamma / dt = 0$ and equation (1.132) is reduced to equation (1.83).

In order to study the stability of nonlinear oscillation - the steady-state periodic regime as the solution of equation (1.132) - it is necessary to linearize this equation at the solution point. To do this, replace the variable $\vec{X}^*_\Gamma$ with a small increment $\Delta\vec{X}^*_\Gamma$ and equate the full differentials of the left and right parts [64]:

$$\frac{d\,\Delta\vec{X}^*_\Gamma}{dt} + \omega D^* \Delta\vec{X}^*_\Gamma + \frac{d\vec{Z}^*_\Gamma}{d\vec{X}^*_\Gamma}\,\Delta\vec{X}^*_\Gamma = 0 \ . \qquad (1.133)$$

Given the notation (1.107), this linearized equation takes the form

$$\frac{d\,\Delta\vec{X}^*_\Gamma}{dt} + (\omega D^* + S^*_{Z\Gamma})\,\Delta\vec{X}^*_\Gamma = 0 \ . \qquad (1.133a)$$

The characteristic equation corresponding to it can be obtained by the formula [48]

$$\det(p\breve{E} + \omega D^* + S^*_{Z\Gamma}) = 0, \qquad (1.134)$$

here p - variable of the characteristic equation and $\breve{E}$ -- a unit matrix of the same size as the matrix D^*.

If the solvable differential equation has the form (1.64), then the linearized equation for small increments of the components of the amplitude vectors is as follows

$$S^*_{Y\Gamma}\frac{d\,\Delta\vec{X}^*_\Gamma}{dt} + (\omega D^*\, S^*_{Y\Gamma} + S^*_{Z\Gamma})\,\Delta\vec{X}^*_\Gamma = 0 \qquad (1.135)$$

and its characteristic equation can be obtained by the formula

$$\det(pS^*_{Y\Gamma} + \omega D^* S^*_{Y\Gamma} + S^*_{Z\Gamma}) = 0 . \qquad (1.136)$$

By revealing the determinants in formulas (1.134) or (1.136) the characteristic equations are reduced to the traditional form of the algebraic equation

$$a_0 p^n + a_1 p^{n-1} + ... + a_{n-1} p + a_n = 0 \qquad (1.137)$$

and expressions for its coefficients $a_0, a_1, ..., a_n$. are obtained.

As is known [2, 7, 8, 30, 48, 52, 56], the necessary condition for the asymptotic stability of the solution is satisfied when all the coefficients of the characteristic equation of the form (1.137) are valid and greater than zero (nonnegative). Thus aperiodic disturbance of stability is impossible, disturbance of stability can have only oscillatory character in the form of so-called self-shaking.

Sufficient stability conditions formulate, for example, Hurwitz criteria (also known as Rauss-Hurwitz criteria): it is necessary that the values of all Hurwitz determinants, the determinant of the Hurwitz matrix formed by a special rule from the coefficients of the characteristic equation, and its diagonal (main) minors) were greater than zero.

From the analysis of Hurwitz criteria it is known [48] that when, starting from some stable region, to change the parameters of the system, then, in case of deterioration of the stability of the regime, the first to change their signs last (n) or $(n-1)$ last Hurwitz inequality. In the first case, this occurs when the sign changes the free term of the characteristic equation, and this means an aperiodic violation of stability. If the sign of the penultimate changes the $(n-1)$ Hurwitz inequality, then there is a change in the sign of the real part of the complex-conjugate pair of roots of the characteristic equation, and this means a violation of stability in the form of self-oscillation.

The free term of the characteristic equation (1.134) is expressed by the formula

$$a_n = \det(\omega D^* + S^*_{Z\Gamma}) \qquad (1.138)$$

and for the characteristic equation (1.136) - by the formula

$$a_n = \det(\omega D^* S^*_{Y\Gamma} + S^*_{Z\Gamma}) \ . \qquad (1.139)$$

When calculating the forced oscillations and obtaining the approximate value of the periodic solution by the method h-characteristics, ie numerical integration of the differential equation (1.102) from the initial conditions at zero value of the composite vector of amplitudes of system variables, to analyze the aperiodic stability free member of the characteristic equation.

Since the h-characteristic begins with the initial stable solution (absence of oscillations), when moving along it (the forcing force increases) the change of the sign of the free member a_n from plus to minus means the transition to the aperiodically unstable part of the characteristic, and the subsequent change of the sign from minus to plus - to restore aperiodic stability.

For the analysis of other types of loss of stability (for example - self-oscillation) at numerical integration of the differential equation (1.102) it is necessary to trace not only a sign of a free term (1.138) or (1.139) of the characteristic equation, but also to analyze other Hurwitz inequalities concerning the characteristic equation (1.134). or (1,136).

As we can see, the values of the matrices of differential harmonic parameters $S^*_{Y\Gamma}$ and $S^*_{Z\Gamma}$ and are used to analyze the stability of nonlinear oscillations calculated by the differential harmonic method. In the analysis of stability, their use is organic, because they, by definition,

are associated with infinitesimal increments of harmonic amplitudes. Their values, especially calculated in the last iteration of the solution refinement, fairly accurately reflect the relationships between the harmonics in the vicinity of the solution, where the stability is analyzed. It is also important that it is not necessary to additionally calculate the values of the matrices of differential harmonic parameters for the analysis of the stability of the solution, because they are calculated in the process of finding the solution.

1.5.7. If there are more then one periodic solutions

Among the set of nonlinear vector differential equations of the form (1.64) - (1.67), the periodic solutions of which are sought, there are equations which at certain values of their parameters can have several periodic solutions. If they are found by the algorithms described above, solving nonlinear systems of finite equations whose amplitudes of harmonics are unknown, then when obtaining the first approximation by calculating the $h-$characteristic - numerical integration of the differential equation of the form (1.102) - there are difficulties. They are due to the fact that for such cases, depending on the parameter h, the components of the vector $\vec{X}^*$, if represented graphically, are ambiguous, in particular - loop-shaped (see Fig. 1.6, there y – one of the components of the vector $\vec{X}^*$). They pass through points that are special: at these points the absolute values of the components of the derivative $d\vec{X}^*/dh$ are infinitely large.

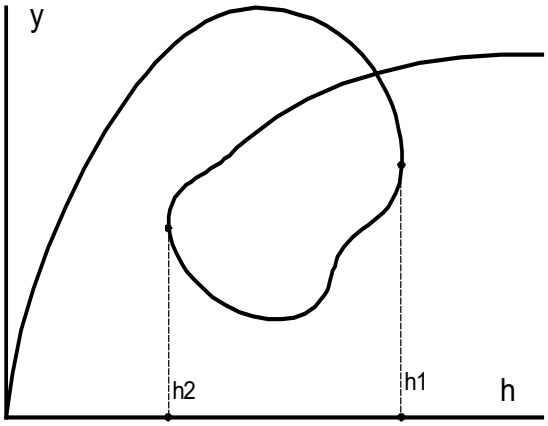


Fig. 1.6. Dependence y=y[h] loop-shaped nature

As a result, numerical integration to obtain the dependence (1,100) when approaching such points (for example, to points $h=h_1$ or $h=h_2$ in Fig. 1.6) becomes impossible. Such difficulties in numerical integration can be avoided if we use the method described in [64] and called inversion of the system of differential equations.

The essence of inversion is that in the course of numerical integration when approaching a particular point, when the absolute values of the derivatives of the independent variable h of all other variables (ie - all components of the vector $\vec{X}^*$) grow strongly (at a particular point they are equal to infinity), the independent variable h it is necessary to make dependent and to make any of the components of the vector $\vec{X}^*$ independent. At this special point, the derivatives of the new independent variable of all other quantities will be zero, and in relation to this new independent variable, the point is no longer special.

Inversion - the replacement of an independent variable - is performed not at the most specific point, but when approaching it. After inversion, numerical integration continues, but only after this new independent variable. In this case, the initial value of the step can be taken as equal to the value of the increment that this variable received in the previous step, when it was still a dependent variable.The new independent variable retains this status as long as the absolute values of the derivatives of the other variables behind it decrease. As soon as they

read to grow, it indicates the possible presence of a special point in front of this variable, and therefore it is necessary to return the status of the independent variable to the parameter h. This alternate exchange of the status of the independent variable between the parameter h and one of the components of the vector $\vec{X}^*$ eliminates the problem of passing special points.

Consider which of the components of the vector $\vec{X}^*$ it is advisable to choose the one that is given the status of an independent variable during inversion. Theoretically, it can be any component of the vector $\vec{X}^*$. However, if we talk about the solution of the nonlinear vector finite equation of the form (1.90), which is a harmonic reflection of the nonlinear vector differential equation of one of the forms (1.64) - (1.67), then the amplitude vectors $\vec{X}_{\Gamma 1}, \ldots, \vec{X}_{\Gamma k}$ as components of the vector $\vec{X}^*$ most accurately calculate first-order harmonics. when replacing certain integrals with sums, their half-waves account for the largest number of nodes (see Section 1.3). Therefore, it is expedient to choose the cosine or sine component of the first harmonic of any of the vectors of amplitudes included in the vector $\vec{X}^*$. Let this be the first amplitude vector corresponding to the variable x_1. In it it is expedient to take the second component, then in case of consideration of constant components and harmonics of all orders it will be a cosine component of amplitude of the first harmonic and in case of consideration only odd harmonics - a sine component of amplitude of the first harmonic.

The algorithm for inverting the vector differential equation (1.102) is significantly simplified if the independent variable h is added to the vector of variables $\vec{X}^*$ as its additional component. In this case, equation (1.102) must be added to the equation

$$dh/dh = 1 \,. \qquad (1.140)$$

At each step of numerical integration, the values of the derivatives of the vector $\vec{X}^*$ by the variable h are calculated by solving (1.102) as a system of linear equations with respect to $d\vec{X}^*/dh$ and the value of the derivative dh/dh by equation (1.140). Let at some value they are as follows:

$$dX_{10}/dh = a;\ dX_{1c1}/dh = b;\ dX_{1s1}/dh = c; \ldots; dh/dh = 1. \qquad (1.141)$$

If there was an inversion (the parameter h became a dependent variable, and the new non-dependent variable - the amplitude X_{1c1}), it is necessary to calculate the values of the derivatives of all variables for this new independent variable. They can be calculated from the values (1.141) taking into account the formulas

$$\frac{dX_{10}}{dX_{1c1}} = \frac{dX_{10}}{dh} \cdot \frac{dh}{dX_{1c1}} = \frac{dX_{10}}{dh} \Big/ \frac{dX_{1c1}}{dh};$$

$$\frac{dX_{1c1}}{dX_{1c1}} = 1;$$

$$\frac{dX_{1s1}}{dX_{1c1}} = \frac{dX_{1s1}}{dh} \cdot \frac{dh}{dX_{1c1}} = \frac{dX_{1s1}}{dh} \Big/ \frac{dX_{1c1}}{dh};$$
$$\vdots \qquad (1.142)$$
$$\frac{dh}{dX_{1c1}} = \frac{dh}{dh} \cdot \frac{dh}{dX_{1c1}} = 1 \Big/ \frac{dX_{1c1}}{dh}.$$

These values will be as follows:

$$\frac{dX_{10}}{dX_{1c1}} = \frac{a}{b}; \quad \frac{dX_{1c1}}{dX_{1c1}} = 1; \quad \frac{dX_{1s1}}{dX_{1c1}} = \frac{c}{b}; \quad \dots \quad ; \frac{dh}{dX_{1c1}} = \frac{1}{b}. \qquad (1.143)$$

The inversion algorithm allows the numerical integration of differential equations to obtain the dependences of the form (1.100), regardless of how many loops have graphs illustrating these dependencies, and how many special points on them. Thus, moving along the $h-$ characteristic (that is, along the dependence $\vec{X}^* = \vec{X}^*[h]$), we go through all the periodic solutions that exist on this trajectory, regardless of which of them are stable and which are unstable.

Finally, the following two remarks should be made when concluding the inversion algorithm, which allows the detection of periodic solutions if there are more than one.

1. If the system of differential equations has more than one periodic solution, then to detect them by the proposed method, it is necessary to set the "correct" maximum value of the parameter h when calculating the $h-$characteristic. Let's take another look at fig. 1.6: the : $h-$characteristic becomes obvious only after the completion of the numerical integration of the differential equation of the form (1.102). In the course of integration, it manifests itself gradually - step by step, and if at its receipt such as it is shown in fig. 1.6, given the maximum value of the parameter $h_{\max} \langle h_1$, we would get only the upper part of the curve and would not find that it further turns back and spins in the form of a loop and that for all values h in the interval $h_2 \langle h \langle h_1$ the system of differential equations has three periodic solutions.

2. It is obvious that the advance along the $h-$characteristic using the inversion algorithm allows to reveal all periodic solutions of the considered vector differential equation only on the condition that this characteristic is continuous. In all the technical problems that the author of this book had to consider, the continuity of $h-$characteristics was confirmed.

If when solving some problems of calculating nonlinear oscillations it turns out that the $h-$characteristic is not continuous, then for such cases we should look for other ways to find all roots (if the root is not one) of a vector finite equation, which is a harmonic reflection of the solvable vector differentialtion equation.

Chapter 2

SOFTWARE OF THE DIFFERENTIAL HARMONIC METHOD

2.1. Structure of the software of differential harmonic method

It should be clear to the reader who has read the previous chapter of this book that the differential harmonic method (DHM) as a tool for calculating nonlinear oscillations (that is, finding periodic solutions of nonlinear systems of differential equations describing these oscillations) is not intended for use in "manual" (non-automated) calculations. His niche is computer modeling of nonlinear oscillations, that is, with the use of a computer, in particular, a personal computer (PC) or a laptop.

The volume of preparatory work for modeling is significantly minimized if you can use pre-prepared special software. We have developed such support for DHM, and its main elements are presented in this and the following chapters. The use of the software of the differential harmonic method makes it possible to create any program for modeling and calculation of nonlinear oscillations in one or another nonlinear system using the proposed method from two blocks of software components, as shown in Fig. 2.1.

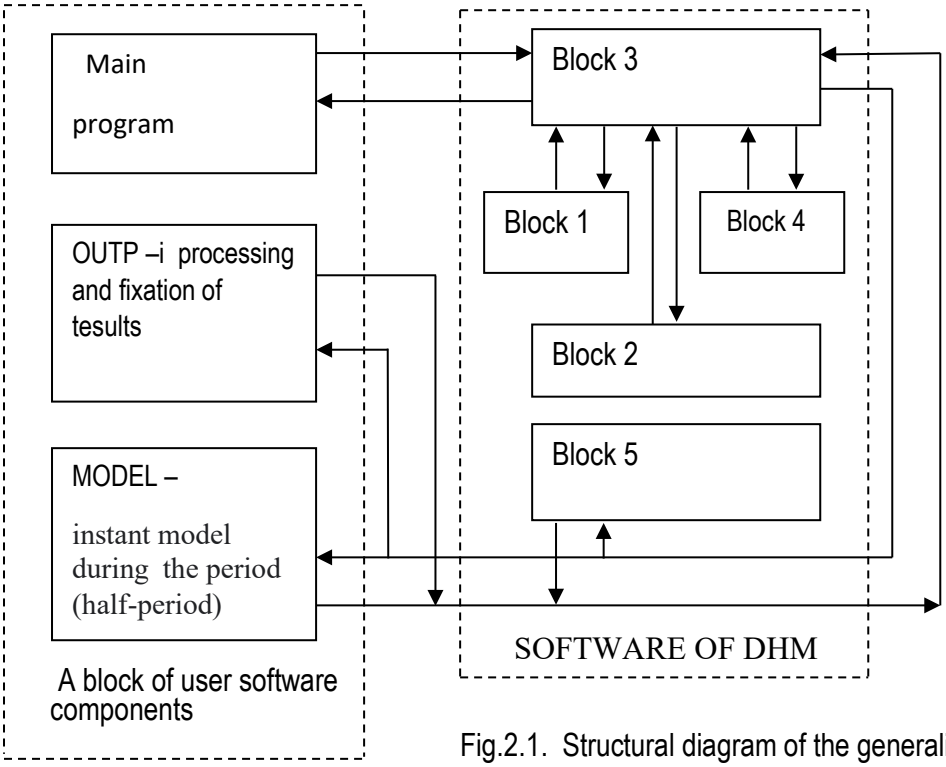


Fig.2.1. Structural diagram of the generalized program calculation of nonlinear oscillations using DHM software

These are the following blocks:

a) a block of ready-made software components (external routines, procedures) of the DHM software (hereinafter – DHM-S), which includes Blocks 1 - 5;

b) a block of user software components that he creates exclusively for this specific task.

All basic operations of the method are implemented in the procedures of blocks 1 – 5 of the DHM-S (they can be considered as macrooperations - calculation of the values of the amplitude vectors by the values of the nodal vectors, the values of the nodal vectors by the values of the amplitude vectors, the values of the matrices of differential harmonic parameters, integration of the system of differential equations, the variables of which there are amplitude vectors, etc.).

User software components implement what individualizes each specific task in the field of modeling nonlinear oscillations.

In this chapter, we will consider the software components included in Blocks 1 - 4 of the DHM-S. The program components of Block 5 of the DHM-S are considered in Chapter 3.

All software components of the DGM-S and illustrative programs given in the book are written in the Fortran-90 algorithmic language [43, 69]. They can be translated by any compiler configured for this version of Fortran and all its younger versions, for example - the compiler of the integrated development environment (IDE) ***Microsoft Developer Studio***.

The choice of programming language during the development of DGM-S was determined by the author's experience and preferences. The Fortran-90 version of the Fortran algorithmic language was popular at the end of the last century among scientific and scientific and technical specialists. Although in recent years the family of algorithmic languages has been replenished with new effective languages (Pascal, C++, Python, etc), Fortran, thanks to its simplicity, ease of learning and close connection with the language of mathematical analysis, has not gone out of use among scientists and engineers. it is intensively developed: its versions Fortran-95, Fortran-2003 [43, 70], Fortran-2008 [71], Fortran-2018 [72] are known, which are supported by many IDEs.

While preparing the book for translation into English, the author had the idea to translate the DHM-S into the C++ language, and a certain part has already been translated, but in the process of translation, the author once again made sure that Fortran is much more effective for those tasks for which DHM is intended.

If the sympathies of the reader of this book are still directed towards another algorithmic language, then he will have to translate the program components of DHM-S written in Fortran language into this other algorithmic language.

Some routines from special libraries could be used in the DHM-S, for example, a routine for solving systems of linear equations from the NAG Mark 20 library for Fortran-90 [43] or from any other similar library. However, the DHM-S proposed in this book is quite autonomous: when solving many problems in the field of nonlinear oscillations, at least those considered in this book, it was possible to do without the use of third-party subroutines.

This chapter provides complete listings (printouts) of software components of DHM-S software, and in the following chapters - printouts of user software components in illustrative programs of test tasks and examples with their input numerical files.[1]

---

[1] The user can avoid the tedious work of entering the text of software modules and data using the keyboard into the memory of his PC and searching for possible errors when entering them manually: copies of these software products can be sent to the reader if they are ordered at the author's e-mail address: **gl.lev42@ gmail.com** . At the same address, you can order the version of the DGM software in the C++ language, modified for the ***Dev C++ 5.11*** integrated environment (platform).

2.2. Program components of DHM-S

The software components of the DHM-S presented below in this chapter are grouped into five software blocks, as shown in Fig. 2.1. A library of subroutines for numerical modeling by the differential harmonic method of nonlinear oscillations can be created from these program blocks.

2.2.1. The first program block

The first block of DHM-S components (let's call it Block 1) contains those procedures that implement operations using matrices of harmonic transformations. These are six procedures: ***SNCS, VGVS, VSVG, VSVGP, KVGVS*** and ***KVSVG***. The ability of these procedures to access matrices of harmonic transformations is implemented here using the module function. For this, the block contains the ***ARRAY*** module, which makes the arrays ***F, G*** and ***GNP***, which store the values of the matrices of harmonic transformations, available to these procedures, in other words, these arrays within the block have the status of global.

To compile the software components of this block, it is advisable to download them all together (in the form of, for example, a single Block1.for file), so as not to cause, in the case of separate compilation of each software component of the block, compiler messages such as ***"Cannot find module Array".***

2.2.1.1. Module ARRAY

Text of module:

```
          Module Array
!--  The module with description of arrays F, G  and  GNP, that store
!--  the matrices of harmonic transformations
!--  that should be available to all procedures of this block
!-------------------------------------------------------
          implicit none
          real,dimension(25,150)::F,G
          real,dimension(49,150)::GNP
!---------------------------------
!--  The dimension of the arrays set here allow you to set to the number N of the highest harmonic
!--  the maximum value 12
!--  (if are taken into account constant components and all harmonics)
!--  or the maximum value 23 (if only odd harmonics are taken into account).
!--  With larger values of N, the sizes of the arrays in module must be  increased.
!---------------
          end module Array
```

The module ***Array*** consists only of a descriptive part. It describes three arrays - ***F, G*** and **GNP**, intended for storing values of matrices of harmonic transformations of the forms (1.39) or (1.44) and (1.42). Their value is calculated by the ***SNCS*** procedure included in Block 1 (it is described further in the next subsection). The values of these matrices are available to all other procedures of this block, due to the presence of the ***Use Array*** operator in each of them, and thus these matrices within the first program block receive the status of global.

The dimensions of the ***F, G*** and ***GNP*** arrays in this block are set in such a way that it is possible to set the maximum order of harmonics up to 12, taking into account constant

components and harmonics of all odd and even orders, and up to 23, taking into account harmonics of only odd orders. if it is necessary to take into account harmonics with higher orders, the dimensions of these matrices in the ***Array*** module must be increased.

These dimensions can be reduced, but at the same time they will become smaller than 12 or 23, respectively, the maximum orders of harmonics that can be set

2.2.1.2. Procedure SNCS

Text of the procedure:

```
          Subroutine SNCS(IG,N,NG,NG1,M)
!--  The procedure for calculating the dimensions M,NG,NG1 and the
!--  values of the matrix F, G  and  GNP of harmonic transformations,
!--   which are stored in the module Array
!---------------------------------
          Use Array
          implicit none
          real::SM,DE,ETA,E
          real,parameter::PI=3.14159
          integer,intent(in)::IG,N
          integer,intent(out)::NG,NG1,M
          integer::i,j
!-----------------
!--   IG - control varoiable:
!--       if =0, then are taken into account constant components and all  harmonics
!--       if =1, then are taken into account only odd harmonics
!--     N - the highest order of harmonics taken into account
!-----------------
!--     NG= 2*N+1 (at IG=0) and N+1 (at IG=1)- the number of rows of  matrices F and G
!--     NG1= 4*N+1(at IG=0) and 2*N+1 (at IG=1)- the number of rows of   matrix GNP
!--     M - the number of nodes per period (at IG=0) or half-period (at  IG=1)
!--           and the number of columns of matrices F, G and GNP
!----------------------------
      if(IG.eq.1.or.IG.eq.0) goto 1;  goto 5
  1   if(IG.eq.0.and.N.gt.12) goto 10
      if(IG.eq.1.and.N.gt.23) goto 20
          if(IG.eq.0) then
      NG=2*N+1; NG1=4*N+1
          else
            NG=N+1;   NG1=2*N+1
          end if
      M=6*NG;   if(M.le.24)M=24;   SM=2./M
        DE=PI/M;   if(IG.eq.0) DE=2.*DE
          if(IG.eq.1) goto 2
!--  when are taken into account constant components and all harmonics
          Do j=1,M
         ETA=DE*(j-1);   F(1,j)=1.;   G(1,j)=.5*SM
              GNP(1,j)=.5*SM
           do i=2,NG,2
            E=ETA*i/2.;   F(i,j)=cos(E);   F(i+1,j)=sin(E)
            G(i,j)=cos(E)*SM;   G(i+1,j)=sin(E)*SM
           end do
```

```
                  do i=2,NG1,2
                    E=ETA*i/2.
                    GNP(i,j)=cos(E)*SM; GNP(i+1,j)=sin(E)*SM
                  end do
        end do; goto 3
!----  when are taken into account only odd harmonics
   2  Do j=1,M
       E=DE*(j-1);   GNP(1,j)=.5*SM
             do i=1,NG,2
               F(i,j)=cos(E*i);   F(i+1,j)=sin(E*i)
               G(i,j)=F(i,j)*SM;  G(i+1,j)=F(i+1,j)*SM
             end do
            do i=2,NG1,2
             GNP(i,j)=cos(E*i)*SM;  GNP(i+1,j)=sin(E*i)*SM
            end do
     end do
   3  return
!--  A message to the console in the event of an abnormal termination
    5 write(*,*)' You specified IG that is neither 0 or 1'
      stop
   10 write(*,*)'  At IG=0 you set N>12'
      write(*,*)' At N>12 need to resize'
      goto 30
   20 write(*,*)'  At IG=1 you set N>23'
      write(*,*)' At N>23 need to resize'
   30 write(*,*)' of matrices F, G  and  GNP'
      write(*,*)' in descriptive part of the module Array'
      stop
      end Subroutine SNCS
```

The ***SNCS*** procedure is designed to calculate the dimensions and values of harmonic transformation matrices of the form (1.39) or (1.44), of the form (1.42) and (see formula (1.57)), which are stored by the arrays ***F, G*** and ***GNP*** described in the ***Array*** module. To access these arrays, the procedure has a ***Use Array*** operator.

The procedure has two input parameters - **IG** and ***N***.

The first input parameter ***IG*** is a control variable that can have a value of 0 or 1. Assigning it a value of 0 means that constant components and harmonics of even and odd orders are taken into account in the calculations for all coordinates of the modeled system, and assigning a value of 1 means that only odd harmonics are taken into account orders The second input parameter of the procedure is the variable ***N*** - the highest harmonic order taken into account; when IG=1, its value can only be an odd integer.

. The output parameters of the procedure are the arrays ***F, G*** and ***GNP***, which store the values of the matrices of harmonic transformations, and the dimensions ***NG, NG1*** and ***M*** of these matrices.

When ***IG*** = 0, the number of ***NG*** rows of matrices of the form (1.39) or (1.44) and matrices of the form (1.42), that is, the used rows of arrays ***F*** and ***G***, is determined by the formula

$$N_g = 2n+1 \ , \qquad (2.1)$$

and the number of rows of the matrix G_{2n} included in the formula (1.57), that is, the rows of the ***GNP*** array used, according to the formula

$$N_{g1} = 4n + 1. \qquad (2.2)$$

In these formulas, n is the highest order of the harmonics taken into account.

Note that the matrix F is stored in memory (array F) in transposed form.

When ***IG*** = 1, the number of rows of matrices F and G (used rows of arrays ***F*** and ***G***) is determined by the formula

$$N_g = n + 1 \qquad (2.3)$$

and the number of rows of the matrix G_{2n} (used rows of the ***GNP*** array) - according to the formula

$$N_{g1} = 2n + 1. \qquad (2.4)$$

The number of columns of arrays ***F , G*** and ***GNP*** , used to store matrices of harmonic transformations F, G and G_{2n} (the same - the number of nodes per period at ***IG*** = 0 or the number of nodes per half period at ***IG*** = 1) is determined by the formula

$$M = 6N_g. \qquad (2.5)$$

At the same time, there are at least six nodes per half-wave of the highest-order harmonic.

It is clear that the number of rows and columns calculated by formulas (2.1) - (2.5) should not exceed the size of the ***F, G*** and ***GNP*** arrays defined in the ***Array*** module. Therefore, the set value of the variable n is controlled by the procedure, and if the value exceeds the maximum allowable (12 at ***IG*** = 0 and 23 at ***IG*** = 1), the calculations are stopped with a corresponding message on the monitor

Next, the ***SNCS*** procedure calculates the values of the matrices F, G and G_{2n}, which are described in the ***Array*** module as arrays ***F, G*** and **GNP** and are available to this procedure thanks to the ***Use Array*** statement.

The values of matrices F and G are calculated when ***IG*** = 0 according to formulas (1.39) and (1.42) and when ***IG*** = 1 according to formula (1.44) and modified formula (1.42). The value of the matrix G_{2n} (***GNP*** array) is calculated using the same formulas as the matrix G. It differs from the matrix G only in that it has not ***NG***, but ***NG1*** rows, because when calculating the values of the matrices of differential harmonic parameters (see ***VSVGP*** and ***GRPAR*** procedures below), it is necessary to expand the function of the distribution of differential parameters to the $2 \cdot n$-th harmonic into a Fourier series.

The ***SNCS*** procedure must be called to execute at the beginning of the calculation, and the values of the matrices ***F , G*** and ***GNP*** calculated by it remain unchanged until the end of the calculation, unless it changes the number of harmonics considered. If the number of harmonics taken into account changes, the ***SNCS*** procedure must be called again for execution at each change.

2.2.1.3. Procedure VGVS

Text of procedure:

```
          Subroutine VGVS(VG,VS,NG,M)
!--  The procedure for calculating the value of the nodal vector VS with
!--  size M  by the value of the amplitude vector VG with size NG
!--  M - the number of nodes per period (at IG=0)   or half-period (at IG=1)
! --   and the number of columns of matrix F
!--  NG - the number of elements of a simple vector of amplitudes
!-------------------
          Use Array
          implicit none
          integer,intent(in)::NG,M
          real,dimension(NG),intent(in)::VG
          real,dimension(M),intent(out)::VS
          integer::i,j
        Do  j=1,M
         VS(j)=0.
          do i=1,NG
             VS(j)=VS(j)+VG(i)*F(i,j)
          end do
        end do
       return
       end  subroutine VGVS
```

The ***VGVS*** procedure is intended for performing calculations according to formula (1.38) - calculating the value of the nodal vector based on the value of the amplitude vector. The value of the matrix ***F*** required for the operation is available from the ***Array*** module.

Before calling the procedure, the formal parameter ***VG*** (vector of amplitude) must have a numerical value, and the ***SNCS*** procedure must first run (so that the matrix F, the value of which is stored in the ***F*** array, has a value).

2.2.1.4. Procedure VSVG

Text of procedure:

```
          Subroutine VGVS(VG,VS,NG,M)
!--  The procedure for calculating the value of the amplitude vector VG
!--  with size NG by the value of the  nodal vectorof VS with size M
!--  M - the number of nodes per period (at IG=0)   or half-period (at IG=1)
!--    and the number of columns of matrix G
!--  NG - the number of elements of a simple vector of amplitudes
!-------------------
          Use Array
          implicit none
          integer,intent(in)::NG,M
          real,dimension(NG),intent(in)::VG
          real,dimension(M),intent(out)::VS
          integer::i,j
        do  j=1,M
         VS(j)=0.
          do i=1,NG
```

```
            VS(j)=VS(j)+VG(i)*F(i,j)
          end do
        end do
        return
        end  subroutine VGVS
```

The ***VSVG*** procedure is designed to perform calculations according to formula (1.41) - the value of the amplitude vector is calculated from the value of the nodal vector. The value of the matrix ***G*** required for the operation is available from the ***Array*** module.

Before calling the procedure, the formal parameter ***VS*** (nodal vector) must have a numerical value and the ***SNCS*** procedure must be executed before that.

2.2.1.5. Procedure VSVGP

Text of procedure:

```
    Subroutine VSVGP(VS,VG,NG1,M)
!--  The procedure for Fourier series expansion of the parameter  distribution function VS(M)
!--  for calculating the matrix of differential harmonic paramrters
!--  M - the number of nodes per period (at IG=0)  or half-period (at IG=1)
!--     and the number of columns of matrix GNP
!-------------------------
          Use Array
          implicit none
          integer,intent(in)::NG1,M
          real,dimension(NG1),intent(out)::VG
          real,dimension(M),intent(in)::VS
          integer::i,j
      do i=1,NG1
         VG(i)=0.
           do j=1,M
              VG(i)=VG(i)+VS(j)*GNP(i,j)
           end do
       end do
    return
    end subroutine VSVGP
```

The ***VSVGP*** procedure is used when calculating the values of the matrices of differential harmonic parameters and performs calculations according to formula (1.57) - decomposes the distribution function on the period (or half-period, when only odd harmonics are taken into account) of the differential parameters into a Fourier series. The value of the **GNP** matrix required for the operation is available from the ***Array*** module by using the ***Use Array*** statement.

Before calling the procedure, the formal parameter ***VS*** (nodal vector, which specifies the distribution on the period or half-period of the differential parameter) must have a numerical value and the ***SNCS*** procedure must be executed.

2.2.1.6. Procedure KVGVS

Text of procedure:

```
    Subroutine KVGVS(K,VG,NG,NK,VS,M,MK)
!--  Процедура перетворення складеного К-кратного  вектора амплітуд VG розміру NK
!--  в складений  К-кратний вузловий вектор  VS розміру MK
!--  M - кількість вузлів на періоді(при IG=0) і напівперіоді
!--   (при IG=1), вона ж - кількість стовпців матриць F, G і GNP
!--  NG - розмірність простого вектора амплітуд
!--------
          Use Array
          implicit none
          integer,intent(in)::K,NG,NK,M,MK
          real,dimension(NK),intent(in)::VG
          real,dimension(MK),intent(out)::VS
          integer::i,j,iN,jM,L
    do i=1,K
     iN=(i-1)*NG
     do j=1,M
      jM=(j-1)*K+i; VS(jM)=0.
        do L=1,NG
         VS(jM)=VS(jM)+VG(L+iN)*F(L,j)
        end do
       end do
    end do
    return
    end subroutine KVGVS
```

The procedure performs calculations according to the formula (1.114a) - based on the value of the composite amplitude vector, it calculates the value of the composite nodal vector, which is formed here according to the formula (1.118). The value of the matrix **F** required for the operation is available from the ***Array*** module by using the ***Use Array*** statement.

Before calling the procedure, the formal parameter ***VG*** (composite vector of amplitudes) must have a numerical value and the ***SNCS*** procedure must work (that is, the values of the matrices of harmonic transformations are calculated).

2.2.1.7. Procedure KVSVG

Text of procedure:

```
Subroutine KVSVG(K,VS,M,MK,VG,NG,NK)
!--  The procedure for calculating the value of the composite vector
!--  of amplitudes  VG with size NK=NG*K
!--  by the value of the nodal composite vector VS with size MK=M*K
!--  M - the number of nodes per period (at IG=0)  or half-period (at IG=1)
!--    and the number of columns of matrix G
!--  NG - the number of elements of a simple vector of amplitudes
!----------------------
          Use Array
          implicit none
          integer,intent(in)::K,NG,NK,M,MK
          real,dimension(NK),intent(out)::VG
```

```
          real,dimension(MK),intent(in)::VS
          integer::i,j,iN,iK,L,LN
    do i=1,K
     iN=(i-1)*NG; iK=K-i
      do L=1,NG
       LN=L+iN; VG(LN)=0.
         do j=1,M
            VG(LN)=VG(LN)+VS(j*K-iK)*G(L,j)
         end do
      end do
    end do
    return
    end  subroutine KVSVG
```

The ***KVSVG*** procedure performs calculations according to the formula (1.115a) or (1.115b) - based on the value of the complex nodal vector, it calculates the value of the complex vector of amplitudes. The value of the matrix ***G*** required for the operation is available from the ***Array*** module by using the ***Use Array*** statement.

Before calling the procedure, the formal parameter ***VS*** (composite nodal vector) must have a numerical value and the ***SNCS*** procedure must be executed.

2.2.2. The second program block

The second block of DHM-S components (Block 2) contains six procedures: **OMA**, **OMV**, **OMAB**, **OMVB**, **GRPAR** and ***GRMAT***. These are procedures that implement operations with the differentiation matrix D of the form (1.16) or (1.17) and calculate the values of the matrices of differential harmonic parameters (1.56) and (1.107).

The procedures of this block do not require access to matrices of harmonic transformations.

2.2.2.1. Procedure OMA

Text of the procedure:

```
      Subroutine OMA(IG,A,B,NG,OM)
!---  Procedure for multiplying circular frequency OM  and matrix D of order NG
!--   by the matrix A of order NG
!--   IG - control variable:
!--       if =0, then constant components and harmonics of all orders  are taken into account
!--       if =1, then only harmonics of odd orders are taken info account
!-------------------------
          Implicit none
          integer,intent(in)::IG,NG
          real,intent(in)::OM
          real,dimension(NG,NG),intent(in)::A
          real,dimension(NG,NG),intent(out)::B
          integer::i,j
          real::C
    if(IG.eq.1) goto 1
!--  when const.components and harmonics of all orders are taken into account
    do j=1,NG
```

```
      B(1,j)=0.
       do i=2,NG,2
         C=A(i,j)
          B(i,j)=A(i+1,j)*OM*i*.5; B(i+1,j)=-C*OM*i*.5
       end do
     end do;  goto 2
!---  when only harmonics of odd orders are taken info account
   1 do  j=1,NG
       do i=1,NG,2
          C=A(i,j)*OM*i
           B(i,j)=A(i+1,j)*OM*i;  B(i+1,j)=-C
       end do
     end do
   2 return
     end subroutine OMA
```

The ***OMA*** procedure is designed to perform the operation of multiplying the product of the circular frequency ω and the differentiation matrix D of the form (1.16) or (1.17) by a square matrix, just as it is performed in formula (1.27).

When calling this procedure, the ***OM, NG*** and ***A*** parameters must have numeric values. The result of this operation is the assignment of the product ωDA to the matrix ***B*** .

2.2.2.2. Процедура OMV

Text of the procedure:

```
     Subroutine OMV(IG,V,V1,NG,OM)
!--   Procedure for multiplying circular frequency OM and matrix D of order NG
!--   by the vector of amplitudes V with the number of elements NG
!--   IG - control variable:
!--        if =0, then constant components and harmonics of all orders  are taken into account
!--        if =1, then only harmonics of odd orders are taken info account
!--------------------
            Implicit none
            integer,intent(in)::IG,NG
            real,intent(in)::OM
            real,dimension(NG),intent(in)::V
            real,dimension(NG),intent(out)::V1
            integer::i
            real::C
            If(IG.eq.1) goto 1
!--  when const.components and harmonics of all orders are taken into account
     V1(1)=0.
     do i=2,NG,2
        C=OM*V(i)*i*.5
        V1(i)=OM*V(i+1)*i*.5;  V1(i+1)=-C
      end do;  goto 2
!---  when only harmonics of odd orders are taken info account
   1 do i=1,NG,2
        C=OM*V(i)*i
        V1(i)=OM*V(i+1)*i;  V1(i+1)=-C
     end do
   2 return
     end subroutine OMV
```

The ***OMV*** procedure is designed to perform the operation of multiplying the product of the circular frequency ω and the differentiation matrix D of the form (1.16) or (1.17) by the vector of amplitudes of the form (1.13), just as it is performed in the formula (1.15).

When calling this procedure, the ***OM, NG*** and ***V*** parameters must have numeric values. The result of this operation is the assignment of the product $\omega D\vec{V}$ to the vector ***V1***.

2.2.2.3. Procedure OMAB

Text of the procedure:

```
      Subroutine OMAB(IG,A,B,NG,K,NK,OM)
!--   Procedure for multiplying the matrix A of order NK=NG*K
!--   by circular frequency OM and the composite matrix DC  on the left (B=OM*DC*A)
!--   NG - the number of elements of simple vector of amplitudes
!--   IG - control variable:
!--       if =0, then const. components and harmonics of all orders  are taken into account
!--       if =1, then only harmonics of odd orders are taken info account
!---------------------------
            Implicit none
            integer,intent(in)::IG,NG,K,NK
            real,intent(in)::OM
            real,dimension(NK,NK),intent(in)::A
            real,dimension(NK,NK),intent(out)::B
            integer::i,j,jK,iK,i1,j1
            real::C
            If(IG.eq.1) goto 1
!-- when const.components and harmonics of all orders are taken into account
     do i1=1,K
      iK=(i1-1)*NG
       do j1=1,K
        jK=(j1-1)*NG
          do j=1,NG
           B(1+iK,j+jK)=0.
             do i=2,NG,2
                C=A(i+iK,j+jK)
                 B(i+iK,j+jK)=OM*A(i+iK+1,j+jK)*i/2.
                 B(i+iK+1,j+jK)=-C*OM*i/2.
              end do
            end do
          end do
     end do;  goto 2
!---  when only harmonics of odd orders are taken info account
   1 do i1=1,K
       iK=(i1-1)*NG
         do j1=1,K
          jK=(j1-1)*NG
            do j=1,NG
              do i=1,NG,2
                 C=A(i+iK,j+jK)*OM*i
```

```
              B(i+iK,j+jK)=A(i+iK+1,j+jK)*OM*i
              B(i+iK+1,j+jK)=-C
            end do
          end do
        end do
      end do
    2 return
      end subroutine OMAB
```

The ***OMAB*** procedure is designed to perform the operation of multiplying the product of the circular frequency ω and the complex differentiation matrix D of the form (1.78) by a square matrix, just as it is performed according to the formula (1.106).

When calling this procedure, the parameters ***OM, NG, K, NK*** and ***A*** must have numeric values. The result of this operation is the assignment of the product $\omega D^* A^*$ to the matrix ***B***.

2.2.2.4. Procedure OMVB

Text of the procedure:

```
      Subroutine OMVB(IG,V,V1,NG,K,NK,OM)
!--   Procedure for multiplying the circular frequency OM and composite matrix D
!--   of order NK=NG*K by composite K-fold vector of amplitudes
!--   with the number of elements NK=NG*K (V1=OM*V*D)
!--   IG - control variable:
!--      if =0, then const.omponents and harmonics of all orders  are taken into account
!--      if =1, then only harmonics of odd orders are taken info account
!------------------------------
           Implicit none
           integer,intent(in)::IG,NG,K,NK
           real,intent(in)::OM
           real,dimension(NK),intent(in)::V
           real,dimension(NK),intent(out)::V1
           integer::i,j,iK
           real::C
           If(IG.eq.1) goto 1
!-- when const. components and harmonics of all orders are taken into account
     do i=1,K
       iK=(i-1)*NG; V1(1+iK)=0.
       do j=2,NG,2
         C=V(j+iK)*j*OM*.5
         V1(j+iK)=V(j+iK+1)*j*OM*.5
         V1(j+iK+1)=-C
       end do
     end do;  goto 2
!--- when only harmonics of odd orders are taken info account
   1 do i=1,K
     iK=(i-1)*NG
       do j=1,NG,2
         C=V(j+iK)*j*OM
         V1(j+iK)=V(j+iK+1)*j*OM
         V1(j+iK+1)=-C
       end do
     end do
   2 return
      end subroutine OMVB
```

The ***OMVB*** procedure is designed to perform the operation of multiplying the product of the circular frequency ω and the complex differentiation matrix D of the form (1.78) by the complex vector of amplitudes of the form (1.79), just as it is performed according to the formula (1.77).

When calling this procedure, the parameters ***OM, NG, K, NK*** and ***V*** must have numeric values. The result of this operation is the assignment of the product $\omega D^* \vec{V}^*$ to the vector ***V1***.

.

2.2.2.5. Procedure GRPAR

Text of the procedure:

```
      Subroutine GRPAR(IG,SY,NG,SYC,M,NG1)
!--   The procedure for calculating the value of the matrix SY
!--   of order NG of differential harmonic parameters
!--   by value of diagonal matrix of parameters in nodes, which is given
!--   by the vector SYC with the number the elements M
!--   IG - control variable:
!--        if =0, then constant components and harmonics of all orders are taken into account
!--        if =1, then only harmonics of odd orders are taken info account
!--   M - the number of nodes per period (at IG=0)  and per half-period (at IG=1),
!--        and number of column of matrices F, G and GNP
!--   NG - the number of elements of simple vector of amplitudes
!----------------------
           Implicit none
           integer,intent(in)::IG,NG,M,NG1
           real,dimension(M),intent(in)::SYC
           real,dimension(NG,NG),intent(out)::SY
           real,dimension(NG1)::GL
           integer::i,j,i1,i2,ij,ji,j1,jk,jk1,jk2,j11,jii,ji1,ji2
!--    GL- working vector with number of elements NG1
!--    NG1=2*N+1 (at IG=1)  and  4*N+1 (at IG=0)
!---------------------------
    Call VSVGP(SYC,GL,NG1,M)
          if(IG.eq.1) goto 1
!-- when const.components and harmonics of all orders are taken into account
    jK=NG-1; jK1=jK-2
      do j=1,NG
        SY(1,j)=GL(j)*.5
       end do
     SY(1,1)=2.*SY(1,1)
      do i=2,jK1,2
        i2=i*2; i1=i+1; SY(i,i)=GL(1)+GL(i2)*.5
        SY(i,i1)=GL(i2+1)*.5; SY(i1,i1)=GL(1)-GL(i2)*.5
        j1=i+2
        do j=j1,jK,2
          j11=j+1; ji=j-i; jii=j+i; ji1=ji+1; ji2=jii+1
          SY(i,j)=(GL(ji)+GL(jii))*.5
          SY(i,j11)=(GL(ji1)+GL(ji2))*.5
          SY(i1,j)=(-GL(ji1)+GL(ji2))*.5
          SY(i1,j11)=(GL(ji)-GL(jii))*.5
```

```
        end do
       end do
        SY(jK,jK)=GL(1)+GL(2*jK)*.5
            SY(jK,NG)=GL(2*jK+1)*.5
        SY(NG,NG)=GL(1)-GL(2*jK)*.5
       do j=1,jK
        i1=j+1
         do i=i1,NG
          SY(i,j)=SY(j,i)
         end do
      end do
       do i=2,NG
        SY(i,1)=2.*SY(i,1)
      end do;  goto 3
!---  when only harmonics of odd orders are taken info account
   1 jK=NG-1;  if(jK.EQ.1) goto 2;  jK1=jK-2
      do i=1,jK1,2
       i2=i*2;  SY(i,i)=GL(1)+GL(i2)*.5
       SY(i,i+1)=GL(i2+1)*.5
       SY(i+1,i+1)=GL(1)-GL(i2)*.5
       j1=i+2
        Do j=j1,JK,2
         ij=i+j;  ji=j-i
         SY(i,j)=(GL(ji)+GL(ij))*.5
         SY(i,j+1)=(GL(ji+1)+GL(ij+1))*.5
         SY(i+1,j)=(-GL(ji+1)+GL(ij+1))*.5
         SY(i+1,j+1)=(GL(ji)-GL(ij))*.5
        end do
      end do
   2 jK2=2*jK
     SY(jK,jK)=GL(1)+GL(jK2)*.5
     SY(jK,NG)=GL(jK2+1)*.5
     SY(NG,NG)=GL(1)-GL(jK2)*.5
    do j=1,jK
     i1=j+1
      do i=i1,NG
       SY(i,j)=SY(j,i)
      end do
     end do
   3 return
     end  subroutine GRPAR
```

The ***GRPAR*** procedure is designed to calculate the value of the matrix of differential harmonic parameters (MDHP) ***SY*** of the form (1.45). It implements the algorithm 1.3 described in section 1.4, according to which the value of MDHP is calculated according to the formula (1.56).

The ***NG, NG1, M*** and ***SYC*** parameters must be set to numeric values before the procedure is invoked. Here, ***SYC*** is a vector of size ***M***, the elements of which are assigned the values of the diagonal matrix of differential parameters of the form (1.49) in ***M*** nodes on a period or half period.

2.2.2.6. Procedure GRMAT

Text of the procedure:

```
      Subroutine GRMAT(IG,SG,NK,SC,MK,K,M,NG,NG1)
!--   The procedure for calculating the value of the composite matrix SG
!--   of differential harmonic parameters of order NK=NG*K
!--   by value of composite martix SC of parametersin nodes with dimensions MK on K  (MK=M*K)
!--   IG - control variable:
!--     if =0, then const.components and harmonics of all orders  are taken into account
!--     if =1, then only harmonics of odd orders are taken info account
!--   NG - the number of elements of simple vector of amplitudes
!--   NG1=2*N+1 (at IG=1)  i  4*N+1 (at IG=0)
!--   M - the number of nodes per period (half-period)
!--   SYC - working vector with number of elements M
!--   S - working array of order NG
!-----------------------------
            Implicit none
            integer,intent(in)::IG,NG,NG1,K,NK,M,MK
            real,dimension(MK,K),intent(in)::SC
            real,dimension(NK,NK),intent(out)::SG
            real,dimension(NG,NG)::S
            real,dimension(M)::SYC
            integer::i,j,iN,jN,jM,ii,jj
      do i=1,K
       iN=(i-1)*NG
        do j=1,K
          jN=(j-1)*NG
            do jM=1,M
              SYC(jM)=SC((jM-1)*K+i,j)
            end do
            Call GRPAR(IG,S,NG,SYC,M,NG1)
          do ii=1,NG
           do jj=1,NG
            SG(ii+iN,jj+jN)=S(ii,jj)
           end do
          end do
        end do
      end do
      return
      end subroutine GRMAT
```

The ***GRMAT*** procedure is designed to calculate the value of the composite matrix ***SY*** of the differential harmonic parameters of the form (1.121), (1.122). It implements algorithm 1.5 described in section 1.5.4. When calculating the value of the ***SY*** matrix, the procedure k^2 times (in a loop) calls the ***GRPAR*** procedure for execution.

Parameters ***K, M, NK, MK, NG, NG1*** and ***SC*** must be assigned numerical values before the ***GRMAT*** procedure is invoked. Here, *SC* is a matrix of size ***MK*** by ***K,*** which contains ***M*** blocks - square matrices of order ***K***. Each of these matrices has the values of the matrix of differential parameters in ***1, … , M*** nodes on a period (semi-period), which are diagonal blocks of matrices of the form (1.127).

2.2.3. The third program block

The third block of DHM-S components called Block 3 contains three procedures: ***HARMOSC, CALCULU*** and ***IMPROVE***, which implement algorithms for determining periodic solutions of nonlinear systems of differential equations - numerical values of composite amplitude vectors representing these solutions, as well as in cases of self-oscillation calculation, the circular frequency of the fundamental harmonic.

2.2.3.1. Procedure HARMOSC

Text of the procedure:

```
        Subroutine HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
!--  The procedure for calculating nonlinear oscillations as a sum of
!--  harmonics - determining the periodic solution (solutions) of a
!--   nonlinear system of differential equations.
!--  The first approximation of the solution is obtained by calculating
!--  the h-characteristic (numerical integration by the Kutt-Merson method
!--  with a variable step and inverting when approaching special points).
!--  The solution is refined using Newton's method.
!------------------
        Implicit none
        integer,intent(in)::K,NK
        integer,dimension(10),intent(in)::KER
        real,intent(in)::HM,H1,EPS1,EPS2
        real,dimension(NK)::E
        real,dimension(NK+2)::Y0,Y1,Y2,F0,F1,F2
        real,dimension((KER(9)/KER(8)+1)*NK+2)::Y22
        real,dimension(NK+1)::U
        real::S,C1,C2,C3,C4,C5,C6,C7,C8,S1,SN,E1,E2,PX0,PH0,PX1,PH1,Y0N,Y2N,AL,AI,AKER
        integer::NG,NG1,M,MK,KH,L,NN,KI,ID,i,ij,KK,INEV,NI,NIT,LST
!------------------
!   Procedure parameters:
!--  K - the order of the system of differential equations
!--   Y0 - a vector of variables with size NK+2, its elements from the 1st
!--        to NK,these are the elements of the composite vector of amplitudes,
!--   NK+1 component is the circular frequency of the first harmonic,
!--    NK+2 (last) component - parameter h;
!--   E - composite vector of amplitudes of forcing force;
!--   NK - the size of the composite vectors of amplitudes;
!--   HM - the maximum value of the parameter h;
!--   H1 - the value of the param. h, at which it is necessary to  clarify  solution by Newton's method;
!--   EPS1 - relative accuracy of h-characteristic calculation;
!--   EPS2 - relative accuracy when refined by Newton's method;
!--   KER - an array of control variables, the values of its components are as follows:
!--   KER(1)=0 –const.components and all harmonics are taken into account
!--   KER(1)=1 - only odd harmonics are taken into account;
!--   KER(2)=1 - differential equations are in written form (1.65)  or (1.67);
!--   KER(2)=0 -              in written form (1.64) or (1.66);
!--   KER(3)=0 -              in written form (1.64) or (1.65)
!--   KER(3)=1 -              in written form (1.66) or (1.67);
!--   KER(4)=0 - forced oscillations;
!--   KER(4)=1 - parametric oscillations;
!--   KER(4)=2 - self-oscillation;
```

```
!--    KER(5)=1 - it is necessary to print (remember) the values of the  vector of variables
!--                at all points of the h-characteristic;
!--    KER(6)=1 - it is necessary to stop the movement along  the h-characteristic after passing
!--                 its first special point;;
!--   KER(7) -  if =0 - there are no hysteresis characteristics;  if equal to a whole positive number,
!--       it means that in  the problem there are hysteresis characteristics and this   whole number is the
!—      number of steps  to expand the loop of the hysteresis characteristic to the real one
!--     KER(8)- the order of the highest harmonic;
!--     KER(9)- the order of the highest harmonic when increasing the  number of harmonics
!--              taken into account;
!--              if KER(9)=0, no scaling is performed
!--     KER(10)- if =0, then  after each increase in the number of harmonics, there are no printouts,
!--             the results are  printed only after taking into account the harmonics with the highest order,
!--             if =1, then printouts are present after each increment
!-----------------------------
      write(1,5) KER(8)
   5 format(2X,'The highest harmonic order=',i2)
            call SNCS(KER(1),KER(8),NG,NG1,M)
!-- calculated the values of matrices of harmonic transformations
            if(KER(4).ne.0)E=0.
            MK=M*K      !-- MK - the size of the composite nodal vectors
            KH=NK+2     !-- KH - number of components of the vector Y0, which
                        !-- is an independent variable
            AL=0.       !-- AL - the narrowing factor of the loops of the
                        !-- characteristics of the hysteresis form (if any)
            KI=0        !-- KI - control variable, with KI=0 – calculation
                        !-- of the h-characteristic, with KI=1 – refinement
                        !-- of the solution by Newton's method
            INEV=1
      !-- INEV - control variable, when INEV=0 - calculation of fluctuations;
      !--       when INEV=1 - calculation of the inconsistencies for the
       !--       initial value of Y0
            call CALCULU(KER,KI,INEV,AL,Y0,F0,U,NK,K,NG,NG1,M,MK,E,KH)
               !-- the CALCULU procedure calculated the inconsistencies F0 for the
                !-- initial value Y0
    if(KER(4).eq.0)goto 4
    write(1,1)
   1 format(3X,'The initial value of the vector of amplitudes:')
     write(1,3)Y0
     write(1,2)
   2 format(3X,'Residua for the initial value of the vector of amplitudes:')
     write(1,3)F0
   3 format(3X,4E12.4)
   4 do i=1,NK+1
        U(i)=F0(i)
     end do
!--- U - vector of residual
     INEV=0;   L=0     !--  L - number of the root at the point h=H1
     if(KER(5).ne.1)goto 31
      write(1,30)
  30 Format(/10X,'The h-characteristic is calculating')
!-------------------------------------------
!---  Two first small steps according to Euler:
  31 do i=1,2
            call CALCULU(KER,KI,INEV,AL,Y0,F0,U,NK,K,NG,NG1,M,MK,E,KH)
```

```
!--- here the CALCULU procedure calculates the derivative vector F0
            Y0=Y0+.0005*F0
      end do
            S=.05     !-- S - the initial length of the integration step
            LST=0
!-------------------------------------------
!-- Step start (Kutt-Merson method)
  40 continue
      LST=LST+1
      NN=0 !-- zeroing of the NN damping, which records the fact of step crushing
!-----------------------------------------------------
!--  step start (if there was a decrease in step length)
  50  C1=S/3.; C2=S/6.;  C3=S/8.; C4=3.*C3
       C5=S/2.; C6=3.*C5; C7=2.*S; C8=2.*C1
            KI=0         !-- updating the KI value
            call CALCULU(KER,KI,INEV,AL,Y0,F0,U,NK,K,NG,NG1,M,MK,E,KH)
             PX0=F0(2); PH0=F0(NK+2)
!-- PX0, PH0 - derivatives of the 2nd and NK+2nd components
!--   at the beginning of the step
         Y1=Y0+C1*F0
            call CALCULU(KER,KI,INEV,AL,Y1,F1,U,NK,K,NG,NG1,M,MK,E,KH)
          Y1=Y0+C2*(F0+F1)
            call CALCULU(KER,KI,INEV,AL,Y1,F1,U,NK,K,NG,NG1,M,MK,E,KH)
          Y1=Y0+C3*F0+C4*F1
            call CALCULU(KER,KI,INEV,AL,Y1,F2,U,NK,K,NG,NG1,M,MK,E,KH)
          Y1=Y0+C5*F0-C6*F1+C7*F2
            call CALCULU(KER,KI,INEV,AL,Y1,F1,U,NK,K,NG,NG1,M,MK,E,KH)
            PX1=F1(2);   PH1=F1(NK+2)
!--  PX1,PH1 - derivatives of the 2nd and NK+2nd components at the end of the step
           Y2=Y0+C2*(F0+F1)+C8*F2
!--  end of calculations in the integration step
!--  Y2 - the value of the vector of variables at the end of the step
!----------------------------------------
!--  step accuracy assessment
        ID=0; E1=0.;  E2=0.
        do i=1,NG
           E1=E1+Y2(i)**2;  E2=E2+(0.2*(Y1(i)-Y2(i)))**2
        end do
        E1=sqrt(E1)*EPS1;   E2=sqrt(E2)
        If(E2.lt.E1) goto 60
        S=S/2.;  NN=1;  goto 50
!--  EPS1 accuracy not reached, step halved,
!--  return to the beginning of the step (at mark 50)
!--  with the fixation that step splitting has occurred (NN=1)
!-------------------------------------------------
!--  EPS1 accuracy achieved
  60 if(E2*10..gt.E1) ID=1
!-- evaluated the obtained accuracy:
!--    satisfactory (ID=1), good (ID=0)
!----------------------------------------
      S1=Y2(2)-Y0(2)
            Y0N=Y0(NK+2);  Y2N=Y2(NK+2);
            SN=Y2N-Y0N; IJ=0; IF(SN.gt.0.) IJ=1
!--   Y0N - the value of the parameter h at the beginning of the step
```

```
!--   Y2N - the value of the parameter h at the end of the step
!--   SN - the increment of the parameter h per step
!--   S1 - increment of the 2nd component of the vector
!--        of amplitudes per step
!--    IJ=1 when there is movement along the h-characteristic in the
!--       direction of increasing h
!        and IJ=0 if it goes in the direction of decreasing
!--------------------------------
!--    A decision is made regarding the need to specify the root,
!       (checking if H1 is within last step)
       if(Y0N.gt.H1.and.Y2N.gt.H1.or.Y0N.lt.H1.and.Y2N.lt.H1) goto 72
        if(abs((H1-Y0N)/SN).gt.0.2) goto 61
        Y1=Y0;  goto 63
  61 if(abs((H1-Y2N)/SN).gt.0.2) goto 62
        Y1=Y2;  goto 63
  62 S=S*abs((Y0N-H1)/SN)*1.1; goto 50
  63 L=L+1;   Y1(NK+2)=H1;  KI=1
      write(*,*) 'Approximate solution is obtained'
      write(1,64)
  64 format(5X,40('-'))
      write(1,77)
  77 format(3X,'An approximate solution:')
      call OUTP(KER(1),Y1,NK+2,K,NG,M,MK,0)
     call NEWT(KI,EPS2,NIT)    !-- refinement of the root at h=H1 by Newton's method
      write(*,*) 'the solution is specified by Newton"s method'
      write(1,65)L,H1,NIT
  65 format(1X,'specified value',I2,' root for h=',F5.3/
     & 2X,'(the solution was obtaind after',i3,'-nd iteration)')
           KK=1;  if(KER(7).eq.0) KK=0
           Call OUTP(KER(1),Y1,NK+2,K,NG,M,MK,KK)
!--   Y1 - the specified value of the variables at the point h=H1
           if(H1.ne.HM)goto 70
      if(KER(7).eq.0) goto 68
           write(1,67)
  67 format(/5X,'taking into accaunt hysteresis')
           AKER=KER(7)
           do NI=1,KER(7)
                 AI=NI;  AL=AI/AKER;  call NEWT(KI,EPS2,NIT)
           end do
           call OUTP(KER(1),Y1,NK+2,K,NG,M,MK,0)
  68 if(KER(9).ne.0) goto 69
           return    !-- output taking into account hysteresis
!-- increasing the number of harmonics taken into account
  69 call IMPROVE(KER,K,KH,Y1,NK+2,NG,E,U,Y22,(KER(9)/KER(8)+1)*NK+2,EPS2,AL)
      write(*,*) 'Increasing the number of harmonics that are taken into account is completed'
           return          !-- output after the completion of increasing the number of harmonics
  70 write(1,64)
      if(Y2N.gt.HM) goto 250
!-----------------------------------
  72 If(KER(5).ne.1)goto 74
      if(KH.eq.NK+2) then
           write (1,75) Y2(NK+2)
      else
           write(1,73) Y2(NK+2),KH
```

```
      end if
  73 format(/2X,'Parameter h=',F8.5,',  at this step variable ',I3,' is independent')
  75 format(/2X,'Parameter h=',F8.5,',  at this step the parameter h  is independent ')
           KK=1;   call OUTP(KER(1),Y2,NK+2,K,NG,M,MK,KK)
!----- The OUTP procedure prints the value of the vector Y2
!----    (if KER(5)=1) at the end of the step
  74 Y0=Y2
!-- end of step = start of next step
!--   A decision is made regarding the necessity of inverting
       if(KER(4).ne.0) goto 140
       If(KH.ne.NK+2) goto 130
        if(abs(PX1).le.abs(PX0)) goto 140
          KH=2;   S=S1;   goto 140
!-- the derivative of the 2nd component of the vector of variables by parameter h increases,
!--  so we make the 2nd component an independent variable
!-- and we move on to integration by the 2nd variable with step S1
 130 if(abs(PH1).le.abs(PH0)) goto 140
       KH=NK+2;  S=SN
!-- the derivative of the parameter h by the 2nd component of the
!-- vector of variables
!-- increases, so we make parameter h an independent variable
!-- and we return to the integration by parameter h with step SN
!-------------------------------------------------
!--  A decision is made to complete the integration
 140 continue
       if(Y0(NK+2).ge.HM) goto 250
        if(KER(6).EQ.1.and.IJ.EQ.0) goto 230
        if(Y0(NK+2).lt.0.) goto 210
!-- exit the procedure if parameter h exceeds HM,
!-- or there was a task to stop the calculation at
!-- passing the first special point (KER(6)=1, h decreases),
!-- or parameter h became negative
          If(NN.eq.1.or.ID.eq.1) goto 40
          S=S*2.5;   goto 40
!-- We continue to calculate the h-characteristics (go to label 40):
! with the same step (if there was step splitting in the previous step
! (NN=1) or the accuracy of the stepwise integration is satisfactory
! (ID=1). Otherwise, the step increases.
!---------------------------------------------------
 210 write(1,220)
 220 format(5X,'Parameter h has moved to the negative region')
       write(*,*) 'Parameter h became negative'
        return
 230 write(1,240)
 240 format(10X,'The first special point of the h-characteristics has been passed,' /10X, 'and therefore
     &   a stop is provided (KER(6)=1)')
          write(*,*) 'Passed first special point'
 250  return
!------------
           Contains
            Subroutine NEWT(KI,EPS,NIT)
!---------------------------
!-- The NEWT internal routine implements Newton's algorithm
!-- solution of a nonlinear system of finite equations
!-------------------------------------------------
```

```
            integer,intent(in)::KI
            real,intent(in)::EPS
            integer::NIT
        NIT=0
    1 Call CALCULU(KER,KI,INEV,AL,Y1,F1,U,NK,K,NG,NG1,M,MK,E,KH)
!-- Here, CALCULU determines the vector of corrections F1
!-- by the value of the vector Y1
!-----------------------
        E1=0.; E2=0.; NIT=NIT+1;  Y1=Y1-F1
                  do i=1,NG
                     E1=E1+Y1(i)**2; E2=E2+F1(i)**2
                   end do
        E1=sqrt(E1)*EPS;  E2=sqrt(E2)
        If(NIT.gt.20) goto 2
         If(E2.gt.E1) goto 1
!----   EPS accuracy is achieved
!--------------------------
            return
   2  write(1,3)
   3  format(10X,'The number of iterations during refinement',/10X,'of the root at the point h=H1
exceeded 20')
     write(*,*) 'Looping during refinement by Newton method'
            stop
            end subroutine NEWT
    end subroutine HARMOSC
```

The ***HARMOSC*** procedure is an improved version of the ***HINVNEWT*** procedure described in [20], it has become more universal and provides:

- modeling (calculation) not only of forced oscillations, but also of parametric oscillations and self-oscillations;
- determination of all periodic solutions of a nonlinear system of differential equations, if there is more than one of them;
- the possibility of modeling in the presence of elements with nonlinear hysteresis characteristics;
- the possibility of modeling by increasing in the process of calculating the number of harmonics taken into account to determine their required number for the sake of modeling accuracy.

The ***HARMOSC*** procedure is called for execution by the main program of the user software component package. It implements the algorithm for solving a nonlinear system of finite equations, which is a harmonic representation of a system of differential equations, the periodic solution of which is sought.

All parameters of the procedure are input, and before the procedure is called, they must receive a value according to its interface defined by the operator

```
Subroutine HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER).
```

The first formal parameter is the variable ***K*** of integer type, it must be given the value of the order of the system of differential equations, the periodic solution of which is sought, this is the value of the variable k of the last component of the vector of the form (1.68).

The second formal parameter is a one-dimensional real array of variables ***Y0***, formed from simple k vectors of amplitudes, the circular frequency ω of the fundamental harmonic, and the parameter h. The value given to it is the initial value of the variable array ***Y0***. When $h = 0$ modeling forced oscillations, simple amplitude vectors are zeroed (because the $h-$ characteristic starts with zero harmonic amplitude values) and the value is set to ω. When modeling self-oscillations or parametric oscillations, it is necessary to set initial values to simple amplitude and frequency ω vectors, the values of which can be calculated using one of the approximate solution methods, for example, the harmonic linearization method, etc.

The third formal parameter is the composite vector of amplitudes ***E*** of the forcing force. It is given value only when calculating forced oscillations. For the case of parametric oscillations and self-oscillations, this vector is zeroed by the ***HARMOSC*** procedure.

The fourth formal parameter is the ***NK*** variable of the integer type, it must be assigned the value of the size of the composite vector of amplitudes of the form (1.79) - (1.82).

The fifth formal parameter is a real ***HM*** variable, which must be assigned the maximum value of the parameter h to which the h-characteristics must be calculated.

The sixth formal parameter is the real variable H1, which must be assigned the value of the parameter h at which the solution must be refined by Newton's iterative method (when modeling self-oscillations and parametric oscillations, the parameters ***HM*** and ***H1*** must be set to the same value 1.0).

The seventh formal parameter is the real variable ***EPS1***, which must be given the value of the relative accuracy of the calculation h-characteristic.

The eighth formal parameter is the real variable ***EPS2***, which must be assigned the value of the relative accuracy to which the refinement of the solution by Newton's iterative method must be performed.

The ninth formal parameter is an integer control vector ***KER*** , which has 10 elements. They have the following content.

KER(1) – this variable is used to set the spectrum of a simple vector of amplitudes: if it includes a constant component and harmonics of both even and odd orders, then this variable must be assigned the number 0; if the simple vector of amplitudes is formed by the amplitudes of harmonics of odd orders only, then this variable must be assigned the number 1. It should be borne in mind that in those cases when it is not known for sure which of these two types of spectra is expected even before the calculation, it is better to set the number 0, and if only odd harmonics are present in the oscillatory process, then in the solution the relative amplitudes of other harmonics and constant components will be zero (close to zero). This will give reason to repeat the calculation by setting the variable ***KER(1)*** to 1.

KER(2) and ***KER(3)*** – these two variables are used to determine which form of the four predicted variants (1.64) – (1.67) the system of differential equations, the periodic solution of which is sought, has. If it fits into option (1.65) or (1.67), that is, when the variables whose periodic dependences are sought are directly under the derivative signs, then the variable ***KER(2)*** must be assigned the number 1, and if there are other variables under the derivative signs that are nonlinear functions from those whose periodic dependences are sought, then the number 0 is assigned. If the system of differential equations fits into variant (1.64) or (1.65), that is, it is solved with respect to the derivatives (in the normal Cauchy form), then the variable ***KER(3)*** must be assigned the number 0, and the number 1 if the derivative vector is preceded by a square

matrix of coefficients (the system of differential equations is not solved with respect to the derivatives). The value of the variables ***KER(2)*** and ***KER(3)*** can also be determined according to the following table:

A type of differential equation	***KER(2)***	***(KER(3)***
(1-64)	0	0
(1-65)	1	0
(1-66)	0	1
(1-67)	1	1

KER(4) – the value of this variable determines what kind of oscillations are sought: forced oscillations – the number 0; parametric oscillations – number 1; self-oscillation - number 2.

KER(5) – this variable controls the memorization of data during calculation h-characteristics: if it is set to 0, only the results that will be remembered (written to the output file whose name is specified in the main user program) correspond to , and if you set 1, then the results will be remembered at each point *h*-characteristics. In some cases, it is advisable to remember the entire *h*-characteristic, because it is the one that is of interest (see further section 4.1.3 – calculation of the characteristic of the ferroresonant circuit). It may also be useful to remember it in the case of an abnormal completion of the calculation and the need to analyze its causes.

KER(6) – this variable determines whether the calculation of characteristics should be continued if its first special point has already been passed (see section 1.5.7): if this variable is assigned the number 0, then after passing the first special point, the calculations will continue.

KER(7) – this variable specifies the features of the calculation when the problem has nonlinear hysteresis characteristics. If this variable has a value of 0, it means that the problem does not have such characteristics. If this variable has the value of a positive integer (for example, 5), then this will mean that the problem has such characteristics, and this integer determines the number of steps for expanding the hysteresis loops when refining the solution from zero area to the real one.

KER(8) – the value of this variable determines the number (order) of the highest harmonic taken into account, this is the value in formula (1.8).

KER(9) – this variable specifies the features of the calculation in which the number of harmonics taken into account is increased. If ***KER(9)*** is assigned the number 0, then the increment of this quantity does not occur, and if it is assigned a number other than zero and greater than that assigned to the variable ***KER(8)***, then the increment occurs from the value of the variable ***KER(8)*** to the value of the variable ***KER(9)*** with step 1.

KER(10) is a variable that controls the recording of results to the output file when increasing the number of harmonics taken into account. If its value is 0, then only the results are recorded when the harmonic number reaches the value of the ***KER(9)*** variable, and if its value is 1, then the results are recorded after each build-up.

The procedure implements the algorithm for solving a nonlinear system of finite equations of the form (1.77) or (1.83) or (1.84) or (1.85), which is a harmonic representation of a system of differential equations whose periodic solution is sought.

Note that the limitation of the forms of recording of the systems of differential equations under consideration to the forms (1.64) - (1.67) relieves the user of this procedure of the need

to perform the harmonic algebraization operation, because it is embedded in the algorithm implemented by the procedure.

Let us briefly describe this algorithm.

First of all, based on the given initial value of the vector of variables $\vec{X}^*$ of the form (1.89), which is stored in the one-dimensional array ***Y0,*** the value of the vector of residua (1.97) is calculated - it is calculated by the ***CALCULU*** procedure called for execution (it is described further), the value of the vector of residua is memorized by a one-dimensional ***U*** array.

Next is the calculation of the *h*-characteristic - the integration of the vector differential equation of the form (1.102) is performed using the Kutt-Merson numerical method [48] with automatic selection of the step size. At the same time, the step size is changed so that the relative accuracy of the ***EPS1*** calculation is satisfied. Integration is performed when the parameter changes from zero to the value that the ***HM*** variable has.

Before starting the Kutt-Merson algorithm, this procedure includes the first two small steps with the step length value $\Delta h = 0{,}0005$, which are performed according to the Euler method without accuracy control. This is due to the fact that in some tasks, in particular when calculating periodic processes in nonlinear electric circuits with valves, at the first step of the calculation *h*-characteristics, when it is determined at which nodes in the period the valves are open and at which they are closed, the Kutt-Merson algorithm with accuracy control without such an introduction can lead to unjustified splitting of the step.

When calculating parametric oscillations or self-oscillation, the value of the variables ***H1*** and ***HM*** is set, as already shown, to be the same and equal to one, then the last point of the *h*-characteristic is specified according to Newton's iterative method [44, 48]. When calculating the forced oscillations, the values of the changed ***H1*** and ***HM*** can also be the same and have equal units, then the solution is refined at the full value of the forcing force. However, they can be different, while the value of the variable ***H1*** must be smaller than the value of the variable ***HM***. For example, if the ***H1*** parameter is assigned a value of 0.4 and the ***HM*** parameter is assigned a value of 0.9, then the procedure will calculate the *h*-characteristic in the range from $h = 0$ to $h = 0.9$ and at $h = 0.4$ specify the solution (or solutions, if there is more than one in this range) according to Newton's method.

Once again, we emphasize the possibilities of the procedure.

1) The algorithm of the ***HARMOSC*** procedure provides for the possibility of inverting the solvable system of differential equations (1.102) during the calculation of forced oscillations in order to ensure the passage of special *h*-characteristic points, if any, during numerical integration. The inversion algorithm is described in section 1.5.7. When inverting, the independent variable – the parameter *h* – becomes the dependent variable, and the independent variable becomes the second component of the vector of variables $\vec{X}^*$.

2) The ***HARMOSC*** procedure assumes the possible presence of nonlinear hysteresis characteristics in the problem. If they are present (while the ***KER(7)*** element is non-zero), the solution is first obtained assuming that all characteristics of the hysteresis form are replaced by single-valued averaged characteristics that pass through the middle of the hysteresis loops. After obtaining this initial solution, the hysteresis loops are gradually expanded to their full values - the number of expansion steps is given by ***KER(7)***. After each expansion, the solution is refined using Newton's method, while the value of the root of the previous stage of expansion is taken as the zero approximation.

3) The ***HARMOSC*** procedure also provides for the possibility of refining the obtained solution by increasing the number of harmonics taken into account. If this option is selected, the value of the ***KER(9)*** element must be non-zero. Then the value of ***KER(9)*** is perceived as the maximum value of the number of the highest harmonic when increasing the number of harmonics taken into account (before the number of harmonics taken into account was increased, a solution was obtained in which the number of the highest harmonic was given by the value of the ***KER(8)*** element).

During its work, the ***HARMOSC*** procedure invokes four external procedures: ***SNCS*** (calculates the values of the matrices of harmonic transformations), ***CALCULU*** (calculates the values of the vector of discontinuities, derivatives or corrections, described later), ***IMPROVE*** (implements the algorithm for increasing the number of harmonics taken into account, described later) and ***OUTP*** (processes the results and writes them to a separate file for subsequent printing) and one internal procedure ***NEWT***, which implements Newton's iterative algorithm for refining the solution.

2.2.3.2. Procedure CALCULU

Text of the procedure:

```
      Subroutine CALCULU(KER,KI,INEV,AL,X,DX,U,NK,K,NG,NG1,M,MK,E,KH)
!--  CALCULU procedure:
!--   at INEV=1 by the value of vector X with size NK+2 determines  the vector of discontinuities and assigns
!--    its value to the vector DX;
!--  at INEV=0 and KI=0 calculates the value of the vector X  vector of derivatives DX with size NK+2 and
!--     divide all components  of this vector to its KH-th component;
!--   at INEV=0 and KI=1 by the value of the vector X determines vector of corrections DX for refinement of
!--       the solution by Newton's method.
!-------------------------------------------------
            Implicit none
            integer,intent(in)::KI,INEV,K,NG,NG1,M,NK,MK,KH
            real,intent(in)::AL
            integer,dimension(10),intent(in)::KER
            real,dimension(NK+2),intent(in)::X
            real,dimension(NK),intent(in)::E
            real,dimension(NK+1)::U,Y1
            real,dimension(NK+2),intent(out)::DX
            real,dimension(NK)::XG,Y,Z,Y11,Y2
            real,dimension(NK,NK)::SY,SZ,SY1,SY2
            real,dimension(NK+1,NK+2)::A
            real,dimension(MK)::XC,YC,ZC
            real,dimension(MK,K)::ZXC,YXC
            real,dimension(K,K)::B
            real::CC,OM,H
            integer::i,j,IG,INFK,IA
!--------------------------------
!--   Procedure parameters:
!--   KER - an array of control variables
!--   KI,INEV - control variables
!--   AL - the narrowing coefficient of the characteristic loop of the hysteresis form
!--   X  - a vector of variables
!--   DX - vector of increments or corrections
!--   U  - a non-coherent vector for the initial approximation of X
!--   NK - the size of the composite amplitude vectors;
!--  K  -  the order of the system of differential equations being solved;
```

```
!--  NG - size of simple amplitude vectors;
!--  NG1 - the number of amplitudes to the 2nd harmonic;
!--  M - size of simple nodal vectors (number of nodes per period or half-period);
!--  MK - the size of the composite nodal vectors;
!--  E  - a composite vector of forcing force amplitudes;
!--  KH - the number of the independent variable in the vector of variables
!---------------------------
            IG=KER(1); INFK=KER(2) ;IA=KER(3); OM=X(NK+1)
!------ OM - circular frequency of the fundamental harmonic
             do i=1,NK
               XG(i)=X(i)
             end do              !-- XG - composite vector of amplitudes
     call KVGVS(K,XG,NG,NK,XC,M,MK)
!--- calculated the value of the composite nodal vector XC
            call Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,B)
!--- calculated composite nodal vectors YC,ZC and matrices YXC,ZXC
         if(INFK.eq.1) goto 50
             call GRMAT(IG,SY,NK,YXC,MK,K,M,NG,NG1)
  50 Call GRMAT(IG,SZ,NK,ZXC,MK,K,M,NG,NG1)
!--  calculated the matrices of differential harmonic parameters SY, SZ
            if(INEV.eq.1) goto 70
            A=0.;         !-- zeroed the matrix A
       if(INFK.ne.1) goto 60
       SY=0.
         do i=1,NK
           SY(i,i)=1.
         end do
  60 call OMAB(IG,SY,SY1,NG,K,NK,OM)
             if(IA.ne.1) goto 61
             SY2=SY1;        call MBDMM(B,K,SY2,SY1,NG,NK)
  61   do i=1,NK
           do j=1,NK
             A(i,j)=SY1(i,j)+SZ(i,j)
           end do
         end do
!---   the main block of matrix A is formed
!-----------------------------------------
!---   we calculate the gaps
  70 if(INFK.eq.1) goto 71
             call KVSVG(K,YC,M,MK,Y,NG,NK)
  71 call KVSVG(K,ZC,M,MK,Z,NG,NK)
!--- calculated the values of the vectors of amplitudes Y and Z
              Y11=Y;    if(INFK.eq.1) Y11=X
!--- calculated values of amplitude vectors:
!------ Y11 is a composite vector of amplitudes Yr or Xr
!------ Z is the composite vector of amplitudes Zr
         call OMVB(IG,Y11,Y,NG,K,NK,OM)
!------ Y is the product OM*D*Y1
            if(IA.ne.1) goto 72
            Y11=Y;  call MBDMV(B,K,Y11,Y,NG,NK)
  72 continue
!------ Y is the product OM*D*Y1 or Br*OM*D*Y1
        H=X(NK+2);   if(INEV.eq.1) H=1.
          do i=1,NK
                Y1(i)=Y(i)+Z(i)-E(i)*H
```

```
                Y2(i)=Y(i)/OM
        end do
           Y1(NK+1)=0.; if(KER(4).eq.2) Y1(NK+1)=X(2)
!------ Y1 is a vector of residua
!------ Y2 is a vector D*Y1
     if(INEV.ne.1) goto 800
           do i=1,NK+1
              DX(i)=Y1(i)
           end do
          return
!------ calculated the residua
 800 do i=1,NK
         A(i,NK+1)=Y2(i)
       end do
           if(KER(4).eq.2) goto 801
           A(NK+1,NK+1)=1.; goto 802
 801 A(NK+1,2)=1.
 802 if(KI.eq.1) goto 803
!--- to calculate increments
      do i=1,NK+1
           A(i,NK+2)=-U(i)
      end do;  goto 804
!--- to calculate corrections
 803 do i=1,NK+1
        A(i,NK+2)=Y1(i)
       end do
 804 continue
      call SYS(A,NK+1,NK+2)
!--  a system of linear equations is solved
     do i=1,NK+1
       DX(i)=A(i,NK+2)
     end do
    If(KI.eq.1) goto 44
          DX(NK+2)=1.
!--- divide by the KH-th component (inversion):
     CC=DX(KH);  DX=DX/CC
      return
  44 DX(NK+2)=0.
      return
          Contains
           Subroutine MBDMV(B,K,X,Y,N,KN)
! Multiplication procedure block matrix
! of the form (1.86) on the composite vector of amplitudes X
!-----------------------
           integer,intent(in)::K,N,KN
           real,dimension(K,K),intent(in)::B
           real,dimension(KN),intent(in)::X
           real,dimension(KN),intent(out)::Y
           integer::i,j,L,iNL,LjK
!---------------
!-- Input values:
!-- B(K,K) is a square matrix of coefficients
!-- X(KN) is a composite vector of amplitudes
!-- Output value:
```

```
!-- Y(KN) is a composite vector of amplitudes - the result
!----------------------------------
            do i=1,K
             do L=1,N
               iNL=(i-1)*N+L; Y(iNL)=0.
                 do j=1,K
                   LjK=L+(j-1)*N
                   Y(iNL)=Y(iNL)+B(i,j)*X(LjK)
                 end do
             end do
            end do
            return
            end Subroutine MBDMV
!---------
            Subroutine MBDMM(B,K,X,Y,N,KN)
! Multiplication procedure block matrix
! of the form (1.86) - on the composite matrix X
!----------------------
            integer,intent(in)::K,N,KN
            real,dimension(K,K),intent(in)::B
            real,dimension(KN,KN),intent(in)::X
            real,dimension(KN,KN),intent(out)::Y
            integer::i,ig,j,L,Lg,iNL,jNL,LjK
!---------------
! Input values:
! B(K,K) is a square matrix of coefficients
! X(KN,KN) is a composite block matrix of K*K blocks,
! each block is a matrix of dimension N
! Output value:
! Y(KN,KN) - matrix - result
!----------------------------------
           do ig=1,K
            do i=1,K
              do Lg=1,N
               jNL=(ig-1)*N+Lg
                do L=1,N
                 iNL=(i-1)*N+L; Y(iNL,jNL)=0.
                   do j=1,K
                   LjK=L+(j-1)*N
                   Y(iNL,jNL)=Y(iNL,jNL)+B(i,j)*X(LjK,jNL)
                   end do
                 end do
               end do
             end do
            end do
            return
            end Subroutine MBDMM
        end Subroutine CALCULU
```

The ***CALCULU*** procedure is called to execute the ***HARMOSC*** procedure described in the previous section and the ***IMPROVE*** procedure described later.

The ***CALCULU*** procedure implements the algorithms 1.6 and 1.7 described in section 1.5.5 - based on the given value of the ***X*** vector (composite vector of the amplitudes of the

unknown $\vec{x}$ and the circular frequency ω) using the instantaneous process model on the period (or half-period) calculates:

1) the value of the vector of residual (1.97), if the control variable ***INEV*** has a value of 1, or

2) the value of the derivatives according to the parameter h of the unknown components of the vector at the step of numerical integration, if the control variables ***INEV*** and ***KI*** have the value 0, or

3) the value of corrections to refine the solution according to Newton's method, if the control variable ***INEV*** has a value of 0 and the control variable ***KI*** has a value of 1.

This procedure is focused on systems of differential equations exclusively in the writing forms (1.64) - (1.67).

The ***CALCULU*** procedure when executed calls the ***KVGVS, KVSVG, OMAB, OMVB*** and ***GRMAT*** procedures described in the previous sections from the corresponding block of standard software components of the DHM-S, the ***SYS*** procedure for solving systems of linear equations (it is given below in this chapter) and the ***Model*** procedure from block of user software components (it specifies an instantaneous process model for a period or a half-period).

If the procedure works according to the second option, then the values of the derivatives of the components of the unknown vector calculated at the integration step are divided by the derivative component of the variable that is the independent variable at this step. This is how the inversion of the system of differential equations described in section 1.5.7 is carried out.

The **CALCULU** procedure contains two internal subroutines - the ***MBDMV*** subroutine, which performs the operation of multiplying a block matrix of the form (1.86) by a composite vector of amplitudes - see formulas (1.84) and (1.85), and the ***MBDMM*** subroutine, which performs the operation of multiplying a block matrix of the form (1.86) by a composite matrix - see formulas (1.110) and (1.111). The routines ***MBDMV*** and ***MBDMM*** are called for execution only when the control variable ***KER(3)*** has the value 1, that is, when a periodic solution (or periodic solutions) of the differential equations of the form (1.66) or (1.67) is sought.

At the end of the description of the ***CALCULU*** procedure, it is necessary to pay attention once again that it is focused on the system of differential equations exclusively in one of the forms of writing (1.64) - (1.67). Harmonic algebraization of differential equations of these forms is already embedded in it. That is why when using this procedure, the volume of preparatory work of the user is minimal - he only needs to develop his ***Model*** procedure for a specific type of system of differential equations. If this system of equations does not fit into any of the forms of writing (1.64) - (1.67), then the user must develop his own ***CALCULU*** procedure, having previously performed the harmonic algebraization of the system of differential equations, the periodic solution of which is sought, and then this procedure from Block 3 DHM-S is transferred to the block of user software components.

The need to develop a variant of the ***CALCULU*** procedure may also arise when the user wants to use some additional capabilities of the method, for example, taking into account the symmetry of a periodic process in a multiphase electrical circuit [15, 17, 20] or the same symmetry in other oscillations, if they are there, - to reduce the number of unknown harmonic amplitude vectors in order to minimize the required amount of computing resources.

2.2.3.3. Procedure IMPROVE

In order for the trigonometric series of the form (1.8) reproduced by the found vector $\vec{X}^*$ to approximate the time dependence of the variables in the periodic solution of the nonlinear system of differential equations with the necessary accuracy, it is necessary to take into account the appropriate number of harmonics. For the most part, in engineering calculations of nonlinear oscillations in systems and devices, this number of harmonics is not too large.

Practically, establishing the required dimension of the amplitude vectors (in other words, the number of harmonics that must be taken into account) can be carried out by means of a numerical experiment, increasing the number of harmonics taken into account. This increase can be done in different ways, for example, as follows.

The first approximation for the iterative process according to scheme (1.112), calculated by the h-characteristics method, should be sought when taking into account a small number of harmonics. So, let n = 3 be taken to obtain the first approximation of the sought-after periodic solution of the system of differential equations, and after refinement according to the scheme (1.112), the vector of amplitudes $\vec{X}_\Gamma$ corresponding to one of the variables, for a given value n , has the form

$$\vec{X}_\Gamma^{(3)} = colon(X_0^{(3)}, X_{c1}^{(3)}, X_{s1}^{(3)}, X_{c2}^{(3)}, X_{s2}^{(3)}, X_{c3}^{(3)}, X_{s3}^{(3)}). \qquad (2.6)$$

We increase the number of harmonics taken into account by one, that is, we take n = 4, and as a first approximation for the iterative scheme (1.112) we take the value of the vector of amplitudes

$$\vec{X}_\Gamma^{(4)} = colon(X_0^{(3)}, X_{c1}^{(3)}, X_{s1}^{(3)}, X_{c2}^{(3)}, X_{s2}^{(3)}, X_{c3}^{(3)}, X_{s3}^{(3)}, 0, 0) , \qquad (2.7)$$

which has two more components and the last two added components have zero values. After the appropriate number of iterations of the solution refinement algorithm by Newton's method, the last two components receive numerical values and the other components slightly change their values.

After that, we again increase the number of components of the amplitude vector $\vec{X}_\Gamma$ by two components and specify the value of the vector. We continue to increase the number of harmonics that are taken into account in this way until the predetermined maximum number of taken into account harmonics is reached.

This algorithm for increasing the number of considered harmonics is implemented by the ***IMPROVE*** procedure, the text of which is given below.

```
      Subroutine IMPROVE(KER,K,KH,Y1,N11,NGP,E1,U,Y2,N12,EPS,AL)
!--   The procedure for increasing the number of considered harmonics
!--   in the sought-after periodic solution of a nonlinear system of differential equations
!-----------------------
      Implicit none
            integer,intent(in)::K,NGP,N11,N12,KH
            integer,dimension(10),intent(in)::KER
            real,intent(in)::EPS,AL
```

```
            real,dimension(K*NGP),intent(in)::E1
            real,dimension(N11),intent(in)::Y1,U
            real,dimension(N12),intent(out)::Y2
            real,dimension(N12)::Y20
            real,dimension(N12-2)::E2,E20
            real,dimension(N12)::F2
            real::S1,S2,H1,OM
            integer::M,MK,i,j,NIT,NB,NBP,NG,NG1,NG0,NK,NKR,IG,N1,KK
!-------------------
!--   Procedure parameters:
!--   KER is an array of control variables, in it:
!--   KER(8) - the number of the highest harmonic before building up;
!--   KER(9) - the number of the highest harmonic to follow build harmonics;
!--   K - the order of the solvable system of differential equations;
!--   KH - the number of the independent variable in the vector of variables
!--  Y1 - a composite vector of the amplitudes of the solution to build-up
!--   N11 - the number of elements of the vector Y1
!--   NGP - the initial value of the size of the simple vector of amplitudes;
!--   E1 - composite vector of forcing force amplitudes;
!--   U - a vector of entanglements
!--   Y2 - the original (refined) composite vector of solution amplitudes;
!--   N12 - the number of elements of the vector Y2
!--   EPS - relative precision for refinement
!--   AL - narrowing coefficient of the hysteresis loop
!-------------
       write(1,110) KER(8),KER(9)
  110 Format(1X,'Refinement of the value of the root by increasing the number of harmonics from
N=',I2,' to N=',I2)
           NG0=NGP;  IG=KER(1)
             Y20=0.;   E20=0.
            do i=1,N11
              Y20(i)=Y1(i)
            end do
            do i=1,N11-2
             E20(i)=E1(i)
            end do
            H1=Y1(N11);  OM=Y1(N11-1)
            NKR=1;  if(IG.eq.1)NKR=2
            NBP=KER(8)+1;   if(IG.eq.1)NBP=KER(8)+2
!--   a cycle in which the number of considered harmonics is increased
     do NB=NBP,KER(9),NKR
             call SNCS(IG,NB,NG,NG1,M)
!-- calculated new values ??of matrices of harmonic transformations
             NK=K*NG; N1=NK+2; MK=M*K
             Y2=0.; E2=0.
             do i=1,K
               do j=1,NG0
                 Y2((i-1)*NG+j)=Y20((i-1)*NG0+j)
                 E2((i-1)*NG+j)=E20((i-1)*NG0+j)
               end do
             end do
       Y2(N1)=H1;  Y2(N1-1)=OM;  NIT=0
  300  Call CALCULU(KER,1,0,AL,Y2,F2,U,NK,K,NG,NG1,M,MK,E2,KH)
!-- The CALCULU procedure determines the value of the Y2 vector
```

```
!--   vector of corrections F2 for refinement of the solution according to Newton's method
       S1=0.; S2=0.
       NIT=NIT+1;
            do i=1,N1
               Y2(i)=Y2(i)-F2(i)
            end do
           S1=Y2(NG-1)**2+Y2(NG)**2
           S2=F2(NG-1)**2+F2(NG)**2
         S1=sqrt(S1)*EPS;  S2=sqrt(S2)
         If(NIT.gt.20) goto 320
!-- if looping, then emergency exit (at mark 320)
         If(S2.gt.S1) goto 300
!-- if the accuracy is worse than EPS, then go to the next iteration
!-- (at mark мітку 300)
            do j=1,N1
              Y20(j)=Y2(j)
            end do;
            do j=1,NK
              E20(j)=E2(J)
            end do
           NG0=NG
           if(KER(10).eq.0.and.NB.ne.KER(9)) goto 310
           write(1,305)NB
 305 format(/1X,'added ',i2,'harmonic')
           KK=1; if(NB.eq.KER(9))KK=0
           call OUTP(KER(1),Y2,N1,K,NG,M,MK,KK)
 310 continue
           end do
!-- the end of the cycle of increasing the number of harmonics
            return
 320 write(1,321)NB
 321 format(10X,'Number of iterations when joining', I2,'-th harmonic exceeded 20')
      write(*,*)'looping in the IMPROVE procedure'
      stop
     end  subroutine IMPROVE
```

The formal parameters of the procedure are described in the comments.

The procedure ***IMPROVE*** when executed calls the procedures ***SNCS*** (called after each change of value n), ***CALCULU*** and ***OUTP***.

* * *

It is possible to increase the number of harmonics considered in larger increments than what is included in the ***IMPROVE*** procedure, for example two at a time.

The number of harmonics taken into account can also be increased in the direction of decreasing numbers of their orders, that is, subharmonics can also be taken into account. So, if we consider the first subharmonic with circular frequency $\omega/2$, then it should be considered as a new first, while setting at the levels (1.77), (1.83), (1.84) or (1.85) a new value of the circular frequency equal to $\omega/2$,, and at the same time former first harmonics of all variable values and forcing forces should be called other harmonics, second thirds, etc.

It should be noted that when increasing the number of taken into account harmonics of variable values, it is important not to fall out those harmonics on which the phenomenon of

resonance may occur. Ignoring resonant harmonics can significantly reduce the accuracy of the obtained periodic solution.

The remarks stated above after the asterisks are not implemented in the ***IMPROVE*** procedure. If the user wants to modify this procedure, he can use the mentioned comments.

2.2.4. The fourth program block

To this block (Block 4 in the DHM-S) we will assign procedures that implement operations that are not operations of the differential harmonic method, but are general and can be used in other tasks. In this sense, the block, although included in the DHM-S, actually does not belong to it.

To begin with, we will introduce two procedures to this block: ***SYS*** for solving systems of linear algebraic equations and ***INTLIN*** for linear interpolation from tables that specify hysteresis-free magnetization curves of ferromagnetic materials.

It is suggested that the user in his work also refers to this block his own procedures of a general nature and application.

2.2.4.1. Procedure SYS

Text of the procedure:

```
      Subroutine SYS(B,KY,KV)
!--   Procedure for solving a system of linear equations
!--   by the Gaussian method with the selection of the main element.
!--   B - extended matrix of coefficients with dimensions KY*KV.
!--   The result is located in the far right column in place of free members
!-----------------------------------------
          integer,intent(in)::KY,KV
          real,dimension(KY,KV)::B
          real::C1,C4
          integer::L,i,j,K,L1
    do L=1,KY
      C1=0.0
     do i=L,KY
       C4=ABS(B(i,L))
       if(C4.GT.C1) goto 2
       goto 3;
  2     K=i;  C1=C4
  3   end do
         do j=L,KV
          C1=B(K,j); B(K,j)=B(L,j); B(L,j)=C1
         end do
     K=L+1
     do j=K,KV
       B(L,j)=B(L,j)/B(L,L)
     end do
      if(K.GT.KY) goto 7
          do i=K,KY
              do j=K,KV
            B(i,j)=B(i,j)-B(i,L)*B(L,j)
```

```
            end do
          end do
        end do
      7 do L=2,KY
          j=KY-L+2; K=j-1
          do L1=1,K
            i=K-L1+1
            B(i,KV)=B(i,KV)-B(i,j)*B(j,KV)
          end do
        end do
        return
        end subroutine SYS
```

The ***SYS*** procedure implements the well-known algorithm for solving a system of linear equations according to the Gauss scheme with the selection of the main element [48].

This procedure could not be included in the DHM-S, and in programs for determining periodic solutions of nonlinear differential equations, a similar procedure from a package of standard subroutines of one or another library could be used. Its presence in the DHM-S assumes the case when such a package of standard routines is unavailable to the user for one reason or another. The presence of this procedure in Block 4 increases the autonomy of the DHM-S.

Before applying this procedure, all its formal parameters must be given values, in particular, parameter ***B*** - the value of the extended matrix of coefficients (free terms of the equations - in the far right column), ***KY*** - the order of the system (the number of scalar equations in the system), ***KV*** - the number of columns matrix ***B (KV=KY+1)***. The procedure places the solution (the value of the unknown system of linear equations) in the far right column, in the places of the free terms (the right-hand sides of the equations).

2.2.4.2. Procedure for linear interpolation magneti curve

Text of the procedure:

```
    Subroutine INTLIN(X,Y,YX,X1,DX,XT,M)
!-- The procedure of linear interpolation of the magnetization curve
!-- X - abscissa; Y - ordinate; YX - a derivative
!-- XT(M) - the table for the non-linear part
!-- X1 - the initial abscissa of the non-linear part
!-- DX - table step
!------------------------
          Implicit none
          integer,intent(in)::M
          real,intent(in)::X,X1,DX
          real,dimension(M),intent(in)::XT
          real,intent(out)::Y,YX
          integer::j
          real::AX,ZX,XM
    ZX=sign(1.,X); AX=abs(X)
    If(AX.GT.X1) goto 1
!   Initial linear part
    YX=XT(1)/X1; Y=ZX*YX*AX
     return
```

```
   1      XM=X1+(M-1)*DX
          if(AX.ge.XM)goto 2
!--------------------
!  The non-linear part
          j=(AX-X1)/DX+1
          YX=(XT(j+1)-XT(j))/DX
          Y=ZX*(XT(j)+YX*(AX-(X1+(j-1)*DX)))
          return
!-----------------------
!  The final linear part
   2 YX=(XT(M)-XT(M-1))/DX
     Y=ZX*(XT(M)+YX*(AX-XM))
     return
     end subroutine INTLIN
```

This procedure is intended for interpolation from the table, which specifies the hysteresis-free characteristic of magnetization - for example, the dependence of the magnetic field induction in a ferromagnet on its stress or the dependence of the flux-coupling of a coil with a ferromagnetic core on its current. It is assumed that this curve is set only for positive values of the abscissa (the curve is symmetric odd) and is divided into three parts: the initial linear, which passes through the origin, the curvilinear (knee) and the final linear (after the saturation knee). With such a breakdown, only the nonlinear part is numerically displayed: it is necessary to set the value of the abscissa (voltage or current) of the beginning of the nonlinear part - ***X1***, the step between nodes (nodes equidistant) - ***DX***, the number of table nodes - ***M*** and the table of ordinate values (induction or flux linkage) in the nodes - ***XT***. At the same time, the first node of the table is the junction point of the initial linear part and the non-linear part (knee), the penultimate node is the junction point of the non-linear part with the final linear part; the last node also lies on the terminal line segment.

The algorithm by which the procedure works is as follows.

If the given value of the abscissa ***X*** (by absolute value) is smaller than ***X1***, then the extrapolation is carried out along a straight line passing through the origin and the first node of the nonlinear part of the magnetization curve. If the given value of the abscissa ***X*** goes beyond the non-linear part to the right, then the extrapolation is carried out along a straight line drawn through the last two nodes of the table (the penultimate node completes the non-linear part and the last one lies on the linear part). If the given value of the abscissa ***X*** is within the nonlinear part of the magnetization curve, then the two nearest nodes are determined, a straight line is drawn through them, and interpolation is carried out behind it.

The found value of the ordinate Y is assigned the sign of the abscissa ***X.*** The value of the derivative is calculated as the tangent of the angle of inclination of the corresponding segment of the broken line approximating the magnetization curve, and its value is assigned to the formal parameter ***YX***

.

Chapter 3
METHODOLOGY OF NUMERICAL MODELING OF NONLINEAR OSCILLATIONS

In this chapter, we will consider the method of creating a numerical model of nonlinear oscillation using the theoretical provisions of Chapter 1 and the DHM-S described in Chapter 2.

First of all, let us emphasize that we will be talking about models that correspond to systems of differential equations exclusively of the forms (1.64) - (1.67).

Creation of a numerical model of nonlinear oscillations involves the following stages.

1. Writing the system of differential equations describing the oscillating system and reducing it to one of the forms (1.64) - (1.67).

2. Analysis of system nonlinearities from the point of view of the classification described further in section 3.2.1. If among the nonlinearities there are those belonging to the second and/or third groups, and they differ from those considered further in sections 3.2.1.1 and 3.2.1.2, then for these nonlinearities it is necessary to develop instantaneous models on a period (semi-period), for example those given in sections 3.2.1.1 and 3.2.1.2, and implement their programmatic implementation, and attach the developed procedures to Block 5 of the DHM-S.

3. Development of a block of user software components for this task (see Fig. 2.1). This block includes:

- the main program;
- the ***Model*** procedure, which implements an instant mathematical model of the process on a period (semi-period);
- the ***OUTP*** procedure, which implements the algorithm for processing the results and writing them to the output file.

Regarding the last procedure. If the user is satisfied with the level of processing of the results implemented by the standard ***OUTP*** procedure described below in section 3.3 (it is assigned to Block 5 of the DHM-S), then the user does not have to create his own version of this procedure. The need for its development arises when larger-scale processing of the results is required, for example, analysis of the stability of the obtained solution, construction of graphs and tables, etc.

3.1. Structure of the main program

The purpose of the main program is input of input data, their initial processing and organization of transmission to the ***HARMOSC*** and ***Model*** procedures and the call to execute the ***HARMOSC*** procedure.

In the descriptive part of the program, it is necessary to describe:

a) a real one-dimensional array that stores the value of the vector of variables of the form (1.89), and the last element of the array is the value of the parameter h;

b) a real one-dimensional array for the composite vector of forcing force amplitudes of the form (1.81);

c) a whole one-dimensional array for the control vector ***KER***, which has 10 elements;

d) a set of real and integer simple variables and arrays, which are necessary to transfer information to the ***HARMOSC*** and ***Model*** procedures.

The descriptive part ends with a description of the shared memory area, which should ensure the transfer of part of the data from the main program to the ***Model*** procedure.

The input data entered by the main program from the input file is divided into two parts:

1) data to be passed to the ***HARMOSC*** procedure (passed through formal parameters);

2) data that needs to be transferred to the Model procedure (this includes, among other things, information about non-linear relationships in the system) in ways other than the way of transfer through formal parameters. In all the following examples, this is a transfer using common memory areas ***(Common)***.

In the part of the program where the primary processing of the entered data is performed, it is necessary to call the ***SizesV*** procedure (described later in this section, included in Block 5 of the DHM-S), which determines the sizes of the simple and compound amplitude vectors, the values of which are necessary for the formation of the compound vector of amplitudes of the forcing force and the initial value of the vector of variables, and are added to the information to be passed to the ***HARMOSC*** procedure. There, an initial value $\vec{X}_0^*$ (see formula (1.97)) is assigned to the sought-after vector of variables (its components are the composite vector of amplitudes of the form (1.82), the circular frequency ω of the fundamental harmonic, and the parameter h) and a value is assigned to the composite vector of amplitudes $\vec{E}_\Gamma^*$ of the form (1.81) of the forcing force (in the case of forced oscillations).

In the final part of the main program, after all the formal parameters of the ***HARMOSC*** procedure have already been assigned the required values (see section 2.2.3.1 for a description of its formal parameters), this procedure is called for execution.

An example of the text of the main program:

```
      Program Main
      Implicit none
      real,dimension(14)::X
      real,dimension(12)::E
      integer,dimension(10)::KER
      real::A,B,C,OM,EPS1,EPS2,H1,HM
      integer::K,NG,NK
     common/MP/A,B,C     !--- shared area with the Model procedure
!--- Entering data from an input file
     open(1,File='DaniIn.dat',status='old')
     read(1,*)A,B,C,Ec
     read(1,*)OM,EPS1,EPS at whytch the root 2,H1,HM
!-----  A,B,C – data to pass to the Model procedure
!-----  Ec – the amplitude of the forcing force
!-----  OM – circular frequency
!-----  H1 – the value of the  h  parameter at which the root must be specified
!-----  HM – the maximum value of the parameter h
!-----  EPS1 – accuracy of h-characteristic calculation
!-----  EPS2 – accuracy for Newton's method
       read(1,*)KER    !---  KER – the array of control variables
       read(1,*)K      !---  K – the order of the system of differential equations
       close(1)
!--- Output of input data to the output file
       open(1,file='DataOutp.dat')
       write(1,1)
   1 format(4X,'Entered data:')
     write(1,2)A,B,C,Ec
   2 format(2X,' A = ',E10.4,' B =',E10.4, ' C = ',E10.4, ' Ec = ',E10.4)
```

```
      write(1,3)OM,EPS1,EPS2,H1,HM
    3 format(2X,' OM=',E10.4,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
      write(1,5)KER
      Write(1,4)K
    4 format(2X,' K=',i2)
    5 format(2X,10I5)
      call SizesV(KER(1),K,KER(8),NG,NK)
      write(1,3)NG,NK
!-----  NG – the size of the simple vector of amplitudes
!-----  NK – the size of the composite vector of amplitudes
      E=0.; E(NG+1)=Ec
      X=0;  X(NK+1)=OM
!---  formed the vector of amplitudes E of forcing fotces
!---  and the initial value of the vector X
     write(1,6)
    6 format(/2X,'C a l c u l a t I o n :')
      call HARMOSC(K,X,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
      end Program Main
```

In the descriptive part of the above main program:

- the real one-dimensional array, which stores the values of the vector of variables of the form (1.89) and the parameter h, is given the name ***X***;

- the real one-dimensional array for the composite vector of forcing force amplitudes is given the name ***E***;

- the whole one-dimensional array (control vector) is given the name ***KER***;

- the names ***A, B, C, OM, EPS1, EPS2, H1,HM, K, NG, NK*** are given to the real and integer simple variables with which data are passed to the ***HARMOSC*** and ***Model*** procedures.

The example of the main program discussed above can be used as a sample in the development of blocks of user software components in all cases of simulation of nonlinear oscillations. As a sample, it is used in all the tests and examples of Chapter 4.

Here is the text of the ***SizesV*** procedure, which is called for execution by the main program. This procedure is based on the given values of the formal parameter ***IG*** (if ***IG***= 0, then the constant components and all harmonics are taken into account, and if ***IG***=1, then only harmonics of odd orders are taken into account; the formal parameter ***IG*** corresponds to the actual parameter ***KER(4)***), the formal parameter ***K*** (the order of the system of differential equations, the periodic solution of which is sought) and the formal parameter ***N*** (the number of the highest considered harmonic, the actual parameter ***KER(8)*** corresponds to it) determines the values of ***NG*** (the number of elements of the simple vector of amplitudes) and ***NK*** (the number of elements of the complex vector of amplitudes).

The ***SizesV*** procedure is assigned to Block 5 in the DHM-S. Its text is as follows:

```
      Subroutine SizesV(IG,K,N,NG,NK)
!--  The procedure for determining the NK size of a composite vector
!--  the amplitude of the forcing force and the unknown vector
!--  and of the NG size of the simple vector of amplitudes
      implicit none
      integer,intent(in)::IG,K,N
      integer,intent(out)::NG,NK
!-----------------
!--  IG - if =0, then all harmonics and constant components are taken into account;
!--       if =1, then only odd harmonics are taken into account
!--  K - the order of the system of differential equations being solved
!--  N - the number of the highest harmonic taken into account
!-------------------------------------------
      NG=2*N+1; if(IG.eq.1)NG=N+1
      NK=K*NG
      return
      end subroutine SizesV
```

3.2. Programming of an instantaneous model of process on one period (semi-period)

When numerically modeling nonlinear oscillations (determining the periodic solution of a nonlinear system of differential levels), they are calculated on one period (or half-period, if only odd harmonics are present in the periodic dependences of the variables) containing m equidistant nodes. The values of the process variables at these nodes are interconnected by superimposed relationships, whether linear or non-linear. By the instantaneous mathematical model of a periodic process on a period (semi-period), we will understand the algorithm for determining the values of one variable at all nodes of a period (or half-period) based on the values of other variables, while the latter are arguments and the former are functions. In relation to equations of the form (1.64) – (1.67), the argument is a vector variable $\vec{x}$ and the functions are the vector variables $\vec{y}$ and $\vec{z}$. This mathematical model is implemented by the ***Model*** procedure, which is called for execution by the ***CALCULU*** procedure (see section 2.2.3.2).

Mostly, the values of the functions $\vec{y}$ and $\vec{z}$ can be determined by the ***Model*** procedure in each of the nodes of the period (semi-period), regardless of what these values are in other nodes. Then the order of traversing the nodes when calculating by the value $\vec{x}$ of the values $\vec{y}$ and $\vec{z}$ in these nodes can be arbitrary, the easiest way is from the first to the last with the number m. However, under certain circumstances, the order of traversing the nodes to calculate their values $\vec{y}$ and $\vec{z}$ or their individual components must be different, for example, starting from some internal node to the last, and then from the first to the one from which the traversal began. Such circumstances may be the specificity of non-linear relationships between variables.

Therefore, before considering the typical structure of the MODEL procedure, it is advisable to first consider the types of nonlinearities that may be present in the problems of calculating nonlinear oscillations.

3.2.1. Types of nonlinearities

Nonlinearities in the relationships between the parameters of the system, in which the oscillations are modeled, can be very diverse. Some types of nonlinearities may have features that affect the way they are presented and used in DHM algorithms. Therefore, it is necessary to classify them from this point of view.

Different authors have different approaches to the classification of nonlinearities [45, 52, 56]. Here, we will conduct it exclusively with regard to the specifics of taking into account nonlinear connections in the numerical modeling of nonlinear oscillations by the proposed method.

Nonlinearities with which we will further operate, that is, functional dependencies of the form

$$y = y[x], \tag{3.1}$$

where x is an argument and y is a function, and these two variables are nonlinearly connected to each other, let's divide into three groups.

Group 1 includes unambiguous nonlinearities - all types of continuous functions, both smooth (here by them we mean functions whose first derivatives do not have discontinuities), and non-smooth, that is, those whose graphs for some values of the argument have breaks, and the graphs of their first derivatives - finite gaps. Examples of such nonlinearities are illustrated in figures 3.1 - 3.6.

In fig. 3.1 shows a graph of the nonlinear dependence of the elastic force of a conical spring as a function of the deviation from the equilibrium state, which can be analytically represented by the formula [56]

$$F_{np} = a\,x + b\,x^2 + c\,x^3. \tag{3.2}$$

In fig. 3.2 shows a typical hysteresis-free magnetization curve of a ferromagnetic material, here the variable x can denote the intensity of the magnetic field or current of the coil, the core of which is made of ferromagnetic material, and variable y - magnetic field induction or coil flux coupling. Most often, this curve is given in the form of a table, from which values are selected by interpolation.

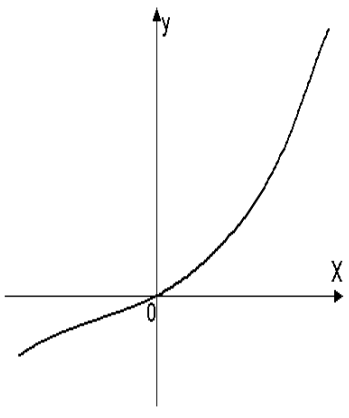


Fig 3.1. The elastic characteristic of the conical rod

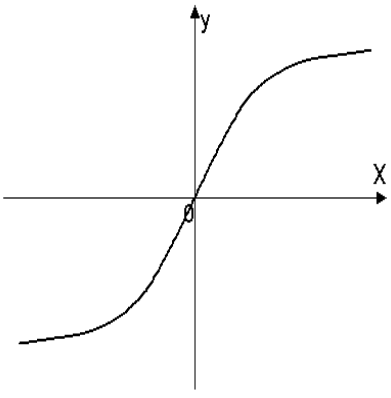


Fig. 3.2. Magnetization curve

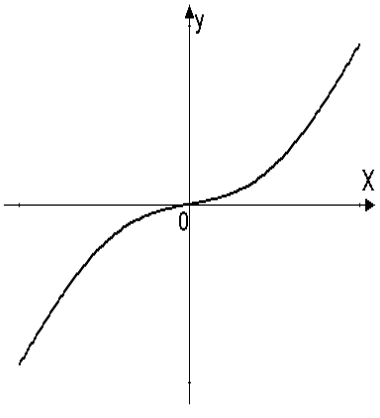


Fig. 3.3. The force of resistance of bodies moving in a gaseous medium

In fig. 3.3 shows the curve of the dependence of the force of resistance to the movement of a solid body in a gaseous medium depending on the speed of movement ***x***. With a certain idealization, this dependence can be given by the expression [56]

$$F_{on} = \alpha |\dot{x}| \dot{x} , \qquad (3.3)$$

where $\dot{x}$ is the derivative of the deviation x of the body over time, that is, the speed of the body's movement.

In fig. 3.4 shows the dependence of the elastic force of a spring with a subspring [56]. In the figure, it is shown as a broken line that has a breaking point and is formed by two straight lines. The breaking point corresponds to the deviation at which the pre-spring begins to be loaded. It could also be a continuous line formed by two curved lines joining at the break point.

In fig. 3.5 shows the restriction function [56], which is implemented, for example,

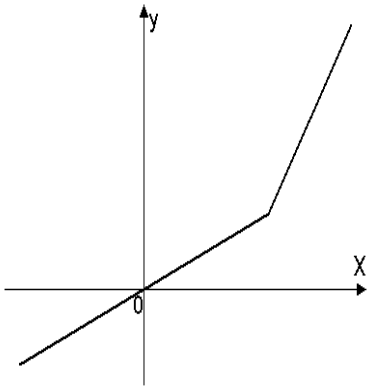


Fig. 3.4. Characteristics of a spring with a spring loaded spring

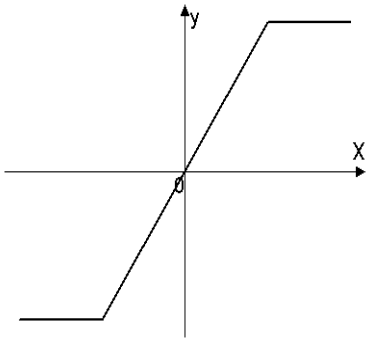


Fig. 3.5. Function limitation

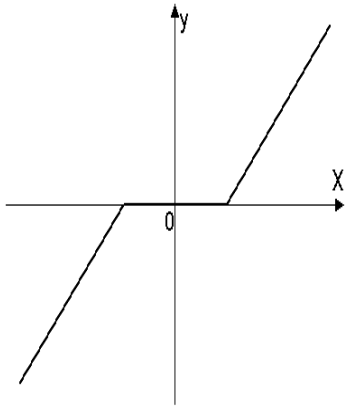


Fig. 3.6. Characteristsc of zone of insesnsintivity

hydraulic servo motor with a control spool. In [45, 52] it is called characteristic of the saturation zone. This is also a broken line that has two breaking points and is formed by three straight lines.

In fig. 3.6 shows the characteristic of the insensitivity zone. This can be, for example [52], the dependence of the speed of a direct current electric motor with independent excitation from the armature voltage in the presence of the magnitude of the static load moment.

We also include the first group of nonlinearities functions having finite discontinuities. An example of this functions are shown in fig. 3.7 relay characteristic ristic (characteristic of an ideal relay) [45, 52]. It reflects the force of Coulomb (dry) friction [56], if the dependence argument is not deviation, and the derivative of the deviation over time, that is, the speed, and then the formula corresponds to it

$$F_к[\dot{x}] = R\frac{\dot{x}}{|\dot{x}|}, \qquad (3.4)$$

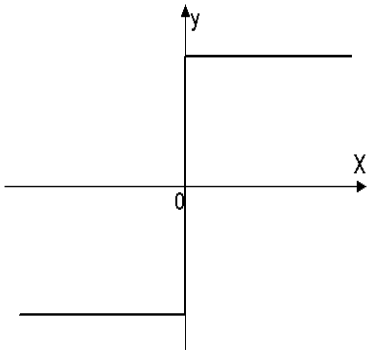


Fig. 3.7. Relay function

where R - modulus of friction force.

If the dependencies belonging to the first group have finite gaps, then it is necessary to determine their values (and the values of their derivatives) at the points of gaps, so that this does not lead to problems during modeling. So, we can assume that the function (3.4) at the point of discontinuity (at $\dot{x} = 0$) has the same value as at a point distant from it by an infinitesimally small distance (to the left or to the right), and then this value is R or $-R$.

Group 2 of nonlinearities includes single-valued nonlinearities with conditions. In this group, the functional relationship between the variables x and y is defined by two or more dependencies

$$\begin{aligned} y &= y_1[x]; \\ &\vdots \\ y &= y_k[x], \end{aligned} \qquad (3.5)$$

each of which can be of the form (3.1), and to determine the relationship between the argument and the function in one or another node of the period (semi-period), one of them is selected depending on the fulfillment of some condition. An example here can be the volt-ampere characteristic of a controlled diode (thyristor), which is formed by two dependencies: one describes the operation of the thyristor when it works as a diode (its resistance at a positive applied voltage is significantly less than at a negative one), and the second, when the resistance of the thyristor is large and constant, regardless of the sign of the applied voltage. According to these dependencies, the resistance of the thyristor is determined not only as a function of the argument (voltage or current), but also of an additional condition - the presence or absence of an opening pulse and whether this pulse is supplied when the applied voltage is positive. These characteristics are considered in more detail in section 3.2.1.1.

Group 3 of nonlinearities includes multivalued nonlinearities of the hysteresis type. Figures 3.8 - 3.10 show samples of such dependencies [45, 52]. In fig. 3.8 shows the real relay characteristic, which has the form of a hysteresis loop. In fig. 3.9 shows an ambiguous (hysteresis) characteristic with a saturation zone. In fig. 3.10 shows the hysteresis characteristic of elements of systems with backlashes or backlashes.

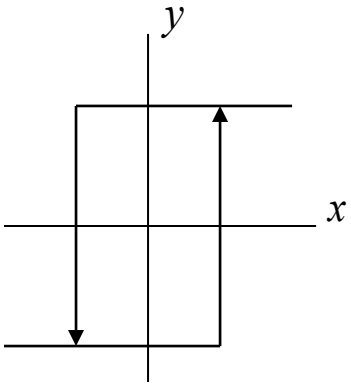


Fig. 3.8. Hysteresis relay characteristic

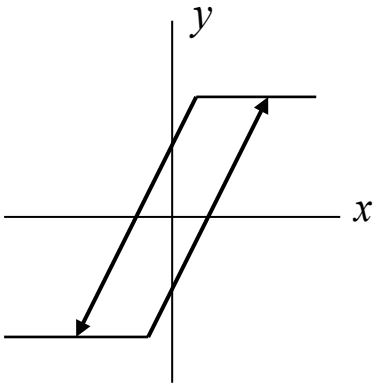


Fig. 3.9. Hysteresis characteristic of the saturation zone

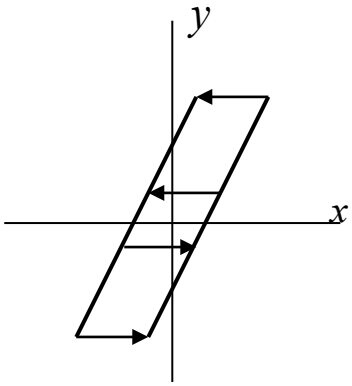


Fig. 3.10. Hysteresis characteristic of backlash (clearance)

Further, in section 3.2.1.2, a mathematical model on the period of a ferromagnetic element with a magnetization hysteresis curve is considered.

Dependencies (3.1), (3.5) in the nonlinearities of the first and second groups and individual branches of hysteresis loops (nonlinearities of the third group) can be specified both analytically and in tabular form using one or another interpolation algorithm.

3.2.1.1. Instantaneous model of the controlled valve in one period

A characteristic example of the nonlinearity of the second group of the form (3.5) is the functional relationship between the voltage u and current i of the controlled valve (thyristor), that is, its voltage-current characteristic

$$u = u[i] \qquad (3.6)$$

and differential (ohm-ampere) characteristic, that is, the dependence of the differential resistance r_∂ of the valve on its current

$$r_\partial = du/di = r_\partial[i]. \qquad (3.7)$$

For a diode (uncontrolled valve), characteristics (3.6) and (3.7) are shown in Fig. 3.11 and 3.12 with solid lines (these are nonlinearities of the first group). When the gate current has a "+" sign, it is open, and then its active resistance has some minimum value $r_{\min}$, and when the current becomes negative, the diode goes to the closed state, and its active resistance gets some maximum value $r_{\max}$. According to such characteristics, at a given value of the current, the voltage and resistance of the valve is determined at any node of the period of the periodic mode, regardless of the current values of this valve at other nodes of the period.

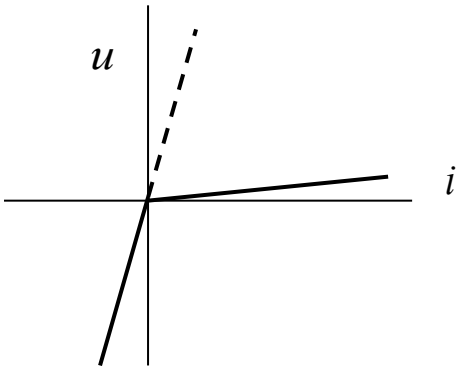
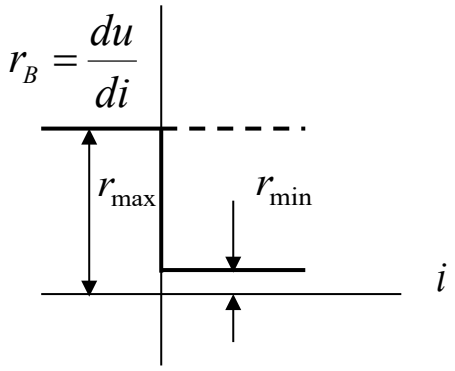


Fig. 3.11. Volt-ampere characteristic of the valve



Figc. 3.12. Dependence on the currenr дifferential resistance of the valve

The characteristics of a controlled valve (thyristor), unlike the characteristics of a diode, are characteristics with conditions. In order for the thyristor to switch from a non-conducting state to a conducting state, it is not enough for the applied voltage and its current to change the sign from minus to plus, an additional condition is the presence of a pulse on the special control electrode. So, there are two conditions for the transition of a thyristor from a non-conducting

state to a conducting state: the "+" sign of its current and the simultaneous presence of an opening pulse on its control electrode. If any of these two conditions are not fulfilled, then the thyristor cannot go from a non-conducting state to a conducting state.

Therefore, for a thyristor in a periodic process, characteristics (3.6) and (3.7) are the same as for a diode, only for that part of the period that begins immediately after the opening pulse (provided that during its action the voltage on the thyristor and its current are positive) and up to the moment when the thyristor current passes through zero into the negative region. In other parts of the period, these characteristics are depicted by straight lines in fig. 3.11 without breaking and in fig. 3.12 - without a break (in these figures, solid lines are continued with dashed lines).

Such a feature of the characteristics of the controlled valve leads to the fact that only information about the sign of its current in this node is not enough to determine the values of the voltage drop on it and its resistance at the node. If the valve current in some node has a “+” sign, then it can be considered open, as already mentioned above, only under the following conditions:

a) the area of action of the opening pulse covers the considered node (the first of such nodes in the area of action of the pulse when considering them from left to right opens the conduction zone of the thyristor; the conduction zone ends with a node in which the current is still positive and changes to negative in the next node that opens the zone of non-conductivity);

b) the pulse area does not cover the considered node, but in the previous node the valve was open, that is, the conduction zone still continues.

It is impossible to check the fulfillment of these conditions separately for any node in the period, while the value of the current in the previous nodes and the time coordinate of the pulse must also be taken into account.

When creating an instantaneous model on the period of the controlled valve, it should be taken into account that the location of the opening impulse is possible according to two options shown in Fig. 3.13. In variant "a", the action of the pulse with angular duration $\Delta\alpha_з$ and distance $\alpha_з$ from the beginning of the period ends before the end of the period. In variant "b", the angular coordinate $\alpha_з$ of the moment of occurrence of the impulse is close to the end of the period, and the action of the impulse with the same duration $\Delta\alpha_з$ ends already in the next period, or, which is the same, after the beginning of the considered period.

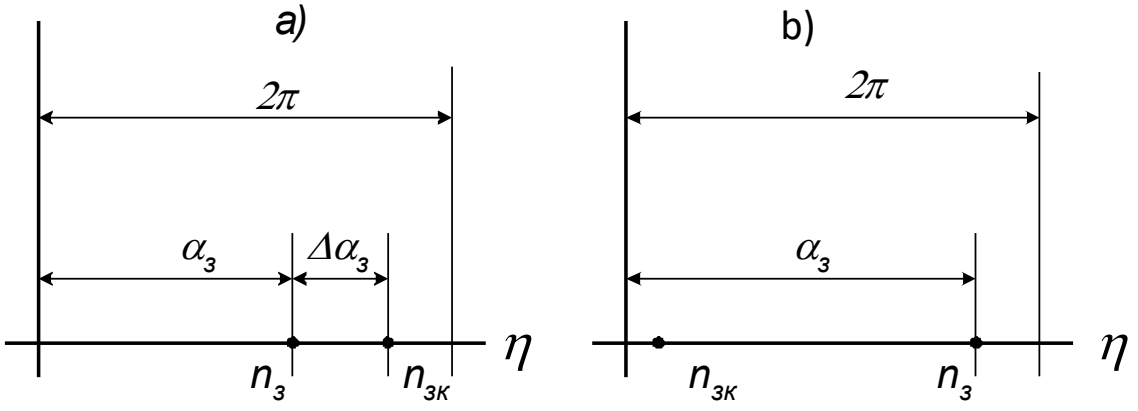


Fig. 3.13. Two options for the location of the ignition pulse on the period

The algorithm of operation of the instantaneous thyristor model on the period is as follows.

We will consider the value of the nodal vector $\vec{i}_в$ of the form (1.37) of the gate current to be given.

In all nodes of the period, the resistance of the valve r_v is set to the maximum $r_{\max}$. Then, based on the specified values of the coordinates $\alpha_з$ and width $\Delta\alpha_з$ of the ignition pulse, we determine: the number $n_з$ of the node in the period that corresponds to the beginning (front) of the pulse and opens the scope of the pulse; node number $n_{зк}$, which corresponds to the final coordinate of the pulse and closes the scope of the pulse. After that, by scanning the nodes from $n_з$ to $n_{зк}$, we determine the first node in which the component of the vector $\vec{i}_в$ has the sign "+". The serial n_z number of this node is the initial coordinate of the conduction zone of the valve. Next, we scan all the nodes, starting from n_z, in order of increasing number, and in all nodes where the values of the components of the vector $\vec{i}_в$ have a "+" sign, the gate resistance is changed to $r_{\min}$. The scan ends when a node is detected in which the value of the component of the vector $\vec{i}_в$ becomes negative, which means that it is outside the conduction zone of the gate.

As a result of performing these operations, the value of the nodal vector $\vec{r}_в$ of the valve resistances is calculated. Knowing the instantaneous values of the valve current and its resistance at all nodes of the period, that is, the values of the vectors $\vec{i}_в$ and $\vec{r}_в$, we calculate, by multiplying their components of the same name, the value of the nodal vector $\vec{u}_в$ of the valve voltage.

The algorithm for calculating the values of the nodal vectors $\vec{r}_в$ and $\vec{u}_в$ by the value of the nodal vector $\vec{i}_в$ and the values of the coordinates $\alpha_з$ and $\Delta\alpha_з$ the ignition pulse is implemented by the ***VENPER*** procedure below (we refer it to Block 5 of the DHM-S).

```
      Subroutine VENPER(STR,M,AZ,DAZ,RMAX,RMIN,UC,RV)
!--   The procedure for calculating the nodal vector UC of the gate voltage
!--   and nodal vector RV of its resistances
!--   by the given nodal current vector STR
!---------------------------------------
      Implicit none
      integer,intent(in)::M
      real,intent(in)::DAZ,RMAX,RMIN
      real,dimension(M),intent(in)::STR
      real,dimension(M),intent(out)::UC,RV
      integer::i,IZ,NZ,NZD
      real::AZ,AM,SM,AZDAZ
      real,parameter::PI2=6.2832
!------------------------------
!--   Input values:
!--   STR - the nodal vector of the gate current
!--   M - the number of points in the period
```

```
!--   AZ - valve ignition angle, rad.
!--   DAZ - ignition pulse width, rad.
!--   RMAX - the resistance of the closed valve
! --  RMIN - the resistance of the open valve
!--   Output values:
!--   UC - the nodal vector of valve voltages
!--   RV - the nodal vector of valve resistances
!---------------------------------------
      AM=M/PI2
  10 if(AZ.le.PI2) goto 11
      AZ=AZ-PI2;  goto 10
  11 AZDAZ=AZ+DAZ;  if(AZDAZ.le.PI2) goto 20
      AZDAZ=AZDAZ-PI2;   goto 11
!---- values of ignition angles led to <= PI2
  20 NZ=AZ*AM+1.5; NZD=AZDAZ*AM+1.5
      If(NZ.gt.M) NZ=NZ-M
      If(NZD.gt.M) NZD=NZD-M
!--  NZ - ignition switch unit number
!--  NZD - number of the ignition switch-off node
      RV=RMAX
!--  The resistance of the valve in all nodes was set to the maximum
      SM=0.;   if(NZD.lt.NZ) goto 30
!--    We are looking for the activation node according to option "a"
            do i=NZ,NZD
              if(STR(i).gt.SM) goto 40
             end do;  goto 50
!  We are looking for the activation node according to option "b"
   30 do i=NZ,M
         if(STR(i).gt.SM) goto 40
       end do
            do i=1,NZD
              if(STR(i).gt.SM) goto 40
             end do;  goto 50
  40  IZ=i
!--  found IZ - the number of the valve activation node
            do i=IZ,M
              if(STR(i).gt.SM) RV(i)=RMIN
              if(STR(i).le.SM) goto 50
            end do
            do i=1,IZ
              if(STR(i).gt.SM) RV(i)=RMIN
              if(STR(i).le.SM) goto 50
            end do
!-- in all nodes of the conduction zone of the valve
!-- its resistance was made equal to RMIN
  50 UC=STR*RV
!--  calculated the nodal vector UC of the gate voltage
      return
      end subroutine VENPER
```

This procedure does not need additional explanations, since the comments in its text are quite enough to track the logic and implement the algorithm.

3.2.1.2. Instantaneous model on the period of nonlinearity of the hysteretic form

Let's consider one of the nonlinearities of the third group - the hysteresis type, in particular - an inductive element with a ferromagnetic core, the Weber-ampere characteristic of which is shown in Fig. 3.14.

This figure shows the main magnetization curve passing through the origin of coordinates, and two branches - the upper and lower ones, which form the limiting hysteresis loop [4, 38]. We denote the abscissa of the left merging point of the upper and lower branches i_L and the abscissa of the right merging point of these branches i_R. We will not take the so-called partial symmetric and asymmetric hysteresis loops into account and will assume that the relationship between the flux coupling and the current of the inductive element in a periodic process is expressed either by the main magnetization curve, if the minimum and maximum current values in the period do not go beyond the values i_L and i_R, or this relationship is expressed by the limit loop of hysteresis, if the minimum and maximum value of the current in the period go beyond these limits.

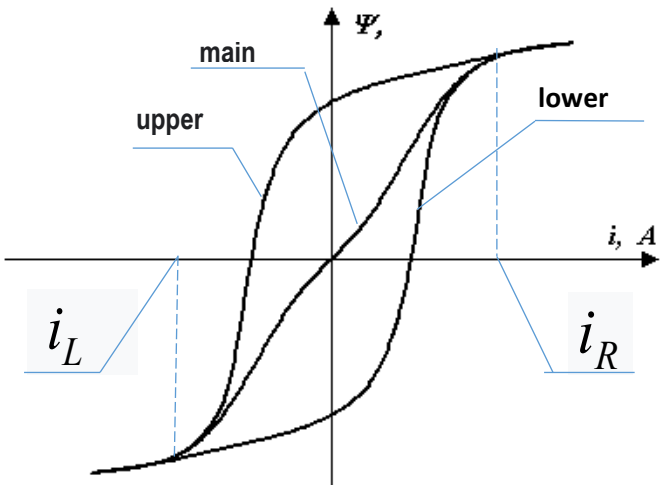


Fig. 3.14. Hysteresis loop

By the instantaneous mathematical model on the period (half-period) of the hysteresis inductive element [26], we will understand the algorithm for calculating the values of the flux linkage ψ and differential inductance $L = d\psi/di$ of the inductive element based on the values of its current i in the same nodes of the period.

The input value of the model is the value $\vec{i}_в$ of the current nodal vector of the inductive element. The algorithm of the instantaneous model on the period (half-period) of the hysteresis inductive element is as follows.

Among all the components of the nodal current vector $\vec{i}_в$, we look for its maximum $i_{\max}$ and minimum $i_{\min}$ components, as well as the number $n_{\min}$ that has the minimum component in the vector. At the same time, the following options are possible:

1) $i_{\min}$ is less than i_L and at the same time $i_{\max}$ is greater than i_R, then the values of the elements of the nodal flux coupling vector are searched using the limit hysteresis loop. At the same time, we start traversing the nodes of the period from the node with the number $n_{\min}$. At the next node ($n_{\min}$+1), the value of the current will already be greater than $i_{\min}$

(the current increases), and therefore, at these two points of the period, the connection between ψ and i is determined by the lower branch of the loop. Moving through the nodes of the period from $n_{\min}$ to the right and using the lower branch as the connection function between ψ and i, in each of the nodes we determine the value of ψ and L and compare the current value i with $i_{\max}$, and as soon as it becomes greater than $i_{\max}$, the upper branch is assigned as the function of the connection between ψ and i. Moving the nodes further to the right and determining the value ψ and L in each node and using the upper branch, in each of the nodes we compare the value of the current i with $i_{\min}$. As soon as i becomes smaller than $i_{\min}$, the lower branch is again assigned as the connection function between ψ and i. These actions in the order described above are repeated in all nodes until the end of the period and then from the first node to the node with number $n_{\min}$ - 1.

2) $i_{\min}$ is not less than i_L or $i_{\max}$ is not greater than i_R, then the main branch is assigned as the connection function between ψ and i at all nodes of the period, and with its use at all nodes of the period (here the nodes of the period can be bypassed in order from first to last) the values of ψ and L are determined by the current values i.

This algorithm is implemented by the following procedure (we refer it to Block 5 of the DHM-S):

```
      Subroutine HISTPER(AL,STR,M,PS,LH,PST,XL,XP,DS,NH)
!--  The procedure for calculating the nodal vector PS of the flow coupling
!--   and the nodal vector LH of the differential inductances of the hysteresis element
!--   by the given nodal current vector STR
!----------------------------------------
!--   AL - narrowing coefficient of the hysteresis loop
!--   M is the number of elements in the nodal vectors STR, PS, LH
!--   PST - the table that specifies the hysteresis loop
!--   XL,XP - the left and right coordinates of the points of convergence of the branches
!--   DS - table step
!--   NH - the number of table nodes
!---------------------------------
      implicit none
      integer,intent(in)::M,NH
      real,intent(in)::AL,XL,XP,DS
      real,dimension(M),intent(in)::STR
      real,dimension(3,NH),intent(in)::PST
      real,dimension(3,NH)::PST1
      real,dimension(M),intent(out)::PS,LH
      integer::i,iN,NK
      real::SN,SX
      NK=2     !  NK - loop branch number
       if(AL.eq.0.) goto 3
       SN=STR(1); SX=STR(1); iN=1
       do i=2,M
         if(STR(i).gt.SN) goto 1
         SN=STR(i);  iN=i
 1   if(STR(i).lt.SX) goto 2
       SX=STR(i)
```

```
   2  continue
          end do
!-- determined the minimum SN and maximum SX value
!-- STR vector component and iN number of the minimum
         if(SN.lt.XL.and.SX.gt.XP) goto 4
!--  by the middle branch
   3 do i=1,M
      call INTHIST(NK,STR(i),PS(i),LH(i),PST,XL,XP,DS,NH)
      end do
          return
!--  obtaining the PST1 table for the narrowed loop
    4 do i=1,NH
        PST1(2,i)=PST(2,i)
        PST1(1,i)=PST(2,i)+AL*(PST(1,i)-PST(2,i))
         PST1(3,i)=PST(2,i)-AL*(PST(2,i)-PST(3,i))
       end do
!-- by the upper and lower branches of the loop
       NK=3
       do i=iN,M
          call INTHIST(NK,STR(i),PS(i),LH(i),PST1,XL,XP,DS,NH)
             if(NK.eq.3.and.STR(i).gt.XP) NK=1
          if(NK.eq.1.and.STR(i).lt.XL) NK=3
         end do
!--  went to the right along the lower branch
          do i=1,iN-1
            call INTHIST(NK,STR(i),PS(i),LH(i),PST1,XL,XP,DS,NH)
             if(NK.eq.3.and.STR(i).gt.XP) NK=1
             if(NK.eq.1.and.STR(i).lt.XL) NK=3
          end do
!--  returned along the upper branch
      return
         Contains
         Subroutine INTHIST(NK,X,Y,YX,YT,XL,XP,DX,M)
!----- The procedure of linear interpolation of the given
!----- hysteresis loop table
!-------------------------
!--   NK - loop branch number:
!--                1 - upper
!--                2 - average (main)
!--                3 - lower
!-- X - the abscissa
!-- Y - the ordinate
!-- YX - derivative
!-- YT - the table by which the loop is specified
!-- XL,XP - left and right coordinates of the points of convergence of the branches
!-- DX - table step
!-- M - the number of table nodes
!----------------------
          Implicit none
          integer,intent(in)::M,NK
          real,intent(in)::X,XL,XP,DX
```

```
          real,dimension(3,M),intent(in)::YT
          real,intent(out)::Y,YX
          integer::j
    If(X.LT.XL) goto 1
          if(X.GT.XP) goto 2
!-----  Interpolation within a loop
          j=(X-XL)/DX+2
    YX=(YT(NK,j+1)-YT(NK,j))/DX
          Y=YT(NK,j)+YX*(X-(XL+DX*(j-2)))
       return
!---- Extrapolation from the left
    1 YX=(YT(NK,2)-YT(NK,1))/DX
          Y=YT(NK,2)+(X-XL)*YX
          return
!----   Extrapolation from the right
    2   YX=(YT(NK,M)-YT(NK,M-1))/DX
        Y=YT(NK,M-1)+(X-XP)*YX
         return
         end subroutine INTHIST
      end subroutine HISTPER
```

The formal parameters of the ***HISTPER*** procedure are described in the comments located immediately after the procedure header. However, several of them require additional clarification.

About the formal parameter ***AL***. The practice of calculations has shown that the numerical simulation of periodic processes in systems containing nonlinear elements of hysteresis type should be carried out in this order.

First, the calculation of the periodic process is performed under the condition that the characteristics of all hysteresis elements are their main branches that pass through the origin of the coordinates (the areas of the loops are reduced to zero). For this, the formal parameter ***AL*** is assigned a zero value, and thus the upper and lower branches of the loops are combined with the main branches. After that, the process of taking into account the hysteresis begins (see the relevant part of the ***HARMOSC*** procedure, section 2.2.2.1): in the loop, the parameter of which takes the value from unity to the value specified by the element ***KER(7)*** of the formal parameter ***KER*** of the ***HARMOSC*** procedure, the value of ***AL*** changes from zero to units, and at the same time the upper and lower branches deviate from the main branch and approach their real values. With each change in ***AL***, the solution is refined using Newton's iterative method. The solution obtained for ***AL*** equal to unity is the one corresponding to the periodic process in the scheme with hysteresis taken into account.

It is precisely in order to implement such an algorithm that the formal parameter ***AL*** is included in the list of formal parameters of the ***HISTPER*** procedure.

The formal parameters ***PST, XL, XP, DS, NH*** of this procedure set the hysteresis loop and have the following meaning:

DS - the step of the tables used to specify the branches of the loop;

XL – coordinate of the left point of convergence of the loop branches;

XP – coordinate of the right point of convergence of the loop branches;

PST(3,NH) - an array that stores three tables of ordinate values of internalthe left of the three branches of the loop (in the ***PST*** array, the first row is the table for the upper branch, the second row is for the main branch, and the third row is the lower branch);

NH - the number of nodes in each table.

The values ***XL*** and ***XP*** are the beginnings of the linear parts of the converging branches, and ***XL*** is the abscissa of the second node of the tables from the left and ***XP*** is the second node of the tables from the right. This arrangement of these points allows linear extrapolation to the left and right of the tables for the two extreme nodes.

The ***HISTPER*** procedure has its own ***INTHIST*** procedure, the purpose of which is linear interpolation from the table specifying one of the loop branches. This procedure does not need additional explanations, as they are all described in sufficient detail in the comments.

3.2.2. Auxiliary procedures for the development of the simplification of procedure *Model*

The purpose of the procedures considered in this section is their use in the development of the ***MODEL*** procedure, with the aim of simplifying the latter, in each specific case, of numerical modeling of nonlinear oscillation. Each of these auxiliary procedures introduces a certain type of macro operation of the method.

Let's break down these procedures (all of them will be assigned to Block 5 of the DHM-S).

3.2.2.1. Procedure DRAWOUTV

This procedure makes it possible to extract the vector of the form (1.68) of the instantaneous values of all process variables in the desired node during the period (half-period) from the composite nodal vector of the form (1.118), formed in a parallel way.

Text of the procedure:

```
            Subroutine DRAWOUTV(K,XC,MK,X,iM)
!--  A procedure that copies a fragment from a composite node vector,
!--  corresponding to the iM-th node on the period (semi-period)
!------------------------------
            implicit none
            integer,intent(in)::K,MK,iM
            real,dimension(MK),intent(in)::XC
            real,dimension(K),intent(out)::X
            integer::i
!------ Procedure parameters:
!--- K - the order of the diff. system. equations, he is -  the number of elements of the X vector
!--- XC - composite nodal vector of the form (1.118)
!--- MK - the number of elements of the vector ЧС
!--- X - the vector of values ??of variables of the form (1.68)
!--- iM - specified number of the node on the period (half-period)
!-------------------------------------------
            do i=1,K
               X(i)=XC((iM-1)*K+i)
            end do
            return
            end subroutine DRAWOUTV
```

The formal parameters of the procedure are sufficiently fully described in the comments.

3.2.2.2. Procedure DRAWOUTXV

This procedure makes it possible to extract from the complex nodal vector of the form (1.118) a simple nodal vector of the form (1.37) of the instantaneous values of one of the variables at all m nodes of the period (semi-period).

Text of the procedure:

```
            Subroutine DRAWOUTXV(K,XC,MK,M,XV,iK)
!--  A procedure that extracts from a composite nodal vector
!--  simple nodal vector for the iK-th variable
!-----------------------------------
            implicit none
            integer,intent(in)::K,MK,M,iK
            real,dimension(MK),intent(in)::XC
            real,dimension(M),intent(out)::XV
            integer::i
!-------------------------
!----- Procedure parameters:
!--- K - the order of the system of differential equations
!--- XC - composite nodal vector of the form (1.118)
!--- MK - the number of elements of the vector ЧС
!--- M - number of nodes per period (semi-period)
!--- XV - a simple nodal vector of the form (1.37)
!--- iK - specified variable number (iK is less than or equal to K)
!--------------------------
            do i=1,M
               XV(i)=XC((i-1)*K+iK)
            end do
            return
            end subroutine DRAWOUTXV
```

The formal parameters of the procedure are sufficiently fully described in the comments.

3.2.2.3. Procedure DRAWUPV

This procedure performs the opposite action of the ***DRAWOUTV*** procedure: it forms the corresponding fragment of the composite node vector from the values of all variables in one node of a period (semi-period).

Text of procedure:

```
          Subroutine DRAWUPV(K,XV,XC,MK,iM)
!--  A procedure that "inserts" into a composite knot vector
!--  fragment corresponding to the iM-th node
!----------------------
          implicit none
          integer,intent(in)::K,MK,iM
          real,dimension(K),intent(in)::XV
          real,dimension(MK)::XC
          integer::i
!---------------
```

```
!------ Procedure parameters:
!--- K - the order of the system of differential equations
!---     and the number of elements of the XV vector
!--- XV - vector of appearance variables (1.68)
!--- XC - composite nodal vector of the form (1.118)
!--- MK - the number of elements of the vector ЧС
!--- iM - specified node number
!----------------------------
          do i=1,K
             XC((iM-1)*K+i)=XV(i)
          end do
          return
          end subroutine DRAWUPV
```

The formal parameters of the procedure are sufficiently fully described in the comments.

3.2.2.4. Procedure DRAWUPM

The result of this procedure is the formation of a fragment of the composite matrix of nodal parameters of the form (1.127) - its diagonal block corresponding to one of the nodes of the period (semi-period).

Text of the procedure:

```
          Subroutine DRAWUPM(K,SX,SC,MK,iM)
!--   A procedure that "inserts" into a composite matrix of nodal parameters
!--   fragment corresponding to the iM-th node
!----------------------
          implicit none
          integer,intent(in)::K,MK,iM
          real,dimension(MK,K)::SC
          real,dimension(K,K)::SX
          integer::i,j
!---------------
!------ Procedure parameters:
!--- K - the order of the diff system. equations and the SX matrix
!--- SX - matrix of instantaneous appearance parameters (1.128)
!--- SC - composite matrix of nodal appearance parameters (1.127)
!--- MK - the number of rows of the SC matrix
!--- iM - number of the node on the period (half-period)
!---------------------------
          do i=1,K
            do j=1,K
               SC((iM-1)*K+i,j)=SX(i,j)
            end do
          end do
          return
          end subroutine DRAWUPM
```

Here it should be borne in mind that the array ***SC*** contains all the diagonal blocks of the composite matrix of nodal differential parameters of the form (1.127), shifted to the left, so that they "stand" on top of each other. This is why the ***SC*** array has ***K*** (not ***MK***) columns. The array ***SC*** contains a composite matrix of nodal differential parameters of the form (1.127) in a packed form.

The formal parameters of the procedure are sufficiently fully described in the comments.

3.2.2.5. Procedure ADDV

The purpose of this procedure is to add the values of the elements of the simple nodal vector of the form (1.37) for one of the variables to the values of the corresponding elements of the complex nodal vector of the form (1.118).

Such an operation is necessary in the case when there are nonlinearities of the second and/or third groups in the oscillating system (see section 3.2.1). The values of the nodal vectors of the variables containing the nonlinearities of these groups must be calculated, as shown in previous sections 3.2.1.1 and 3.2.1.2, separately from the values of the composite nodal vectors for the group of variables that are connected by linear dependencies and/or nonlinearities of the first group. After their separate calculation, it becomes necessary to add the values of the elements of these nodal vectors to the values of the corresponding elements of the previously calculated composite nodal vectors and obtain the final values of the composite nodal vectors.

The text of the procedure is as follows:

```
            Subroutine ADDV(K,XV,M,XC,MK,iK)
!--  A procedure that adds to the elements of a compound node
!--  vector elements of a simple nodal vector of the iK-th variable
!-------------------------------------------------------
            implicit none
            integer,intent(in)::K,MK,M,iK
            real,dimension(MK)::XC
            real,dimension(M)::XV
            integer::i
!----- Procedure parameters:
!--- K is the order of the system of differential equations
!--- XV is a simple nodal vector
!--- M is the number of elements of the XV vector
!--- XC is a composite nodal vector
!--- MK - the number of elements of the XC vector
!--- iK - variable number
!-------------------------------------
     do i=1,M
        XC((i-1)*K+iK)=XC((i-1)*K+iK)+XV(i)
     end do
     return
     end subroutine ADDV
```

The formal parameters of the procedure are sufficiently fully described in the comments.

3.2.2.6. Procedure ADDM

The purpose of this procedure is to add the values of the elements of the matrix of nodal parameters of the form (1.49) for one of the variables to the values of the corresponding elements of the composite matrix of the nodal parameters of the form (1.127).

Such an operation is necessary in the case when there are nonlinearities of the second and/or third groups in the oscillating system. The values of the nodal parameter matrices for the variables containing the nonlinearities of these groups must be calculated, as shown in the previous sections, separately from the values of the composite nodal parameter matrices for the

group of variables that are connected by linear dependencies and/or nonlinearities of the first group. After their separate calculation, it is necessary to add the values of the elements of these nodal parameter matrices to the values of the corresponding elements of the previously calculated nodal parameter matrices.

Text of the procedure:

```
            Subroutine ADDM(K,XM,M,XMC,MK,iK)
!--  A procedure that adds to a compound node matrix of parameters
!--  a simple nodal diagonal matrix of parameters for the iK-th variable
!-------------------------------
            implicit none
            integer,intent(in)::K,M,MK,iK
            real,dimension(M)::XM
            real,dimension(MK,K)::XMC
            integer::i
!----- Procedure parameters:
!--- K - the order of the system of differential equations
!--- XM - a simple nodal diagonal matrix
!--- M - the number of elements of the XM matrix
!--- XMC - composite matrix of nodal parameters
!--- MK - the number of rows of the XMC matrix
!--- iK - variable number
            do i=1,M
              XMC((i-1)*K+iK,iK)=XMC((i-1)*K+iK,iK)+XM(i)
            end do
            return
            end subroutine ADDM
```

The formal parameters of the procedure are sufficiently fully described in the comments.

3.2.3. Sample procedure Model

Now we can consider the structure of the ***Model*** procedure, which will serve as a sample for its development in each specific case of modeling.

The sample text of this procedure is as follows.

```
    Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
! The subroutine implements the instantaneous pricess model
!------------------------
          Implicit none
          real,intent(in)::AL
          integer,intent(in)::M,K,MK
          real,dimension(MK),intent(in)::XC
          real,dimension(MK),intent(out)::YC,ZC
          real,dimension(MK,K),intent(out)::YXC,ZXC
          real,dimension(K)::X,Y,Z
          real,dimension(K,K)::YX,ZX,BM
          real,dimension(M)::XV,UV,RV,UV1,RV1
          real,dimension(3,3)::B
          real::C1,C2,C3
```

```
      integer::i
      common/MPM/C1,C2,C3,B
!-- through Common/MPM/ data from the main program is transferred
      BM=B
      do i=1,M
       call DRAWOUTV(K,XC,MK,X,i)
       Y(1)=    ; Y(2)=   ;  Y(3)=
       call DRAWUPV(K,Y,YC,MK,i)
       Z(1)=    ; Z(2)    ;   Z(3)=
       call DRAWUPV(K,Z,ZC,MK,I)
        YX(1,1)=    ; YX(1,2)=   ; YX(1,3)=
        YX(2,1)=    ; YX(2,2)=   ;  YX(2,3)=
       YX(3,1)=    ; YX(3,2)=    ;   YX(3,3)=
       call DRAWUPM(K,YX,YXC,MK,i)
      ZX(1,1)=    ; ZX(1,2)=    ;   ZX(1,3)=
       ZX(2,1)=    ; ZX(2,2)=     ;   ZX(2,3)=
       ZX(3,1)=    ; ZX(3,2)=     ;   ZX(3,3)=
       call DRAWUPM(K,ZX,ZXC,MK,i)
      end do
      call DRAWOUTXV(K,XC,MK,M,XV,3)
      call TIMEMOD(XV,M,…,UV,RV)
      call ADDV(K,UV,M,ZC,MK,3)
      call ADDM(K,RV,M,ZXC,MK,3)
      return
      end  subroutine Model
```

First of all, we note that the title of the procedure (its name and the number of formal parameters and their types) cannot be changed, because it is specified by the ***HARMOSC*** and ***IMPROVE*** procedure call operators (see sections 2.2.2.1 and 2.2.2.3).

The formal parameters of this procedure are:

AL is a real variable, the narrowing coefficient of the hysteresis loop (in case there are hysteresis-type nonlinearities in the problem);

M is an integer variable, the number of nodes per period (half-period) of the process, it is also the size of a simple nodal vector;

K is an integer variable, the order of the system of differential equations describing oscillations;

MK is an integer variable, the size of the complex nodal vector;

XC is a real one-dimensional array, a composite nodal vector for a vector variable, the time dependence of which is the desired periodic solution of the system of differential equations of the forms (1.64) - (1.67);

YC is a real one-dimensional array, a composite nodal vector for a vector variable in equations of the form (1.64) or (1.66);

ZC is a real one-dimensional array, a composite nodal vector for a vector variable in equations of the form (1.64) - (1.67);

YXC is a real two-dimensional array, stores a complex matrix of nodal differential parameters of the form (1.127a), whose diagonal blocks are located one under the other - for the purpose of denser packing;

ZXC is a real two-dimensional array that stores a complex matrix of nodal differential parameters of the form (1.127b), whose diagonal blocks are located one below the other;

BM is a real two-dimensional array, for a matrix of the form (1.69).

The set of formal parameters described above assumes the case when the solvable system of differential equations has the most general form (1.66). For cases when the form of the system of differential equations is different - (1.64), (1.65) or (1.67) - some of the formal parameters will be unused.

The ***Model*** procedure also receives data from the main program using a shared memory area, the name of this area (here ***MPM***) must be the same as in the main program. It is given by the operator ***Common/MPM/C1,C2,C3,B***. In this shared memory area, there are three real variables ***C1,C2,C3*** and a real two-dimensional array ***B(3,3)*** containing a matrix of the form (1.69) for the case when the system of differential equations to be solved is of the third order.

The first executed statement ***BM=B*** assigns the value of the matrix ***B*** to the formal parameter ***BM*** .

Next in the procedure is a loop (let's call it the main loop of the ***Model*** procedure), whose parameter i changes from 1 to ***M***. In this loop:

- the ***call DRAWOUTV(K,XC,MK,X,i)*** operator calls the ***DRAWOUTV*** procedure (see section 3.2.2.1), which copies the fragment corresponding to the i-th node in the period (semi-period) from the ***XC*** array and assigns it to the X array, which is intended for storing the vector (1.68a);
- operators ***Y(1)= ; Y(2)= ; Y(3)=*** calculate the value of the array ***Y***, which stores the value of the vector (1.68b);
- the operator ***call DRAWUPV(K,Y,YC,MK,i)*** (see section 3.2.2.3) the value of the ***Y*** array is inserted as a fragment into the ***YC*** array;
- then similar actions are performed with respect to arrays ***Z*** and ***ZC***;
- Operators ***YX(1,1)= ; YX(1,2)= ... ; YX(3,3)=*** calculate the value of the elements of the diagonal block corresponding to the ith node on the period (semi-period) of the matrix (1.127a);
- by the operator ***call DRAWUPM(K,YX,YXC,MK,i)*** (see section 3.2.2.4), the value of the ***YX*** array is inserted as a fragment into the ***YXC*** array;
- then similar actions are performed for ***ZX*** and ***ZXC*** arrays.

The cycle described above forms the arrays ***YC , ZC , YXC*** and ***ZXC*** in the part that corresponds to all linear elements of the system and nonlinear ones, except for those containing nonlinearities of the second and third groups.

The procedure ends with a group of operators that finalize the arrays ***YC , ZC , YXC*** and ***ZXC*** in the part that corresponds to the nonlinear elements of the system containing the nonlinearities of the second and third groups. These operators are:

- operator ***call DRAWOUTXV(K,XC,MK,M,XV,3)*** (see section 3.2.2.2), which copies a fragment of the ***XC*** array and forms an ***XV*** array (a simple nodal vector of the form (1.37) corresponding to the third element of the vector); it is believed that in this problem it is related to the nonlinearity of the second or third group);

- operator c***all TIMEMOD(XV,M,...,UV,RV)***, which calls some ***TIMEMOD*** procedure (here the name of the procedure is conditional), which implements an instantaneous model during the period of this nonlinearity from the second or third group;

- operators ***call ADDV(K,UV,M,ZC,MK,3)*** and ***call ADDM(K,RV,M,ZXC,MK,3)*** (see sections 3.2.2.5 and 3.2.2.6), which add to the elements of the complex nodal vector ***ZC*** and the complex matrix of nodal parameters ***ZXC*** the corresponding elements of the simple nodal vector ***UV*** and

the matrix of nodal parameters ***RV*** (it is considered that in this problem the arrays ***YC*** and ***YXC*** do not need).

If the oscillating system does not contain nonlinearities of the second or third groups, then there are no other operators between the end of the main cycle of the ***Model*** procedure and its ***Return*** operator.

The sample procedure ***Model*** described in this section is fairly stereo-typical. It individualizes in each specific case of modeling only the body of the main loop (operators between ***" do i=1,M "*** and ***"end do"***), and, in the case of nonlinearities of the second and/or third groups, operators between ***"end do "*** and ***"return".*** This is illustrated by all the examples in Chapter 4.

3.3. Procedure OUTP

As already mentioned in section 2.1 and in the introductory part of this chapter, the ***OUTP*** procedure for processing and saving simulation results belongs to the block of user software components. This procedure can be developed individually for each simulation case. However, it is possible to develop a typical ***OUTP*** program and assign it to Block 5 of the DHM-S. And only in the case when the operations included in the typical ***OUTP*** procedure do not satisfy the user, he will be forced to develop his own version of this procedure. However, the title of the procedure (name, list of formal parameters and their types) cannot be changed, because it is determined by the call operators in the ***HARMOSC*** and ***IMPROVE*** procedures

The text of a typical (standard) ***OUTP*** procedure is as follows.

```
      Subroutine OUTP(IG,Y,N,K,NG,M,MK,KK)
!--   The procedure for processing and memorizing (recording) the results to a file
!--   with results (with KK=1, the recording volume is minimal)
!-------------------------------------
          Implicit none
          integer,intent(in)::IG,N,K,NG,M,MK,KK
          real,dimension(N),intent(in)::Y
          real,dimension(MK)::XC
          real,dimension(M)::XV
          integer::i,j,KG,N1,N2,N3,KNG
          real::AC,AS,AA
          KG=(NG-1)/2
          If (IG.eq.0) then
             N1=KG; N2=1; N3=2
          else
             N1=NG; N2=2; N3=1
          end if
          KNG=K*NG
          call KVGVS(K,Y,NG,KNG,XC,M,MK)
       do i=1,K
          write(1,10)i
          if(IG.ne.0) goto 1
          write(1,11) Y((i-1)*NG+1)
    1  do j=1,N1,N2
             AC=Y((i-1)*NG+N3*j);  AS=Y((i-1)*NG+N3*j+1)
             AA=sqrt(AC**2+AS**2)
             write(1,12)i,j,AC,i,j,AS,i,j,AA
         end do
```

```
        if(KK.eq.1)goto 2
        if (IG.eq.0) then
          write(1,13)M
        else
     write(1,14)M
        end if
        call DRAWOUTXV(K,XC,MK,M,XV,i)
        write(1,15)XV
  2  continue
     end do
     write(1,16)Y(K*NG+1)
      return
  10 format(2X,'Amplitudes of harmonics of ',i2,'-th variable:')
  11 format(2X,'constant component = ',E11.4)
  12 format(2X,'X',i1,'(c',i2,')=',E11.4,' X',i1,'(s',i2,')=',E11.4,'  X',i1,'(',i2,')=',E11.4)
  13 format(2X,'The value of the variable in nodes of the period, M=',i3')
  14 format(2X,'The value of the variable in nodes of the half-period, M =',i3)
  15 format(2X,6E11.4)
  16 format(2X,'circular frequency of the fundamental harmonic =',E11.4)
     end Subroutine OUTP
```

The formal parameters of the procedure are:

IG – variable of integer type; if its value is zero, then it is considered that the variables of the problem have constant components and harmonics of both even and odd orders; if its value is one, then it is considered that the variable problems have only odd harmonics;

Y - a real one-dimensional array formed from the composite vector of amplitudes of variables of the form (1.82), the circular frequency of the fundamental harmonic and the parameter ;

N - a variable of the integer type, the size of the array ***Y***;

K - a variable of integer type, the order of the system of differential equations, the periodic solution of which is sought;

NG - an integer type variable, the size of a simple vector of amplitudes;

M - an integer variable, the number of nodes per period (semi-period), the size of a simple nodal vector;

MK - an integer type variable, the size of the complex nodal vector;

KK - a variable of integer type; if its value is one, then not all data is saved (written to the output file), but only part of it.

* * *

The main program of the block of user software components and the ***OUTP*** procedure can contain statements that implement dialog (when entering data) and multimedia (when outputting data in the form of graphs, diagrams, tables, etc.) capabilities of the latest versions of Fortran (or other languages - through compatible programming or under "unification of object modules as compilation products from these languages). The author leaves these possibilities to the user who owns such software tools.

Chapter 4

TESTS AND EXAMPLES

In this chapter, a number of tests and examples are considered, the purpose of which is to illustrate the application of DHM and its software, as well as to perform their verification. Among them are both problems that have approximate solutions by analytical methods, and then such problems play the role of tests, and problems that cannot be solved by analytical methods.

4.1. Tests and examples of calculations of forced oscillations

This section presents several examples of numerical modeling of forced oscillations in nonlinear systems with various types of nonlinear connections in them. These are examples B.1 - B.5. In each example of this group, the fourth element ***KER(4)*** of the ***KER*** control array is set to 0 (meaning oscillation is forced) before calling the ***HARMOSC*** procedure in the main program of the user program component block.

4.1.1. Example (project) B.1

As a test calculation for the case when the nonlinearity in the system is unambiguously unconditional (belongs to the first group of nonlinearities) and it is given analytically, we will consider in this project the calculation of forced oscillations of a body in a gaseous medium. A similar example is given in [56] under number 4.2.1.

With a certain idealization, it can be assumed that the resistance force of a moving body in a gas medium is proportional to the square of the velocity and has the same sign as the velocity of the body (see formula (3.3) and Fig. 3.3). Taking this equation into account, the motion of the body has the form

$$\ddot{x} + \omega_0^2 x + \alpha |\dot{x}| \dot{x} = h \sin(\omega t + \varphi). \qquad (4.1)$$

In [56], when considering this example and determining the oscillation parameters, the method of harmonic balance is used (as analytical) and the solution is sought in the form

$$x = a \sin \omega t \ . \qquad (4.2)$$

As a result of substituting (4.2) into (4.1) and a series of analytical transformations, the value of the amplitude a of oscillations is obtained as the root of the biquadratic equation

$$\frac{64 \alpha^2 \omega^4}{9\pi^2} a^4 + (\omega_0^2 - \omega^2)^2 a^2 - h^2 = 0 \qquad (4.3)$$

and is expressed by the formula (at $\omega < \omega_0$)

$$a = \sqrt{\frac{-b_2 + \sqrt{b_2^2 + 4b_1 h^2}}{2b_1}}\ , \qquad (4.4)$$

here

$$b_1 = \frac{64\alpha^2 \omega^4}{9\pi^2}\ ; \qquad (4.5)$$

$$b_2 = (\omega_0^2 - \omega^2)^2\ , \qquad (4.6)$$

and the initial phase of the forcing force - by the formula

$$\varphi = arctg \frac{8\alpha a\omega^2}{3\pi(\omega_0^2 - \omega^2)}\ . \qquad (4.7)$$

Formulas (4.1) – (4.7) use the same notation as in [56] when considering example 4.2.1 there. In particular, in the notation of differential equation (4.1), as before in formula (3.4), one dot above the variable means that it is the first derivative of this variable in time, and two dots above the variable means that it is the second derivative.

If specified by numerical values

$$\omega = 10.0 \quad 1/c; \qquad \omega_0 = 25.0 \quad 1/c;$$

$$\alpha = 12.0 \quad 1/м\ ; \qquad h = 100.0 \quad м/c^2\ ,$$

then by formulas (4.4) and (4.7) we obtain $a = 0{,}1797$ m; $\varphi = 0{,}335$ rad.

We will calculate these same oscillations using the proposed numerical polyharmonic modeling using DHM-S. At the same time, we will use the numbers obtained by formulas (4.1) - (4.7) as standards for comparison and at the same time evaluate the admissibility of neglecting higher harmonics in the analytical solution.

Preparation of the task for modeling.

We write equation (4.1) in the form

$$\frac{dx_1}{dt} = x_2; \quad \frac{dx_2}{dt} + \omega_0^2 x_1 + \alpha |x_2| x_2 = h_c \cos\omega t + h_s \sin\omega t \qquad (4.8)$$

given that

$$h\sin(\omega t + \varphi) = h_c \cos\omega t + h_s \sin\omega t\ . \qquad (4.9)$$

We calculate the value of the forcing force amplitudes

$$h_c = h\sin\varphi = 100.0\sin(0.335) = 32.9\ м/c^2;$$

$$h_s = h\cos\varphi = 100.0\cos(0.335) = 94.4\ м/c^2. \qquad (4.10)$$

We reduce the system of equations (4.8) to the form (1.65)

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0 \qquad (4.11)$$

with designations

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\|; \qquad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ \omega_0^2 x_1 + \alpha |x_2| x_2 \end{matrix} \right\|; \qquad (4.12а,б)$$

$$\vec{e} = \left\| \begin{matrix} e_1 \\ e_2 \end{matrix} \right\| = \left\| \begin{matrix} 0 \\ h_c \cos \omega t + h_s \sin \omega t \end{matrix} \right\|. \qquad (4.12в)$$

The value of the derivative $d\vec{z}/d\vec{x}$, which is the diagonal block of the composite matrix of nodal differential parameters $S^*_{zв}$ of the form (1.127b) and is used in the formation of an instantaneous mathematical model of the process on a half-cycle (due to the oddity of the nonlinear dependence of the resistance force on the speed of body movement in the problem, harmonics of only odd orders are present) is as follows

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 0 & -1 \\ \omega_0^2 & 2\alpha |x_2| \end{matrix} \right\|. \qquad (4.\ 13)$$

The project "Example B.1" is implemented in the integrated program development environment (platform) ***Microsoft Developer Studio*** and its appearance in the window of this platform is shown in fig. 4.0.

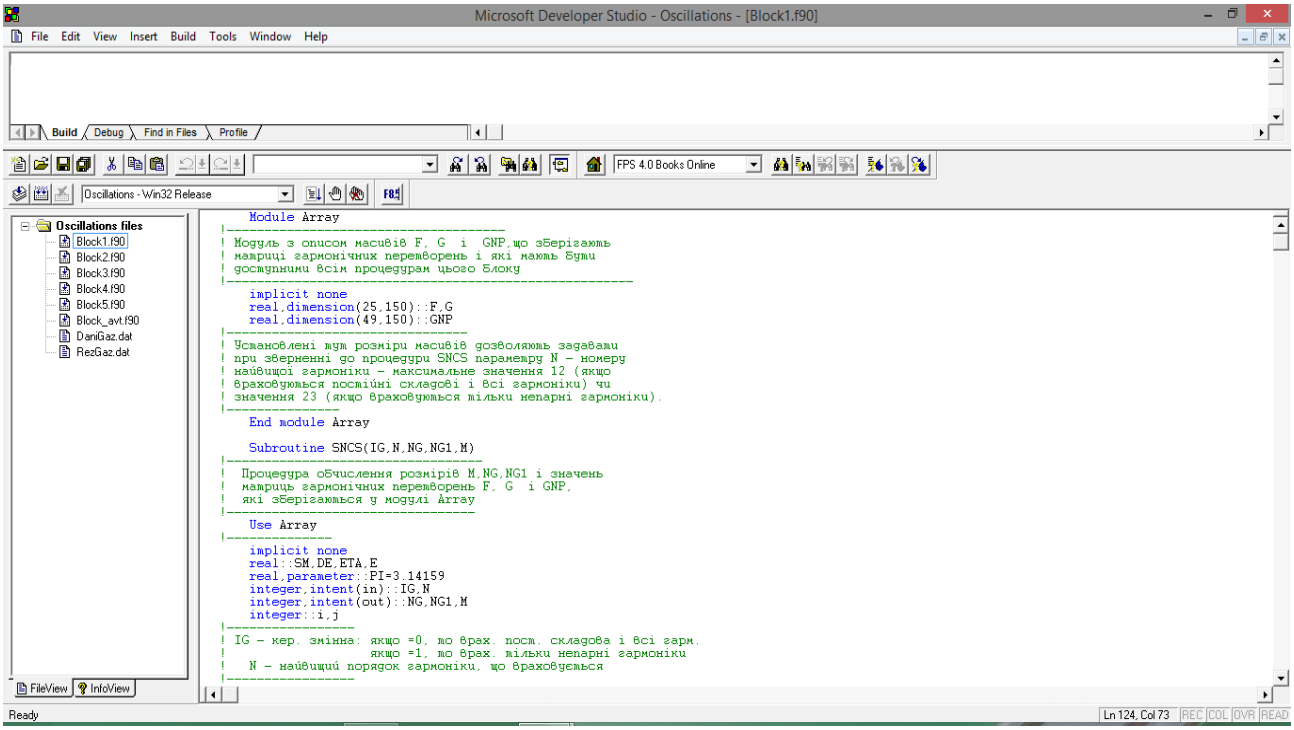


Fig. 4.0. Project B.1 foldere in the window of the integrated environment (platform) ***Microsoft Developer Studio***

Here, the blocks of software components (files ***Block1.f90, Block2.f90, Block3.f90, Block4.f90, Block5.f90***), which are described above in chapters 2 and 3, are not formed in the form of a separate library (the reader of this book and the user of the method can do it independently), so they are present in the project program package.

The block of user program components for this project consists of the main ***Program Gaz*** program and the ***Model*** procedure of the instant process model.

The main program of the user block is as follows:

```
      Program Gaz
!--  The program for determining the periodic solution
!--   of the differential equation describing the forced
!--   vibrations of a body in a gaseous environment
!--      dX/dt+Z=E
!--      X=colon(x1,x2)
!--      Z=colon(z1,z2)
!--      E=colon(e1,e2)
!--      z1=-x2;    z2=c*x1+Alfa*x2*abs(x2)
!--      e1=hc*cos(om*t);  e2=hs*sin(om*t).
!--    (variables contain only odd harmonics)
!----------------------------------------
          Implicit none
          real,dimension(18)::Y0
          real,dimension(16)::E
          integer,dimension(10)::KER
          real::C,Alfa,OM,EPS1,EPS2,H1,HM,Hc,Hs
          integer::K,NG,NK
          common/MP/C,Alfa     !---  shared memory area with the procedure Model
          open(1,File='DaniGaz.dat',status='old')
          read(1,*)C,Alfa,Hc,Hs
          read(1,*)OM,EPS1,EPS2,H1,HM
!-----  OM – circular frequency of  the fundamental harmonic
!-----  EPS1 – accuracy of integration
!-----  EPS2 – accuracy for Newton's method
!-----  H1 -  the value of h, at which it is nessesery to specify the root
!-----  HM -  maximum value h
          read(1,*)KER
          read(1,*)K
!-----  KER – the array of control variables
!-----  K – the order of the system of differential equations
       close(1)
       open(1,file='RezGaz.dat')
       write(1,5)
  5  format(4X,'Periodic solution of the equation of vibrations of a body in a gaz'/
    & 10X,'Entered data:')
      write(1,14)C,Alfa,Hc,Hs
 14 format(2X,' C = ',E10.4,' Alfa =',E10.4,' Hc = ',E10.4,' Hs = ',E10.4)
      write(1,15)OM,EPS1,EPS2,H1,HM
 15 format(2X,' OM=',E10.4,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
      write(1,16)KER
```

```
  16 format(2X,'KER =',10i5)
     write(1,17)K
  17 format(2X,' K=',i2)
     call SizesV(KER(1),K,KER(8),NG,NK)
     write(1,3)NG,NK
   3 format(2X,2i5)
!-----  NG – ther size of the simple vector of amplitudes
!-----  NK - ther size of the composite vector of amplitudes
        E=0.;  E(NG+1)=Hc;  E(NG+2)=Hs
        Y0=0;  Y0(NK+1)=OM
!--  formed a composite vector E of amplitudes of forces forcing
!--  and the initial value of the Y0 vector
     write(1,18)
  18 format(/2X,'C a l c u l a t I o n :')
     call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
     close(1)
     stop
    end Program Gaz
```

The following objects are described in the descriptive part of the main program:

– a real one-dimensional array ***Y0*** , designed to store a column vector formed from two simple vectors of amplitudes $\vec{X}_{1\Gamma}$ and $\vec{X}_{2\Gamma}$ form (1.13a), the circular frequency ω of the fundamental harmonic and the parameter h;

– real one-dimensional array ***E,*** designed to store a composite vector of amplitudes of the form (1.81), formed from two simple vectors of amplitudes $\vec{E}_{1\Gamma}$ and $\vec{E}_{2\Gamma}$ form (1.13g);

– integer one-dimensional array ***KER*** of control variables, which has 10 elements;

– real variables ***C, Alfa, OM*** - respectively, coefficients ω_0^2, α and frequency ω from equation (4.1);

– real variables ***EPS1, EPS2*** – relative accuracy of $h-$characteristic calculation and solution refinement by Newton's iterative method;

– real variables ***H1,HM*** – the value of h at which the solution should be refined and the maximum value of h;

– real variables ***Hc,Hs*** – cosine and sine amplitudes from formula (4.9);

– integer variables ***K,NG,NK*** – the order of the system of differential equations (4.8), the number of elements of the simple vector of amplitudes and the number of elements of the composite vector of amplitudes, respectively.

Next, a shared memory area named ***MP*** is described, with which the values of ***C*** and ***Alfa*** are passed from the main program to the ***Model*** procedure.

The operator part of the program begins by opening the ***DaniGaz.dat*** file, which contains the input data (a printout of this file is given below). Input data is read from this file by read statements, and then the input file is closed.

The output file ***RezGaz.dat*** is then opened and the input data just entered is written to it. Then, by calling the ***SizesV*** procedure (see section 3.1), the ***NG*** and ***NK*** values are determined and written to the output file.

Next, values are assigned to the composite vector of the amplitudes of the forcing force ***E*** and the initial value of the vector of variables ***Y0*** - the amplitudes of the harmonics are given a zero value and the NK+1 element is assigned the value of the circular frequency ω.

Let's comment on the values assigned to the elements of the ***KER*** array (see ***DaniGaz.dat*** file printout and section 3.1 below):

KER(1)= 1 (because variables x_1 and x_2 contain only odd harmonics);

KER(2)= 1 and ***KER(3)***=0 (because equation (4.8) corresponds to the form of entry (1.65));

KER(4)= 0 (because the oscillation is forced);

KER(5)= 0 (assumes that it is not necessary to print the results at each point of the $h-$ characteristic);

KER(6)= 0 (presupposes that if the $h-$characteristic had special points, then after passing the first special point, the calculation should be continued);

KER(7)= 0 (because the nonlinearity of the problem is hysteresis-free);

KER(8)= 1 (assumes that only the first harmonic is taken into account at the first stage of the calculation);

KER(9)= 7 (presupposes that after obtaining a solution taking into account only the first harmonic, the number of considered harmonics should increase from 1 to 7);

KER(10)= 1 (suggests that the results should be printed after each increment of the number of considered harmonics).

By the values of the 8th and 9th elements of the ***KER*** array, the following solution search strategy is selected (it is embedded in the ***HARMOSC*** procedure): the oscillation in the first approximation should be sought as harmonic (only the first harmonic is taken into account), and then the solution should be refined by increasing number of considered harmonics.

The main program ends with a call to the ***HARMOSC*** procedure.

The ***Model*** procedure, which implements the instantaneous model of the half-cycle process, has the following form:

```
      Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!—The procedure of the instantaneous (half-period) process model
      implicit none
      real::AL
      integer,intent(in)::M,K,MK
      real,dimension(MK)::XC,YC,ZC
      real,dimension(MK,K)::YXC,ZXC
      real,dimension(K)::X,Z
      real,dimension(K,K)::ZX,BM
      real::C,Alfa
      integer::i
      common/MP/C,Alfa !—a shared area of memory  with the main program
      BM(1,1)=0.; AL=0.      !—statements for blocking
      YC(1)=0.; YXC(1,1)=0.  !—compiler notes
      do i=1,M
        call DRAWOUTV(K,XC,MK,X,i)
        Z(1)=-X(2);  Z(2)=C*X(1)+Alfa*X(2)*abs(X(2))
        ZX(1,1)=0.;  ZX(1,2)=-1.; ZX(2,1)=C;  ZX(2,2)=2.*Alfa*abs(X(2))
        call DRAWUPV(K,Z,ZC,MK,i)
```

```
        call DRAWUPM(K,ZX,ZXC,MK,i)
      end do
      return
      end  subroutine Model
```

The interface of the Model procedure (name and list of formal parameters) is defined by the ***CALCULU*** procedure (see section 2.2.3.2).

The formal parameters of the Model procedure are:

- real variable ***AL*** is expansion coefficient of the hysteresis loop; is not used in this problem, because there are no hysteresis nonlinearities of the third group in the problem;
- integer variable ***M*** is the number of nodes per half-cycle;
- the integer variable ***K*** is the order of the system of equations (4.8);
- integer variable ***MK*** is the size of the composite nodal vector;
- real one-dimensional arrays ***XC,YC,ZC*** – respectively, composite nodal vectors $\vec{x}_в^*$, $\vec{y}_в^*$ and $\vec{z}_в^*$ of the form (1.118); in this problem, the ***YC*** array is not used, because equation (4.11) has the form (1.65) and does not contain the variable y;
- real two-dimensional arrays ***YXC, ZXC*** – composite matrices of nodal differential parameters $S_{yв}^*$ and $S_{zв}^*$ form (1.127); in this problem, the ***YXC*** matrix is not used, because, as already mentioned above, equation (4.11) has the form (1.65);
- two-dimensional array ***BM*** – matrix of coefficients of the system of differential equations, if they are of the form (1.66) or (1.67); in this problem, this matrix is not used, because equation (4.11) has the form (1.65).

These formal parameters are used to pass data from the CALCULU procedure to the ***Model*** procedure. Some data is also passed to the *Model* procedure from the main program using a shared memory area called MP.

The operative part of the procedure is opened by two operators

```
BM(1,1)=0.;    AL=0.
```

.

These operators perform unnecessary assignments. Their purpose is to block the compiler's message when compiling the procedure that the formal parameters ***BM*** and ***AL*** in the body of the procedure are not used. It was already noted above that the ***AL*** parameter is intended for use only in the case when the problem has hysteresis-type nonlinearities. There are none in this problem, so this parameter is not used. The ***BM*** parameter is provided for use in the case when the system of differential equations is not written in the normal Cauchy form. In the notation (4.11), the system of equations (4.8) has the normal Cauchy form, so the ***BM p***arameter is not used here.

The same purpose (to block compiler messages) and the following two operators

```
YC(1)=0.;    YXC(1,1)=0.
```

In this case, the equation (4.11) does not contain the vector , so the complex nodal vector ***YC*** and the matrix of nodal parameters ***YXC*** provided by the procedure are not used.

Since the nonlinearity of the problem belongs to the first group - it is unconditionally unique (see section 3.2.1), the calculation of the values of the elements of the composite nodal

vector ***ZC*** and the composite matrix of nodal parameters ***ZXC*** can occur in an arbitrary sequence. This sequence is specified by the loop operator, in which the loop parameter (variable **i** – the number of the node in the half-cycle) changes from 1 to ***M***. In the loop, for each value of **i**, the following actions are performed. The ***call DRAWOUTV(K,XC,MK,X,i)*** operator calls the ***DRAWOUTV*** procedure (it is included in Block 5 of the DHM-S), which extracts the values of the variables in the i-th node from the composite nodal vector ***XC***, which has ***MK*** elements in the form of a vector ***X*** with ***K*** elements (here - two). Next, values are calculated and assigned to the elements of the vector ***Z*** (4.12b) and the derivative matrix ***ZX*** (4.13). After that, ***call DRAWUPV(K,Z,ZC,MK,i)*** and ***call DRAWUPM(K,ZX,ZXC,MK,i)*** operators are called to perform the procedure ***DRAWUPV*** and ***DRAWUPM*** (they are included in Block 5 of the DHM-S), which assign the values of the elements of the matrices ***Z*** and ***ZX*** to the corresponding elements of the matrices ***ZC*** and ***ZXC*** (see sections 3.2.2.1, 3.2.2.3 and 3.2.2.4).

The ***DaniGaz.dat*** input file looks like this:

```
  625.    12.    32.9     94.4
  10.    0.001    0.0001    1.0     1.0
 1  1  0  0  0  0  0  1  7  1
  2
```

The ***RezGaz.dat*** source data file - calculation results after the program has been run is as follows:

```
Periodic solution of the edquation of vibrations of a body in gaz
          Entered data:
   C = .6250E+03 Alfa = .1200E+02  Hc = .3290E+02  Hs = .9440E+02
   OM= .1000E+02 EPS1= .1000E-02 EPS2= .1000E-03 H1= .1000E+01
   HM = .1000E+01
   KER =  1    1    0    0    0    0    0    1    7    1
   K= 2
      2    4

  C a l c u l a t i o n :
  Number of the highes harmonic =  1
      Refined value of 1 root for h=1.000
  (the solution was obtained after the 3th iteration)
  Amplitudes of harmonics of the 1st variable:
  X1(c1)= -.5008E-04  X1(s1)=  .1798E+00  X1(1)=  .1798E+00

  The value of the variable in nodes of half-period, M = 24
 -.5008E-04 .2342E-01 .4649E-01  .6876E-01  .8985E-01  .1094E+00
  .1271E+00 .1426E+00 .1557E+00  .1661E+00  .1737E+00  .1782E+00
  .1798E+00 .1783E+00 .1737E+00  .1661E+00  .1557E+00  .1427E+00
  .1272E+00 .1095E+00 .8994E-01  .6885E-01  .4658E-01  .2352E-01

  Amplitudes of harmonics of the 2st variable:
  X2(c1)=  .1798E+01  X2(s1)=  .5005E-03  X2(1)=  .1798E+01

  The value of the variable in nodes of half-period, M = 24
  .1798E+01 .1783E+01 .1737E+01  .1661E+01  .1557E+01  .1427E+01
  .1272E+01 .1095E+01 .8994E+00  .6885E+00  .4658E+00  .2352E+00
  .5028E-03-.2342E+00-.4649E+00 -.6876E+00 -.8985E+00 -.1094E+01
 -.1271E+01-.1426E+01-.1557E+01 -.1661E+01 -.1737E+01 -.1782E+01
```

```
  Refinement of the value of the root by increasing the number of
  harmonics from N= 1 to N= 7

 Added 3rd harmonic
  Amplitudes of harmonics of the 1st variable:
  X1(c 1)=  .9549E-03  X1(s1)=  .1815E+00  X1(1)=  .1815E+00
  X1(c 3)=  .2189E-02  X1(s3)= -.6959E-02  X1(3)=  .7296E-02

  Amplitudes of harmonics of the 2st variable:
  X2(c 1)=  .1815E+01  X2(s1)= -.9549E-02  X2(1)=  .1815E+01
  X2(c 3)= -.2088E+00  X2(s3)= -.6567E-01  X2(3)=  .2189E+00

added 5rd harmonic
  Amplitudes of harmonics of the 1st variable:
  X1(c1)=  .6234E-03  X1(s1)=  .1821E+00  X1(1)=  .1821E+00
  X1(c3)=  .2712E-02  X1(s3)= -.6942E-02  X1(3)=  .7453E-02
  X1(c5)= -.1029E-02  X1(s5)=  .4554E-03  X1(5)=  .1126E-02

  Amplitudes of harmonics of the 2st variable:
  X2(c1)=  .1821E+01  X2(s1)= -.6234E-02  X2(1)=  .1821E+01
  X2(c3)= -.2083E+00  X2(s3)= -.8136E-01  X2(3)=  .2236E+00
  X2(c5)=  .2278E-01  X2(s5)=  .5147E-01  X2(5)=  .5628E-01

added 7rd harmonic
  Amplitudes of harmonics of the 1st variable:
  X1(c1)=  .5420E-03  X1(s1)=  .1822E+00  X1(1)=  .1822E+00
  X1(c3)=  .2783E-02  X1(s3)= -.6892E-02  X1(3)=  .7433E-02
  X1(c5)= -.1051E-02  X1(s5)=  .3961E-03  X1(5)=  .1123E-02
  X1(c7)=  .2339E-03  X1(s7)=  .4659E-04  X1(7)=  .2385E-03

  The value of the variable in nodes of half-period, M = 48
  .2508E-02 .1320E-01 .2383E-01  .3439E-01  .4486E-01  .5523E-01
  .6548E-01 .7559E-01 .8554E-01  .9529E-01  .1048E+00  .1140E+00
  .1229E+00 .1315E+00 .1396E+00  .1473E+00  .1545E+00  .1612E+00
  .1674E+00 .1730E+00 .1779E+00  .1821E+00  .1854E+00  .1879E+00
  .1894E+00 .1898E+00 .1891E+00  .1872E+00  .1840E+00  .1797E+00
  .1743E+00 .1678E+00 .1604E+00  .1522E+00  .1434E+00  .1340E+00
  .1243E+00 .1143E+00 .1041E+00  .9368E-01  .8319E-01  .7262E-01
  .6198E-01 .5128E-01 .4054E-01  .2978E-01  .1900E-01  .8233E-02

  Amplitudes of harmonics of the 2st variable:
  X2(c1)=  .1822E+01  X2(s1)= -.5420E-02  X2(1)=  .1822E+01
  X2(c3)= -.2068E+00  X2(s3)= -.8349E-01  X2(3)=  .2230E+00
  X2(c5)=  .1981E-01  X2(s5)=  .5257E-01  X2(5)=  .5617E-01
  X2(c7)=  .3263E-02  X2(s7)= -.1638E-01  X2(7)=  .1670E-01

  The value of the variable in nodes of half-period, M = 48
  .1638E+01 .1630E+01 .1619E+01  .1607E+01  .1592E+01  .1576E+01
  .1556E+01 .1533E+01 .1506E+01  .1473E+01  .1433E+01  .1387E+01
  .1334E+01 .1275E+01 .1210E+01  .1140E+01  .1065E+01  .9840E+00
  .8969E+00 .8018E+00 .6968E+00  .5800E+00  .4498E+00  .3053E+00
  .1470E+00-.2346E-01-.2029E+00 -.3870E+00 -.5705E+00 -.7478E+00
 -.9135E+00-.1063E+01-.1194E+01 -.1303E+01 -.1391E+01 -.1460E+01
 -.1511E+01-.1548E+01-.1575E+01 -.1595E+01 -.1609E+01 -.1621E+01
 -.1630E+01-.1638E+01-.1643E+01 -.1646E+01 -.1646E+01 -.1644E+01
```

The results of the calculations are summarized in Table 4.1.

Table 4.1. Results of calculation of body vibrations in a gaseous environment

n	$X_{1(1)}$	$X_{1(3)}$	$X_{1(5)}$	$X_{1(7)}$	$X_{2(1)}$	$X_{2(3)}$	$X_{2(5)}$	$X_{2(7)}$
1	0,1798				1,798			
3	0,1815	0,0073			1,815	0,2188		
5	0,1821	0,0074	0,0011		1,821	0,2236	0,0563	
7	0,1822	0,0074	0,0011	0,0002	1,822	0,2230	0,0562	0,0167

As we can see from table 4.1, taking into account only the first harmonic, the obtained value of the amplitude up to the third sign coincides with the one obtained by formula (4.4). Taking into account the higher harmonics up to and including the seventh, the amplitude of the first harmonic increases slightly - by 1.3%. Higher harmonics here are more pronounced in the variable x_2, that is, in the speed of movement: the amplitude of the third harmonic of the speed of movement is 12% of the value of the amplitude of the first harmonic. The table illustrates that in this problem harmonics with orders greater than 7 can be neglected in calculations.

The above result of the calculation of the forced vibrations of a body in a gas medium, taking into account the 7th harmonics inclusive, can be represented by formulas. Thus, the time dependence of a variable x_2 (body movement speed) can be expressed by a formula

$$x_2[t] = 1.822\cos(10.0\,t) - 0.00542\sin(10.0\,t) - 0.2068\cos(30.0\,t) - 0.08349\sin(30.0\,t) + 0.01981\cos(50.0\,t) + 0.0527\sin(50.0\,t) + 0.003263\cos(70.0\,t) - 0.01638\sin(70.0\,t)\,.$$

4.1.2. Example B.2

This example illustrates the calculation of forced oscillations when the nonlinearity in the system is given analytically and is uniquely unconditional (belongs to the first group of nonlinearities, see section 3.2.1), and the differential equation describing the oscillations has not one periodic solution, but three periodic solutions. This is an equation

$$\ddot{x} + b_1\dot{x} + b_2x^3 = h\cos\omega t\ , \qquad (4.14)$$

which is known in the literature as one of the Duffing equations [65]. In particular, it describes [76] in mechanics - the movement of a load on a spring with nonlinear stiffness, in nuclear physics - the movement of a particle in a potential field with two potential wells.

We reduce equation (4.14) to the form (1.65):

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0$$

with designation

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\|; \qquad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ b_1x_2 + b_2x_1^3 \end{matrix} \right\|; \qquad (4.15a,6)$$

$$\vec{e} = \left\| \begin{matrix} e_1 \\ e_2 \end{matrix} \right\| = \left\| \begin{matrix} 0 \\ h\cos\omega t \end{matrix} \right\| . \qquad (4.15в)$$

The value of the derivative $d\vec{z}/d\vec{x}$, which is necessary in the formation of an instantaneous (half-period) mathematical model, is as follows

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 0 & -1 \\ 3b_2x_1^2 & b_1 \end{matrix} \right\| . \qquad (4.16)$$

Let's set the following values of coefficients and circular frequency:

$$b_1 = 0.2; \quad b_2 = 1.0; \quad \omega = 1.0$$

and obtain the solution by numerical simulation using DHM-S.

The ***Program Duffing*** main program and the ***Model*** procedure from the user program component block for this case look like this:

```
          Program Duffing
!--  Program for determining periodic solutions of the Duffing's equation
!--    dX/dt+Z=E
!--    X=colon(x1,x2)
!--    Z=colon(z1,z2)
!--    E=colon(e1,e2)
!--    z1=-x2;   z2=b1*x2+b2*x1**3
!--    e1=0;  e2=h*cost.
!--  (variables contain only odd harmonics)
          Implicit none
          real,dimension(14)::Y0
          real,dimension(12)::E
          integer,dimension(10)::KER
          real::B1,B2,OM,EPS1,EPS2,H1,HM
          integer::K,NG,NK
          common/MP/B1,B2    !----- shared memory area with the procedure Model
          open(1,File='DaniDuf.dat',status='old')
          read(1,*)B1,B2
          read(1,*)OM,EPS1,EPS2,H1,HM
!-----  OM – circular frequency of the fundamental harmonic
!-----  EPS1 -  accuracy of integfration
!-----  EPS2 – accuracy for Newton's method
!-----  H1 – the value of h, at which the ruquired duck root
!-----  HM – maximum value h
          read(1,*)KER
          read(1,*)K
!-----  KER – array of control variables
!-----  K – the order of the system of differential equations
          close(1)
        open(1,file='RezDuf.dat')
          write(1,5)
```

```
    5 format(4X,'Periodic solutions of the Duffing's equation' /10X,'Entered data:')
          write(1,14)B1,B2
   14 format(2X,' B1=',E10.4,' B2=',E10.4)
          write(1,15)OM,EPS1,EPS2,H1,HM
   15 format(2X,' OM=',E10.4,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
          write(1,16)KER
   16 format(2X,'KER=',10i5)
          write(1,17)K
   17 format(2X,' K=',i2)
    3  format(2X,10I5)
         call SizesV(KER(1),K,KER(8),NG,NK)
         write(1,3)NG,NK
!-----  NG – the size of the simple vector of amplitudes
!-----  NK - the size of the composite vector of amplitudes
          E=0.;   E(NG+1)=1.0
          Y0=0;  Y0(NK+1)=OM
!--  Formed the vector of amplitudes  E  of forcing forces
!--  and the initial value of the Y0 vector
     write(1,13)
   13 format(/2X,'C a l c u l a t I o n :')
      call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
      cose(1)
      stop
      end Program Duffing
!--------------------------
     Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--  The procedure of the instantaneous half-period process model
          Implicit none
          real::AL
          integer,intent(in)::M,K,MK
          real,dimension(MK)::XC,YC,ZC
          real,dimension(MK,K)::YXC,ZXC
          real,dimension(K)::X,Z
          real,dimension(K,K)::ZX,BM
          real::B1,B2
          integer::i
          common/MP/B1,B2     !--  A shared area of memory with the main program
          BM(1,1)=0.;  AL=0.
          YC(1)=0.;  YXC(1,1)=0.
          do i=1,M
           call DRAWOUTV(K,XC,MK,X,i)
           Z(1)=-X(2)
           Z(2)=B1*X(2)+B2*X(1)**3
           ZX(1,1)=0.;      ZX(1,2)=-1.
           ZX(2,1)=3.*B2*X(1)**2; ZX(2,2)=B1
           call DRAWUPV(K,Z,ZC,MK,i)
           call DRAWUPM(K,ZX,ZXC,MK,i)
          end do
          return
          end  subroutine Model
```

The ***RezDuf.dat*** file, designed to save the results, receives the following data after the program has been run:

```
       Periodic solutions of the Duffing's equation
          Entered data:
   B1= .2000E+00 B2= .1000E+01
   OM= .1000E+01 EPS1= .1000E-02 EPS2= .1000E-03
   H1= .3000E+00 HM= .6000E+00
   KER =  1    1    0    0    0    0    0    5    0    0
     K = 2
        6   12
  C a l c u l a t i o n :
  Number of the highest harmonic =  5
     ---------------------------------------
  Refined value 1 root at h= .300
  (the solution was obtained after the  3-rd iteration)

  Amplitudes of harmonics  of the  1-st variable:
  X1(c1)= -.3100E+00   X1(s1)=  .6709E-01   X1(1)=  .3172E+00
  X1(c3)= -.6841E-03   X1(s3)=  .5845E-03   X1(3)=  .8998E-03
  X1(c5)= -.1062E-05   X1(s5)=  .2510E-05   X1(5)=  .2725E-05

  Amplitudes of harmonics  of the  2-st variable:
  X2(c1)=  .6709E-01   X2(s1)=  .3100E+00   X2(1)=  .3172E+00
  X2(c3)=  .1753E-02   X2(s3)=  .2052E-02   X2(3)=  .2699E-02
  X2(c5)=  .1262E-04   X2(s5)=  .5290E-05   X2(5)=  .1369E-04
       ---------------------------------------
 Refined value 2 root at h= .300
  (the solution was obtained after the  2-rd iteration)

  Amplitudes of harmonics  of the  1-st variable:
  X1(c1)= -.7394E+00  X1(s1)=  .6751E+00   X1(1)=  .1001E+01
  X1(c3)=  .2232E-01  X1(s3)=  .2497E-01   X1(3)=  .3349E-01
  X1(c5)=  .8579E-03  X1(s5)= -.6993E-03   X1(5)=  .1107E-02

  Amplitudes of harmonics  of the  2-st variable:
  X2(c1)=  .6751E+00   X2(s1)=  .7394E+00   X2(1)=  .1001E+01
  X2(c3)=  .7491E-01   X2(s3)= -.6696E-01   X2(3)=  .1005E+00
  X2(c5)= -.3495E-02   X2(s5)= -.4290E-02   X2(5)=  .5533E-02
     ---------------------------------------
 Refined value 3 root at h= .300
  (the solution was obtained after the  3-rd iteration)

   Amplitudes of harmonics  of the  1-st variable:
     X1(c1)=  .6864E+00   X1(s1)=  .9841E+00  X1(1)=  .1200E+01
  X1(c3)= -.5973E-01   X1(s3)=  .2146E-01  X1(3)=  .6347E-01
  X1(c5)= -.1277E-03   X1(s5)= -.3154E-02  X1(5)=  .3157E-02

  Amplitudes of harmonics  of the  2-st variable:
  X2(c1)=  .9841E+00   X2(s1)= -.6864E+00  X2(1)=  .1200E+01
  X2(c3)=  .6438E-01   X2(s3)=  .1792E+00  X2(3)=  .1904E+00
  X2(c5)= -.1577E-01   X2(s5)=  .6382E-03  X2(5)=  .1578E-01
```

It should be noted that in this case, the ***KER(5)*** element of the ***KER*** control array was assigned a zero value, and therefore all $h-$characteristic points, except for $h=h_1$, were not recorded in the ***RezDuf.dat*** array. If we set the value 1 to the element ***KER(5),*** then it would be possible to trace that the $h-$characteristic when changing h from zero to 0.6 has a loop-like character and two special points, which the algorithm of the ***HARMOSC*** procedure successfully passed, using the technique of inverting differential equations (see section 1.5.7). The $h-$characteristic passed through the point $h=h_1=0.3$ three times, and thus all three periodic solutions of equation (4.14) were found. Each of the solutions is refined by Newton's iterative method.

When solving this problem, the ***KER(9***) element of the ***KER*** control array was assigned a value of zero, and the ***KER(8)*** element was assigned a value of 5, i.e., the following strategy for finding a solution was set: as a polyharmonic oscillation with the first, third, and nth harmonics without further increasing the number of harmonics taken into account.

4.1.3. Example B.3

In the previous section, the numerical modeling of forced oscillations was considered, when the nonlinearity in the system is given analytically and is unambiguously unconditional (belongs to the first group), and the differential equation to be solved has three periodic solutions. In the example considered in this section, forced nonlinear oscillations in an object with unambiguous unconditional nonlinearity are also modeled, but here it is not specified analytically, but by a table. The differential equations describing the considered oscillations, at certain parameter values, also have, as in the previous example, three periodic solutions.

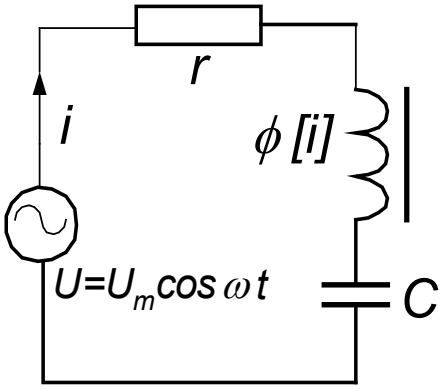


Fig. 4.1. Sequential ferro-resonance circuit

This example is a calculation of the characteristics of a simple series ferroresonant circuit shown in Fig. 4.1, which is formed from a linear active resistance r, a nonlinear inductance (saturation choke) with magnetization characteristic $\phi=\phi[i]$ and capacitance C [11]. The sought characteristic of the circuit is the dependence of the effective value I of its current on the amplitude of the applied voltage U_m.

The processes in this circuit are described by the following equations:

$$\begin{aligned} &\frac{d\phi}{dt}+ri+u_c=hU_m\cos\omega t;\\ &\frac{du_c}{dt}-\frac{1}{C}i=0, \end{aligned} \qquad (4.17)$$

where i – circuit current; ϕ – flux coupling of the coil; u_c – capacitor voltage.

This system of differential equations is reduced to the form of notation of the vector differential equation (1.64), if we accept

$$\vec{x}=\left\|\begin{matrix}x_1\\x_2\end{matrix}\right\|=\left\|\begin{matrix}i\\u_c\end{matrix}\right\|;\quad \vec{y}=\left\|\begin{matrix}y_1\\y_2\end{matrix}\right\|=\left\|\begin{matrix}\phi\\u_c\end{matrix}\right\|;\quad \vec{e}=\left\|\begin{matrix}e_1\\e_2\end{matrix}\right\|=\left\|\begin{matrix}h\,U_m\cos\omega t\\0\end{matrix}\right\|;$$
$$\vec{z}=\left\|\begin{matrix}z_1\\z_2\end{matrix}\right\|=\left\|\begin{matrix}r\,i+u_c\\-\dfrac{1}{C}i\end{matrix}\right\|=\left\|\begin{matrix}r\,x_1+x_2\\-\dfrac{1}{C}x_1\end{matrix}\right\|. \qquad (4.18)$$

The matrices of "instant" differential parameters for this case have the form

$$\frac{d\,\vec{y}}{d\,\vec{x}}=\left\|\begin{matrix}\dfrac{d\phi}{di} & 0\\0 & 1\end{matrix}\right\|;\quad \frac{d\,\vec{z}}{d\,\vec{x}}=\left\|\begin{matrix}r & 1\\-\dfrac{1}{C} & 0\end{matrix}\right\|. \qquad (4.19)$$

In [39], the calculation of nonlinear oscillations in the figure shown in Fig. 4.1 circuit (in [39] it is Fig. 3.29) by the analytical method of slowly changing amplitudes when approximating the magnetization characteristic of choke by the formula

$$\phi\,[i]=L_0(i-\gamma_0\,i^3), \qquad (4.20)$$

where L_0,γ_0 – constants. It is also noted that with such an approximation of the magnetization curve, the result can be considered adequate for the process in the circuit only when the current changes in the interval of its values, in which dependence (4.20) satisfactorily approximates the real magnetization curve. However, before calculating the oscillations, the range of possible current values is unknown in advance. Difficulties with the selection of analytical approximations of nonlinearities of magnetization curves progressively increase with the complexity of the circuit, which can be multi-circuit and contain several chokes (coils).

When numerically modeling periodic processes in the scheme of fig. 4.1 with the use of DHM-S, the non-linearity of the scheme - the dependence of the flux lincage ϕ of choke on the circuit current i - is given by the table using interpolation.

It should be noted that the algorithms for calculating periodic processes by the differential-harmonic method are insensitive to the smoothness of the functions used to approximate the characteristics of nonlinear elements. And if in the numerical integration over time of nonlinear differential equations (calculation of transient processes) it is necessary to use sufficiently smooth interpolation polynomials of the second and higher degrees, including quadratic and cubic splines [48], then in the calculation of periodic processes by the differential-harmonic method for the approximation of nonlinear characteristics it is perfectly permissible to use tables and perform linear interpolation, i.e. approximate them with a broken lines (see clause 2.2.4.2).

The block of user software components for calculating the characteristics of the ferroresonance circuit consists of: the main program ***Program Ferorez*** and the ***Model*** procedure of the instantaneous model of the process on one half-cycle.

The standard ***OUTP*** procedure is used to process the results and store them (it is included in Block 5 of the DHM-S), so it is not included in the block of user software components.

Next is a printout of the main ***Program Ferorez*** program of the block of user software components.

```
     Program Ferorez
!-- Characteristic calculation program  ferroresonance circuit
!-- by determining periodic solutions X=X(t) vector nonlinear differential equation
!-- states of the ferroresonance circuit
!--    dY/dt+Z=E
!--    X=colon(x1,x2)
!--    Y=colon(y1,y2)
!--    Z=colon(z1,z2)
!--    E=colon(e1,e2)
!--    x1=i (current);  x2=Uc (capacitor voltage)
!--    y1=Psi (flux lincage);  y2=Uc
!--    z1=R*i+Uc;    z2=-i/C,  C - capacitor,  R - resistance
!--    e1=h*Um*sin(OM*t);  e2=0.
!--  (the loop current contains only odd harmonics)
!----------------------------------------
          Implicit none
          real,dimension(14)::Y0
          real,dimension(12)::E
          integer,dimension(10)::KER
          real,dimension(30)::H
          real::B0,STEP,Um,R,C,OM,H1,HM,EPS1,EPS2
          integer::NT,K,NG,NK,i
          common/MP/H,NT,B0,STEP,R,C   !----- common area with procedure Model
          open(1,File='DaniFer.dat',status='old')
          Read(1,*)NT
          Read(1,*)B0,STEP
          Read(1,*)(H(i),i=1,NT)
          Read(1,*)Um,R,C,OM
          Read(1,*)H1,HM,EPS1,EPS2
          Read(1,*)KER
          Read(1,*)K
          Close(1)
          open(1,file='RezFer.dat')
          write(1,5)
    5 format(4X,'Entered data:')
          write(1,10)
   10 format(2X,'Weber-ampere characteristic of the choke:')
          write(1,3)NT
          write(1,4)B0,STEP
          write(1,4)(H(i),i=1,NT)
          write(1,11)
```

```
  11 format(2X,'Other circle parameters:')
          write(1,14)Um,R,C,OM
  14 format(2X,'Um=',E10.4,' R=',E10.4,' C=',E10.4,'OM=',E10.4)
          write(1,12)
  12 format(2X,'Other data:')
          write(1,15)H1,HM,EPS1,EPS2
  15 format(2X,' H1=',E10.4,' HM=',E10.4 /2X,'EPS1=',E10.4,' EPS2=',E10.4)
          write(1,3)KER
          write(1,17)K
  17 format(2X,' K=',i2)
   3 format(2X,10I5)
   4 format(2X,4E12.4)
          call SizesV(KER(1),K,KER(8),NG,NK)
          write(1,3)NG,NK
!-----  NG – the size of the simple vector of amplitudes
!-----  NK - the size of the composite vector of amplitudes
          E=0.; E(1)=Um
          Y0=0;  Y0(NK+1)=OM
!-- Formed the amplitude vector E of the coercive forces
!-- and the initial value of the Y0 vector.
      write(1,13)
  13 format(/2X,'C a l c u l a t I o n :')
       call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
      end Program Ferorez
```

The main program of this package performs:

- description of variables, including description and declaration of sizes of two arrays: ***Y0(14)*** and ***E(12). Y0*** is a column matrix formed from a vector of current amplitudes, a vector of capacitance voltage amplitudes, a circular frequency ω and a parameter h. The specified dimensions of these arrays allow you to set the maximum order of the harmonics taken into account no higher than 5 (harmonics of odd orders only are taken into account). If a higher value of the maximum harmonic order is specified, the sizes of these arrays must be increased;

- a description of the shared memory area ***/MP/,*** which is used to transfer data to the ***Model*** procedure: the magnetization curve table (***H*** – the table of the nonlinear part of the characteristic, ***NT*** – the number of nodes in the table, ***B0*** – the first value of the abscissa of the nonlinear part, ***STEP*** – the step of the table), active resistance ***R*** and capacitance ***C***;

- entering input data from the ***DaniFer.dat*** file;

- output to the ***RezFer.dat*** output file of input data;

- appeal to the ***SizesV*** procedure;

- formation of the forcing force vector ***E***;

- assignment of the initial value of the ***Y0*** array;

- referring to the *HARMOSC* procedure for calculating the h-characteristic and specifying the solution for the given value of h.

Next is a printout of the ***Model*** procedure, which implements an instantaneous (half-period) model of the periodic process in the scheme.

```
     Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--  The procedure of the "instant" model of the process on one half-cycle
!-------------------
          implicit none
          real::AL
          integer,intent(in)::M,K,MK
          real,dimension(MK)::XC,YC,ZC
          real,dimension(MK,K)::YXC,ZXC
          real,dimension(K)::X,Y,Z
          real,dimension(K,K)::YX,ZX,BM
          real,dimension(30)::H
          integer::NT,i
          real::B0,STEP,R,C,LD
          common/MP/H,NT,B0,STEP,R,C
!--  A shared area of memory with the main program
          BM(1,1)=0.; AL=0.
          do i=1,M
            call DRAWOUTV(K,XC,MK,X,i)
            call INTLIN(X(1),Y(1),LD,B0,STEP,H,NT)
!----- by interpolation calculated Y(1)=Psi and derivative LD=dPsi/di
      Y(2)=X(2)
            Z(1)=R*X(1)+X(2)
            Z(2)=-X(1)/C
            YX(1,1)=LD
            YX(1,2)=0.
            YX(2,1)=0.
            YX(2,2)=1.
            ZX(1,1)=R
            ZX(1,2)=1.
            ZX(2,1)=-1./C
            ZX(2,2)=0.
            call DRAWUPV(K,Y,YC,MK,i)
            call DRAWUPV(K,Z,ZC,MK,i)
            call DRAWUPM(K,YX,YXC,MK,i)
            call DRAWUPM(K,ZX,ZXC,MK,i)
          end do
          return
          end  Subroutine Model
```

The first two executed statements of the procedure ***BM(1,1)=0.*** and ***AL=0.*** perform unnecessary operations and are present only in order not to provoke comments by the compiler, because the formal parameters ***BM*** and ***AL*** are not used here: the parameter ***BM*** is not used because equation (4.17) does not have a matrix of coefficients, that is, it does not have the form (1.66) or (1.67), and the ***AL*** parameter is not used, because there are no elements in the problem with hysteresis characteristics.

The main operator of the model is the cycle operator, in which the variable ***i*** is the number of a node in a half-cycle. In this cycle, the following is performed:

– the ***call DRAWOUTV(K,XC,MK,X,i)*** operator calls the ***DRAWOUTV*** procedure (it is included in Block 5 of the DHM-S) for execution, which “extracts” the values of the variables x_1 (circuit

current) and x_2 (condenser voltage) from the composite nodal vector) in the i-th node of the half-period and forms a vector $\vec{x}$ from them;

- the operator ***call INTLIN(X(1),Y(1),LD,B0,STEP,H,NT)*** is called for execution the ***INTLIN*** procedure (it is included in Block 4 of the DHM-S), which, by means of linear interpolation, determines from the table that the choke magnetization curve, the value of flux lincage of choke and differential inductance is specified by it;

– a group of assignment operators gives values to arrays ***Y, Z, YX*** and ***ZX***, according to formulas (4.18) and (4.19);

– operators ***call DRAWUPV(K,Y,YC,MK,i); call DRAWUPV(K,Z,ZC,MK,i)*** the elements of the arrays ***Y*** and ***Z*** are "inserted" into the arrays ***YC*** and ***ZC*** (the latter store the values of the composite nodal vectors of the form (1.118)) in places corresponding to the i-th node of the half-period;

– operators ***call DRAWUPM(K,YX,YXC,MK,i); call DRAWUPM(K,ZX,ZXC,MK,i)*** the elements of the ***YX*** and ***ZX*** arrays are "inserted" into the ***YXC*** and ***ZXC*** arrays (the latter store the values of the composite matrices of nodal differential parameters of the form (1.127)) in the places corresponding to the i-th node of the half-period.

Printout of ***DaniFer.dat*** file with input data:

```
11
1.0 0.2
0.1   0.115   0.126   0.135   0.142   0.148   0.153   0.157
0.16   0.162   0.1635
100.   0.3   0.001   314.16
0.32   0.5   0.01   0.001
  1   0   0   0   1   0   0   5   0   0
  2
```

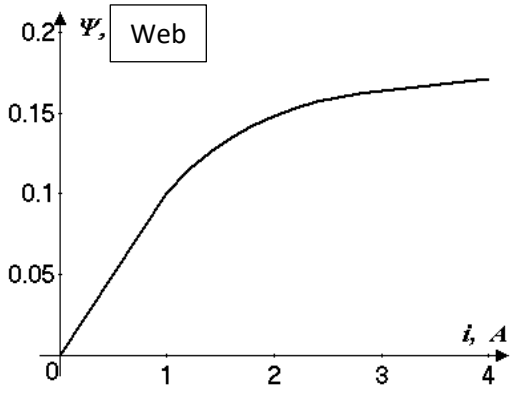


Fig. 4.2. Choke magnetization curve

The first four lines of this file are the numerical data of the tabulated choke magnetization curve shown in Fig. 4.2. The number 11 in the first line is the number of table nodes. Two numbers in the second line: 1.0 – the value of the current from which the nonlinear part begins; 0.2 - table step (table with equidistant nodes). The next 11 numbers in the third and fourth lines are the value of the flow coupling of the throttle in the nodes of the table. Fifth line: 100.0 - for U_m; 0.3 – for r; 0.001 – for C; 314.16 - for ω. The sixth line: 0.32 – for ***H1*** (the value of the parameter h at which the solution should be refined by Newton's method); 0.5 – for ***HM*** (maximum parameter value h); 0.01 - for the accuracy ε_1 of the characteristic calculation; 0.001 - for accuracy ε_2 when refining by Newton's method. The seventh line is the value of the elements of the ***KER*** control array: the first number is 1 - the variables contain only odd harmonics; the second and third numbers are 0 and 0 - the system of equations (4.17) is reduced to the form (1.64); fourth number 0 – oscillations are forced; fifth

number 1 - it is necessary to print data for all h – characteristic points; the sixth number is 0 – there is no need to stop the calculation upon reaching the first special point; the seventh number is 0 – there are no hysteresis characteristics in the problem; the eighth number 5 is the number of the highest harmonic taken into account; the ninth and tenth numbers 0 and 0 – no increase in the number of considered harmonics is assumed, the solution is immediately sought as a polyharmonic oscillation with harmonics from the first to the fifth without a subsequent increase in the number of considered harmonics.

The following is a printout of the ***RezFer.dat*** file with the output data (result):

```
    Entered data:
Weber-ampere characteristic of the choke:
   11
   .1000E+01   .2000E+00
   .1000E+00   .1150E+00   .1260E+00   .1350E+00
   .1420E+00   .1480E+00   .1530E+00   .1570E+00
   .1600E+00   .1620E+00   .1635E+00
Other circle parameters:
Um = .1000E+03 R = .3000E+00 C = .1000E-02 OM = .3142E+03
Other data:
 H1 = .3200E+00 HM = .5000E+00
EPS1 = .1000E-01  EPS2 = .1000E-02
 KER =   1    0    0    0    1    0    0    5    0    0
 K = 2
 NG = 6     NK = 12

C a l c u l a t i o n :
Number of the highest harmonic =  5
 The h-characteristic is calculating

Parameter h=  .05100, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .1919E-02  X1(s1)=  .1806E+00  X1(1)=  .1806E+00
X1(c3)= -.1000E-07  X1(s3)= -.1282E-07  X1(3)=  .1626E-07
X1(c5)= -.9916E-08  X1(s5)=  .6184E-08  X1(5)=  .1169E-07
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.5748E+00  X2(s1)=  .6106E-02  X2(1)=  .5748E+00
X2(c3)= -.8978E-06  X2(s3)=  .1498E-06  X2(3)=  .9102E-06
X2(c5)= -.8929E-06  X2(s5)=  .2132E-06  X2(5)=  .9180E-06

Parameter h=  .17600, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .6621E-02  X1(s1)=  .6232E+00  X1(1)=  .6233E+00
X1(c3)= -.3453E-07  X1(s3)= -.4424E-07  X1(3)=  .5612E-07
X1(c5)= -.3422E-07  X1(s5)=  .2134E-07  X1(5)=  .4033E-07
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.1984E+01  X2(s1)=  .2107E-01  X2(1)=  .1984E+01
X2(c3)= -.3098E-05  X2(s3)=  .5171E-06  X2(3)=  .3141E-05
X2(c5)= -.3081E-05  X2(s5)=  .7359E-06  X2(5)=  .3168E-05
     ----------------------------------------
Refined value 1 root at h= .320
(the solution was obtained after the  3-rd iteration)

Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .1244E-01  X1(s1)=  .1154E+01  X1(1)=  .1154E+01
X1(c3)= -.4680E-03  X1(s3)= -.1506E-01  X1(3)=  .1506E-01
```

```
X1(c5)=  .4823E-03  X1(s5)=  .9395E-02  X1(5)=  .9407E-02
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.3671E+01  X2(s1)=  .3959E-01  X2(1)=  .3672E+01
X2(c3)=  .1597E-01  X2(s3)= -.4957E-03  X2(3)=  .1597E-01
X2(c5)= -.5986E-02  X2(s5)=  .3083E-03  X2(5)=  .5994E-02
   ----------------------------------------
Parameter h=  .33225, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .1277E-01  X1(s1)=  .1199E+01  X1(1)=  .1199E+01
X1(c3)= -.1131E-03  X1(s3)= -.1322E-01  X1(3)=  .1322E-01
X1(c5)=  .7192E-04  X1(s5)=  .3720E-02  X1(5)=  .3720E-02
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.3815E+01  X2(s1)=  .4064E-01  X2(1)=  .3816E+01
X2(c3)=  .1402E-01  X2(s3)= -.1189E-03  X2(3)=  .1402E-01
X2(c5)= -.2374E-02  X2(s5)=  .4723E-04  X2(5)=  .2374E-02

Parameter h=  .41790, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .2017E-01  X1(s1)=  .1774E+01  X1(1)=  .1775E+01
X1(c3)= -.2506E-02  X1(s3)= -.1671E+00  X1(3)=  .1671E+00
X1(c5)=  .8612E-03  X1(s5)=  .4959E-01  X1(5)=  .4959E-01
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.5647E+01  X2(s1)=  .6419E-01  X2(1)=  .5648E+01
X2(c3)=  .1773E+00  X2(s3)= -.2657E-02  X2(3)=  .1773E+00
X2(c5)= -.3157E-01  X2(s5)=  .5502E-03  X2(5)=  .3158E-01

Parameter h=  .45149, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .3560E-01  X1(s1)=  .2350E+01  X1(1)=  .2350E+01
X1(c3)= -.2027E-01  X1(s3)= -.4279E+00  X1(3)=  .4284E+00
X1(c5)=  .1296E-01  X1(s5)=  .1409E+00  X1(5)=  .1415E+00
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.7479E+01  X2(s1)=  .1133E+00  X2(1)=  .7480E+01
X2(c3)=  .4539E+00  X2(s3)= -.2150E-01  X2(3)=  .4544E+00
X2(c5)= -.8968E-01  X2(s5)=  .8255E-02  X2(5)=  .9006E-01

Parameter h=  .46160, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .2130E+00  X1(s1)=  .5228E+01  X1(1)=  .5232E+01
X1(c3)= -.2734E+00  X1(s3)= -.1986E+01  X1(3)=  .2005E+01
X1(c5)=  .1482E+00  X1(s5)=  .6047E+00  X1(5)=  .6225E+00
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.1664E+02  X2(s1)=  .6778E+00  X2(1)=  .1665E+02
X2(c3)=  .2107E+01  X2(s3)= -.2901E+00  X2(3)=  .2127E+01
X2(c5)= -.3849E+00  X2(s5)=  .9433E-01  X2(5)=  .3963E+00

Parameter h=  .45161, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .6507E+00  X1(s1)=  .8826E+01  X1(1)=  .8849E+01
X1(c3)= -.8147E+00  X1(s3)= -.3381E+01  X1(3)=  .3477E+01
X1(c5)=  .2731E+00  X1(s5)=  .6126E+00  X1(5)=  .6708E+00
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.2809E+02  X2(s1)=  .2071E+01  X2(1)=  .2817E+02
X2(c3)=  .3586E+01  X2(s3)= -.8643E+00  X2(3)=  .3689E+01
X2(c5)= -.3900E+00  X2(s5)=  .1739E+00  X2(5)=  .4270E+00

Parameter h=  .34430, at this step variable 14 is independent
```

```
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .6777E+01  X1(s1)=  .2530E+02  X1(1)=  .2619E+02
X1(c3)= -.4689E+01  X1(s3)= -.4414E+01  X1(3)=  .6440E+01
X1(c5)= -.1396E+01  X1(s5)= -.5603E+00  X1(5)=  .1504E+01
Amplitudes of harmonics of the 2-st variable
X2(c1)= -.8052E+02  X2(s1)=  .2157E+02  X2(1)=  .8336E+02
X2(c3)=  .4683E+01  X2(s3)= -.4975E+01  X2(3)=  .6832E+01
X2(c5)=  .3566E+00  X2(s5)= -.8885E+00  X2(5)=  .9574E+00
     ----------------------------------------
Refined value 2 root at h= .320
(the solution was obtained after the  3-rd iteration)

Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .8393E+01  X1(s1)=  .2787E+02  X1(1)=  .2910E+02
X1(c3)= -.5414E+01  X1(s3)= -.4329E+01  X1(3)=  .6932E+01
X1(c5)= -.1726E+01  X1(s5)= -.2838E+00  X1(5)=  .1749E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.8870E+02  X2(s1)=  .2671E+02  X2(1)=  .9263E+02
X2(c3)=  .4592E+01  X2(s3)= -.5744E+01  X2(3)=  .7354E+01
X2(c5)=  .1805E+00  X2(s )= -.1099E+01  X2(5)=  .1114E+01
----------------------------------------
Parameter h=  .31757, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .9745E+01  X1(s1)=  .2882E+02  X1(1)=  .3043E+02
X1(c3)= -.5887E+01  X1(s3)= -.3668E+01  X1(3)=  .6936E+01
X1(c5)= -.2023E+01  X1(s5)= -.1108E+00  X1(5)=  .2026E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.9174E+02  X2(s1)=  .3102E+02  X2(1)=  .9684E+02
X2(c3)=  .3891E+01  X2(s3)= -.6246E+01  X2(3)=  .7359E+01
X2(c5)=  .7037E-01  X2(s5)= -.1287E+01  X2(5)=  .1289E+01

Parameter h=  .25073, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .2195E+02  X1(s1)=  .3566E+02  X1(1)=  .4187E+02
X1(c3)= -.8121E+01  X1(s3)=  .1117E+01  X1(3)=  .8198E+01
X1(c5)= -.1599E+01  X1(s5)=  .2856E+01  X1(5)=  .3273E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.1135E+03  X2(s1)=  .6986E+02  X2(1)=  .1333E+03
X2(c3)= -.1185E+01  X2(s3)= -.8616E+01  X2(3)=  .8697E+01
X2(c5)= -.1818E+01  X2(s5)= -.1018E+01  X2(5)=  .2084E+01

Parameter h=  .20046, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .4593E+02  X1(s1)=  .3062E+02  X1(1)=  .5520E+02
X1(c3)= -.1614E+01  X1(s3)=  .9243E+01  X1(3)=  .9383E+01
X1(c5)=  .3957E+01  X1(s5)= -.7017E+00  X1(5)=  .4018E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.9745E+02  X2(s1)=  .1462E+03  X2(1)=  .1757E+03
X2(c3)= -.9806E+01  X2(s3)= -.1713E+01  X2(3)=  .9954E+01
X2(c5)=  .4465E+00  X2(s5)=  .2519E+01  X2(5)=  .2558E+01

Parameter h=  .19796, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .5010E+02  X1(s1)=  .2761E+02  X1(1)=  .5720E+02
X1(c3)=  .6282E+00  X1(s3)=  .9425E+01  X1(3)=  .9446E+01
X1(c5)=  .3361E+01  X1(s5)= -.2444E+01  X1(5)=  .4156E+01
Amplitudes of harmonics of the 2-st variable:
```

```
X2(c1)= -.8787E+02  X2(s1)=  .1594E+03  X2(1)=  .1821E+03
X2(c3)= -.9999E+01  X2(s3)=  .6664E+00  X2(3)=  .1002E+02
X2(c5)=  .1555E+01  X2(s5)=  .2139E+01  X2(5)=  .2645E+01

Parameter h=  .19628, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .5770E+02  X1(s1)=  .2008E+02  X1(1)=  .6109E+02
X1(c3)=  .4950E+01  X1(s3)=  .8005E+01  X1(3)=  .9412E+01
X1(c5)=  .2322E+00  X1(s5)= -.4465E+01  X1(5)=  .4471E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)= -.6392E+02  X2(s1)=  .1836E+03  X2(1)=  .1944E+03
X2(c3)= -.8493E+01  X2(s3)=  .5251E+01  X2(3)=  .9985E+01
X2(c5)=  .2842E+01  X2(s5)=  .1478E+00  X2(5)=  .2846E+01

Parameter h=  .24834, at this step variable 2 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .7383E+02  X1(s1)= -.1923E+02  X1(1)=  .7629E+02
X1(c3)=  .6192E+01  X1(s3)= -.5864E+01  X1(3)=  .8528E+01
X1(c5)= -.1529E+01  X1(s5)=  .4633E+01  X1(5)=  .4879E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .6121E+02  X2(s1)=  .2350E+03  X2(1)=  .2428E+03
X2(c3)=  .6222E+01  X2(s3)=  .6568E+01  X2(3)=  .9047E+01
X2(c5)= -.2949E+01  X2(s5)= -.9731E+00  X2(5)=  .3106E+01

Parameter h=  .30532, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .7583E+02  X1(s1)= -.3982E+02  X1(1)=  .8565E+02
X1(c3)=  .5660E+00  X1(s3)= -.8466E+01  X1(3)=  .8485E+01
X1(c5)=  .3588E+01  X1(s5)=  .3239E+01  X1(5)=  .4833E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .1267E+03  X2(s1)=  .2413E+03  X2(1)=  .2726E+03
X2(c3)=  .8982E+01  X2(s3)=  .6002E+00  X2(3)=  .9002E+01
X2(c5)= -.2061E+01  X2(s5)=  .2284E+01  X2(5)=  .3076E+01
     ----------------------------------------
 Refined value 3 root at h= .320
(the solution was obtained after the  3-rd iteration)

Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .7516E+02  X1(s1)= -.4765E+02  X1(1)=  .8899E+02
X1(c3)= -.1461E+01  X1(s3)= -.9241E+01  X1(3)=  .9355E+01
X1(c5)=  .4776E+01  X1(s5)=  .1572E+01  X1(5)=  .5028E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .1517E+03  X2(s1)=  .2392E+03  X2(1)=  .2832E+03
X2(c3)=  .9803E+01  X2(s3)= -.1550E+01  X2(3)=  .9925E+01
X2(c5)= -.1001E+01  X2(s5)=  .3040E+01  X2(5)=  .3201E+01
----------------------------------------
Parameter h=  .35726, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .7658E+02  X1(s1)= -.5233E+02  X1(1)=  .9275E+02
X1(c3)= -.2501E+01  X1(s3)= -.8261E+01  X1(3)=  .8631E+01
X1(c5)=  .4818E+01  X1(s5)=  .8092E+00  X1(5)=  .4885E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .1665E+03  X2(s1)=  .2437E+03  X2(1)=  .2952E+03
X2(c3)=  .8764E+01  X2(s3)= -.2653E+01  X2(3)=  .9157E+01
X2(c5)= -.5149E+00  X2(s5)=  .3067E+01  X2(5)=  .3110E+01

Parameter h=  .43297, at this step variable 2 is independent
```

```
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .7764E+02  X1(s1)= -.6656E+02  X1(1)=  .1023E+03
X1(c3)= -.5058E+01  X1(s3)= -.7139E+01  X1(3)=  .8749E+01
X1(c5)=  .4549E+01  X1(s5)= -.1651E+01  X1(5)=  .4840E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .2118E+03  X2(s1)=  .2471E+03  X2(1)=  .3255E+03
X2(c3)=  .7574E+01  X2(s3)= -.5366E+01  X2(3)=  .9282E+01
X2(c5)=  .1051E+01  X2(s5)=  .2896E+01  X2(5)=  .3081E+01

Parameter h=  .50740, at this step variable 14 is independent
Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .7908E+02  X1(s1)= -.7798E+02  X1(1)=  .1111E+03
X1(c3)= -.6292E+01  X1(s3)= -.6074E+01  X1(3)=  .8746E+01
X1(c5)=  .3726E+01  X1(s5)= -.2837E+01  X1(5)=  .4683E+01
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .2482E+03  X2(s1)=  .2517E+03  X2(1)=  .3535E+03
X2(c3)=  .6444E+01  X2(s3)= -.6676E+01  X2(3)=  .9279E+01
X2(c5)=  .1806E+01  X2(s5)=  .2372E+01  X2(5)=  .2981E+01
```

The ***HARMOSC*** procedure provided writing to the output file of the entire h-characteristic (because ***KER(5)***=1 was set), because it exactly reflects the characteristic of the ferroresonant circuit. This characteristic is shown in fig. 4.3 with a solid line.

Here it should be taken into account that the figure shows the dependence on the amplitude of the applied voltage, not of the effective value of the current, but of the amplitude of its first harmonic. This is due to the fact that the standard ***OUTP*** procedure from Block 5 of the DHM-S was used, which does not provide for the calculation of the actual values of variables. If such additional calculations and printouts are necessary, the user must develop the ***OUTP*** procedure independently.

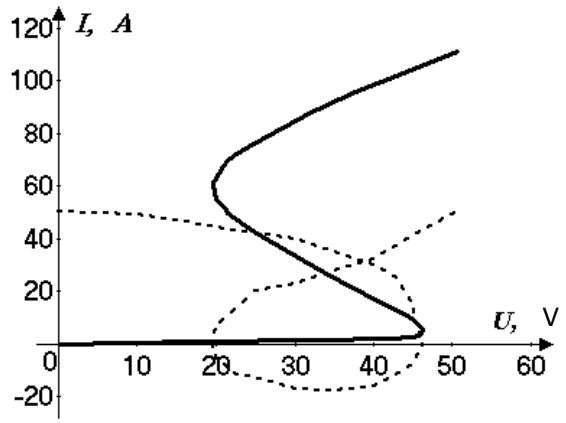


Fig. 4.3. Characteristic of the ferroresonance circuit

In fig. 4.3 also shows the dependence on the amplitude of the applied voltage of the value proportional to the value of the free term of the characteristic equation of the form (1.138). This curve (it is shown dotted) illustrates that all stable modes (nonlinear oscillations) that lie on the first part of the characteristic (variation of the parameter h from zero to 0.4616) are aperiodically stable, because the free term of the characteristic equation has a plus sign. Steady processes belonging to the second part of the characteristic (the parameter decreases from 0.4616 to 0.196) are aperiodically unstable, because when moving to this part, the free term of the characteristic equation changes its sign from plus to minus and keeps this sign throughout this part. The steady processes belonging to the third part of the characteristic are aperiodically stable, because when moving to this part, the free term again changes its sign from minus to plus.

Calculation of the value of the free term of the characteristic equation can be provided in the ***OUTP*** procedure, for this the user needs to modify this procedure.

In fig. 4.3 is shown the dependence of the parameter h on the amplitude of the first harmonic of the current, it has two special points $h \approx 0.4616$ and $h \approx 0.196$. Their passage in the process of calculating the $h-$characteristic was carried out by the ***HARMOSC*** procedure using the inversion algorithm described in section 1.5.7. This can be tracked by analyzing the printout of the output file: there you can see when the independent variable when moving along the characteristic is a parameter h (here it is the 14th component of the vector of variables), and when the independent variable is the sine amplitude of the first harmonic of the current (here it is the 2th component of the vector variables).

Three periodic solutions of the system of equations (4.17) at $h = 0.32$ using the ***HARMOSC*** procedure were refined by Newton's iterative method (this value of h corresponds $U_m = 32.0\ V$). In the refined first periodic solution, the content of higher harmonics in the current curve is insignificant: the third harmonic is 1.3%; the fifth harmonic is 0.8% (in this stable mode, the choke saturation is negligible). In the refined second periodic solution (aperiodically unstable), the content of higher harmonics is as follows: third – 23.8% and fifth – 5.8%. In the refined third periodic solution, the content of higher harmonics is as follows: the third – 10.5% and the fifth – 5.6%.

4.1.4. Example B.4

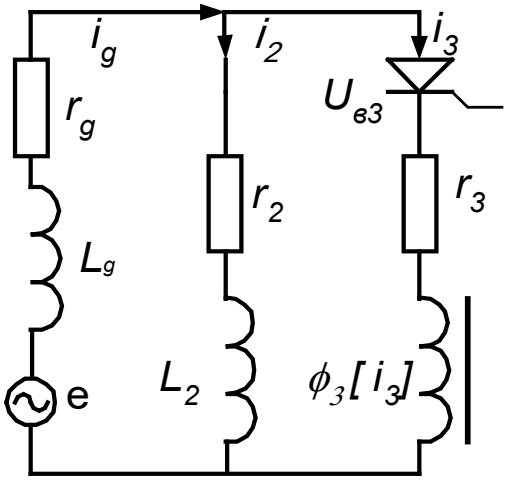


Fig.4.4 The circuit with choke and thyristor

This example considers the calculation of a periodic process in an electrical circuit containing a controlled valve (see Fig. 4.4). The problem uses the instantaneous model described in section 3.2.1.1 on the period of the controlled valve, the volt-ampere characteristic of which belongs to the second group - is unique with the condition.

The considered scheme of fig. 4.4 is non-linear, since its third branch contains two non-linear elements: a controlled valve and a saturation choke.

This scheme is described by the following system of algebraic-differential equations:

$$\begin{aligned} & i_g - i_2 - i_3 = 0; \\ & \frac{d\phi_g}{dt} + r_g i_g + \frac{d\phi_2}{dt} + r_2 i_2 = e; \\ & \frac{d\phi_2}{dt} + r_2 i_2 - \frac{d\phi_3}{dt} + r_3 i_3 - u_{в3} = 0, \end{aligned} \qquad (4.21)$$

here i_g, i_2, i_3 - circuit branch currents; $u_{в3}$ - valve voltage (is a non-linear function of the current i_3); L_g, L_2 - inductances of the first two branches; ϕ_g, ϕ_2, ϕ_3 - flux coupling of the inductive elements of the branches, at the same time

$$\phi_g = L_g\, i_g; \quad \phi_2 = L_2\, i_2 \ ; \tag{4.22}$$

$$\phi_3 = \phi_3[\, i_3\,] \tag{4.23}$$

- magnetization curve of the magnetic choke of the third branch;

$$e = E_m \sin\omega t \tag{4.24}$$

- electromotive force in the first branch.

Equation (4.21) is reduced to the form (1.66) if we denote:

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \\ x_3 \end{matrix} \right\| = \left\| \begin{matrix} i_g \\ i_2 \\ i_3 \end{matrix} \right\|; \quad \vec{y} = \left\| \begin{matrix} y_1 \\ y_2 \\ y_3 \end{matrix} \right\| = \left\| \begin{matrix} \phi_g \\ \phi_2 \\ \phi_3 \end{matrix} \right\|; \quad \vec{e} = \left\| \begin{matrix} 0 \\ e \\ 0 \end{matrix} \right\|;$$

$$\vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \\ z_3 \end{matrix} \right\| = \left\| \begin{matrix} i_g - i_2 - i_3 \\ r_g\, i_g + r_2\, i_2 \\ r_2\, i_2 - r_3\, i_3 - u_{в3}[i_3] \end{matrix} \right\| = \left\| \begin{matrix} x_1 - x_2 - x_3 \\ r_g\, x_1 + r_2\, x_2 \\ r_2\, x_2 - r_3\, x_3 - u_{в3}[x_3] \end{matrix} \right\|. \tag{4.25}$$

At the same time, the matrix of the form (1.69) is as follows:

$$B = \left\| \begin{matrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & -1 \end{matrix} \right\|. \tag{4.26}$$

The matrices of differential parameters necessary for building an instantaneous process model (on a period) have the form

$$d\vec{y} \big/ d\vec{x} = diag(L_g, L_2, L_3^{\partial}[i_3]); \tag{4.27}$$

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 1 & -1 & -1 \\ r_g & r_2 & 0 \\ 0 & r_2 & -r_3 - r_в \end{matrix} \right\|, \tag{4.28}$$

here $L_3^{\partial} = d\phi_3 / di_3 = L_3^{\partial}[i_3]$ – the differential inductance of the choke of the third branch, it is a nonlinear function of the current i_3; $r_в = du_{в3} / di_3 = r_в[i_3]$ – the differential active resistance of the valve, it is a nonlinear function of the current i_3.

The text of the main program ***Program Ven*** of the block of user software components:

```
    Program Ven
! The program of calculation of periodic process  in the scheme with a valve (Fig. 4.4)
!-------------------------
          implicit none
          real,dimension(53)::X
          real,dimension(51)::E
          integer,dimension(10)::KER
          real,dimension(3,3)::B
          real,dimension(20)::PS3
          real::ALG,AL2,RG,R2,R3,ST31,DS3,Rmax,Rmin,AZ3,DAZ3
          real::Em,OM,EPS1,EPS2,H1,HM
          integer::i,NT3,K,NG,NK
          common/MPM/B,ALG,AL2,RG,R2,R3, NT3,ST31,DS3,PS3,Rmax,Rmin,AZ3,DAZ3
!----  common/MPM/ passes data to the Model procedure
         open(1,File='DaniV.dat',status='old')
          read(1,1)Em,OM,EPS1,EPS2,H1,HM
          read(1,1)B
!---  Em,OM - the amplitude of the applied voltage and its frequency
!---   B  - the matrix of coefficients of the system of differential equations
          read(1,1)ALG,AL2,RG,R2,R3
!---  ALG,AL2,RG, R2, R3 - inductances and active resistors of the scheme
    Read(1,2)NT3
          read(1,1)ST31,DS3
          read(1,1) (PS3(i),i=1,NT3)   !--  magnetiz. curve of the choke
          read(1,1)Rmax,Rmin,AZ3,DAZ3  !-- valve parameters
          read(1,2)KER
          read(1,2)K
  1 format(4E10.4)
  2 format(10I3)
          close(1)
     open(1,FILE='RezV.DAT')
     write(1,10)
 10 format(/5X, 'Input data of the task:'/)
         write(1,4)Em,OM,EPS1,EPS2,H1,HM
          write(1,4)B
          write(1,4)ALG,AL2,RG,R2,R3
          write(1,3)NT3
          write(1,4)ST31,DS3
          write(1,4)(PS3(i),i=1,NT3)
          write(1,4)Rmax,Rmin,AZ3,DAZ3
          write(1,3)KER
          write(1,3)K
  3 format(2X,10I5)
```

```
   4 format(2X,4E12.4)
          write(1,20)
  20 format(/5X,'T h e  r e s u l t s  :'/)
       call SizesV(KER(1),K,KER(8),NG,NK)
          write(1,3)NG,NK
!---  a composite vector of amplitudes of forcing forces
!---  (a sinusoid with amplitude Em) and the initial value of the vector X are formed
       E=0.;  E(NG+3)=Em
       X=0.;  X(NK+1)=OM
      call HARMOSC(K,X,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
      end Program Ven
```

The main program of this package performs:

- description of variables, including:

a) description and declaration of the size of arrays ***X(53), E(51)***. Here ***X*** is a column matrix formed from the composite vector of current amplitudes of three branches, circular frequency and parameter h ; ***E*** is the composite vector of forcing force amplitudes. The set sizes of these two arrays allow you to set the maximum order of the considered harmonics no higher than 6 (the problem takes into account constant components and harmonics of all even and odd orders);

b) description of the ***KER(10)*** array - control variables of the integer type;

c) description of the array ***B(3,3)*** - for the matrix (4.26) of the coefficients of the algebraic-differential system of equations (4.21), reduced to the form (1.66);

d) description of the ***PSN(20)*** array - for the table, which specifies the nonlinear part of the choke magnetization curve (obviously, the number of nodes in the table should not be more than 20);

e) description of scalar variables of integer and real types;

– a description of the shared memory area called ***MPM***, with its help, data is transferred to the ***Model*** procedure),

– entering input data from the *DaniV.dat* file;

– output of this data to the ***RezV.dat*** source file;

– appeal to the ***SizesV*** procedure (included in Block 5 of the DHM-S);

– assigning a value to the composite vector of amplitudes ***E***;

– appeal to the ***HARMOSC*** procedure (included in Block 3 of the DHM-S).

The text of the *Model* procedure of the instantaneous process model (on the period) in the scheme of fig. 4.4:

```
    Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!-- The subroutine implements the instantaneous model of process in the circuit with valve
          Implicit none
          integer,intent(in)::M,K,MK
          real,dimension(MK)::XC,YC,ZC
          real,dimension(MK,K)::YXC,ZXC
          real,dimension(K)::X,Y,Z
          real,dimension(K,K)::YX,ZX,BM
```

```
      real,dimension(M)::XV,UV,RV,UV1,RV1
      real,dimension(3,3)::B
      real,dimension(20)::PS3
      real::ALG,AL2,RG,R2,R3,ST31,DS3,YX3,Rmax,Rmin,AZ3,DAZ3,AL
      integer::NT3,i
      common/MPM/B,ALG,AL2,RG,R2,R3,NT3,ST31,DS3,PS3,Rmax,Rmin,AZ3,DAZ3
!--  through Common/MPM/ the data from the main program is transferred
       BM=B;  AL=0.
      do i=1,M
        call DRAWOUTV(K,XC,MK,X,i)
        Y(1)=ALG*X(1);  Y(2)=AL2*X(2)
        call INTLIN(X(3),Y(3),YX3,ST31,DS3,PS3,NT3)
        Z(1)=X(1)-X(2)-X(3)
        Z(2)=RG*X(1)+R2*X(2)
        Z(3)=R2*X(2)-R3*X(3)
        ZX(1,1)=1.;  ZX(1,2)=-1.;   ZX(1,3)=-1.
        ZX(2,1)=RG;    ZX(2,2)=R2;    ZX(2,3)=0.
        ZX(3,1)=0.;    ZX(3,2)=R2;    ZX(3,3)=-R3
        YX(1,1)=ALG;   YX(1,2)=0.;    YX(1,3)=0.
        YX(2,1)=0.;    YX(2,2)=AL2;   YX(2,3)=0.
        YX(3,1)=0.;    YX(3,2)=0.;    YX(3,3)=YX3
        call DRAWUPV(K,Y,YC,MK,i)
        call DRAWUPV(K,Z,ZC,MK,i)
        call DRAWUPM(K,YX,YXC,MK,i)
        call DRAWUPM(K,ZX,ZXC,MK,i)
      end do
      call DRAWOUTXV(K,XC,MK,M,XV,3)
      call Venper(XV,M,AZ3,DAZ3,Rmax,Rmin,UV,RV)
      UV1=-UV;  RV1=-RV
      call ADDV(K,UV1,M,ZC,MK,3)
      call ADDM(K,RV1,M,ZXC,MK,3)
      return
      end    subroutine Model
```

The ***Model*** procedure based on the value of the vector $\vec{x}$ of instantaneous values of the independent variables - currents of the circuit branches - calculates in a cycle at all nodes of the period the values of vectors $\vec{y}, \vec{z}$ and matrices $d\vec{y}/d\vec{x}$ (YX) and $d\vec{z}/d\vec{x}$ (ZX) according to formulas (4.25), (4.27) and (4.28). However, in this case, the value "$-u_{в3}$" is not added to the element ***Z(3)*** and the value "$-r_3$" is not added to the element ***ZX(3,3)***, as required by formula (4.25), because these data belong to the controlled gate - nonlinearities of the second group. These additional data are determined already after the end of the cycle by calling the ***Venper*** procedure from Block 5 of the DHM-S, which implements the instant-on-period model of the controlled valve. To do this, the ***DRAWOUTXV*** procedure is first called for execution, which extracts a simple nodal vector ***XV*** (the value of the currents of the third branch in the nodes of the period) from the complex nodal vector of currents ***XC***. The ***Venper*** procedure is then run, which calculates the values of the simple node vectors ***UV*** (valve voltage at period nodes) and

RV (valve resistance at period nodes). And, at the end, it enters this data using the ***ADDV*** and ***ADDM*** procedures (see sections 3.2.2.5 and 3.2.2.6) into the ***ZC*** and ***ZXC*** arrays.

Input data (***DaniV.dat*** file):

```
+.2000E 01+.3142E 03+.1000E-01+.3000E-02
+.1000E 01+.1000E 01
+.0000E 00+.1000E 01+.0000E 00+.0000E 00
+.1000E 01+.1000E 01+.0000E 00+.0000E 00
-.1000E 01
+.3000E-03+.9000E-03+.1000E 00+.3000E 00
+.4000E 00
011
+.1000E 01+.2000E 00
+.1500E-02+.1730E-02+.1890E-02+.2025E-02
+.2130E-02+.2220E-02+.2295E-02+.2355E-02
+.2400E-02+.2430E-02+.2445E-02
+.1000E 04+.1000E-03+.5000E 00+.3000E 00
00000000100000000000008000000
003
```

In this file:

The first line is data for assignment to variables E_m, ω, ε_1 (accuracy of $h-$ characteristic calculation) and ε_2 (accuracy for iterations when refining the connection).

The second line is for assigning values to the variables Δh (initial value of the step), h_1 - (value of the parameter h at which the solution should be refined), $h_{\max}$ (final value of the parameter h).

The third, fourth and fifth lines are the matrix B according to the formula (4.26), read column by column.

The sixth and seventh lines are for variables L_g, L_2, r_g, r_2, r_3.

The next five lines are the magnetization curve of the choke: the number of points, the coordinate of the start of the non-linear part, the step of the table and the table.

The thirteenth line is for variables $r_{\max}, r_{\min}, \alpha_з$ (ignition angle), $\Delta\alpha_з$ (ignition pulse width, its value should not be less than the angular distance between neighboring nodes in the period).

The last two lines are numbers of the integer type: for the ***KER*** array and for the variable k - the order of the system of equations (4.21).

Calculation results (***RezV.dat*** file):

```
 Input data of the task:
 .2000E+01   .3142E+03   .1000E-01   .3000E-02
 .1000E+01   .1000E+01
 .0000E+00   .1000E+01   .0000E+00   .0000E+00
 .1000E+01   .1000E+01   .0000E+00   .0000E+00
-.1000E+01
 .3000E-03   .9000E-03   .1000E+00   .3000E+00
 .4000E+00
 11
 .1000E+01   .2000E+00
```

```
    .1500E-02   .1730E-02   .1890E-02   .2025E-02
    .2130E-02   .2220E-02   .2295E-02   .2355E-02
    .2400E-02   .2430E-02   .2445E-02
    .1000E+04   .1000E-03   .5000E+00   .3000E+00
     0    0    1    0    0    0    0    8    0    0
     3
    T h e  r e s u l t s :

    17   51
 Number of the highest harmonic = 8
    ----------------------------------------
 Refined value 1 root at h= 1.000
 (the solution was obtained after the 12-rd iteration)

 Amplitudes of harmonics of the 1-st variable:
 Constant component =   .5783E+00
 X1(c1)= -.3057E+01    X1(s1)=  .3380E+01    X1(1)=  .4558E+01
 X1(c2)= -.1332E+00    X1(s2)= -.4176E+00    X1(2)=  .4384E+00
 X1(c3)=  .9837E-01    X1(s3)= -.1036E-01    X1(3)=  .9891E-01
 X1(c4)=  .2848E-01    X1(s4)=  .4142E-01    X1(4)=  .5027E-01
 X1(c5)= -.2363E-01    X1(s5)= -.2090E-01    X1(5)=  .3155E-01
 X1(c6)=  .1114E-01    X1(s6)=  .1083E-01    X1(6)=  .1554E-01
 X1(c7)=  .4702E-02    X1(s7)= -.9364E-02    X1(7)=  .1048E-01
 X1(c8)= -.2168E-03    X1(s8)=  .9578E-02    X1(8)=  .9581E-02

The value of the variable in nodes of period, M =102
 .2493E+01-.2326E+01-.2151E+01 -.1968E+01 -.1778E+01 -.1579E+01
-.1371E+01-.1153E+01-.9237E+00 -.6828E+00 -.4305E+00 -.1675E+00
 .1048E+00 .3850E+00 .6713E+00  .9622E+00  .1257E+01  .1554E+01
 .1854E+01 .2156E+01 .2460E+01  .2766E+01  .3072E+01  .3377E+01
 .3676E+01 .3967E+01 .4245E+01  .4506E+01  .4745E+01  .4960E+01
 .5147E+01 .5306E+01 .5435E+01  .5535E+01  .5605E+01  .5647E+01
 .5661E+01 .5648E+01 .5608E+01  .5542E+01  .5451E+01  .5336E+01
 .5201E+01 .5047E+01 .4878E+01  .4697E+01  .4507E+01  .4310E+01
 .4108E+01 .3900E+01 .3685E+01  .3462E+01  .3228E+01  .2981E+01
 .2721E+01 .2446E+01 .2160E+01  .1864E+01  .1562E+01  .1260E+01
 .9606E+00 .6692E+00 .3886E+00  .1205E+00 -.1346E+00 -.3773E+00
-.6093E+00-.8324E+00-.1048E+01 -.1259E+01 -.1465E+01 -.1666E+01
-.1863E+01-.2053E+01-.2235E+01 -.2409E+01 -.2573E+01 -.2726E+01
-.2868E+01-.3000E+01-.3120E+01 -.3229E+01 -.3326E+01 -.3411E+01
-.3484E+01-.3543E+01-.3588E+01 -.3619E+01 -.3636E+01 -.3639E+01
-.3628E+01-.3603E+01-.3566E+01 -.3515E+01 -.3451E+01 -.3375E+01
-.3286E+01-.3183E+01-.3068E+01 -.2941E+01 -.2802E+01 -.2652E+01

 Amplitudes of harmonics of the 2-st variable:
 Constant component =  -.1928E+00
 X2(c1)= -.2308E+01  X2(s1)=  .2403E+01   X2(1)=  .3332E+01
 X2(c2)=  .4440E-01  X2(s2)=  .1392E+00   X2(2)=  .1461E+00
 X2(c3)= -.3280E-01  X2(s3)=  .3455E-02   X2(3)=  .3298E-01
 X2(c4)= -.9498E-02  X2(s4)= -.1381E-01   X2(4)=  .1676E-01
 X2(c5)=  .7872E-02  X2(s5)=  .6966E-02   X2(5)=  .1051E-01
 X2(c6)= -.3719E-02  X2(s6)= -.3610E-02   X2(6)=  .5183E-02
 X2(c7)= -.1573E-02  X2(s7)=  .3123E-02   X2(7)=  .3496E-02
 X2(c8)=  .6675E-04  X2(s8)= -.3191E-02   X2(8)=  .3192E-02
```

```
 The value of the variable in nodes of period, M =102
-.2497E+01-.2329E+01-.2152E+01 -.1966E+01 -.1773E+01 -.1574E+01
-.1371E+01-.1163E+01-.9536E+00 -.7431E+00 -.5328E+00 -.3234E+00
-.1158E+00  .8959E-01  .2921E+00  .4911E+00  .6859E+00  .8755E+00
 .1059E+01  .1235E+01  .1404E+01  .1563E+01  .1713E+01  .1852E+01
 .1983E+01  .2104E+01  .2216E+01  .2320E+01  .2417E+01  .2507E+01
 .2590E+01  .2667E+01  .2736E+01  .2798E+01  .2853E+01  .2898E+01
 .2935E+01  .2963E+01  .2981E+01  .2990E+01  .2989E+01  .2977E+01
 .2954E+01  .2919E+01  .2873E+01  .2812E+01  .2739E+01  .2651E+01
 .2550E+01  .2436E+01  .2310E+01  .2173E+01  .2028E+01  .1875E+01
 .1715E+01  .1551E+01  .1381E+01  .1206E+01  .1027E+01  .8416E+00
 .6505E+00  .4532E+00  .2498E+00  .4067E-01 -.1731E+00 -.3901E+00
-.6087E+00-.8273E+00-.1044E+01 -.1257E+01 -.1466E+01 -.1668E+01
-.1864E+01-.2052E+01-.2233E+01 -.2405E+01 -.2568E+01 -.2722E+01
-.2866E+01-.2999E+01-.3120E+01 -.3230E+01 -.3327E+01 -.3411E+01
-.3482E+01-.3540E+01-.3585E+01 -.3616E+01 -.3634E+01 -.3638E+01
-.3628E+01-.3604E+01-.3567E+01 -.3516E+01 -.3451E+01 -.3373E+01
-.3283E+01-.3180E+01-.3065E+01 -.2939E+01 -.2802E+01 -.2655E+01

 Amplitudes of harmonics of the 3-st variable:
 Constant component =   .7711E+00
 X3(c1)= -.7486E+00   X3(s1)=  .9771E+00   X3(1)=  .1231E+01
 X3(c2)= -.1776E+00   X3(s2)= -.5568E+00   X3(2)=  .5845E+00
 X3(c3)=  .1312E+00   X3(s3)= -.1382E-01   X3(3)=  .1319E+00
 X3(c4)=  .3798E-01   X3(s4)=  .5523E-01   X3(4)=  .6703E-01
 X3(c5)= -.3150E-01   X3(s5)= -.2786E-01   X3(5)=  .4206E-01
 X3(c6)=  .1486E-01   X3(s6)=  .1444E-01   X3(6)=  .2073E-01
 X3(c7)=  .6274E-02   X3(s7)= -.1249E-01   X3(7)=  .1397E-01
 X3(c8)= -.2835E-03   X3(s8)=  .1277E-01   X3(8)=  .1277E-01

 The value of the variable in nodes of period, M =102
 .3340E-02  .2912E-02 .1035E-02-.1773E-02 -.4301E-02 -.4705E-02
-.7083E-03  .1009E-01  .2987E-01 .6029E-01 .1022E+00  .1559E+00
 .2206E+00 .2954E+00  .3792E+00 .4711E+00  .5708E+00  .6783E+00
 .7945E+00 .9202E+00  .1056E+01 .1203E+01  .1360E+01  .1524E+01
 .1693E+01 .1863E+01  .2029E+01 .2186E+01  .2328E+01  .2453E+01
 .2557E+01 .2639E+01  .2699E+01 .2736E+01  .2752E+01  .2749E+01
 .2726E+01 .2685E+01  .2626E+01 .2552E+01  .2462E+01  .2359E+01
 .2247E+01 .2127E+01  .2005E+01 .1884E+01  .1768E+01  .1659E+01
 .1558E+01 .1464E+01  .1375E+01 .1289E+01  .1200E+01  .1107E+01
 .1005E+01 .8959E+00  .7790E+00 .6574E+00  .5354E+00  .4179E+00
 .3100E+00 .2160E+00  .1388E+00 .7980E-01  .3846E-01  .1274E-01
-.5431E-03-.5090E-02 -.4538E-02-.1953E-02  .5018E-03  .1672E-02
 .1313E-02-.1600E-03 -.2006E-02-.3483E-02 -.4090E-02 -.3696E-02
-.2524E-02-.1040E-02  .2268E-03 .8600E-03  .6805E-03 -.2104E-03
-.1479E-02-.2676E-02 -.3385E-02-.3357E-02 -.2585E-02 -.1307E-02
 .6975E-04 .1102E-02  .1452E-02 .1001E-02 -.1008E-03 -.1474E-02
-.2620E-02-.3080E-02 -.2595E-02-.1219E-02  .6630E-03  .2416E-02
```

Comment on the **RezV.dat** file with the rezults*:*

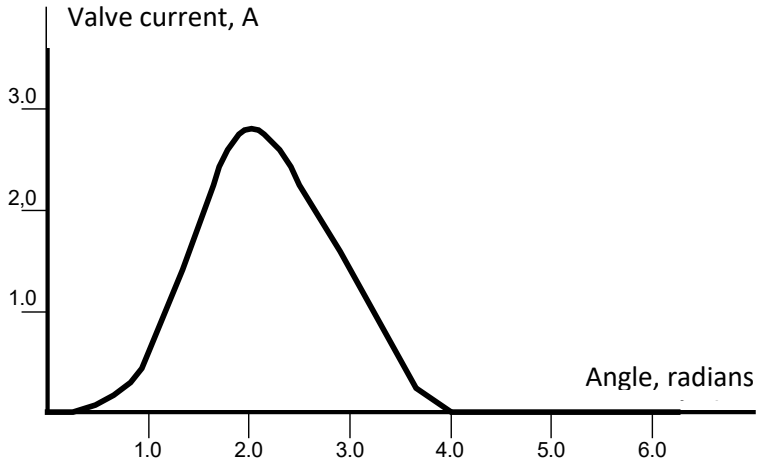


Fig. 4.5 Valve current per period

After completing the calculation of the $h-$characteristics, the obtained value of the composite vector of current amplitudes of the branches of the circuit was refined according to Newton's method (the specified accuracy was obtained after the 12th iteration). For each variable (circuit branch currents), the constant components and amplitudes of harmonics, cosine and sine components of these amplitudes, as well as the values of these variables in period nodes are printed.

The dependence of the third variable on one period of the angular coordinate - the current of the third branch with a controlled valve - is shown in Fig. 4.5. The value of the ignition angle was set to $\alpha_3 = 0.5$. Due to the presence of a fairly significant inductance in the branch with the valve, the valve is in a state of conduction from 0.5 to almost 4 radians (the applied voltage is a sinusoid).

4.1.5. Example B.5

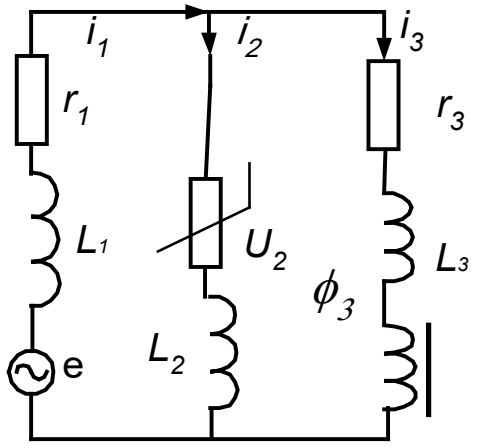


Fig. 4.6 Scheme with magnetic choke

In this example, nonlinear oscillations are simulated in a system that contains an ambiguous nonlinearity of the hysteresis type (nonlinearity of the third group), which is described in section 3.2.1.2. Such a system is the electric circuit shown in fig. 4.6. In the third branch of the circuit there is a choke (nonlinear inductance), the Weber-ampere characteristic of which has a hysteresis form. In addition, in the second branch of the circuit there is a nonlinear active resistance with a current-current characteristic that is given analytically and is unambiguously unconditional (nonlinearity of the first group).

This scheme is described by the following algebraic-differential system of equations:

$$
\begin{aligned}
& i_1 - i_2 - i_3 = 0; \\
& \frac{d\phi_1}{dt} + r_1 i_1 + \frac{d\phi_2}{dt} + u_2 = e; \\
& \frac{d\phi_2}{dt} + u_2 - \frac{d\phi_3}{dt} - r_3 i_3 = 0,
\end{aligned}
\tag{4.29}
$$

here
$$e = E_m \sin\omega t \tag{4.30}$$

- external electromotive force;

i_1, i_2, i_3 - circuit branch currents;

$u_2 = u_2[i_2]$ - the voltage on the active resistance of the second branch is a non-linear function of the current i_2 ;

ϕ_1, ϕ_2, ϕ_3 - flux coupling of the inductances of the branches, at the same time

$$\phi_1 = L_1 i_1; \quad \phi_2 = L_2 i_2 \ , \tag{4.31}$$

L_1 , L_2 - constant inductances of the first two branches;

$$\phi_3 = L_3 i_3 + \phi_Д [i_3] \ , \tag{4.32}$$

L_3 - constant inductance of the third branch;

$$\phi_Д = \phi_Д [i_3] \tag{4.33}$$

- the magnetization curve of the choke in the third branch, it is hysteretic.

Equations (4.29) are reduced to the form (1.66) if we denote:

$$
\begin{aligned}
& \vec{x} = \begin{Vmatrix} x_1 \\ x_2 \\ x_3 \end{Vmatrix} = \begin{Vmatrix} i_1 \\ i_2 \\ i_3 \end{Vmatrix}; \quad \vec{y} = \begin{Vmatrix} y_1 \\ y_2 \\ y_3 \end{Vmatrix} = \begin{Vmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Vmatrix}; \quad \vec{e} = \begin{Vmatrix} 0 \\ e \\ 0 \end{Vmatrix}; \\
& \vec{z} = \begin{Vmatrix} z_1 \\ z_2 \\ z_3 \end{Vmatrix} = \begin{Vmatrix} i_1 - i_2 - i_3 \\ r_1 i_1 + u_2 [i_2] \\ u_2 [i_2] - r_3 i_3 \end{Vmatrix} = \begin{Vmatrix} x_1 - x_2 - x_3 \\ r_1 x_1 + u_2 [x_2] \\ u_2 [x_2] - r_3 x_3 \end{Vmatrix}.
\end{aligned}
\tag{4.34}
$$

At the same time, the matrix B of the form (1.69) is as follows:

$$B = \begin{Vmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & -1 \end{Vmatrix}. \qquad (4.35)$$

The matrices of differential parameters necessary for building an instant model of the process have the form

$$d\vec{y}/d\vec{x} = diag(L_1, L_2, L_3 + L_Д^{\partial}[i_3]); \qquad (4.36)$$

$$\frac{d\vec{z}}{d\vec{x}} = \begin{Vmatrix} 1 & -1 & -1 \\ r_1 & r_2^{\partial} & 0 \\ 0 & r_2^{\partial} & -r_3 \end{Vmatrix}, \qquad (4.37)$$

here $L_Д^{\partial} = d\phi_Д / di_3 = L_Д^{\partial}[i_3]$ – the differential inductance of the choke of the third branch is a non-linear function of the current i_3; $r_2^{\partial} = du_2 / di_2 = r_2^{\partial}[i_2]$ - the differential active resistance of the second branch is a nonlinear function of the current i_2.

Values are accepted for calculation

$E_m = 15000$ v $\omega = 314.161/C$; $L_1 = L_2 = L_3 = 0.000$ Hn;

$r_1 = 0.01$ Om; $r_3 = 0.001$ Om

Let the nonlinear dependence $u_2 = u_2[i_2]$ be defined analytically and have the form

$$u_2 = r_2(i_2 + \alpha i_2^3) \qquad (4.38)$$

at

$$r_2 = 7.5 \text{ Om}; \qquad \alpha = 0.1 \cdot 10^{-4} 1/A^2,$$

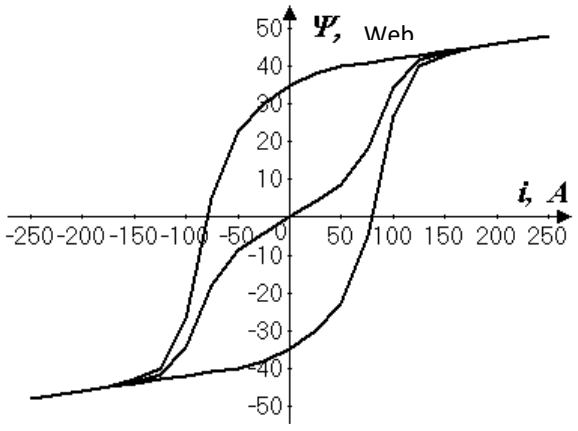


Fig. 4.7. Weber-Ampere characteristic of choke

and the Weber-ampere characteristic of the choke is given graphically (see Fig. 4.7), its three branches (upper, middle (main) and lower) are given for calculation by tables.

The block of user program components in this task consists of the main ***Program Hister*** program and the ***Model*** procedure of the instantaneous half-period model of the process.

To process the results and write them to the output file, the standard ***OUTP*** procedure from Block 5 of the DHM-S is used, so it is not included in the user block here

The main program of the user software component block is as follows.

```
      PROGRAM Hister
!---   a program for calculating of periodic process
!---   in a circuit with a hysteresis inductance
!---   (only odd harmonics are present in the currents)
!------------------------
            Implicit none
            real,dimension(32)::X
            real,dimension(30)::E
            integer,dimension(10)::KER
            real,dimension(3,3)::B
            real,dimension(3,20)::PS3
            real::AL1,AL2,AL3,R1,R2,ALF,R3,XL,XP,DI
            real::Em,OM,EPS1,EPS2,DH,H1,HM
            integer::i,NT3,K,NG,NK,j
            Common/MPM/B,AL1,AL2,AL3,R1,R2,ALF,R3,NT3,XL,XP,DI,PS3
!----   Common/MPM/ passes data to the Model procedure
           open(1,File='DaniV.dat',status='old')
            read(1,1)Em,OM,EPS1,EPS2,DH,H1,HM
            read(1,1)B
            read(1,1)AL1,AL2,AL3,R1,R2,ALF,R3
            read(1,2)NT3
            read(1,1)XL,XP,DI
            read(1,5) ((PS3(i,j),i=1,3),j=1,NT3)
!-----  input of choke magnetization loop data
            read(1,2)KER
            read(1,2)K
!------  KER -  array of control variables
!--------  K - the order of the system of differential equations
    1 format(4E10.4)
    2 format(10I3)
    5 format(3E10.4)
      close(1)
      open(1,FILE='RezV.DAT')
      write(1,10)
   10 format(/5X, 'Input data of the task :'/)
        write(1,4)Em,OM,EPS1,EPS2,DH,H1,HM
        write(1,4)B
         write(1,4)AL1,AL2,AL3,R1,R2,ALF,R3
         write(1,3)NT3
         write(1,4)XL,XP,DI
         write(1,6)((PS3(i,j),i=1,3),j=1,NT3)
        write(1,3)KER
        write(1,3)K
    3 format(2X,10I5)
    4 format(2X,4E12.4)
    6 format(2X,3E12.4)
       call SizesV(KER(1),K,KER(8),NG,NK)
        write(1,3)NG,NK
!---  a composite vector of amplitudes of forcing force
!---  and the initial value of the vector X  are formed
```

```
      E=0.;  E(NG+2)=Em
      X=0.;  X(NK+1)=OM
     write(1,13)
  13 format(/2X,'C a l c u l a t I o n :')
     call HARMOSC(K,X,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
       end Program Hister
```

The main program of this package performs:

– a description of the variables, including a description and declaration of the sizes of two arrays: ***X(32)*** is a matrix-column formed from the composite vector of current amplitudes of three branches of the circuit, circular frequency ω and parameter h, and matrix-column ***E(30)*** is a composite vector of amplitudes coercive force. The specified sizes of these arrays (32 and 30) allow you to set the maximum order of the harmonics taken into account no higher than 9 (harmonics of odd orders only are taken into account). If a higher value of the maximum harmonic order is specified, the sizes of these arrays must be increased;

– description of the shared memory area ***/MPM/***, with the help of which data is transferred to the ***Model*** procedure: matrix ***B*** of the form (3.26); inductances of three branches ***AL1, AL2, AL3***; active supports ***R1, R3*** of the first and third branches; ***R2*** and ***ALF*** coefficients of formula (4.38); the number of ***NT3*** nodes of the table, which is used to set the hysteresis loop; left ***XL*** and right ***XP*** abscissas of the nodes where the upper, middle and lower branches of the loop converge; step between nodes ***DI*** (nodes equidistant); the ***PS3*** array, which contains a table that specifies the three branches of the hysteresis loop - see section 3.2.1.2;

– entering input data from the **DaniV.dat** file;

– output to the ***RezV.dat*** source file of input data;

– appeal to the ***SizesV*** procedure;

– formation of the forcing force vector ***E***;

– assignment of the initial value of the array ***X***;

– appeal to the ***HARMOSC*** procedure.

The ***Model*** procedure, which implements an instantaneous (half-period) model of a periodic process in the scheme, looks like this:

```
    Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!-- The routine implements the instant model
!-- of a periodic process in the scheme (Fig. 4.6)
!-----------------------
         implicit none
         real,intent(in)::AL
         integer,intent(in)::M,K,MK
         real,dimension(MK)::XC,YC,ZC
         real,dimension(MK,K)::YXC,ZXC
         real,dimension(K)::X,Y,Z
         real,dimension(K,K)::YX,ZX,BM
         real,dimension(M)::XV,PSH,LH
         real,dimension(3,3)::B
```

```
      real,dimension(3,20)::PS3
      real::AL1,AL2,AL3,R1,R2,ALF,R3,XL,XP,DI,U2
      integer::NT3,i
      common/MPM/B,AL1,AL2,AL3,R1,R2,ALF,R3,NT3,XL,XP,DI,PS3
!-- through Common/MPM/ data from the main program is transferred
        BM=B
      do i=1,M
       call DRAWOUTV(K,XC,MK,X,i)
       Y(1)=AL1*X(1);  Y(2)=AL2*X(2);  Y(3)=AL3*X(3)
      Z(1)=X(1)-X(2)-X(3)
       U2=R2*(X(2)+ALF*X(2)**3)
       Z(2)=R1*X(1)+U2
       Z(3)=U2-R3*X(3)
       ZX(1,1)=1.;   ZX(1,2)=-1.;   ZX(1,3)=-1.
       U2=R2*(1.+3.*ALF*X(2)**2)
       ZX(2,1)=R1;   ZX(2,2)=U2;    ZX(2,3)=0.
       ZX(3,1)=0.;   ZX(3,2)=U2;    ZX(3,3)=-R3
       YX(1,1)=AL1;    YX(1,2)=0.;    YX(1,3)=0.
      YX(2,1)=0.;     YX(2,2)=AL2;   YX(2,3)=0.
       YX(3,1)=0.;     YX(3,2)=0.;    YX(3,3)=AL3
       call DRAWUPV(K,Y,YC,MK,i)
       call DRAWUPV(K,Z,ZC,MK,i)
       call DRAWUPM(K,YX,YXC,MK,i)
       call DRAWUPM(K,ZX,ZXC,MK,i)
      end do
      call DRAWOUTXV(K,XC,MK,M,XV,3)
      call HISTPER(AL,XV,M,PSH,LH,PS3,XL,XP,DI,NT3)
      call ADDV(K,PSH,M,YC,MK,3)
      call ADDM(K,LH,M,YXC,MK,3)
      return
      end    subroutine Model
```

The first operator of the procedure ***BM=B*** assigns to the formal parameter ***BM*** the value of the matrix ***B*** from the ***MPM*** common area of the memory. Further in the procedure there is a loop, in which the loop parameter is the variable ***i*** - the number of the node in the half period. In this cycle, the following is performed:

– the call ***DRAWOUTV(K,XC,MK,X,i)*** operator calls the ***DRAWOUTV*** procedure (from Block 5 of the DHM-S) for execution, which "extracts" from the composite nodal vector the values of the variables x_1 (current of the first branch), x_2 (current of the second branch) and x_3 (the current of the third branch) in the i -th node of the half-cycle and forms a vector from them;

– the values of the components of the vectors $\vec{y}$ and $\vec{z}$ (4.34) are calculated based on the values of the currents and the formulas (4.34) – the ***Y*** and ***Z*** arrays and differential parameter matrices $d\vec{y}/d\vec{x}$ (4.36), $d\vec{z}/d\vec{x}$ (4.37) – the ***YX*** and ***ZX*** arrays;

– the operators ***call DRAWUPV(K,Y,YC,MK,i)*** and ***call DRAWUPV(K,Z,ZC,MK,i)*** the values of the elements of the arrays ***Y*** and ***Z*** are "inserted" into the arrays ***YC*** and ***ZC*** (the values of the composite nodal shape vectors are stored in the last (1.118)) in places corresponding to the *i*-th node of the half-period;

– operators ***call DRAWUPM(K,YX,YXC,MK,i)*** and ***call DRAWUPM(K,ZX,ZXC,MK,i)*** are inserting the values of the elements of the ***YX*** and ***ZX*** arrays into the ***YXC*** and ***ZXC*** arrays (the latter store the values of the composite matrices of nodal differential parameters of the form (1.127)) in the places corresponding to the i-th node on the half-period.

In the cycle, the values of the elements of the arrays ***YC, ZC, YXC*** and ***ZXC*** are calculated, which correspond to all the linear connections of the diagram in fig. 4.6 and nonlinearities,which are included in the first group of nonlinearities (nonlinear active resistance in the second branch of the circuit).

In the final part of the procedure (after the cycle), the values of the elements of the arrays ***YC, ZC, YXC*** and ***ZXC*** are calculated taking into account the nonlinearity included in the third group - the hysteresis characteristic of the magnetization of the choke in the third branch of the circuit.

In this part of the procedure:

- the operator ***call DRAWOUTXV(K,XC,MK,M,XV,3)*** "draws" from the complex nodal vector ***XC*** of currents the simple nodal vector ***XV*** of the current of the third branch;

- the operator ***call HISTPER(AL,XV,M,PSH,LH,PS3,XL,XP,DI,NT3)*** calls the ***HISTPER*** procedure (see section 3.2.1.2), which calculates the values of flux coupling (***PSH*** matrix) and differential inductances (***LH*** matrix) of the choke based on the values of the coefficient ***AL*** of the expansion of the hysteresis loop and the value of the nodal current vector of the third branch ***XV*** in the nodes on the half-cycle;

– operators ***call DV(K,PSH,M,YC,MK,3)*** and **call ADDM(K,LH,M,YXC,MK,3)** are adding the value of the flux coupling of the inductive choke at the nodes of half-period to the corresponding components of the matrices ***YC*** (composite nodal vector of the variable $\vec{y}$) and ***YXC*** (composite matrix of nodal differential parameters of the form (1.127)) .

Printout of ***DaniV.dat*** file with input data:

```
+.1500E 05  +.3142E 03  +.1000E-01  +.3000E-02
+.1000E 00  +.1000E 01  +.1000E 01
+.0000E 00  +.1000E 01  +.0000E 00  +.0000E 00
+.1000E 01  +.1000E 01  +.0000E 00  +.0000E 00
-.1000E 01
+.1000E-03  +.1000E-03 +.1000E-03  +.1000E-01
+.7500E 01  +.1000E-04  +.1000E-02
017
-.1750E 03  +.1750E 03  +.2500E 02
-.4600E 02  -.4600E 02   -.4600E 02
-.4500E 02  -.4500E 02   -.4500E 02
-.4300E 02  -.4350E 02   -.4400E 02
-.4000E 02  -.4150E 02   -.4300E 02
-.2700E 02  -.3450E 02   -.4200E 02
+.5000E 01  -.1800E 02   -.4100E 02
+.2300E 02  -.8500E 01   -.4000E 02
+.3000E 02  -.4000E 01   -.3800E 02
+.3500E 02  +.0000E 00   -.3500E 02
+.3800E 02  +.4000E 01   -.3000E 02
+.4000E 02  +.8500E 01   -.2300E+02
+.4100E+02  +.1800E+02   -.5000E+01
+.4200E+02  +.3450E+02   +.2700E+02
+.4300E+02  +.4150E+02   +.4000E+02
```

```
+.4400E+02  +.4350E+02   +.4300E+02
+.4500E+02  +.4500E+02   +.4500E+02
+.4600E 02  +.4600E 02   +.4600E+02
00100000100000000010009000000
003
```

The ***RezV.dat*** file with the output data (result) after running the program is as follows:

```
      Input data of the task :
     .1500E+05   .3142E+03   .1000E-01   .3000E-02
     .1000E+00   .1000E+01   .1000E+01
     .0000E+00   .1000E+01   .0000E+00   .0000E+00
     .1000E+01   .1000E+01   .0000E+00   .0000E+00
    -.1000E+01
     .1000E-03   .1000E-03   .1000E-03   .1000E-01
     .7500E+01   .1000E-04   .1000E-02
     17
    -.1750E+03   .1750E+03   .2500E+02
    -.4600E+02  -.4600E+02  -.4600E+02
    -.4500E+02  -.4500E+02  -.4500E+02
    -.4300E+02  -.4350E+02  -.4400E+02
    -.4000E+02  -.4150E+02  -.4300E+02
    -.2700E+02  -.3450E+02  -.4200E+02
     .5000E+01  -.1800E+02  -.4100E+02
     .2300E+02  -.8500E+01  -.4000E+02
     .3000E+02  -.4000E+01  -.3800E+02
     .3500E+02   .0000E+00  -.3500E+02
     .3800E+02   .4000E+01  -.3000E+02
     .4000E+02   .8500E+01  -.2300E+02
     .4100E+02   .1800E+02  -.5000E+01
     .4200E+02   .3450E+02   .2700E+02
     .4300E+02   .4150E+02   .4000E+02
     .4400E+02   .4350E+02   .4300E+02
     .4500E+02   .4500E+02   .4500E+02
     .4600E+02   .4600E+02   .4600E+02
      1    0    1    0    0    0   10    9    0    0
      3
     10   30
  C a l c u l a t I o n :
   Refined value 1 root at h= 1.000
  (the solution was obtained after the 3-rd iteration)

  Amplitudes of harmonics of the 1-st variable:
  X1(c1)= -.1862E+03  X1(s1)=  .5968E+03  X1(1)=  .6252E+03
  X1(c3)= -.2321E+02  X1(s3)=  .9510E+02  X1(3)=  .9789E+02
  X1(c5)= -.3233E+02  X1(s5)=  .3852E+02  X1(5)=  .5029E+02
  X1(c7)= -.5870E+01  X1(s7)=  .1935E+02  X1(7)=  .2022E+02
  X1(c9)= -.5311E+01  X1(s9)=  .9524E+01  X1(9)=  .1090E+02

 The value of the variable in nodes of period, M =60
-.2529E+03-.1830E+03 -.1110E+03-.4050E+02  .2562E+02  .8501E+02
 .1363E+03 .1791E+03  .2139E+03 .2422E+03  .2654E+03  .2854E+03
 .3037E+03 .3211E+03  .3382E+03 .3548E+03  .3706E+03  .3850E+03
```

```
 .3976E+03 .4081E+03  .4166E+03 .4238E+03  .4305E+03  .4376E+03
 .4461E+03 .4564E+03  .4689E+03 .4832E+03  .4989E+03  .5149E+03
 .5304E+03 .5445E+03  .5565E+03 .5662E+03  .5734E+03  .5786E+03
 .5820E+03 .5841E+03  .5851E+03 .5851E+03  .5839E+03  .5814E+03
 .5772E+03 .5713E+03  .5639E+03 .5554E+03  .5465E+03  .5380E+03
 .5309E+03 .5254E+03  .5217E+03 .5191E+03  .5160E+03  .5107E+03
 .5007E+03 .4837E+03  .4577E+03 .4214E+03  .3744E+03  .3176E+03

  Amplitudes of harmonics of the 2-st variable:
  X2(c1)= -.2024E+01  X2(s1)=  .5971E+03  X2(1)=  .5971E+03
  X2(c3)= -.1578E+01  X2(s3)=  .9529E+02  X2(3)=  .9530E+02
  X2(c5)= -.1241E+01  X2(s5)=  .3880E+02  X2(5)=  .3882E+02
  X2(c7)= -.9343E+00  X2(s7)=  .1948E+02  X2(7)=  .1950E+02
  X2(c9)= -.5976E+00  X2(s9)=  .9601E+01  X2(9)=  .9620E+01

 The value of the variable in nodes of period, M =60
-.6376E+01 .6135E+02  .1264E+03  .1861E+03  .2385E+03  .2823E+03
 .3174E+03 .3445E+03  .3652E+03  .3810E+03  .3940E+03  .4057E+03
 .4173E+03 .4294E+03  .4419E+03  .4546E+03  .4668E+03  .4779E+03
 .4872E+03 .4947E+03  .5004E+03  .5047E+03  .5080E+03  .5110E+03
 .5141E+03 .5175E+03  .5211E+03  .5247E+03  .5277E+03  .5299E+03
 .5307E+03 .5301E+03  .5282E+03  .5251E+03  .5215E+03  .5178E+03
 .5142E+03 .5109E+03  .5077E+03  .5044E+03  .5003E+03  .4948E+03
 .4877E+03 .4786E+03  .4678E+03  .4557E+03  .4428E+03  .4300E+03
 .4175E+03 .4056E+03  .3938E+03  .3810E+03  .3658E+03  .3464E+03
 .3209E+03 .2877E+03  .2460E+03  .1957E+03  .1377E+03  .7372E+02

  Amplitudes of harmonics of the 3-st variable:
  X3(c1)= -.1842E+03  X3(s1)= -.2318E+00  X3(1)=  .1842E+03
  X3(c3)= -.2164E+02  X3(s3)= -.1888E+00  X3(3)=  .2164E+02
  X3(c5)= -.3109E+02  X3(s5)= -.2769E+00  X3(5)=  .3109E+02
  X3(c7)= -.4935E+01  X3(s7)= -.1220E+00  X3(7)=  .4937E+01
  X3(c9)= -.4713E+01  X3(s9)= -.7771E-01  X3(9)=  .4714E+01

 The value of the variable in nodes of period, M =60
-.2465E+03-.2443E+03 -.2375E+03 -.2266E+03 -.2128E+03 -.1973E+03
-.1811E+03-.1655E+03 -.1512E+03 -.1389E+03 -.1286E+03 -.1203E+03
-.1136E+03-.1083E+03 -.1038E+03 -.9983E+02 -.9621E+02 -.9283E+02
-.8966E+02-.8668E+02 -.8381E+02 -.8086E+02 -.7753E+02 -.7341E+02
-.6807E+02-.6110E+02 -.5225E+02 -.4144E+02 -.2887E+02 -.1494E+02
-.2760E+00 .1440E+02  .2838E+02  .4102E+02  .5190E+02  .6083E+02
 .6785E+02 .7324E+02  .7739E+02  .8073E+02  .8368E+02  .8655E+02
 .8952E+02 .9269E+02  .9607E+02  .9969E+02  .1036E+03  .1081E+03
 .1133E+03 .1198E+03  .1280E+03  .1381E+03  .1502E+03  .1643E+03
 .1799E+03 .1960E+03  .2116E+03  .2256E+03  .2367E+03  .2439E+03
  Take hysteresis into account
  Amplitudes of harmonics of the 1-st variable:
  X1(c1)= -.1561E+03  X1(s1)=  .6739E+03  X1(1)=  .6917E+03
  X1(c3)= -.6535E+02  X1(s3)=  .8633E+02  X1(3)=  .1083E+03
  X1(c5)= -.2328E+02  X1(s5)=  .2769E+02  X1(5)=  .3618E+02
  X1(c7)= -.6093E+01  X1(s7)=  .1434E+02  X1(7)=  .1558E+02
```

```
 X1(c9)= -.1202E+01  X1(s9)=  .8852E+01  X1(9)=  .8933E+01

The value of the variable in nodes of period, M =60
-.2520E+03-.1846E+03-.1151E+03 -.4610E+02  .2006E+02  .8173E+02
 .1380E+03 .1885E+03 .2337E+03  .2744E+03  .3114E+03  .3455E+03
 .3775E+03 .4075E+03 .4355E+03  .4612E+03  .4843E+03  .5043E+03
 .5211E+03 .5348E+03 .5459E+03  .5548E+03  .5624E+03  .5693E+03
 .5761E+03 .5828E+03 .5896E+03  .5961E+03  .6020E+03  .6066E+03
 .6098E+03 .6113E+03 .6113E+03  .6101E+03  .6082E+03  .6061E+03
 .6043E+03 .6027E+03 .6015E+03  .6002E+03  .5982E+03  .5951E+03
 .5905E+03 .5840E+03 .5760E+03  .5670E+03  .5576E+03  .5488E+03
 .5413E+03 .5353E+03 .5308E+03  .5269E+03  .5222E+03  .5147E+03
 .5023E+03 .4829E+03 .4550E+03  .4175E+03  .3705E+03  .3148E+03

 Amplitudes of harmonics of the 2-st variable:
 X2(c1)= -.2077E+01  X2(s1)=  .5971E+03  X2(1)=  .5971E+03
 X2(c3)= -.1552E+01  X2(s3)=  .9521E+02  X2(3)=  .9523E+02
 X2(c5)= -.1193E+01  X2(s5)=  .3881E+02  X2(5)=  .3883E+02
 X2(c7)= -.8949E+00  X2(s7)=  .1948E+02  X2(7)=  .1950E+02
 X2(c9)= -.5816E+00  X2(s9)=  .9626E+01  X2(9)=  .9644E+01

The value of the variable in nodes of period, M =60
-.6298E+01 .6143E+02 .1265E+03  .1862E+03  .2384E+03  .2822E+03
 .3173E+03 .3444E+03 .3650E+03  .3808E+03  .3938E+03  .4055E+03
 .4171E+03 .4292E+03 .4418E+03  .4545E+03  .4668E+03  .4778E+03
 .4872E+03 .4947E+03 .5004E+03  .5046E+03  .5080E+03  .5110E+03
 .5141E+03 .5175E+03 .5212E+03  .5247E+03  .5278E+03  .5300E+03
 .5308E+03 .5302E+03 .5282E+03  .5252E+03  .5216E+03  .5178E+03
 .5142E+03 .5109E+03 .5078E+03  .5044E+03  .5003E+03  .4949E+03
 .4877E+03 .4786E+03 .4678E+03  .4557E+03  .4428E+03  .4300E+03
 .4175E+03 .4056E+03 .3938E+03  .3810E+03  .3658E+03  .3464E+03
 .3209E+03 .2877E+03 .2460E+03  .1957E+03  .1376E+03  .7366E+02

 Amplitudes of harmonics of the 3-st variable:
 X3(c1)= -.1540E+03  X3(s1)=  .7682E+02  X3(1)=  .1721E+03
 X3(c3)= -.6380E+02  X3(s3)= -.8881E+01  X3(3)=  .6441E+02
 X3(c5)= -.2209E+02  X3(s5)= -.1112E+02  X3(5)=  .2473E+02
 X3(c7)= -.5198E+01  X3(s7)= -.5145E+01  X3(7)=  .7313E+01
 X3(c9)= -.6208E+00  X3(s9)= -.7740E+00  X3(9)=  .9922E+00

 The value of the variable in nodes of period, M =60
-.2457E+03-.2460E+03-.2415E+03 -.2323E+03 -.2184E+03 -.2005E+03
-.1793E+03-.1559E+03-.1312E+03 -.1064E+03 -.8244E+02 -.5999E+02
-.3964E+02-.2171E+02-.6293E+01  .6706E+01  .1752E+02  .2646E+02
 .3388E+02 .4012E+02 .4547E+02  .5018E+02  .5442E+02  .5832E+02
 .6193E+02 .6531E+02 .6847E+02  .7140E+02  .7413E+02  .7664E+02
 .7895E+02 .8107E+02 .8303E+02  .8487E+02  .8662E+02  .8833E+02
 .9005E+02 .9184E+02 .9373E+02  .9576E+02  .9794E+02  .1003E+03
 .1027E+03 .1054E+03 .1082E+03  .1113E+03  .1148E+03  .1188E+03
 .1238E+03 .1297E+03 .1371E+03  .1459E+03  .1563E+03  .1683E+03
 .1814E+03 .1952E+03 .2089E+03  .2218E+03  .2329E+03  .2411E+03
```

Based on the values of the third variable (current i_3) in the m nodes of the half-period, the dependence of this current on time (angle η) is constructed at one period.

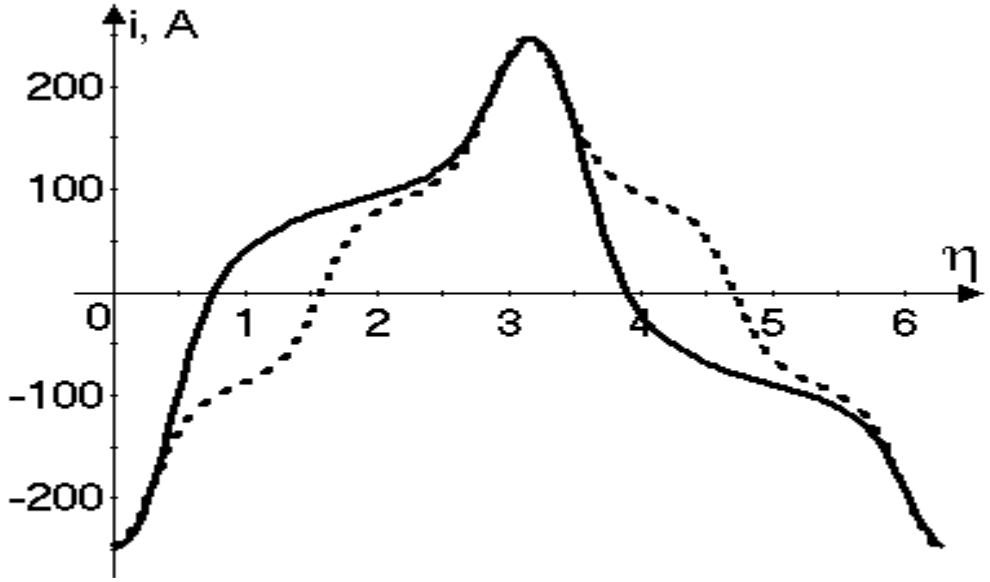


Fig. 4.8. Current of 3-th branch

This dependence is shown in fig. 4.8. The dashed line shows the current i_3 before taking into account the hysteresis (calculation on the main branch of the loop), and the solid line shows the same current taking into account the hysteresis (calculation on the full loop).

Analyzing these curves, as well as the value of the vector of current amplitudes before and after taking into account hysteresis, it is necessary to note the appearance of an active component of the first harmonic of the current, which is a reflection of power losses due to remagnetization of the choke. Indeed, without taking into account the hysteresis, the amplitude of the first harmonic of the current of the third branch with a choke is as follows (see the printout of the RezV.DAT file) $I_{3(c1)} = -184.2\,A; \quad I_{3(s1)} = -0.23\,A$. Given that the electromotive force in the first branch is a sine wave (see formula (4.30)), the current of the third branch is almost a pure cosine wave, that is, it is inductive in nature (lags behind the electromotive force by 90 electric degrees), because the resistance of the third branch in is mainly inductive. After taking into account the hysteresis, the following values of the amplitude of the first harmonic of the current were obtained $I_{3(c1)} = -154.0\,A; \quad I_{3(s1)} = 76.8\,A$..

We see the appearance of a significant sinusoidal component in the first harmonic of the current, which is in phase with the electromotive force, that is, it is active and causes losses due to remagnetization.

The reader is invited to conduct a numerical experiment on the described model: to specify several variants of the hysteresis loop with different areas and to make sure that the calculated power losses due to remagnetization of the choke core will be proportional to the area of the hysteresis loop [4, 41].

4.2. Tests and examples of self-oscillation calculations

In this section, several examples of numerical simulation of self-oscillations in nonlinear systems with various types of nonlinear connections are given. These are examples A.1 - A.4. In each example of this group, the value 2 is assigned to the fourth element ***KER(4)*** of the control array ***KER*** before calling the ***HARMOSC*** routine in the main program of the user program component block.

4.2.1. Example A.1

As the first problem, as an example of numerical modeling of self-oscillations, we will consider the determination of the periodic solution of a nonlinear differential equation

$$\ddot{x} + \mu (x^2 - 1)\dot{x} + x = 0, \qquad (4.39)$$

which describes, in particular, the motion of a pendulum with nonlinear damping. This equation is known in the literature as Van der Pol's equation. Its nonlinearity is unconditionally unambiguous (nonlinearity of the first group) and is given analytically.

It is shown in [27] that the determination of the periodic solution of equation (4.39) can be carried out by the asymptotic Bogolyubov-Mitropolsky method, but only for values of the coefficient $\mu << 1$, when the oscillations described by this equation are almost harmonic. When the values of this coefficient are close to 1, and especially when $\mu >> 1$ the oscillations are polyharmonic and have a relaxation character, it is recommended in [27] to use other methods, in particular, the method of A. Dorodnitsin (asymptotic approximation by powers of $1/\mu$).

This example shows that during the numerical modeling by the differential harmonic method of nonlinear self-oscillations described by equation (4.39), no restrictions are imposed on the value of the coefficient μ .

We reduce equation (4.39) to form (1.65):

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0 \qquad (4.40)$$

with designations

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\|; \qquad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ x_1 + \mu(x_1^2 - 1)x_2 \end{matrix} \right\|; \quad \vec{e} = 0. \qquad (4.41а,б)$$

The value of the derivative $d\vec{z}/d\vec{x}$, which is necessary in the formation of the instantaneous (half-period) mathematical model of the system, is as follows

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 0 & -1 \\ 1+2\mu x_1 x_2 & \mu(x_1^2-1) \end{matrix} \right\|. \qquad (4.42)$$

The text of the main program of the block of user software components for numerical simulation of self-oscillations described by equation (4.39) has the form:

```
    Program Van_der_Pol
!--  Program for determination of periodic solutions
!--  van der Pol's equation
!--      dX/dt+Z=E
!--      X=colon(x1,x2)
!--      Z=colon(z1,z2)
!--      E=colon(e1,e2)
!--      z1=-x2;   z2=x1-mju*(1-x1**2)*x2
!--      e1=0;  e2=0
!--  (variables contain only odd harmonics)
!----------------------------------------
          Implicit none
          real,dimension(42)::Y0
          real,dimension(40)::E
          integer,dimension(10)::KER
          real::mju,OM,EPS1,EPS2,H1,HM
          real::X1C1,X1S1,X2C1,X2S1
          integer::K,NG,NK
          common/MP/mju   !--- area of memory shared with the Model procedure
           open(1,File='DaniVan.dat',status='old')
          read(1,1)mju
          read(1,1)EPS1,EPS2,H1,HM
!-----  EPS1 – accuracy of integration
!-----  EPS2 – accuracy for Newton's method
!-----  H1 – the value of  h  at which it is nessesary todetermine the root more precisely
!-----  HM – the maximum value of h
          read(1,2)KER
          read(1,2)K
          read(1,1)OM
          read(1,1)X1C1,X1S1,X2C1,X2S1
!-----  KER – an array of controlvariables
!-----  K – the order of the system of differentialequations
!-----  X1C1,X1S1,X2C1,X2S1 – initial approximations of the amplitudes
!-----      of first harmonics of variablts  X1  and   X2
   1 Format(4E10.4)
   2 Format(10I3)
     close(1)
      open(1,file='RezVan.dat')
```

```
      write(1,5)
    5 format(2X,'Periodic solution of the van der Pol's equation' /10X,'Entered data:')
      write(1,14)mju
   14 format(2X,' mju=',E10.4)
      write(1,15)EPS1,EPS2,H1,HM
   15 format(2X,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
      write(1,16)KER
   16 format(2X,'KER=',10i5)
      write(1,17)K
   17 format(2X,' K=',i2)
      write(1,18)OM
   18 format(2X,'OM=',E11.4)
      write(1,19)X1C1,X1S1,X2C1,X2S1
   19 format(2X,'X1C1=',E11.4,' X1S1=',E11.4,' X2C1=',E11.4,' X2S1=',E11.4)
    3 Format(2X,8I5)
      call SizesV(KER(1),K,KER(8),NG,NK)
        write(1,3)NG,NK
!-----  NG – the number of elements of a simple vector of amplitudes
!-----  NK – the number of elements of a composite vector of amplitudes
        Y0=0;  Y0(NK+1)=OM
        Y0(1)=X1C1; Y0(2)=X1S1; Y0(NG+1)=X2C1; Y0(NG+2)=X2S1
!----  formed the initial of the Y0  vector
      write(1,13)
   13 format(/2X,'C a l c u l a t I o n :')
      call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
      end Program Van_der_Pol
```

The text of the ***Model*** program, which implements the instantaneous (half-cycle) process model:

```
    Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--  The procedure of the instaneous model (on the half-period of the process,
!--   which is described by equation (4.39)
        Implicit none
        real::AL
        integer,intent(in)::M,K,MK
        real,dimension(MK)::XC,YC,ZC
        real,dimension(MK,K)::YXC,ZXC
        real,dimension(K)::X,Z
        real,dimension(K,K)::ZX,BM
        real::mju
        integer::i
```

```
      Common/MP/mju   !--  area of memory shared with the main program
      BM(1,1)=0.;  AL=0.
      YC(1)=0.;  YXC(1,1)=0.
      do i=1,M
        call DRAWOUTV(K,XC,MK,X,i)
        Z(1)=-X(2)
        Z(2)=X(1)-mju*(1.-X(1)**2)*X(2)
        ZX(1,1)=0.;    ZX(1,2)=-1.
        ZX(2,1)=1.+2.*mju*X(1)*X(2)
        ZX(2,2)=mju*(X(1)**2-1.)
        call DRAWUPV(K,Z,ZC,MK,i)
        call DRAWUPM(K,ZX,ZXC,MK,i)
      end do
      return
      end     Subroutine Model
```

In this procedure, the formal parameters ***AL*** and ***BM*** are not used, because the nonlinearity of equation (4.39) does not belong to the third group - it is not hysteretic - and equation (4.39) in the notation (4.40) is in the normal Cauchy form, that is, it is reduced to the form (1.65). For the same reason, the first four operators

```
BM(1,1)=0.;   AL=0.
YC(1)=0.;   YXC(1,1)=0.
```

perform unnecessary operations, their purpose is to block the compiler's message when compiling the procedure that the specified formal parameters in the body of the procedure are not used.

Since there are no nonlinearities of the second or third groups in the problem, all the actions of the procedure (calculation of the values of the composite nodal vector ***ZC*** and the composite matrix of nodal parameters ***ZXC***) are performed in the main cycle of the procedure.

Printout of the ***DaniVan.dat*** file with input data:

```
.3000E+01
.1000E-02   .1000E-03   .1000E+01   .1000E+01
1 1 0 2 0 0 0 9 0 0
  2
.8000E+00
.2000E+01   .0000E_00   .0000E+00    -.2000E+01
```

Printout of the RezVan.dat file with the calculation results:

```
 Periodic solution of the van der Pol's equation

          Entered data:
   mju= .3000E+01
   EPS1=.1000E-02  EPS2=.1000E-03  H1=.1000E+01  HM=.1000E+01
   KER=   1    1    0    2    0    0    0    9   0    0
   K= 2
  OM = .8000E+00
  X1C1=.2000E+01  X1S1=.0000E+00  X2C1=.0000E+00 X2S1=-.2000E+01
     10   20

  C a l c u l a t I o n :

  Refined value 1 root at h= 1.000
  (the solution was obtained after the 2-rd iteration)

  Amplitudes of harmonics of the 1-st variable:
  X1(c1)=  .2074E+01   X1(s1)= -.7973E-10   X1(1)=  .2074E+01
  X1(c3)= -.3668E+00   X1(s3)= -.3408E+00   X1(3)=  .5007E+00
  X1(c5)=  .1400E-01   X1(s5)=  .2272E+00   X1(5)=  .2276E+00
  X1(c7)=  .8226E-01   X1(s7)= -.9589E-01   X1(7)=  .1263E+00
  X1(c9)= -.8103E-01   X1(s9)=  .1747E-01   X1(9)=  .8289E-01

  The value of the variable in nodes of period, M =60
  .1723E+01 .1706E+01 .1698E+01  .1694E+01  .1691E+01  .1682E+01
  .1665E+01 .1638E+01 .1602E+01  .1560E+01  .1516E+01  .1476E+01
  .1443E+01 .1420E+01 .1405E+01  .1396E+01  .1386E+01  .1372E+01
  .1346E+01 .1307E+01 .1253E+01  .1189E+01  .1119E+01  .1050E+01
  .9875E+00 .9356E+00 .8940E+00  .8581E+00  .8189E+00  .7644E+00
  .6813E+00 .5578E+00 .3858E+00  .1629E+00 -.1062E+00 -.4104E+00
 -.7328E+00-.1053E+01-.1351E+01 -.1609E+01 -.1813E+01 -.1957E+01
 -.2042E+01-.2075E+01-.2069E+01 -.2039E+01 -.1998E+01 -.1959E+01
 -.1930E+01-.1914E+01-.1910E+01 -.1913E+01 -.1917E+01 -.1917E+01
 -.1907E+01-.1886E+01-.1855E+01 -.1819E+01 -.1781E+01 -.1748E+01

  Amplitudes of harmonics of the 2-st variable:
  X2(c1)= -.5166E-06  X2(s1)= -.1472E+01  X2(1)=  .1472E+01
  X2(c3)= -.7254E+00  X2(s3)=  .7808E+00  X2(3)=  .1066E+01
  X2(c5)=  .8060E+00  X2(s5)= -.4967E-01  X2(5)=  .8075E+00
  X2(c7)= -.4763E+00  X2(s7)= -.4086E+00  X2(7)=  .6275E+00
  X2(c9)=  .1115E+00  X2(s9)=  .5175E+00  X2(9)=  .5293E+00

The value of the variable in nodes of period, M =60
 -.2841E+00 -.1624E+00 -.7238E-01 -.3898E-01 -.7235E-01 -.1654E+00
 -.2959E+00 -.4318E+00 -.5397E+00 -.5929E+00 -.5787E+00 -.5015E+00
 -.3826E+00 -.2546E+00 -.1545E+00 -.1137E+00 -.1503E+00 -.2643E+00
 -.4361E+00 -.6309E+00 -.8068E+00 -.9244E+00 -.9579E+00 -.9026E+00
 -.7784E+00 -.6279E+00 -.5092E+00 -.4843E+00 -.6055E+00 -.9031E+00
```

```
-.1376E+01 -.1990E+01 -.2678E+01 -.3352E+01 -.3917E+01 -.4289E+01
-.4404E+01 -.4236E+01 -.3801E+01 -.3149E+01 -.2364E+01 -.1542E+01
-.7787E+00 -.1535E+00  .2833E+00  .5162E+00  .5641E+00  .4733E+00
 .3063E+00  .1279E+00 -.7587E-02 -.6567E-01 -.3667E-01  .6542E-01
 .2090E+00  .3543E+00  .4638E+00  .5113E+00  .4881E+00  .4039E+00

 Circular frequency of the fundamental harmonic =  .7095E+00
```

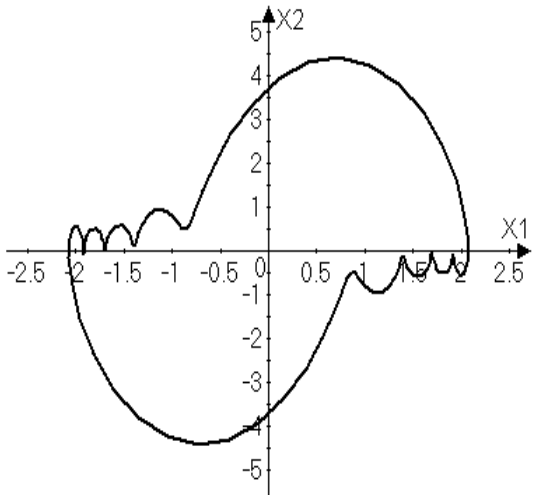


Fig. 4.9 Limit cycle

These results refer to the version of calculations at $\mu = 3.0$ and $n = 9$..

According to the obtained values of the variables x_1 and x_2 at the nodes on the half-period in fig. 4.9, the limit cycle for simulated self-oscillations (dependence of the variable x_2 (speed) on the variable x_1 (deviation)) is constructed.

For the same value of μ, calculations were performed for other values of n, their results are shown in table 4.2. As can be seen from this table, the amplitude of the 15th harmonic of the variable does not exceed 1% of the value of the amplitude of the 1st harmonic, and it makes almost no sense to further increase the number of harmonics taken into account.

Table 4.2. Results of self-oscillation calculation at μ = 3.0

n	$X_{1(1)}$	$X_{1(3)}$	$X_{1(5)}$	$X_{1(7)}$	$X_{1(9)}$	$X_{1(11)}$	$X_{1(13)}$	$X_{1(15)}$	ω
1	2.0								1.0
3	2.102	0.5329							0.8212
5	2.087	0.5391	0.2741						0.7304
7	2.074	0.5068	0.2433	0.1499					0.7105
9	2.074	0.5007	0.2276	0.1263	0.0829				0.7095
11	2.075	0.5001	0.2249	0.1197	0.0704	0.0433			0.7094
13	2.075	0.5000	0.2244	0.1183	0.0673	0.0407	0.0276		0.7093
15	2.075	0.5000	0.2243	0.1179	0.0665	0.0391	0.0241	0.0164	0.7093

4.2.2. Example A.2

In this example, the definition of self-oscillation parameters in the third-order automatic control system described by the equation is considered

$$T^2 \dddot{x} + 2\xi T \ddot{x} + \dot{x} + b\, sign\, x = 0. \qquad (4.43)$$

To determine the periodic solution of this equation, the method of harmonic linearization was used in [56], it is presented there as example 4.2.2. At the same time, fluctuations of the variable are sought in the form

$$x = A \sin \omega t. \qquad (4.44)$$

By the method of harmonic linearization in [56] obtained

$$\omega = \frac{1}{T}; \qquad A = \frac{2bT}{\pi\xi}. \qquad (4.45)$$

If specified by numerical values

$$T = 1.0; \quad b = 10.0; \quad \xi = 0.07,$$

then by formulas (4.45) we obtain: $\omega = 1.0$ and $A = 90.95$.

Let's solve the same problem by means of numerical modeling using DGM software. For this equation (4.43) we reduce to the form (1.65):

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0, \qquad (4.46)$$

here

$$\vec{x} = \begin{Vmatrix} x_1 \\ x_2 \\ x \end{Vmatrix}; \quad \vec{z} = \begin{Vmatrix} z_1 \\ z_2 \\ z_3 \end{Vmatrix} = \begin{Vmatrix} -x_2 \\ -x_3 \\ c_1 x_3 + c_2 x_2 + c_2\, sign\, x_1 \end{Vmatrix}; \qquad (4.47)$$

$$c_1 = \frac{2\xi}{T}; \quad c_2 = \frac{1}{T^2} \qquad (4.48)$$

The software components of the user block in this example are as follows.

1. The main program:

```
      Program SAR
!-- Program for determining periodic solutions
!-- the equation describing the self-oscillating mode
!-- automatic regulation systems
!--      dX/dt+Z=0
!--     X=colon(x1,x2,x3)
!--    Z=colon(z1,z2,z3)
!--     z1=-x2; z2=-x3; z3=c1*x3+c2*x2+c3*sign(x1)
!--  (variables contain only odd harmonics)
!----------------------------------------
           Implicit none
           real,dimension(42)::Y0
           real,dimension(40)::E
           integer,dimension(10)::KER
           real::A,D,T,C1,C2,B,OM,EPS1,EPS2,H1,HM
           real::X1C1,X1S1,X2C1,X2S1,X3C1,X3S1
           integer::K,NG,NK
           common/MP/C1,C2,B   !--- area of memory shared with the Model procedure
           open(1,File='DaniVan.dat',status='old')
           read(1,*)D,T,B
           read(1,*)EPS1,EPS2,H1,HM
!-----  H1 – the value of  h  at which it is nessesary todetermine the root more precisely
!-----  HM – the maximum value of h
!-----  EPS1 – accuracy of integration
!-----  EPS2 – accuracy for Newton's method
           read(1,*)KER
           read(1,*)K
!-----  KER – an array of controlvariables
!-----  K – the order of the system of differentialequations
           close(1)
          open(1,file='RezVan.dat')
           write(1,5)
   5 format(2X,'Periodic solution of the SAR equation' /10X,'Entered data:')
       write(1,14)D,T,B
  14 format(2X,'Dzeta=',E10.4,' T=',E10.4,' B=',E10.4)
           write(1,15)EPS1,EPS2,H1,HM
  15 format(2X,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
           write(1,16)KER
  16 format(2X,' KER=',10i5)
           write(1,17)K
  17 format(2X,' K=',i2)
       OM=1./T;  A=2.*B*T/(3.14159*D)
           write(1,18)OM,A
  18 format(2X,'OM=',E11.4,'  A=',E11.4)
           X1C1=A; X1S1=0.; X2C1=0.; X2S1=-OM*A
       X3C1=-OM**2*A; X3S1=0.
!--- X1C1,X1S1,X2C1,X2S1,X3C1,X3S1 – initial values
!--------- of the amplitudes of the first harmonics of variables
           write(1,19)X1C1,X1S1,X2C1,X2S1,X3C1,X3S1
  19 format(2X,'X1C1=',E11.4,' X1S1=',E11.4,' X2C1=',E11.4,
```

```
     &  ' X2S1=',E11.4,' X3C1=',E11.4,' X3S1=',E11.4)
   3 Format(2X,10I5)
          C1=2.*D/T;  C2=1./T**2
          call SizesV(KER(1),K,KER(8),NG,NK)
          write(1,3)NG,NK
!-----  NG – the number of elements of a simple vector of amplitudes
!-----  NK – the number of elements of a composite vector of amplitudes
          Y0=0;  Y0(NK+1)=OM
          Y0(1)=X1C1; Y0(2)=X1S1; Y0(NG+1)=X2C1; Y0(NG+2)=X2S1
          Y0(2*NG+1)=X3C1; Y0(2*NG+2)=X3S1
!----  formed the initial of the Y0  vector
       write(1,13)
  13 format(/2X,'C a l c u l a t o I n :')
      call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
      end Program SAR
```

In this program, the real variables ***D, T, B*** correspond to the variables ξ , T and b in equations (4.43) and the variables ***OM, A*** correspond to the variables ω and A from the formulas (4.45). Based on the values of the last two variables, the initial values of the amplitudes of the first harmonics of the variables x_1, x_2, x_3 are calculated.

2. The procedure that implements the instantaneous (on half-period) model of the system:

```
     Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--  The procedure of the instantaneous model of the ssystem on half-period
!-------------------
          Implicit none
          real::AL
          integer,intent(in)::M,K,MK
          real,dimension(MK)::XC,YC,ZC
          real,dimension(MK,K)::YXC,ZXC
          real,dimension(K)::X,Z
          real,dimension(K,K)::ZX,BM
          real::C1,C2,B
          integer::i
          common/MP/C1,C2,B    !--  area of memory shared with the main program
          BM(1,1)=0.;  AL=0.
          YC(1)=0.;  YXC(1,1)=0.
          do i=1,M
           call DRAWOUTV(K,XC,MK,X,i)
           Z(1)=-X(2)
           Z(2)=-X(3)
           Z(3)=C1*X(3)+C2*X(2)+C2*sign(B,X(1))
           ZX(1,1)=0.;  ZX(1,2)=-1.;  ZX(1,3)=0.
           ZX(2,1)=0.;  ZX(2,2)=0.;   ZX(2,3)=-1.
           ZX(3,1)=0.;  ZX(3,2)=C2;   ZX(3,3)=C1
           call DRAWUPV(K,Z,ZC,MK,i)
           call DRAWUPM(K,ZX,ZXC,MK,i)
```

```
      end do
     return
     end  Subroutine Model
```

As in the previous example A1, the formal parameters ***AL*** and ***BM*** are not used in the ***Model*** procedure for this example, because the nonlinearity of equation (4.43) is not hysteretic (does not belong to the third group) and equation (4.43) in the notation (4.46) is in to the normal Cauchy form, that is, it does not reduce to the form (1.67). For the same reason, the first four operators perform unnecessary operations, their purpose is to block the compiler's message when compiling the procedure that the specified formal parameters in the body of the procedure are not used.

Since there are no nonlinearities of the second or third groups in the problem, all the actions of the procedure (calculation of the values of the composite nodal vector ***ZC*** and the composite matrix of nodal parameters ***ZXC***) are performed in the main cycle of the procedure.

After carrying out the calculation using this program, taking into account only the first harmonic for a small value of ξ = 0.07, the result *A* = 91.65 and ω = 0.9954 was obtained, which is quite close to the one obtained in [56] by the method of harmonic linearization. The deviation in amplitude does not exceed 0.5% and in frequency - 0.3%.

The calculation of the same option, but with $n = 5$, gave the following result:

```
  Periodic solution of the SAR equation
          Entered data:
  Dzeta= .7000E-01 T= .1000E+01 B= .1000E+02
   EPS1= .1000E-02 EPS2= .1000E-03 H1= .1000E+01 HM= .1000E+01
   KER =  1    1    0    2    0    0    0    5    0    0
   K= 3
  OM=  .1000E+01  A=  .9095E+02
X1C1= .9095E+02 X1S1= .0000E+00 X2C1= .0000E+00 X2S1= -.9095E+02
X3C1= -.9095E+02 X3S1=  .0000E+00
      6   18
  C a l c u l a t i o n :
    Number of the highest harmonic = 5
 Refined value 1 root at h= 1.000
  (the solution was obtained after the 2-rd iteration)

  Amplitudes of harmonics of the 1-st variable:
  X1(c1)=  .9145E+02  X1(s1)=  .0000E+00  X1(1)=  .9145E+02
  X1(c3)=  .1396E-01  X1(s3)= -.1783E+00  X1(3)=  .1789E+00
  X1(c5)= -.4060E-02  X1(s5)=  .2120E-01  X1(5)=  .2158E-01

  Amplitudes of harmonics of the 2-st variable:
  X2(c1)=  .8884E-05  X2(s1)= -.9117E+02  X2(1)=  .9117E+02
  X2(c3)= -.5333E+00  X2(s3)= -.4176E-01  X2(3)=  .5350E+00
  X2(c5)=  .1057E+00  X2(s5)=  .2023E-01  X2(5)=  .1076E+00

  Amplitudes of harmonics of the 3-st variable:
  X3(c1)= -.9089E+02  X3(s1)= -.2267E-05  X3(1)=  .9089E+02
  X3(c3)= -.1250E+00  X3(s3)=  .1595E+01  X3(3)=  .1600E+01
  X3(c5)=  .1007E+00  X3(s5)= -.5268E+00  X3(5)=  .5363E+00

    Circular frequency of the fundamental harmonic = .9969E+00
```

As we can see, at ξ = 0.07 the content of higher harmonics in the solution is negligible.

For large values ξ, the calculation results obtained by the harmonic linearization method and the proposed method differ more. If we set $\xi = 7.0$ (with the same $T = 1.0; b = 10.0$) then by formulas (4.45) we obtain $\omega = 1.0$ and $A = 0.9095$.

Next is a printout of the results of numerical simulation at ξ=7.0 and taking into account the first, third and fifth harmonics (printout of the ***RezVan.dat*** file):

```
Periodic solution of the SAR equation
        Entered data:
Dzeta= .7000E+01 T= .1000E+01 B= .1000E+02
 EPS1= .1000E-02 EPS2= .1000E-03 H1= .1000E+01 HM= .1000E+01
 KER =   1    1    0    2    0    0    0    5    0    0
 K= 3
OM=  .1000E+01  A=  .9095E+00
X1C1=.9095E+00 X1S1=.0000E+00 X2C1=.0000E+00 X2S1= -.9095E+00
X3C1= -.9095E+00 X3S1=  .0000E+00
    6   18

C a l c u l a t i o n :
   Number of the highest harmonic = 5
Refined value 1 root at h= 1.000
(the solution was obtained after the 2-rd iteration)

Amplitudes of harmonics of the 1-st variable:
X1(c1)=  .1660E+01  X1(s1)=  .0000E+00  X1(1)=  .1660E+01
X1(c3)= -.5919E-01  X1(s3)= -.1553E-01  X1(3)=  .6119E-01
X1(c5)=  .1166E-01  X1(s5)=  .5756E-02  X1(5)=  .1301E-01

Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .4941E-07  X2(s1)= -.1228E+01  X2(1)=  .1228E+01
X2(c3)= -.3448E-01  X2(s3)=  .1314E+00  X2(3)=  .1359E+00
X2(c5)=  .2130E-01  X2(s5)= -.4316E-01  X2(5)=  .4813E-01

Amplitudes of harmonics of the 3-st variable:
X3(c1)= -.9089E+00  X3(s1)= -.3584E-07  X3(1)=  .9089E+00
X3(c3)=  .2917E+00  X3(s3)=  .7656E-01  X3(3)=  .3016E+00
X3(c5)= -.1597E+00  X3(s5)= -.7880E-01  X3(5)=  .1781E+00

Circular frequency of the fundamental harmonic =  .7400E+00
```

As you can see, the amplitude of the first harmonic of the variable x_1 (taking into account the first, third and fifth harmonics) is 1.66 and the circular frequency is 0.74. Here, the calculation results differ from those obtained by the harmonic linearization method (0.905 and 1.0), respectively, by 82% and 26%.

The results depend significantly on the number of higher harmonics taken into account, which is illustrated in Table 4.3.

Table 4.3. The results of solving equation (4.43)

n	$X_{1(1)}$	$X_{1(3)}$	$X_{1(5)}$	$X_{1(7)}$	$X_{1(9)}$	$X_{1(11)}$	ω
1	2.207						0.6414
3	2.207	0.082					0.6414
5	1.66	0.061	0.013				0.74
7	1.432	0.053	0.011	0.004			0.7968
9	1.31	0.048	0.01	0.0036	0.0016		0.8334
11	1.233	0.045	0.0095	0.0033	0.0015	0.0008	0.8589

This problem, unlike others given in this section, confirms the admissibility of applying the harmonic linearization method to its solution: when increasing the number of considered harmonics, the value of the amplitude of the first harmonic approaches 0.9095 and the value of the circular frequency - to 1.0, that is, the values obtained by the method of harmonic linearization. Thus, when harmonics up to and including the 23rd are taken into account, these variables take, respectively, the values of 1.059 (16% deviation) and 0.9266 (7% deviation).

The obtained result can be explained by the specificity of nonlinearity - it is a discontinuous function $y = b\,sign\,x$ that is sharply nonlinear (discontinuity of the first kind) only in the vicinity of the zero value of the variable x. The discontinuity of the function is the reason for the poor convergence of the Fourier series approximating the periodic dependence of the variables of this problem (many harmonics must be taken into account).

4.2.3. Example A .3

Consider example 4.1.2 from [56], this is the calculation of the self-oscillating mode described by the equation

$$\ddot{x} + \omega_0^2 x + \alpha(x^2 - b^2)\dot{x} = 0. \qquad (4.49)$$

When considering this example in [56], the analytical method of harmonic balance is used and the solution is sought in the form

$$x = a\sin\omega t. \qquad (4.50)$$

When applying this method, the amplitude a and circular frequency ω values were obtained in [56] by solving algebraic equations

$$\begin{aligned} &\alpha\, a\omega\left(\frac{a^2}{4} - b^2\right) = 0; \\ &-a\omega^2 + a\omega_0^2 = 0, \end{aligned} \qquad (4.51)$$

from which it is obtained

$$\omega = \omega_0; \quad a = 2b. \tag{4.52}$$

If given values

$$\alpha = 0.5;\ b = 1.5;\ \omega_0 = 0.5\ , \tag{4.53}$$

then we will get it

$$a = 3.0; \qquad \omega = 0.5\ . \tag{4.54}$$

We will calculate these self-oscillations (considering them as polyharmonic) by means of numerical simulation using DHM-S. For this equation (4.49) we write in the form (1.65):

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0, \tag{4.55}$$

here

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\|; \qquad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ \omega_0^2 x_1 + \alpha(x_1^2 - b^2)x_2 \end{matrix} \right\|; \qquad \vec{e} = \left\| \begin{matrix} e_1 \\ e_2 \end{matrix} \right\| = \left\| \begin{matrix} 0 \\ 0 \end{matrix} \right\|. \tag{4.56}$$

The expression for the derivative $d\vec{z}/d\vec{x}$, which is necessary for constructing an instantaneous (half-cycle) mathematical model of the process, has the form

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 0 & -1 \\ 2\alpha\, x_1 x_2 + \omega_0^2 & \alpha(x_1^2 - b^2) \end{matrix} \right\|. \tag{4.57}$$

The block of user program components for this case consists of the main ***Program Avto*** program and the ***Model*** procedure of the instantaneous process model.

To process the results and write them to the output file, the standard ***OUTP*** procedure from Block 5 of the DHM-S is used, so it is not included in the user block here.

The main ***Program Avto*** program and the ***Model*** procedure of this block look like this:

```
    Program Avto
!--   dX/dt+Z=E
!--   X=colon(x1,x2)
!--   Z=colon(z1,z2)
!--   E=colon(e1,e2)
!--   z1=-x2;   z2=A*(x1**2-B**2)*x2+C*x1
!--   e1=0;  e2=0
!--  (variables contain only odd harmonics)
!----------------------------------------
          Implicit none
          real,dimension(42)::Y0
          real,dimension(40)::E
          integer,dimension(10)::KER
          real::A,B,C,OM,EPS1,EPS2,H1,HM
```

```
          real::X1C1,X1S1,X2C1,X2S1
          integer::K,NG,NK
          common/MP/A,B,C  !--- area of memory shared with the Model procedure
          open(1,File='DaniVan.dat',status='old')
          read(1,*)A,B,C
          read(1,*)EPS1,EPS2,H1,HM
!-----  H1 – the value of  h  at which it is nessesary todetermine the root more precisely
!-----  HM – the maximum value of h
!-----  EPS1 – accuracy of integration
!-----  EPS2 – accuracy for Newton's method
          read(1,*)KER
          read(1,*)K
          read(1,*)OM
          read(1,*)X1C1,X1S1,X2C1,X2S1
!-----  KER – an array of controlvariables
!-----  K – the order of the system of differentialequations
!-----  X1C1,X1S1,X2C1,X2S1 – initial approximations of the amplitudes
!-----                              of first harmonics of variablts  X1  and   X2
   1 format(4E10.4)
   2 format(10I3)
          close(1)
      open(1,file='RezVan.dat')
          write(1,5)
   5 format(2X,'Solution of the equations of the self-oscillating mode'/10X,'Entered data:')
          write(1,14)A,B,C
  14 format(2X,' A=',E10.4,' B=',E10.4,' C=',E10.4)
          write(1,15)EPS1,EPS2,H1,HM
  15 format(2X,' EPS1=',E10.4, ' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
          write(1,16)KER
  16 format(3X,' KER =',10i5)
          Write(1,17)K
  17 format(2X,' K=',i2)
          write(1,18)OM
  18 format(2X,'OM=',E11.4)
          write(1,19)X1C1,X1S1,X2C1,X2S1
  19 format(2X,'X1C1=',E11.4,' X1S1=',E11.4,' X2C1=',E11.4,' X2S1=',E11.4)
   3 format(2X,10I5)
!-----  занесли введені дані до вихідного файлу
          call SizesV(KER(1),K,KER(8),NG,NK)
          write(1,3)NG,NK
!-----  NG – порядок простого вектора амплітуд
!-----  NK - порядок складеного вектора амплітуд
          Y0=0;  Y0(NK+1)=OM
          Y0(1)=X1C1; Y0(2)=X1S1; Y0(NG+1)=X2C1; Y0(NG+2)=X2S1
      write(1,13)
  13 format(/2X,'C a l c u l a t I o n :')
       call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
       close(1)
```

```
      stop
    end Program Avto
!-----------------------
    Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--  The procedure of the instantaneous model of the ssystem on half-period
        Implicit none
        real::AL
        integer,intent(in)::M,K,MK
        real,dimension(MK)::XC,YC,ZC
        real,dimension(MK,K)::YXC,ZXC
        real,dimension(K)::X,Z
        real,dimension(K,K)::ZX,BM
        real::A,B,C
        integer::i
        common/MP/A,B,C   !--  area of memory shared with the main program
        BM(1,1)=0.;  AL=0.
        YC(1)=0.;  YXC(1,1)=0.
        do i=1,M
          call DRAWOUTV(K,XC,MK,X,i)
          Z(1)=-X(2)
          Z(2)=A*(X(1)**2-B**2)*X(2)+C*X(1)
          ZX(1,1)=0.;     ZX(1,2)=-1.
          ZX(2,1)=2.*A*X(1)*X(2)+C
          ZX(2,2)=A*(X(1)**2-B**2)
          call DRAWUPV(K,Z,ZC,MK,i)
          call DRAWUPM(K,ZX,ZXC,MK,i)
        end do
        return
        end  Subroutine Model
```

As in previous examples A.1 and A.2, formal parameters ***AL*** and ***BM*** are not used in the MODEL procedure for this example, because the nonlinearity of equation (4.49) is not hysteretic (does not belong to the third group) and equation (4.49) in notation (4.55) is in the normal Cauchy form (1.65). For the same reason, the first four operators

```
BM(1,1)=0.;   AL=0.
YC(1)=0.;     YXC(1,1)=0.
```

perform unnecessary operations, their purpose is to block the compiler's message when compiling the procedure that the specified formal parameters in the body of the procedure are not used.

Since there are no nonlinearities of the second or third groups in the problem, all the actions of the procedure (calculation of the values of the composite nodal vector ***ZC*** and the composite matrix of nodal parameters ***ZXC***) are performed in the main cycle of the procedure.

Printout of the ***RezVan.dat*** file with the calculation results:

```
   Solution of the equations of the self-oscillating mode

   Entered data:
   A= .5000E+00  B= .1500E+01  C= .2500E+00
   EPS1= .1000E-02 EPS2= .1000E-03 H1= .1000E+01 HM= .1000E+01
  KER =   1    1    0    2    0    0   0   9   0   0
  K = 2
  OM=  .4000E+00
 X1C1=.3300E+01 X1S1= .1000E-01 X2C1= .1000E-01 X2S1= -.2000E+01
     10    20

  C a l c u l a t i o n :
    Amplitudes of harmonics of the 1-st variable:
  X1(c1)=  .3083E+01   X1(s1)=  .0000E+00   X1(1)=  .3083E+01
    X1(c3)= -.4235E+00   X1(s3)= -.4977E+00   X1(3)=  .6535E+00
  X1(c5)= -.4779E-01   X1(s5)=  .2488E+00   X1(5)=  .2533E+00
  X1(c7)=  .1034E+00   X1(s7)= -.5143E-01   X1(7)=  .1155E+00
  X1(c9)= -.5647E-01   X1(s9)= -.2426E-01   X1(9)=  .6146E-01

    Amplitudes of harmonics of the 2-st variable:
  X2(c1)= -.5489E-06   X2(s1)= -.1223E+01   X2(1)=  .1223E+01
  X2(c3)= -.5922E+00   X2(s3)=  .5038E+00   X2(3)=  .7775E+00
  X2(c5)=  .4934E+00   X2(s5)=  .9476E-01   X2(5)=  .5024E+00
  X2(c7)= -.1428E+00   X2(s7)= -.2872E+00   X2(7)=  .3207E+00
  X2(c9)= -.8659E-01   X2(s9)=  .2016E+00   X2(9)=  .2194E+00

  Circular frequency of the fundamental harmonic =  .3966E+00
```

Table 4.4 illustrates the dependence of the calculated values of the amplitudes of the harmonics of variable x_1 and circular frequency ω oscillations on the number of harmonics taken into account.

Table 4.4. Calculation results for equation (4.49)

n	$X_{1(1)}$	$X_{1(3)}$	$X_{1(5)}$	$X_{1(7)}$	$X_{1(9)}$	ω
1	3.0					0.5
3	3.120	0.7281				0.4206
5	3.085	0.6782	0.3042			0.3970
7	3.082	0.6549	0.2601	0.1328		0.3966
9	3.083	0.6535	0.2533	0.1155	0.0615	0.3966

As you can see, it is enough to take into account harmonics up to and including the seventh in the calculation. At the same time, the obtained amplitude and frequency values differ from values (4.54) by 2.6% and 20.7%, respectively. The time dependence of the variable contains sufficiently pronounced higher harmonics: the third (24%) and the fifth (8.3%).

4.2.4. Example A .4

In this section, we present the results of simulation of oscillations of a pendulum with a pushing force under viscous damping, which are described by the equation

$$\ddot{x} + \omega_0^2 x = f\ sign(\dot{x}) - 2h\dot{x}. \qquad (4.58)$$

Here, the pushing force is a kind of negative dry friction: when the speed is positive, the pushing force is also constant and positive; when the speed is negative, then it is also constant and negative.

In [56], taking

$$x = a\sin(\omega_0 t + \varphi), \qquad (4.59)$$

solved this problem by the averaging method (here it is - example 4.3.3) and got

$$a = \frac{2f}{\pi h \omega_0}. \qquad (4.60)$$

Having asked

$$f = 0.9; \quad h = 0.455; \quad \omega_0 = 1.0, \qquad (4.61)$$

by formula (4.60) we have

$$a = 1.259. \qquad (4.62)$$

Now let's calculate this nonlinear oscillation by means of numerical modeling using DGM software. For this purpose, we reduce equation (4.58) to the form (1.65)

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0, \qquad (4.63)$$

here

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\|; \quad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ \omega_0^2 x_1 + 2hx_2 - f\ sign\, x_2 \end{matrix} \right\|; \quad \vec{e} = \left\| \begin{matrix} e_1 \\ e_2 \end{matrix} \right\| = \left\| \begin{matrix} 0 \\ 0 \end{matrix} \right\|. \qquad (4.64)$$

The expression for the derivative $d\vec{z}/d\vec{x}$, which is necessary for constructing an instantaneous (on half-period) mathematical model of the process, has the form

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 0 & -1 \\ \omega_0^2 & 2h \end{matrix} \right\|. \qquad (4.65)$$

The block of user program components for this case consists of the main ***Program Majatnyk*** program and the ***Model*** procedure (instant model on the half-period), which have the form:

```
      Program Majatnyk
!---  Calculation of pendulum oscillations with a force that pushes
!--    dX/dt+Z=E
!--    X=colon(x1,x2)
!--    Z=colon(z1,z2)
!--    E=colon(e1,e2)
!--    z1=-x2;   z2=C*x1+2*h*x2-f*sign(x2)
!--    e1=0;  e2=0
!--  (variables contain only odd harmonics)
!----------------------------------------
          Implicit none
          real,dimension(42)::Y0
          real,dimension(40)::E
          integer,dimension(10)::KER
          real::C,H,F,OM,A,EPS1,EPS2,H1,HM
          real::X1C1,X1S1,X2C1,X2S1
          integer::K,NG,NK
          common/MP/C,H,F     !--- area of memory shared with the Model procedure
         open(1,File='DaniVan.dat',status='old')
          Read(1,1)C,H,F
          Read(1,1)EPS1,EPS2,H1,HM
          Read(1,2)KER
          Read(1,2)K
   1 Format(4E10.4)
   2 Format(10I3)
          Close(1)
      open(1,file='RezVan.dat')
          write(1,5)
   5 format(2X,'Calculation of pendulum oscillations ' /10X,'Entered data:')
          write(1,14)C,H,F
  14 format(2X,' C=',E10.4,' H=',E10.4,' F=',E10.4)
          write(1,15)EPS1,EPS2,H1,HM
  15 format(2X,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
          write(1,3)KER
          write(1,17)K
  17 format(2X,' K=',i2)
          OM=sqrt(C);  A=2.*F/(3.14159*H*OM)    !--- formula (4.60)
          write(1,18)OM,A
  18 format(2X,'OM=',E11.4,'  A=',E11.4)
          X1C1=A; X1S1=0.; X2C1=0.; X2S1=-A*OM
!--- X1C1,X1S1,X2C1,X2S1 – initial values of the amplitudes of the
!---                              first harmonics of variables
          write(1,19)X1C1,X1S1,X2C1,X2S1
  19 format(2X,'X1C1=',E11.4,' X1S1=',E11.4,' X2C1=',E11.4,' X2S1=',E11.4)
   3 Format(2X,8I5)
          call SizesV(KER(1),K,KER(8),NG,NK)
          Write(1,3)NG,NK
!-----  NG – the order of a simple vector of amplitudes
!-----  NK - the order of the composite vector of amplitudes
          Y0=0;  Y0(NK+1)=OM
```

```
          Y0(1)=X1C1; Y0(2)=X1S1; Y0(NG+1)=X2C1; Y0(NG+2)=X2S1
      write(1,13)
   13 format(/2X,'C a l c u l a t I o n :')
      call HARMOSC(K,Y0,E,NK,HM,H1,EPS1,EPS2,KER)
      close(1)
      stop
      end Program Majatnyk
!--------------------------
    Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--  Instant model of the process (at half period)
          Implicit none
          real::AL
          integer,intent(in)::M,K,MK
          real,dimension(MK)::XC,YC,ZC
          real,dimension(MK,K)::YXC,ZXC
          real,dimension(K)::X,Z
          real,dimension(K,K)::ZX,BM
          real::C,H,F
          integer::i
          Common/MP/C,H,F   !--  A shared area of memory with the main program
          BM(1,1)=0.;  AL=0.
          YC(1)=0.;  YXC(1,1)=0.
          do i=1,M
            call DRAWOUTV(K,XC,MK,X,i)
            Z(1)=-X(2)
            Z(2)=C*X(1)+2.*H*X(2)-sign(F,X(2))
            ZX(1,1)=0.;   ZX(1,2)=-1.
                ZX(2,1)=C;    ZX(2,2)=2.*H
            call DRAWUPV(K,Z,ZC,MK,i)
            call DRAWUPM(K,ZX,ZXC,MK,i)
          end do
          return
          end  Subroutine Model
```

And in this example, as in several previous ones, the formal parameters ***AL*** and ***BM*** are not used in the ***Model*** procedure, because the nonlinearity of equation (4.58) is not hysteretic (does not belong to the third group) and equation (4.58) in the notation (4.63) is in the normal Cauchy form (1.65). For the same reason, the first four operators

```
BM(1,1)=0.;   AL=0.
YC(1)=0.;   YXC(1,1)=0.
```

are performing unnecessary operations, their purpose is to block the compiler's message when compiling the procedure that the specified formal parameters in the body of the procedure are not used.

Since there are no nonlinearities of the second or third groups in the problem, the values of the composite nodal vector ***ZC*** and the composite matrix of nodal parameters ***ZXC*** are performed in the main cycle of the procedure.

Printout of the ***RezVan.dat*** source file with the calculation results:

```
   Calculation of pendulum oscillations
   Entered data:
   C= .1000E+01 H= .4550E+00 F= .9000E+00
   EPS1= .1000E-02 EPS2= .1000E-03 H1= .1000E+01 HM= .1000E+01
   KER =   1    1    0    2    0    0     0   7   0   0
   K = 2
  OM=  .1000E+01  A=  .1259E+01
X1C1= .1259E+01 X1S1= .0000E+00 X2C1= .0000E+00  X2S1=-.1259E+01
      8     16
   C a l c u l a t i o n :
  Amplitudes of harmonics of the 1-st variable:
  X1(c1)=  .1340E+01    X1(s1)=  .7881E-14    X1(1)=  .1340E+01
  X1(c3)= -.6973E-02    X1(s3)=  .5264E-01    X1(3)=  .5310E-01
  X1(c5)= -.6343E-02    X1(s5)=  .8976E-02    X1(5)=  .1099E-01
  X1(c7)= -.3347E-02    X1(s7)=  .2136E-02    X1(7)=  .3971E-02

  Amplitudes of harmonics of the 2-st variable:
  X2(c1)=  .3444E-06   X2(s1)= -.1243E+01    X2(1)=  .1243E+01
  X2(c3)=  .1465E+00   X2(s3)=  .1941E-01    X2(3)=  .1478E+00
  X2(c5)=  .4163E-01   X2(s5)=  .2942E-01    X2(5)=  .5098E-01
  X2(c7)=  .1387E-01   X2(s7)=  .2174E-01    X2(7)=  .2578E-01

  Circular frequency of the fundamental harmonic =  .9277E+00
```

The results of solving this problem, depending on the value n, are shown in Table 4.5.

Table 4.5. Calculation results for equation (4.58)

n	a_1	a_3	a_5	a_7	ω
1	1.296				0.9706
3	1.296	0.048			0.9706
5	1.326	0.051	0.011		0.9419
7	1.340	0.053	0.011	0.004	0.9277

Let's modify the problem discussed above - instead of viscous damping, we will install damping that creates a gas medium: let the resistance force be proportional to the product $\dot{x}|\dot{x}|$. For this case, the oscillation of the pendulum with the pushing force is described by the equation

$$\ddot{x} + \omega_0^2 x = f\ sign(\dot{x}) - \alpha\, \dot{x}|\dot{x}|. \qquad (4.66)$$

Taking what

$$x = a\sin(\omega_0 t + \varphi)\ , \qquad (4.67)$$

by averaging we get

$$a = \sqrt{\frac{3f}{2\alpha\omega_0}}\ . \qquad (4.68)$$

Having accepted the value

$$f = 0.9; \quad \alpha = 1.3; \quad \omega_0 = 1.0, \tag{4.69}$$

we have

$$a = 1.019\ . \tag{4.70}$$

Now let's calculate this nonlinear oscillation by numerical simulation using DHM-S. For this, equation (1) is reduced to the form

$$\frac{d\vec{x}}{dt} + \vec{z} - \vec{e} = 0, \tag{4.71}$$

here

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\|; \qquad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} -x_2 \\ \omega_0^2 x_1 + \alpha x_2 |x_2| \end{matrix} \right\|; \qquad \vec{e} = \left\| \begin{matrix} e_1 \\ e_2 \end{matrix} \right\| = \left\| \begin{matrix} 0 \\ 0 \end{matrix} \right\|. \tag{4.72}$$

The expression for the derivative $d\vec{z}/d\vec{x}$ required for constructing an instantaneous mathematical model of the process at the half-period has the form

$$\frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} 0 & -1 \\ \omega_0^2 & 2\alpha|x_2| \end{matrix} \right\| \ . \tag{4.73}$$

The block of user program components for this case consists of the Main program **`Program Majat1`** and the **`MODEL`** procedure of the instantaneous process model (on a half-cycle), which have the form:

```
    Program Majat1
!-- Calculation of pendulum oscillations in a gaseous environment
!--    dX/dt+Z=E
!--    X=colon(x1,x2)
!--    Z=colon(z1,z2)
!--    E=colon(e1,e2)
!-   z1=-x2;
!--    z2=C*x1+h*x2*abs(x2)-f*sign(x2)
!--    e1=0;  e2=0
!--  (variables contain only odd harmonics )
!----------------------------------------
          Implicit none
          real,dimension(42)::Y
          real,dimension(40)::E
          integer,dimension(10)::KER
          real::C,H,F,OM,A,EPS1,EPS2,H1,HM
          real::X1C1,X1S1,X2C1,X2S1
```

```
          integer::K,NG,NK
          common/MP/C,H,F    !--- area of memory shared with the Model procedure
            open(1,File='DaniMaj.dat',status='old')
          read(1,*)C,H,F
          read(1,*)EPS1,EPS2,H1,HM
          read(1,*)KER
          read(1,*)K
          close(1)
            open(1,file='RezMaj.dat')
          write(1,5)
   5 format(2X,'Calculation of pendulum oscillations in gas' /10X,'Entered data:')
          write(1,14)C,H,F
  14 format(2X,' C=',E10.4,' H=',E10.4,' F=',E10.4)
          write(1,15)EPS1,EPS2,H1,HM
  15 format(2X,' EPS1=',E10.4,' EPS2=',E10.4,' H1=',E10.4,' HM=',E10.4)
          write(1,16)KER
  16 format(2X,' KER =',10i5)
          Write(1,17)K
  17 format(2X,' K=',i2)
          OM=sqrt(C);  A=sqrt(3.*F/(2.*H*OM**2))
          write(1,18)OM,A
  18 format(2X,'OM=',E11.4,'  A=',E11.4)
          X1C1=A; X1S1=0.; X2C1=0.; X2S1=-A*OM
!--- X1C1,X1S1,X2C1,X2S1 – initial values of the amplitudes of the
!---                                first harmonics of variables
          write(1,19)X1C1,X1S1,X2C1,X2S1
  19 format(2X,'X1C1=',E11.4,' X1S1=',E11.4,' X2C1=',E11.4,' X2S1=',E11.4)
          call SizesV(KER(1),K,KER(8),NG,NK)
          write(1,20)NG,NK
  20 format(2X,'NG=',i3,'  NK=',i3)
          Y=0;  Y(NK+1)=OM
          Y(1)=X1C1; Y(2)=X1S1; Y(NG+1)=X2C1; Y(NG+2)=X2S1
       write(1,21)
  21 format(/2X,'C a l c u l a t I o n :')
       call HARMOSC(K,Y,E,NK,HM,H1,EPS1,EPS2,KER)
       close(1)
       stop
       end Program Majat1
!-------------------------------------------------

     Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!  Процедура миттєвої моделі процесу на півперіоді
!-------------------
          Implicit none
          real::AL
```

```
      integer,intent(in)::M,K,MK
      real,dimension(MK)::XC,YC,ZC
      real,dimension(MK,K)::YXC,ZXC
      real,dimension(K)::X,Z
      real,dimension(K,K)::ZX,BM
      real::C,H,F
      integer::i
      common/MP/C,H,F   !--  A shared area of memory with the main program
       BM(1,1)=0.;  AL=0.
       YC(1)=0.;  YXC(1,1)=0.
       do i=1,M
        call DRAWOUTV(K,XC,MK,X,i)
        Z(1)=-X(2)
        Z(2)=C*X(1)+H*X(2)*abs(X(2))-sign(F,X(2))
        ZX(1,1)=0.
        ZX(1,2)=-1.
        ZX(2,1)=C
        ZX(2,2)=2*H*abs(X(2))
        call DRAWUPV(K,Z,ZC,MK,i)
        call DRAWUPM(K,ZX,ZXC,MK,i)
       end do
       return
       end Subroutine Model
```

Printout of the source file ***RezMaj.dat*** with the calculation results:

```
   Calculation of pendulum oscillations in gas

        Entered data:
   C= .1000E+01 H= .1300E+01 F= .9000E+00
   EPS1= .1000E-02 EPS2= .1000E-03 H1= .1000E+01 HM= .1000E+01
   KER =  1    1    0    2    0    0    0    5    9    1
   K= 2
OM = .1000E+01   A = .1019E+01
X1C1= .1019E+01 X1S1= .0000E+00 X2C1= .0000E+00 X2S1= -.1019E+01
       6   11   12   36

  C a l c u l a t i o n :

          The h-characteristic is calculating
     ---------------------------------------
     Refined value 1 root at h= 1.000
  (the solution was obtained after the 4-rd iteration)

  Amplitudes of harmonics of the 1-st variable:
  X1(c1)=  .1216E+01  X1(s1)= -.6529E-12  X1(1)= .1216E+01
  X1(c3)=  .1452E-01  X1(s3)=  .8249E-01  X1(3)= .8376E-01
  X1(c5)= -.1268E-01  X1(s5)=  .1592E-01  X1(5)= .2035E-01
```

```
Amplitudes of harmonics of the 2-st variable:
X2(c1)=  .4739E-06  X2(s1)= -.9979E+00  X2(1)=  .9979E+00
X2(c3)=  .2031E+00  X2(s3)= -.3574E-01  X2(3)=  .2062E+00
X2(c5)=  .6532E-01  X2(s5)=  .5202E-01  X2(5)=  .8350E-01

Circular frequency of the fundamental harmonic =  .8207E+00
```

The results of solving this problem, depending on the value n, are shown in Table 4.6.

Table 4.6. Calculation results for equation (4.66)

n	a_1	a_3	a_5	a_7	ω
1	1.057				0.9639
3	1.197	0.084			0.8398
5	1.216	0.084	0.020		0.8207
7	1.223	0.084	0.020	0.007	0.8134

4.3. Example of calculating of parametric oscillations

Consider, as an example, the calculation of parametric oscillation in an electric circuit given in [39]. There, a simple electric circuit is considered, formed by series-connected inductance L, nonlinear active resistance r, the value of which, depending on the circuit current, is determined by the formula

$$r = R_0(1 + \beta_0 i^2), \tag{4.74}$$

here R_0 and β_0 - constant values, and with a variable capacity C, the value of which is a periodic function of time and is determined by the formula

$$C = \frac{C_0}{1 + m\cos 2\omega t}, \tag{4.75}$$

here C_0, m, ω - constant values.

To calculate the amplitude of current fluctuations in this circuit (in other words, the amplitude of the alternating current) in [39], the method of slowly changing amplitudes was used. It is assumed that due to the generation of parametric oscillations, the current in the circuit changes harmonically with the frequency ω according to the expression

$$i = I_c \cos\omega t + I_s \sin\omega t. \tag{4.76}$$

Taking into account that the circuit is an oscillating system with low dissipation, neglecting the terms of the second order of smallness in the equations describing the circuit and averaging the values of the circuit parameters over the oscillation period, in [39] the so-called shortened equations were obtained, in which the signs of the derivatives are amplitudes I_c and I_s. An expression for the current amplitude is obtained from the stationary solution of these equations

$$I = \sqrt{\frac{4}{3\beta_0}\left(-1+\frac{\omega L}{R_0}\sqrt{m^2/4-\xi^2}\right)}\ , \qquad (4.77)$$

here

$$\xi = 1-\frac{1}{\omega^2 L C_0}\ . \qquad (4.78)$$

Formulas (4.74)–(4.78) use the same notation as in paragraph 4.5 of the source [39].

Given numerical values of $L = 0.025$ Hn; $R_0 = 0.09$ ohms; $\beta_0 = 0.1\ 1/A^2$; $C_0 =$ 0.004 F; $m =$ 0.13; $\omega =$ 100 1/s, by formulas (4.77) and (4.78) we get $\xi = 0$ 3.277 and $I =$ 3,277 A.

We will solve the same problem by means of numerical polyharmonic modeling using DHM-S.

The equations describing the electrical circuit under consideration, in the form of entry (1.64), have the form

$$\frac{d\,\vec{y}}{dt} + \vec{z} - \vec{e} = 0\,, \qquad (4.79)$$

here

$$\vec{x} = \left\| \begin{matrix} x_1 \\ x_2 \end{matrix} \right\| = \left\| \begin{matrix} i \\ u_c \end{matrix} \right\|; \quad \vec{y} = \left\| \begin{matrix} y_1 \\ y_2 \end{matrix} \right\| = \left\| \begin{matrix} \phi \\ u_c \end{matrix} \right\|; \quad \vec{z} = \left\| \begin{matrix} z_1 \\ z_2 \end{matrix} \right\| = \left\| \begin{matrix} ri + u_c \\ -\dfrac{i}{u_c} \end{matrix} \right\|; \quad \vec{e} = 0; \qquad (4.80)$$

u_c - capacitor voltsge;

$$\phi = L\,i \qquad (4.81)$$

- flux coupling of the inductive element of the circuit.

The derivatives $d\vec{y}/d\vec{x}$ and $d\vec{z}/d\vec{x}$ are as follows

$$\frac{d\vec{y}}{d\vec{x}} = \left\| \begin{matrix} L & 0 \\ 0 & 1 \end{matrix} \right\|; \quad \frac{d\vec{z}}{d\vec{x}} = \left\| \begin{matrix} R_0(1+3\beta_0 i^2) & 1 \\ -\dfrac{1+m\cos 2\eta}{C_0} & 0 \end{matrix} \right\|. \qquad (4.82а,б)$$

The text of the main program and the *Model* procedure from the block of user software components:

```
    PROGRAM ParamOscil
!--  A program for calculating parametric oscillations
!-- in an electric circuit with elements:
!-- periodically variable electrical capacity,
!-- nonlinear active resistance and nonlinear inductance
!------------------------
        Implicit none
        real,dimension(22)::X
        real,dimension(20)::E
        integer,dimension(10)::KER
```

```
          real::OM,EPS1,EPS2,H1,HM
          real::R0,BET0,C0,AM,ST1,DST
          real,dimension(8)::PT
          real::CI1,CI2,CU1,CU2
          integer::K,NG,NK,NT,i
          common/MP/R0,BET0,C0,AM,NT,ST1,DST,PT
          open(1,File='DaniV.dat',status='old')
          read(1,1)OM,EPS1,EPS2,H1,HM
          read(1,1)R0,BET0,C0,AM
          read(1,2)NT
          read(1,1)ST1,DST
          read(1,1) (PT(i),i=1,NT)
          read(1,2)KER
          read(1,2)K
          read(1,1)CI1,CI2,CU1,CU2
!------ CI1,CI2,CU1,CU2 – initial values of the amplitudes of the
!---                          first harmonics of variables
   1 format(5E10.4)
   2 format(10I3)
          close(1)
      open(1,FILE='RezV.DAT')
      write(1,10)
  10 format(/5X, 'Incoming data :'/)
       write(1,4)OM,EPS1,EPS2,H1,HM
          write(1,4)R0,BET0,C0,AM
          write(1,3)NT
          write(1,4)ST1,DST
          write(1,4) (PT(i),i=1,NT)
          write(1,11)KER
  11 format(2X.'KER = ',10i5)
          write(1,12)K
  12 format(2X,'K = ',i5)
          write(1,4)CI1,CI2,CU1,CU2
   3 format(1X,10I5)
   4 format(1X,5E11.4)
          write(1,20)
  20 format(/5X,'T h e   r e s u l t s   :'/)
    call SizesV(KER(1),K,KER(8),NG,NK)
          write(1,3)NG,NK
          X=0.;  X(1)=CI1;    X(2)=CI2;
               X(NG+1)=CU1; X(NG+2)=CU2;   X(NK+1)=OM
          call HARMOSC(K,X,E,NK,HM,H1,EPS1,EPS2,KER)
          close(1)
    stop
    end Program  ParamOscil
!--------------------------
     Subroutine Model(AL,M,K,MK,XC,YC,ZC,YXC,ZXC,BM)
!--   The subroutine implements the instantaneous model on a half-period:
!--   for the value of the composite nodal vector XC
!--  determines the value of the composite nodal vectors YC,ZC
!--   and  matrices YXC and ZXC of derivatives
!--- AL - hysteresis loop narrowing factor (not used)
!--- M is the number of nodes per half cycle
!--- K is the order of the system of differential equations
!--- MK=M*K
```

```
!--- BM - matrix of coefficients (not used)
!------------------------
      Implicit none
      real::AL
      integer,intent(in)::M,K,MK
      real, dimension(MK)::XC,YC,ZC
      real,dimension(MK,K)::YXC,ZXC
      real,dimension(K)::X,Y,Z
      real,dimension(K,K)::YX,ZX,BM
      integer::i,NT
      real::R0,BET0,C0,AM,ST1,DST,LD,ET,AI
      real,dimension(8)::PT
      common/MP/R0,BET0,C0,AM,NT,ST1,DST,PT
      BM(1,1)=0.; AL=0.
      do i=1,M
        call DRAWOUTV(K,XC,MK,X,i)
        AI=i-1; ET=3.1416*AI/M   !--- angular coordinate of the node
        call INTLIN(X(1),Y(1),LD,ST1,DST,PT,NT)
        Y(2)=X(2)
        Z(1)=R0*X(1)+R0*BET0*X(1)**3+X(2)
        Z(2)=-X(1)*(1.+AM*cos(2.*ET))/C0
        YX(1,1)=LD
        YX(1,2)=0.
        YX(2,1)=0.
        YX(2,2)=1.
        ZX(1,1)=R0+3.*R0*BET0*X(1)**2
        ZX(1,2)=1.
        ZX(2,1)=-(1.+AM*cos(2.*ET))/C0
        ZX(2,2)=0.
        call DRAWUPV(K,Y,YC,MK,i)
        call DRAWUPV(K,Z,ZC,MK,i)
        call DRAWUPM(K,YX,YXC,MK,i)
        call DRAWUPM(K,ZX,ZXC,MK,i)
      end do
      return
      end    subroutine Model
```

Although in this problem the inductance L of the electric circuit is set constant, in the main program it is provided that it can also be variable - a function of the current, and it is provided to set values for the variables ***ST1, DST, PT, NT***, which set the tabular curve of magnetization of the inductive element (ST1 is the value of the current at which the initial linear part of the magnetization curve ends; ***DST*** is the step of the table; ***PT*** is the table specifying the value of flux coupling in the nodes of the table; ***NT*** is the number of nodes of the table), and then instead of (4.81) we have

$$\phi = \phi[i]. \qquad (4.83)$$

The value L in formula (4.82a) is calculated as

$$L = d\phi / di. \qquad (4.84)$$

To determine the values ϕ and L and in the ***Model*** procedure, the ***INTLIN*** procedure from Block 4 of the DHM-S (see section 2.2.4.2) is called for execution, which performs linear interpolation from the ***PT*** table.

According to the described program, calculations were performed for the case of constant inductance, the magnetization curve is presented as a straight line, the tangent of the angle of inclination of which is equal to L=0.025 Hn, and this straight line is presented by table with three nodes.

The calculation results (only the first harmonic is taken into account) are as follows:

```
      Incoming data:
    .1000E+03  .1000E-01  .3000E-02  .1000E+01  .1000E+01
    .9000E-01  .1000E+00  .4000E-02  .1300E+00
      3
    .1000E+01  .5000E+00
    .2500E-01  .3750E-01  .5000E-01
      KER = 1    0    0    1    0    0    0    1    0    0
      K = 2
    .1500E+01 -.1500E+01  .3000E+01  .4000E+01

      T h e    r e s u l t s :
      2    4
    Number of the highest harmonic =  1
    The initial value of the vector of amplitudes :
       .1500E+01  -.1500E+01   .3000E+01   .4000E+01
       .1000E+03   .0000E+00
    Inconsistencies for the initial value of the vector of
amplitudes:
      -.5694E+00   .6944E-01   .6253E+00   .5063E+02
       .0000E+00   .0000E+00
      ----------------------------------------
      Refined value 1 root at h= 1.000
   (the solution was obtained after the 2-rd iteration)

   Amplitudes of harmonics of the 1-st variable:
   X1(c1)=  .2317E+01  X1(s1)= -.2317E+01  X1(1)=  .3277E+01

   Amplitudes of harmonics of the 2-st variable:
   X2(c1)=  .5417E+01  X2(s1)=  .6170E+01  X2(1)=  .8211E+01
```

Here it is necessary to pay attention that in the input data the 4th element of the ***KER*** array is given the value 1, thereby specifying that the simulated oscillation is parametric.

The above calculation results show that the obtained value of the amplitude of the 1st harmonic of the loop current is equal to 3.277 A, and this coincides with the accuracy of the third sign with the results of the analytical calculation by the method of slowly changing amplitudes using shortened differential equations. This confirms the theoretical correctness of using the method of slowly varying amplitudes to solve this problem with the replacement of the differential equations describing the processes in the circuit with shortened equations: the results of the calculation based on the shortened equations are the same as the numerical solution of the full differential equations without them simplifications and neglect by individual members.

Repeating the same calculations, but taking into account the 3rd, 5th and 7th harmonics in addition to the 1st harmonic, it was obtained that the 3rd harmonic of the current is 0.4%, the 5th and 7th harmonics of the current practically zero. The practical absence of variable higher harmonics in time dependences is explained as follows. In this problem, the generation of higher harmonics can only be carried out by an active resistance, because it is nonlinear, but, given that the voltage drop on it at the given input data is much smaller than the voltage drops on the capacitor and inductance, therefore, the higher harmonics of the current generated by its nonlinearity and voltages are negligible.

Let's complicate the task. Let the inductance also be nonlinear and the dependence of flux coupling on the current (magnetization curve, similar to Fig. 4.2) given by the table. Its initial part is linear, and it determines the same inductance (0.025 H) as in the case considered above. After the loop current reaches a value of 1.0 A, the nonlinear part of the magnetization curve begins.

The calculation results in this case, taking into account harmonics up to and including the 7th, are as follows:

```
      Incoming data:
   .1000E+03  .1000E-01  .3000E-02  .1000E+01  .1000E+01
   .9000E-01  .1000E+00  .4000E-02  .1300E+00
     8
   .1000E+01  .2500E+00
   .2500E-01  .2830E-01  .3120E-01  .3320E-01  .3450E-01
   .3550E-01  .3620E-01  .3670E-01
     1    0    0    1    0    0    0    7    0    0
     2
   .1500E+01 -.1500E+01  .3000E+01  .4000E+01
     T h e    r e s u l t s :
     8   16
      Number of the highest harmonic =  7
   The initial value of the vector of amplitudes :
      .1500E+01  -.1500E+01   .0000E+00   .0000E+00
      .0000E+00   .0000E+00   .0000E+00   .0000E+00
      .3000E+01   .4000E+01   .0000E+00   .0000E+00
      .0000E+00   .0000E+00   .0000E+00   .0000E+00
      .1000E+03   .0000E+00
   Inconsistencies for the initial value of the vector of
amplitudes:
      .3870E+00   .1026E+01   .1067E+01  -.1098E+01
      .1689E+00   .1690E+00   .1288E+00  -.1287E+00
      .6253E+00   .5063E+02  -.2437E+02   .2438E+02
      .9936E-03   .1982E-02   .1569E-02   .3033E-02
      .0000E+00   .0000E+00
     Refined value 1 root at h= 1.000
  (the solution was obtained after the 2-rd iteration)
  Amplitudes of harmonics of the 1-st variable:
  X1(c1)=  .4170E+00  X1(s1)= -.1150E+01  X1(1)=  .1223E+01
  X1(c3)= -.4553E-01  X1(s3)=  .1917E-01  X1(3)=  .4940E-01
  X1(c5)=  .3108E-01  X1(s5)=  .6536E-02  X1(5)=  .3176E-01
  X1(c7)= -.7244E-02  X1(s7)= -.9633E-02  X1(7)=  .1205E-01

  Amplitudes of harmonics of the 2-st variable:
  X2(c1)=  .2685E+01  X2(s1)=  .1103E+01   X2(1)=  .2902E+01
  X2(c3)=  .4596E-01  X2(s3)= -.1367E-01   X2(3)=  .4795E-01
  X2(c5)= -.3574R-02  X2(s5)=  .1383E-01   X2(5)=  .1428E-01
  X2(c7)=   .3293E-02  X2(s7)= -.1867E-02  X2(7)=  .3785E-02
```

As you can see, the calculation results are significantly different from the previous case, when the inductance is constant: 1st harmonic current = 1.223 A (it differs by 2.68 times!). Noticeable values of higher harmonics: 3rd – 4%, 5th – 2.6% and 7th – 0.9%.

In this case, the maximum values of the loop current, and, therefore, the amplitude of the first harmonic of the current are significantly limited by the nonlinearity (saturation) of the inductance.

The method of slowly changing amplitudes is no longer suitable for solving this problem with a nonlinear inductance.

Afterword

Examples of numerical modeling of nonlinear oscillations, considered in the last chapter of this book, are simple: with a small number of variables (no more than three), with nonlinearities that are functions of only one variable, etc. The selection of simple examples was determined by the author's efforts to focus the reader's attention on ways of using the proposed method and its software. It is advisable to undertake modeling of nonlinear oscillations in more complex systems only after thoroughly mastering the method itself and its software.

Having worked through this book, the user is already able to tackle such more complex tasks. And here, the main focus should be on the development of an instantaneous (period or half-period) model of the system process with the most detailed consideration of nonlinearities.

Many problems for calculating periodic processes in such complex objects as electric machines (synchronous, asynchronous, direct current) were solved by the method described in the book, taking into account their nonlinearities, in particular, the saturation of the magnetic circuit, the presence of semiconductor windings in their circuits and others nonlinear elements, when considering instantaneous models of electric machines both from the standpoint of the theory of electric circuits and from the standpoint of the theory of the electromagnetic field [9, 11 – 19, 21 – 26]. And the experience gained in solving these problems can be used in the development of models of nonlinear oscillations in objects of a different physical nature - mechanical, acoustic, radio-electronic, etc.

This book does not consider some additional possibilities of the proposed method, in particular, taking into account the symmetry of periodic processes (such symmetry exists, for example, in symmetrical multiphase electric circuits), which allows minimizing the number of unknown amplitude vectors in the problem. If necessary, the reader can familiarize himself with such possibilities in [15, 17, 20]. This also applies to the search for periodic solutions of nonlinear systems of differential equations with partial derivatives with periodization in time and one of the spatial coordinates (here calculation of the stationary electromagnetic field in massive ferromagnetic media [19, 23, 25]).The proposed method and its software do not impose any restrictions on the dimension of the problem (the number of variables) and the number of harmonics taken into account, the only limitation here is only the performance of the computer used - its speed and memory capacity.

However, the author, promoting his method, would consider it unreasonable to contrast it with other methods of calculating (modeling) nonlinear oscillations. This is only an alternative that has its own niche in this area. And the most expedient is often the use of not one, but several methods in their interaction. Thus, in particular, in several examples given in Chapter 4, initial approximations of amplitudes of harmonics of oscillations are obtained by analytical methods (harmonic linearization, slowly changing amplitudes, asymptotic Bogolyubov-Mitropolsky, harmonic balance, etc.).

The author wishes the reader and user of this book, who decides to use the proposed differential harmonic method and its software in their scientific or engineering developments, and may even improve them (the author would welcome this), success in solving various problems, both simple and complex , in numerical modeling of nonlinear oscillations in systems of different physical nature.

References

1. Andreev V.S.. Theory of non-linear electrical circuits: a textbook for universities. - Moscow: Radio and communication, 1982. - 280 p. [in Russ.]
2. Andronov A.A., Witt A.A., Khaikin S.E. Theory of vibrations. – Moscow: Nauka, 1981. – 568 p. (Russian)
3. Babitskyi V.I., Krupenin V.L. Oscillations in strongly nonlinear systems. – Mockow: Nauka, 1985. – 235 p. (Russian)
4. Bessonov L.A. Nonlinear electrical circuits. - Moscow: Higher. school, 1977. - 343 p. (Russian)
5. Bogolyubov N.N.,.Mitropolskyi Yu.L. Asymptotic Methods in the Theory of Nonlinear Oscillations. – Moscow: Nauka, 1974, - 504 p. (Russian)
6. Bondarenko V.M. Questions of analysis of nonlinear circuits. - Kyiv.: Nauk. Dumka, 1967. - 159 p. (Russian)
7. Bozhko A.E., Golub N.M. Dynamic-energy connections of oscillatory systems. - Kyiv: Nauk. Dumka, 1980. - 188 p. (Russian)
8. Butenin N.V., Neimark Yu.I., Fufaev N.L. Introduction to the theory of nonlinear oscillations. – Moscow.: Nauka, 1976. - 384 p. (Russian).
9. Gavdyo I.R., Glukhivskyi L.Y., Malyar V.S.. “Calculation of stable modes and static characteristics of asynchronous motors with shielded poles by the differential harmonic method”, Visnyk NU “Lvivska politechnika” “Elektroenergetychni ta elektromehanichni systemy” No. 449. Lviv, Ukraine: Vyshcha shkola, 2002. - pp. 48 – 53. (Ukrainian).
10. Garasimiv I.I., Mandziy B.A., Felyshtyn O.I. Machine modeling of electronic and electrical devices. - Lviv: Svit, 1991. - 133 p. (Russian)
11. Glukhivskyi L.Y. “Calculation of the characteristics of a ferroresonant circuit by a differential harmonic method”. Izvestija vuzov, Energetika, Minsk,1979, No. 11. (Russian)
12. Glukhivskyi L.Y. “The main provisions of the differential harmonic method for calculating periodic processes in nonlinear circuits”, Elektrichestvo, Moskov,1980, No. 11, pp. 17 - 22. (Russian)
13. Glukhivskyi L.Y. “Differential harmonic method for calculating periodic processes in circuits with valves” // Izv. universities Electromechanics, 1980, No. 1. (Russian)
14. Glukhivskyi L.Y. “Calculation of large oscillations of a clear-pole synchronous machine by the differential harmonic method” // Technical Electrodynamics, 1980, No. 1. (Russian)
15. Glukhivskyi L.Y. “Calculation of the periodic process in the bridge rectifier circuit by the differential harmonic method” // Technical Electrodynamics, 1980, No. 2. (Russian)
16. Glukhivskyi L.Y. “Determination of the asymptotic stability of periodic modes of nonlinear electric circuits by the differential harmonic method” // Izv. universities Energy, 1981, No. 1. (Russian)
17. Glukhivskyi L.Y. Calculation of periodic processes of electrotechnical devices (differential harmonic method). – Lviv: Higher School. Publishing House at Lviv University, 1984. - 164 p. (Ukrainian)
18. Glukhivskyi L.Y. “Calculation by the differential harmonic method of the steady-state modes of a clear-pole synchronous machine with excitation from an additional stator winding” // Technical Electrodynamics, 1984, No. 1. (Russian)

19. Glukhivskyi L.Y. “Calculation of the stationary electromagnetic field in multilayer media by the differential harmonic method” // Theoretical electrical engineering, vol. 40, 1986. (Russian)

20. Glukhivskyi. L.Y. Periodic processes in nonlinear electrical engineering (differential harmonic method and its software). - Kyiv: Alfa P&K, 2005. - 159 p. (Ukrainian)

21. Glukhivskyi LY., Bilyakovskyi I.E. “Numerical modeling of thyristor voltage regulators by the differential harmonic method” // Technical Electrodynamics, 1986, No. 4. (Russian)

22. Glukhivskyi L.Y., Bilyakovskyi I.E.. “Modeling by the differential harmonic method of polyharmonic asymmetric modes of operation of asynchronous machines taking into account saturation” // Proceedings of the Ph.D. conference dedicated to the 100th anniversary of the birth of Prof. T. Gubenko. Electromechanics, theory and practice. - Lviv: DULP publishing house, 1996. (Ukrainian)

23. Glukhivskyi L.Y., Kostiv A.P. Calculation of the magnetic field of an asynchronous machine with a massive rotor. Lviv: Higher School, 1983. – 128 p. (Russian)

24. Glukhivskyi LY., Malyar V.S. “Calculation of steady-state asynchronous motion of a clearly poled synchronous machine by the differential harmonic method” // Izv. universities Energy, 1984, No. 2. (Russian)

25. Glukhivskyi L.Y., Muzychka I.M., Filts R.V. “Determination of the periodic in time electromagnetic field in a conducting ferromagnetic plate by the differential harmonic method” // Mathematical methods and physical and chemical fields, vol. 20. - Kyiv: Naukova dumka, 1984. (Russian)

26. Glukhivskyi L. “Algorithm of the mathematical model of a nonlinear hysteresis inductive element in a periodic process // Theoretical electrical engineering. – 2007. – Issue 59. – P. 81 – 86. (Ukrainian)

27. Gulyaev V.Y., Bazhenov V.A., Popov S.L. Applied problems of the theory of nonlinear oscillating mechanical systems. - M.: Higher. school, 1989. - 383 p. (Russian)

28. Gorbachev M.N. Finding periodic solutions in some problems of the theory of electric circuits // Electronic modeling, 1999, vol. 21. - No. 3. - pp. 26-35. (Russian)

29. Zakrzhevskyi M.V. Oscillations of essentially non-linear mechanical systems. - Riga: Zinatne, 1980. - 190 p. (Russian)

30. Kuzmin P.A. Small oscillations and stability of motion. - M.: Nauka, 1973. - 208 p. (Russian)

31. Kolev L., Radev N., Stanchev K. Calculation of periodic regimes in nonlinear electric circuits // Elektrychestvo, 1986, No. 6. - pp. 69-71. (Russian)

32. Landa P.S. Self-oscillations in systems with a finite number of degrees of freedom. - M.: Nauka, 1980. - 359 p. (Russian)

33. Lvov E.L. Harmonic balance taking into account an arbitrary number of higher odd harmonics // Elektrychestvo, 1982, No. 2 - p. 32-37. (Russian)

34. Malyar V.S., Malyar A.V. “Mathematical modeling of periodic modes of operation of electrotechnical devices” // Electronic modeling, 2005, vol. 27, no. 3. - pp. 39 - 53. (Russian)

35. Malyar V.S. “Calculation of static characteristics of periodic processes by the differential spline method” // Teoretichna elektrotechnika, 2000. – Vol. 55. – P. 37-42. (Ukrainian)

36. Malyar V.S., Filts R.V.. “A method of calculating the modes of a saturated open-pole synchronous machine with large rotor oscillations” // Izv. universities Energy, 1978, No. 6. - P. 9 - 15. (Russian)

37. Mandelshtam L.I. Lectures on the theory of vibrations. - M: 1972. (Russian)

38. Mathanov P.N. Fundamentals of electric circuit analysis. Nonlinear chains. - M.: Vyssh. school, 1977. 272 p. (Russian)

39. Mygulin V.V., Medvedev V.I., Mustel E.R., Parygin V.N. Fundamentals of the oscillating theory. - M.: Nauka, 1978. - 392 p (Russian)

40. Mitropolskyi Yu. A Nonlinear Mechanics. Single-frequency oscillations. - K.: Institute of Mathematics of the National Academy of Sciences of Ukraine, 1997. - 384 p. (Russian)

41. Neuman L.R., Demirchyan K.S. Theoretical fundamentals of electrical engineering, in 2 volumes - L.: Energoizdat, 1981. Volume 1 - 536 p., Volume 2 - 415 p. (Russian)

42. Neimark Yu.P. The point method is reflected in the theory of non-linear oscillating. - M.: Nauka, 1972. - 471 p. (Russian)

43. Nemnyugin S.A., Stesyk O.L. Modern FORTRAN. - St. Petersburg: BHV-Petersburg, 2004. - 496 p. (Russian)

44. Ortega D., Reinbolt V. Iterative methods of solving nonlinear systems of equations with many unknowns. - M: Mir, 1975. - 558 p. (Russian)

45. Paltov I.P. Non-linear methods of research of automatic systems. - L.: Energy, 1976. - 128 p. (Russian)

46. Panovko J.P. Introduction to the theory of mechanical vibrations: textbook. - M.: Nauka, 1980. - 272 p. (Russian)

47. Paranchuk R.Ya., Synytskyi L.A. “Research of ferroresonance, subharmonic and asymmetric modes in nonlinear circuits” // Bulletin of the Eastern Ukrainian National. University named after V. Dalya. – 2003, No. 6(64). - P. 146 - 152. (Ukrainian)

48. Perkhach V.S. Mathematical problems of electric power engineering. – Lviv: View near Lviv. State University, 1989. - 464 p. (Ukrainian)

49. Petrenko A.I., Humen N.B., Smirnov A.M. “Modeling of periodic processes in non-linear circuits on ECM” // Automation of design in electronics, 1980, vol. 21. – P. 12 – 15. (Russian)

50. Polyakovskyi Yu.V., Sinitskyi L.A. “On the calculation of periodic regimes in non-linear autonomous circuits on ECM” // Theory. electrical engineering, 1971, vol. 13, p. 71-75. (Russian)

51. Popov E.P. Theory of nonlinear systems of automatic regulation and control. - M.: Nauka, 1979. - 256 p. (Russian)

52. Popovych M.G., Kovalchuk O.V. Theory of automatic control. - Kyiv: Lybid, 1997. - 544 p. (Ukrainian)

53. Pukhov G.E. Methods of analysis and synthesis of quasi-analog electronic circuits. - K.: Nauk. dumka, 1967. - 568 p. (Russian)

54. Pukhov G.E. Mathematical modeling of periodic processes by joint application of differential T-transformed and trigonometric series. – In the book: Machine modeling of electric and electronic circuits. - K.: Nauk. Dumka, 1981. - P. 3 - 20. (Russian)

55. Samoilenko A.M., Ronto N.Y. Numerical and analytical methods of research of periodic solutions. - K.: Nauk. dumka, 1976. - 180 p. (Russian)

56. Simonovskyi V. I. Oscillations of nonlinear systems: Study. manual for students / Sumy state Univ. — Sumy: Publishing House of Sumy State University, 1999. — 130 p. (Ukrainian)

57. Sinytskyi L.A., Shumkov Yu.M. “On the search for periodic regimes in nonlinear circuits by numerous methods //” Theoret. electrical engineering, 1970, vol. 9, p. 110-115. (Russian)

58. Sinytskyi L.A., Khvishtun I.A., Shmygelskyi Y.A. “On a reliable algorithm for searching the periodic regime in nonlinear circuits” // Theory. electrical engineering, 1981, vol. 30. - p. 114-125. (Russian)

59. Sokil B.I.. Nonlinear oscillations of mechanical systems and analytical methods of their research: Author's abstract. dis... Dr. Tech. Sciences: 05.02.09 / Lviv Polytechnic National University.—Lviv, 2001. — 36 p. (Ukrainian)

60. Stoker J. Nonlinear oscillations in mechanical and electrical systems. - M.: Mir, 1953. - 256 p. (Russian)

61. Taft V.A. ”On the analysis of established periodic regimes in nonlinear systems with many degrees of freedom” // Avtomatika i Telemehanika, 1959, No. 9. - P. 1163 - 1170. (Russian)

62. Tondl A. Nonlinear oscillations of mechanical systems. - M.: Mir, 1973. (Russian)

63. Feygin M. I. Forced oscillations of systems with discontinuous nonlinearities. — Moscow: Nauka, 1994. — 288 p. (Russian)

64. Filts R.V. Mathematical foundations of the theory of electromechanical converters. - K.: Nauk. dumka, 1979. - 208 p. (Russian)

65. Hayasi T. Nonlinear oscillations in physical systems. - M.: Mir, 1968. – 432 p. (Russian)

66. Hale. J. Oscillations in nonlinear systems. - M.: Mir, 1966. - 230 p. (Russian)

67. Aprill T., Trik T. Analysis of the stationary mode of nonlinear circuits with periodic input signals. – In the book: Automation in design. - M.: Mir, 1972. - p. 148 - 155. (Russian)

68. Yavorskyi Ya.Ya., Glukhivskyi L.Y., Filts R.V. “Mathematical modeling of the magnetic field and periodic processes in a controlled reactor with a rotating magnetic field based on the differential harmonic method” // Electronic modeling, 1983, No. 6. (Russian)

69. Hahn B. Fortran 90 for Scientists and Engineers. - Arnold, 1997. - 368 p.[2]

70. https://fortranwiki.org/fortran/show/Fortran+2003

71. https://fortranwiki.org/fortran/show/Fortran+2008

72. https://fortranwiki.org/fortran/show/Fortran+2018

73. Fundamentals of computer science. Structured programming in Fortran: a study guide / V.A. Kalion, I.Yu. Semenova, M.V. Lavrenyuk. - K: Publishing and Printing Center "Kyiv University", 2015. - 207 p. (Ukrainian)

74. Glukhivskyi L. “Process modeling nonlinear electric circuits and devices using differential harmonic method” // Computational problems of electrical engineering, vol. 3 , N2, 2013. – pp. 20 -23. (English)

75. Glukhivskyi L.Y. Nonlinear oscillations: numerical polyharmonic modeling. - K.: Alfa P&K, 2008. - 204 p. (Ukrainian)

76. Chuyko G.P., Dvornyk O.V., Yaremchuk O.M. Mathematical modeling of systems and processes. Study guide. - Mykolaiv, publishing house of the Black Sea National University, 2015. - 244 p. (Ukrainian)

[2] The list of sources after number 69 was supplemented by the author during the translation.

Content

Preface to the translation of the book …… 3
The foreword .... …….. .. 4
Introduction 5
Chapter 1. Mathematical bases of the differential harmonic method ……..... 14
1.1 Harmonic algebraization of differential equations 14
1.2 Determining the periodic solution 19
1.3 Algorithm for calculating harmonic characteristics 22
1.4 Algorithm for calculating matrix values of differential harmonic parameters 27
1.5 Periodic solutions of nonlinear systems of differential equations 35
1.5.1 The notation form a nonlinear system of differential equations 35
1.5.2 Harmonic algebraization of a vector differential equation 38
1.5.3 Determination of the periodic solution of a nonlinear differential system equations 40
1.5.4 Harmonic characteristics and their differential parameters 45
1.5.5 Algorithms of search of periodic solution 51
1.5.6 Investigation of the stability of nonlinear oscillations 53
1.5.7 If there are more than one periodic solution 56
Chapter 2 Software of the differential harmonic method …… ……..... 59
2.1 Structure of the software of differential harmonic method..... 59
2.2 Program components of DGM-S 61
2.2.1 The first program block …… ……. .61
2.2.1.1 Module ARRAY 61
2.2.1.2 Procedure SNCS 62
2.2.1.3 Procedure VGVS 65
2.2.1.4 Procedure VSVG 65
2.2.1.5 Procedure VSVGP 66
2.2.1.6 Procedure KVGVS 67
2.2.1.7 Procedure KVSVG 67
2.2.2 The second program block 68
2.2.2.1 Procedure OMA 68
2.2.2.2 Procedure OMV 69
2.2.2.3 Procedure OMAB 70
2.2.2.4 Procedure OMVB 71
2.2.2.5 Procedure GRPAR 72
2.2.2.6 Procedure GRMAT 74
2.2.3 The third program unit 75
2.2.3.1 Procedure HARMOSC 75
2.2.3.2 Procedure CALCULU..... 84
2.2.3.3 Procedure IMPROVE 89
2.2.4 The fourth program block..... 92
2.2.4.1 Procedure SYS 92
2.2.4.2 Procedure for linear interpolation 93

Chapter 3. Methodology of numerical modeling of nonlinear oscillations 95
3.1 Structure of the main program 95
3.2 Programming of an instantaneous model of process on one period (semi-period) .. 98
3.2.1 Types of nonlinearities 99
3.2.1.1 Instantaneous model of the controlled valve in one period 102
3.2.1.2 Instantaneous model on the period of nonlinearity of the hysteresis form 106
3.2.2 Auxiliary procedures for the development of the simplification of procedure Model 110
3.2.2.1 Procedure DRAWOUTV 110
3.2.2.2 Procedure DRAWOUTXV 111
3.2.2.3 Procedure DRAWUPV 111
3.2.2.4 Procedure DRAWUPM 112
3.2.2.5 Procedure ADDV 113
3.2.2.6 Procedure ADDM 113
3.2.3 Sample procedure Model 114
3.3 Procedure OUTP 117
Chapter 4. Tests and examples 119
4.1 Tests and examples of calculations of forced oscillations 119
4.1.1 Example (project) B.1 119
4.1.2 Example B.2 128
4.1.3 Example B.3 132
4.1.4 Example B.4 143
4.1.5 Example B.5 151
4.2 Tests and examples of self-oscillation calculations 162
4.2.1 Example A.1 162
4.2.2 Example A.2 168
4.2.3 Example A.3 173
4.2.4 Example A.4 178
4.3 Example of calculation of parametric oscillations 185
Afterword 191
References 192

SCIENTIFIC, TECHNICAL AND EDUCATIONAL EDITION

GLUKHIVSKYI LEV YOSYPOVYCH, *D. Sc. in Engineering,* Professor

ГЛУХІВСЬКИЙ ЛЕВ ЙОСИПОВИЧ, д-р техн.наук, професор

NONLINEAR OSCILLATIONS: NUMERICAL POLYHARMONIC MODELING

НЕЛІНІЙНІ КОЛИВАННЯ: ЧИСЕЛЬНЕ ПОЛІГАРМОНІЧНЕ МОДЕЛЮВАННЯ

Матеріали друкуються в авторській редакції.
Формат: PDF. Об'єм даних 2,6 МБ.

Редакція: ДНУ «Український інститут науково-технічної експертизи та інформації» (УкрІНТЕІ) 03150, м. Київ, вул. Антоновича, 180 Тел. (044) 521-00-10, e-mail: uintei@uintei.kiev.ua Свідоцтво суб'єкта видавничої справи ДК № 5332 від 12.04.2017 р