UDC 62.83

A DIGITAL SETTING DEVICE FOR OPTIMAL CONTROL OF POSITIONAL ELECTRIC DRIVES

O. I. Tolochko, P. I. Rozkaryaka

Donetsk National Technical University

Abstract

This paper is a theoretical study of digital control algorithms for the systems of positional electric drives which would ensure minimum energy losses. In our research we took into account the effect of time discreetness, extrapolation, peculiarities of numerical integration and the necessity of real time calculations.

Keywords: electric drive, optimal control, positioning, heat loss, energy saving, parabolic tachogram, time discreteness, real time system

Introduction

At present, when energy costs are growing, the problem of energy saving by means of electric drives becomes more and more important. In positional electric drives working in recursive short-time mode the increase of motion playback time will help to significantly reduce the heat losses from the dynamic component of armature current for DC drives and from moment-generating component of stator current in the systems of AC drives vector control [1-3]. There are many diagrams describing the changes of position, speed, acceleration and jerk in the process of mechanism motion from one given position to another. The basic diagrams are those optimal by speed and by heat losses in DC motor armatures or in asynchronous motor rotors, if speed constraints, acceleration, motor current and jerk are taken into account.

Review of Previous Studies

A great contribution to the development of electric drive control theory was made by Yu. P. Petrov, K. I. Kozhevnikov, V. P. Chistov, V. A. Svyatoslavskiy and others. Many published works [3-7] are devoted to the rational choice of appropriate laws of positional drive coordinates variation; these publications also deal with the algorithms for developing proper control actions.

Implementation of position adjustment systems (optimal by heat losses) is based on the use of digital setting devices which help to implement the most complicated control laws taking into account various constraints. In these works the optimal control algorithms are provided in continuous form.

The goal of this paper is to develop a reliable technique of setting device (SD) software implementation which would ensure the optimal (by heat losses) control of positional electric drives. When considering this problem we took into account acceleration and speed constraints, the permanent moment of static resistance at the shaft of the actuating mechanism, time quantization effects, signal extrapolation and digital integration.

Results and Discussion

Let us consider the amount of heat losses conditioned by dynamic armature current in case when magnetic flux is constant. In these paper ε, ω, Δφ stand for acceleration, speed and motion respectively: U, $E_{Д}$, I_a stand for voltage, EMF and motor armature current; R_a is armature circuit resistance; M_j, I_j, M_c, I_c are dynamic and static components of the moment and armature current; J is the drive inertia moment reduced to the motor shaft; $\omega_{м}$ is maximum speed, $\varepsilon_{\max\text{ p}}$ is maximum speed-up acceleration; $\varepsilon_{\max\text{ t}}$ is maximum braking acceleration; I_0 is the value of current limitation, t_0 is the time of motion playback.

Electric energy consumed from power supply network

$$A = \int_0^t U(t) I_{\text{a}}(t) dt = A_{\text{M}} + A_{\text{T}} \qquad (1)$$

is spent on mechanical work

$$A_{\text{М}} = \int_0^t E_{\text{Д}}(t) I_{\text{a}}(t) dt = \int_0^t \omega(t) M(t) dt \tag{2}$$

and heat loses in armature circuit (the losses in exciting winding are ignored)

$$A_{\text{Т}} = \int_0^t I_{\text{a}}^2(t) R_{\text{a}} dt \tag{3}$$

Let us decompose the electromagnetic moment of the motor and armature current (proportional to electromagnetic moment if magnetic flux is constant) into static and dynamic components:

$$M(t) = M_j(t) + M_{\text{c}} = J\varepsilon(t) + M_{\text{c}} = cI_j(t) + cI_{\text{c}} \tag{4}$$

Then

$$A_{\text{М}} = \int_0^t \omega(t) M_j(t) dt + M_{\text{c}} \int_0^t \omega(t) dt \tag{5}$$

$$A_{\text{Т}} = \int_0^t I_{\text{a}}^2(t) R_{\text{a}} dt = \int_0^t (I_j(t) + I_{\text{c}})^2 R_{\text{a}} dt = R_{\text{a}} \left(I_{\text{c}}^2 t + \int_0^t I_j^2(t) dt + 2I_{\text{c}} \int_0^t I_j(t) dt \right) \tag{6}$$

Taking into account that $\int_0^t I_j(t) dt = \omega(t_0) = 0$, $\int_0^t \omega(t) dt = \Delta\varphi$ and $\int_0^t \omega(t) M_j(t) dt = 0$, we obtain finally:

$$A_{\text{М}} = M_{\text{c}} \Delta\varphi \tag{7}$$

$$A_{\text{Т}} = R_a \left(I_c^2 t + \int_0^t I_j^2(t) dt \right) = A_{\text{cТ}} + A_{j\text{Т}} = R_a I_c^2 t_0 + R_a \int_0^t I_j^2(t) dt \tag{8}$$

Then the electrical energy heat loss from the effect of armature current dynamic component is

$$A_{jT} = R_a \int_0^t I_j^2(t) dt \tag{9}$$

In [1] it is proved that the loss A_{jT} is minimal when speed diagram is parabolic and acceleration diagram is triangular. In this case speed and current can exceed their permissible values. In the process of creating diagrams acceleration constraint (dynamic moment) should be taken into account. However, if static moment is present the constraint of dynamic moment can change when the system reaches current limitation level. That is why dynamic moment constraint leads to acceleration limitation at level ε_1 in case of speed up and at level ε_2 in case of braking. Thus the acceleration diagram and the tachogram become asymmetric.

Formulas for Calculating Optimal Diagrams in the Continuous Form

Possible types of optimal diagrams for the given problem are provided in Figure 1. Three signals *ε(t), ω(t)* and *φ(t)* are interconnected through integro-differential dependences, acceleration *ε(t)* being the simplest one. So when a control action is being implemented in digital form it will be enough to generate this particular signal and the rest coordinates *ω(t)* and *φ(t)* can be obtained by means of successive numerical integration of acceleration task code.

$$\omega(t) = \int \varepsilon(t)dt \; ; \; \Delta\varphi(t) = \int \omega(t)dt \qquad (10)$$

From the above mentioned it follows that node points coordinates of the graphs *ε(t)* are sufficient for creating the optimal diagrams. However, if we need to select the required diagram we should know the formula for calculating the maximum value of speed $\omega_м$ without taking into account the constraint. The methods of defining the coordinates of the characteristic points of diagrams (Figure 1) are discussed in [5, 6]. Besides, these works provide the formulas for calculating the above parameters (see Table 1).

It's worth mentioning that incomplete equation of the forth degree in the variable ε_2, obtained for diagram 1*d*, has no analytical solution. That's why its roots should be found by means of a numerical method.

Formation of Discrete Standard Diagrams

The drawback of the formulas from Table 1 lies in the fact that in the process of the setting device digital implementation the x-values of the curve *ε(t)* breakpoints are not multiple of the sampling period *T* and therefore should be rounded. It leads to significant deviations of the formed control actions from the desired ones. In this case, the steady value of motion task changes, whereas the accuracy of positioning should be as high as possible. That is why the given diagrams should be corrected so that the change of abscissas $t_i=k_iT$ (t_i are integers) could not distort the value φ_3.

The suggested technique of acceleration diagram node points coordinates correction comprises the following stages:

1) The total desired time of the motion playback t_0 is rounded to the nearest larger value multiple of the sampling period *T*:

$$t_{0new} = T \cdot ceil\left(\frac{t_{0old}}{T}\right) \qquad (11)$$

where t_{0old} is the time before rounding and t_{0new} is the time after rounding;

2) The times $t_1 \ldots t_5$ are calculated according to the formulas presented in Table 1 and aimed at SD implementation in continuous form;

3) The times t_1, t_3 and t_5 are rounded to the nearest larger values multiple of the sampling period *T* as in (11), and the time t_3 is rounded according to the common rules (that is to the nearest integer):

$$t_{2new} = T \cdot round\left(\frac{t_{2old}}{T}\right); \qquad (12)$$

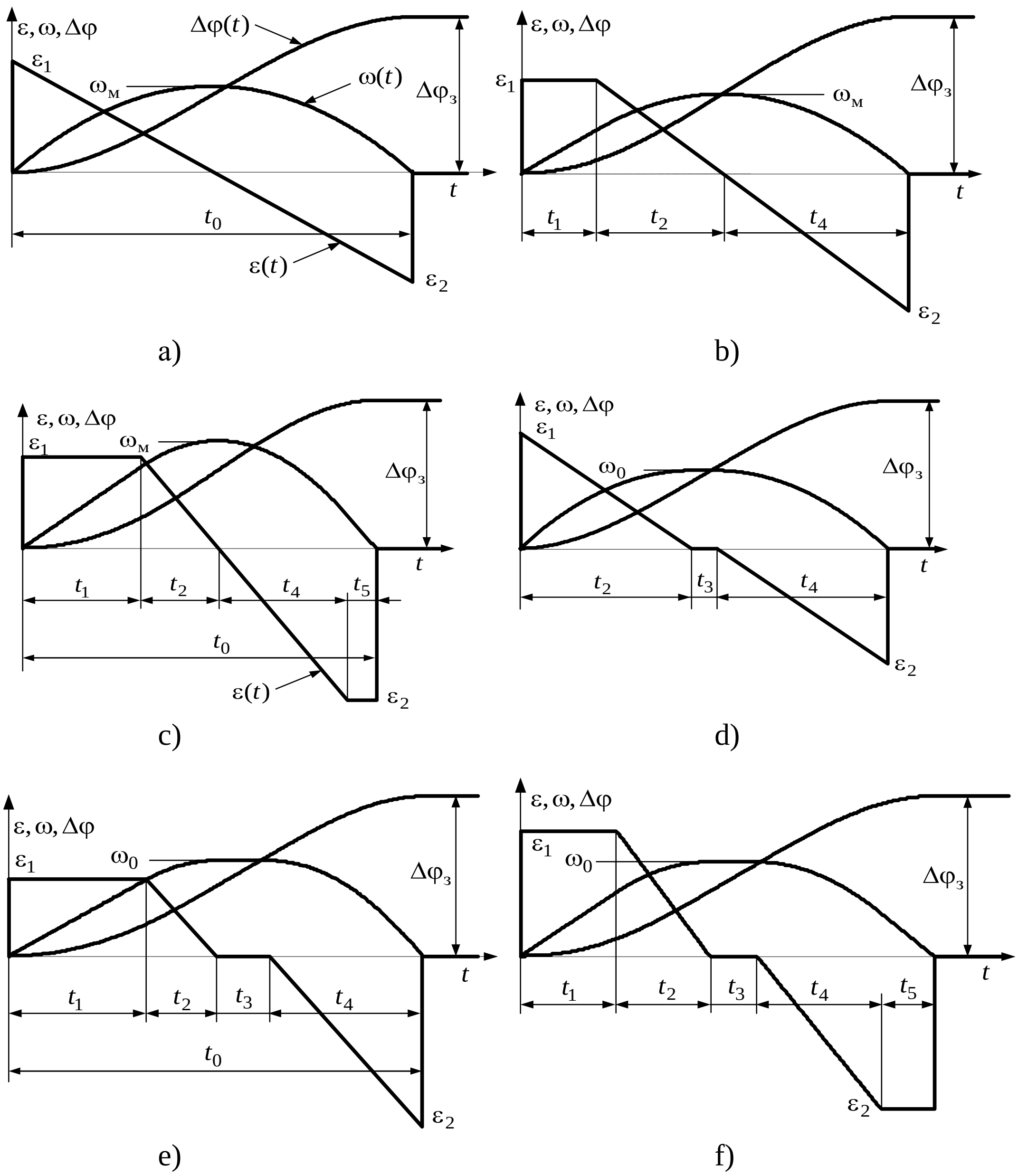
4) Then we find the values of the rest of times $t_2 + t_4$ or t_2, without changing the total time t_0:

$$t_2 + t_4 = t_0 - t_1 - t_5$$

$$t_4 = t_0 - t_1 - t_2 - t_3 - t_5\,; \qquad (13)$$

5) At this stage we recalculate the values of accelerations ε_1 and ε_2 by solving the equation set (ES):

$$\begin{cases} \omega(\varepsilon_1, \varepsilon_2, t_1, \ldots, t_5)\big|_{t=t_0} = 0, \\ \Delta\varphi(\varepsilon_1, \varepsilon_2, t_1, \ldots, t_5)\big|_{t=t_0} = \Delta\varphi_3. \end{cases} \qquad (14)$$



a) b)

c) d)

e) f)

Figure 1. Motion diagrams optimal by heat losses

It should be mentioned that the suggested technique does not presuppose re-examination of the corrected parameters aimed at finding out whether these parameters have reached limiting levels and whether the diagram has changed if compared to the original one. Such a procedure can break the equality of jerks in the intervals t_2 and t_4. We should also take into account that the corrected values $\varepsilon_{1,}$ ε_2 and ω_0 can exceed their limiting levels. If it is not acceptable each time interval should be rounded up, and that will lead to the increase of time t_0.

The suggested technique of deriving expressions for the left sides of ES (14) is the following:

- We put down the equations for the graph $\varepsilon(t)$ in each interval t_1, t_2, …, t_5, combining the initial reference point of time with the beginning of the respective interval;
- Then we integrate twice the right sides of the obtained equations taking into account that

$$\omega(t) = \int \varepsilon(t)\,dt\,, \qquad \Delta\varphi(t) = \int \omega(t)dt \tag{15}$$

Under initial conditions (IC)

$$\omega_{0i} = \omega(t_{i-1}), \qquad \Delta\varphi_{0i} = \Delta\varphi(t_{i-1}), \qquad \omega_{01} = 0, \qquad \Delta\varphi_{01} = 0 \tag{16}$$

we find successively the expressions for the graphs of speed and motion variation in different time intervals of the considered diagram as well as the values of these signals at interval extremities (i. e. in break points) substituting them in the expressions obtained from (15) taking into account IC (16), $t = t_i$;

- From the expressions obtained for the values of the considered signals at the end of the last interval the left sides of ES (14) are formed.

The suggested technique helps to derive the formulas for recalculating maximum speed up and braking accelerations. The results for all considered diagrams are given in Table 2.

The Symmetry of Acceleration Task Extrapolated Signal

In case of digital implementation of control algorithms optimal by heat losses (parabolic character of speed variation being their distinctive feature) the above mentioned initial continuous algorithm correction is obviously not enough. We illustrate it by comparing the control actions (Figure 2) formed by means of SD with two considered types of tachograms (triangular and parabolic) using numerical integration (NI) methods with automatic step selection (ASS) and NI methods with constant step equal to the sampling period ($h_И = T$). In both cases speed and acceleration do not reach their limiting levels and maximum values of acceleration are obtained from the condition of a given motion playback in a given time, i. e. termination control problem is solved:

$$\varepsilon_{m\text{т}} = 4\Delta\varphi_{з}\big/t_{\text{о}}^2, \quad \varepsilon_m = 6\Delta\varphi_{з}\big/t_{\text{о}}^2 .$$

From the given figure it is evident that the acceleration task diagrams have pieces with infinite gain factor, and the accuracy of their representation decreases significantly if NI methods with constant step are used. In this or that way it influences the operation of SDs under comparison. In case of a triangle tachogram it will lead to the one-NI-step increase of signal formation time only. But if the diagram is parabolic the symmetry of the graph $\varepsilon_{зп}(t)$ (with respect to the time axis) will be broken. As a result at the moment of the equality of the acceleration task to zero the speed task will be negative and the motion task $\Delta\varphi_{зп}(t)$ will decrease (with constant rate) deviating more and more from the required value $\Delta\varphi_{з}$. Even the forced zeroing of the signal $\omega_{зп}(t)$ at the moment $t_{\text{о}}$ or $t_{\text{о}} + h_И$ will not solve the given problem.

The same takes place if the analog integrators are substituted for the discrete ones with zero order extrapolation (see Figure 3a). To eliminate this drawback we should correct the initial continuous diagram of acceleration task change so that by means of extrapolation we could obtain from it the symmetric discrete diagram $\varepsilon_{зпэ}(t)$.

Such a result can be obtained by means of defining the difference between the maximum and the minimum values of accelerations of the discrete diagram, dividing this difference by half and shifting the continuous diagram up this value:

$$\Delta\varepsilon_m = |\varepsilon_{\max}| - |\varepsilon_{\min}| = 2\varepsilon_m \frac{T}{t_{\text{о}}} \tag{17}$$

$$\Delta\varepsilon_{д} = \frac{\Delta\varepsilon_m}{2} = \varepsilon_m \frac{T}{t_{\text{о}}} \tag{18}$$

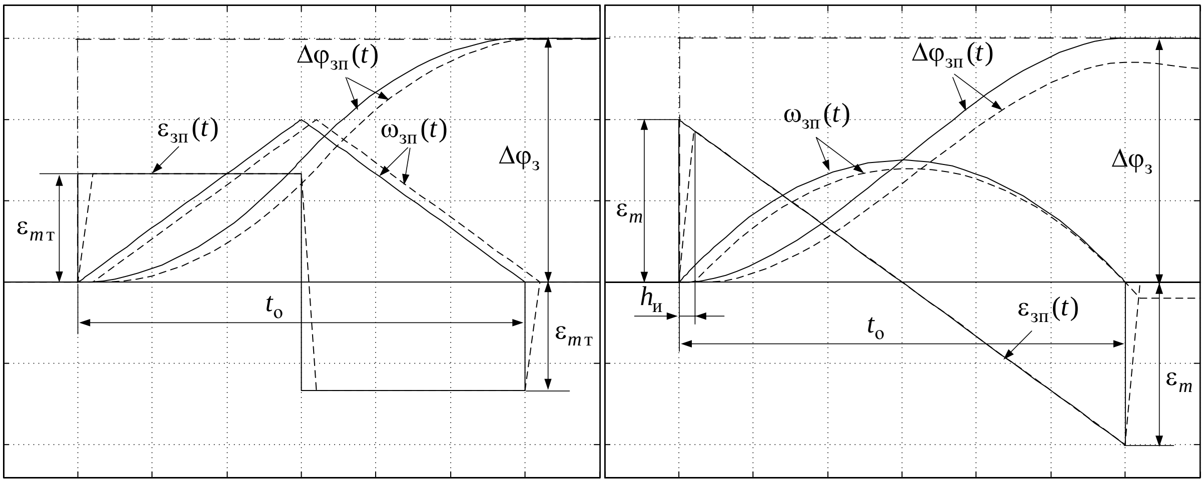
$$\varepsilon_{\max\text{к}} = \varepsilon_m + \Delta\varepsilon_{д}, \qquad \varepsilon_{\min\text{к}} = -\varepsilon_m + \Delta\varepsilon_{д} \tag{19}$$

Table 1. Formulas for calculating the parameters of continuous diagrams from Figure 1.

Figure	$\varepsilon_{\max\text{р}}$	$\varepsilon_{\max\text{т}}$	$\omega_{\text{м}}$	t
1a	$\varepsilon_1=\frac{6\Delta\varphi_3}{t_0^2}$	$\varepsilon_2=\frac{6\Delta\varphi_3}{t_0^2}$	$\omega_{\text{м}}=\frac{3}{2}\frac{\Delta\varphi_{\text{з}}}{t_o}$	t_0
1b	ε_1	$\varepsilon_2=\frac{6\varepsilon_1\Delta\varphi_{\text{з}}+t_0^2\varepsilon_1^2}{3\varepsilon_1 t_0^2-6\Delta\varphi_{\text{з}}}$	$\omega_{\text{м}}=\varepsilon_1 t_1+\frac{\varepsilon_1^2(t_0-t_1)}{2(\varepsilon_2+\varepsilon_1)}$	$t_1=t_0\frac{\varepsilon_2-\varepsilon_1}{\varepsilon_2+\varepsilon_1}$; $t_2=(t_0-t_1)\frac{\varepsilon_1}{\varepsilon_2+\varepsilon_1}$; $t_4=t_0-t_1-t_2$
1c	ε_1	ε_2	$\omega_{\text{м}}=\varepsilon_1 t_1+\frac{\varepsilon_1 t_2}{2}$	$t_1=\frac{t_0\varepsilon_2-\sqrt{3\varepsilon_1\varepsilon_2 t_0^2-6\Delta\varphi_{\text{з}}(\varepsilon_1+\varepsilon_2)}}{\varepsilon_1+\varepsilon_2}$; $t_4=t_2\frac{\varepsilon_2}{\varepsilon_1}$; $t_2=\frac{\varepsilon_2 t_0-t_1(\varepsilon_1+\varepsilon_2)}{\varepsilon_2+\varepsilon_1/2+\varepsilon_2^2/2\varepsilon_1}$; $t_5=t_0-t_1-t_2-t_4$
1d	$\varepsilon_1=\frac{2\omega_0}{t_2}$	$\varepsilon_2=\frac{2\omega_0}{t_2}$	ω_0	$t_2=t_4=1{,}5\left(t_0-\frac{\Delta\varphi_{\text{з}}}{\omega_0}\right)$; $t_3=3\frac{\varphi_{\text{з}}}{\omega_0}-2t_0$
1e	ε_1	$\varepsilon_2^4\left(-3\omega_0^2+6\omega_0 t_0\varepsilon_1-6\Delta\varphi_{\text{з}}\varepsilon_1\right)-$ $-\varepsilon_2^3\left(4\omega_0^2\varepsilon_1\right)-\varepsilon_1^4\omega_0^2=0$	ω_0	$t_4=\frac{2\omega_0}{\varepsilon_2}$; $t_2=t_4\frac{\varepsilon_1}{\varepsilon_2}$; $t_1=\frac{\omega_0}{\varepsilon_1}-\frac{t_2}{2}$; $t_3=t_0-t_1-t_2-t_4$
1f	ε_1	ε_2	ω_0	$t_4=\frac{2}{\varepsilon_1^3+\varepsilon_2^3}\sqrt{\frac{3\varepsilon_2(\varepsilon_1^3+\varepsilon_2^3)g}{j_1}}$; $g=2\varepsilon_1\varepsilon_2(\omega_0 t_0-\Delta\varphi_{\text{з}})-\omega_0^2(\varepsilon_1+\varepsilon_2)$; $t_5=\frac{\omega_0}{\varepsilon_2}-\frac{t_4}{2}$; $t_2=\frac{\varepsilon_1}{\varepsilon_2}t_4$; $t_1=\frac{\omega_0}{\varepsilon_1}-\frac{t_5}{2}$; $t_3=t_0-t_1-t_2-t_4-t_5$.

Table 2. Formulas for recalculation of maximum acceleration at speed up ε_1 and at braking ε_2

Diagram	Values of acceleration
a	$\varepsilon_1 = \varepsilon_2 = 6\Delta\varphi_з / t_о^2$
b	$\varepsilon_1 = \dfrac{6\Delta\varphi_з}{3t_1^2 + 4t_1(t_2 + t_4) + (t_2 + t_4)^2}$; $\varepsilon_2 = \varepsilon_1 \dfrac{2t_1 + (t_2 + t_4)}{(t_2 + t_4)}$;
c	$\varepsilon_1 = \dfrac{6\Delta\varphi_з}{3t_1^2 + 2(t_2 + t_4)^2 + 6t_1(t_5 + (t_2 + t_4)) + 3(t_2 + t_4)t_5 - ((t_2 + t_4)^2 + 3t_5^2 + 3(t_2 + t_4)t_5)\dfrac{2t_1 + (t_2 + t_4)}{2t_5 + (t_2 + t_4)}}$; $\varepsilon_2 = \varepsilon_1 \dfrac{2t_1 + (t_2 + t_4)}{2t_5 + (t_2 + t_4)}$;
d	$\varepsilon_1 = \dfrac{6\Delta\varphi_з}{2t_2^2 + 3t_2t_3 + 2t_2t_4}$; $\varepsilon_2 = \varepsilon_1 \dfrac{t_2}{t_4}$;
f	$\varepsilon_1 = \dfrac{6\Delta\varphi_з}{3t_1^2 + 2t_2^2 + 6t_1t_2 + 6t_1t_3 + 4t_1t_4 + 3t_2t_3 + 2t_2t_4}$; $\varepsilon_2 = \varepsilon_1 \dfrac{2t_1 + t_2}{t_4}$;
j	$\varepsilon_1 = \dfrac{6\Delta\varphi_з}{3t_1^2 + 2t_2^2 + 6t_1(t_2 + t_3 + t_4 + t_5) + 3t_2(t_3 + t_4 + t_5) - (t_4^2 + 3t_5^2 + 3t_4t_5)\dfrac{2t_1 + t_2}{2t_5 + t_4}}$; $\varepsilon_2 = \varepsilon_1 \dfrac{2t_1 + t_2}{t_4 + 2t_5}$.



a) b)

Figure 2. Diagram of motion with the triangle and parabolic tachograms formed by the analog model: a) – NI with ASS; b) – NI with constant step (Euler method)

The analysis of the original graphs and those corrected by means of the suggested formulas leads to the conclusion that the same result can be achieved by shifting the signal $\varepsilon_{зп}(t)$ to the right by half of the sampling period, as it is shown in Figure 3. (See graph $\varepsilon_{зпк}(t)$).

Besides, such a method of correction is also true for quasi-optimal diagrams that have intervals of motion with constant values of speed and (or) acceleration which appear if these coordinates constraints are taken into account.

From the last figure it is evident that the suggested method of acceleration task signal correction allows eliminating the discrepancy in the times when transient process are coming to an end and creating the basic control action without any error and in stable regime.

It is worth mentioning that such correction technique can also be applied for the diagrams optimal by heat losses obtained under the condition that the constant static moment is present at the motor shaft [8].

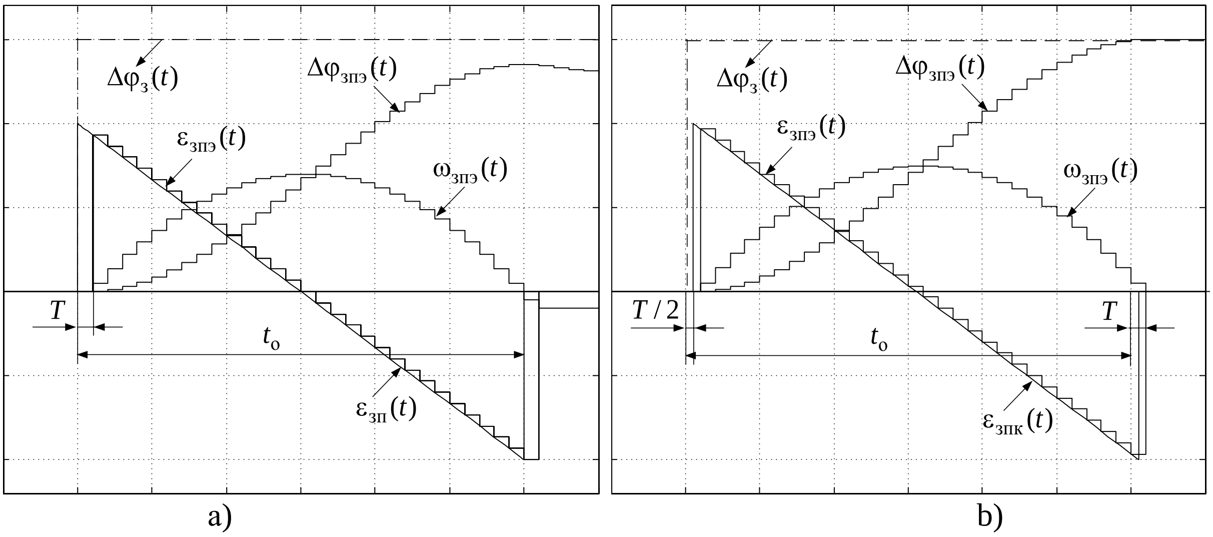


Figure 3. An optimal by heat loss diagram formed by a discrete model: a) – without correction, b) – with the correction of the diagram $\varepsilon_{\text{зп}}(t)$

Numerical Integration Method Selection

The choice of operation algorithms for digital integrators (DI) is very important in the practical implementation of digital setting devices. It predetermines the degree of discrete diagrams approximation to the respective analog diagrams taken as standard.

The simplest NI methods are right-side rectangles method (*Forward Euler*), left-side rectangles method (*Backward Euler*) and the method of trapezoids (*Trapezoidal*) which have respective DIs of the first order with transfer functions (TF).

$$W_{FE}(z) = \frac{T}{z-1}, \tag{20}$$

$$W_{BE}(z) = \frac{Tz}{z-1}, \tag{21}$$

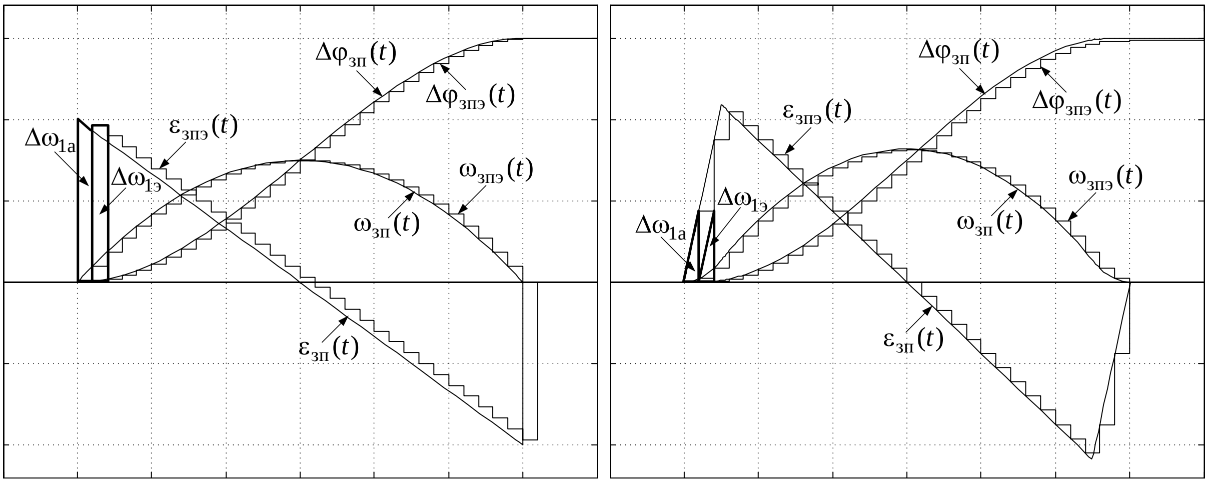
$$W_{T}(z) = \frac{T(z+1)}{2(z-1)}. \tag{22}$$

These are the methods that can be installed in the unit *Discrete-Time Integrator* of the program *Simulink* in the package *MATLAB*, which (due to the presence of *Real Time Workshop*) can be regarded as one of the possible versions of digital control devices software implementation [9].

Usually in the intervals with constant value of the integrated signal we use *Forward Euler*, whereas in the intervals where the signal changes according to the linear law we should apply the method of trapezoids, and that is quite natural. Following such a logic, in SD with triangular tachogram we must use the integrator with TF (20) (as the first integrator which forms acceleration task); and in SD with parabolic tachogram we should apply the integrator with TF (22). Nevertheless the modeling results have shown that integrator (21) is the best for both cases, because it ensures the coincidence of the discrete extrapolated tachogram $\omega_{\text{зпэ}}(t)$ with the standard analog tachogram $\omega_{\text{зп}}(t)$ at the moments of time multiple of the interrupt period. It is confirmed by the graphs provided in Figure 4a. This effect can be explained by the equality of the respective areas of the trapezoid and the rectangle added in case of analog and discrete integration. The first of them ($\Delta\omega_{1a}$ and $\Delta\omega_{1э}$) are marked with thick lines in Figure 4a. The analysis of the provided diagrams shows that with the use of the above-mentioned acceleration diagram correction method the altitude of the marked rectangle is equal to the half-sum of the bases of the marked trapezoid. This property is typical of all the subsequent pairs of the considered geometrical figures.

It is quite different with the diagrams, which take into account the jerk constraint (the examples are in Figure 4b). In this case, in order to achieve the coincidence of the extrapolated discrete tachogram with the standard one we need to carry out NI of the acceleration task signal by means of the trapezoid method, i. e. to use the DI with transfer function as the first integrator (22).

The second DI of the position setter integrates the speed task. For SDs with triangular and trapezoidal tachograms it is better to use the integrator, which employs the trapezoid method. If the integrators employing the rectangles method (Euler method) are used instead, it will reduce the dynamic accuracy of acceleration task signal reproduction without changes in its steady-state value.



a) b)

Figure 4. Diagrams of motion with parabolic tachograms

a) – without constraints, b) – with jerk constraint during signal $\varepsilon_{зпэ}(t)$ integration by left-side rectangles method and signal $\omega_{зпэ}(t)$ integration by trapezoid method

If SDs with the parabolic tachogramm are used, the integrators will form the acceleration task with the same steady-state error. It follows from the well known NI formulas if condition (23) is satisfied

$$\omega_{зп}(0) = \omega_{зп}(t_о) = 0\,. \tag{23}$$

Let us find an analytical expression for this error. Let the interval $t_о$ contain an even number of sampling periods:

$$t_о = 2kT\,, \qquad k = 1, 2, \ldots\,. \tag{24}$$

The analog signal of the acceleration task changes according to:

$$\omega_{зп}(t) = \int\left(\varepsilon_m - 2\varepsilon_m \frac{t}{t_о}\right)dt = \varepsilon_m\left(t - \frac{t^2}{t_о}\right). \tag{25}$$

The exact value of the motion developed according to the tachogram specified by (25) is defined from:

$$\Delta\varphi_{\text{з}} = \int_0^{t_{\text{o}}} \omega_{\text{зп}}(t)\,dt = \frac{\varepsilon_m t_{\text{o}}^2}{6}\,, \tag{26}$$

and the approximate value calculated with rectangles method and trapezoid method with step T is defined from the formula

$$\Delta\varphi_{\text{уст}} = T\sum_{i=1}^{2k-1}\omega_{\text{зп}}(iT) = \varepsilon_m T^2\sum_{i=1}^{2k-1} i - \frac{\varepsilon_m T^3}{t_{\text{o}}}\sum_{i=1}^{2k-1} i^2 = \varepsilon_m T^2 k(2k-1) - \frac{\varepsilon_m T^3}{t_{\text{o}}}\cdot\frac{(2k-1)k(4k-1)}{3}\,.$$

After substitution $k = \frac{t_{\text{o}}}{2T}$ and transformations we have finally

$$\Delta\varphi_{\text{уст}} = \frac{\varepsilon_m (t_{\text{o}}^2 - T^2)}{6}\,. \tag{27}$$

Subtracting (27) from (26) we come to:

$$\varepsilon_\varphi = \Delta\varphi_{\text{з}} - \Delta\varphi_{\text{уст}} = \frac{\varepsilon_m T^2}{6}\,. \tag{28}$$

If the sampling period is small the error (28) will be insignificant and can be compensated, if necessary.

The diagrams from Figure 4 are obtained by using (as the second DI) the integrator which involves the method of trapezoids. They show that the discrete signals of speed task as well as the discrete signals of position task coincide with standard curves at the moments of time multiple of the sampling period. The values of steady-state errors obtained in the process of modeling agree with the values obtained from formula (28). Meanwhile, the accurate integration of a parabolic signal (speed task signal is parabolic) can be achieved by means NI algorithms of a higher order, for example, Simpson algorithm [11].

Equation Numerical Solution in Real Time

Let us consider the possibilities of real time implementation of the given diagrams in the software environment *MatLab → Simulink → Real Time WorkShop.*

The formulas from Table 1 can be easily represented as the units of *Simulink*. But for calculating the maximum value of acceleration ε_2 for the diagram (Figure 1 e) we need to solve an incomplete algebraic equation of the foreth degree with real coefficients

$$\varepsilon_2^4\left(6\omega_0 t_0\varepsilon_1 - 3\omega_0^2 - 6\varphi_{\text{з}}\varepsilon_1\right) - \varepsilon_2^3\left(4\omega_0^2\varepsilon_1\right) - \varepsilon_1^4\omega_0^2 = 0\,. \tag{29}$$

From the physical sense of the given problem it follows that we need to obtain the real root belonging to the interval $[0,\ (I_0 + I_c)c/J]$.

The simplest methods of solving algebraic and transcendental equations are iteration half-division method, chord method and tangent method [10]. The analysis of the polynomial function $f(\varepsilon_2)$ in the right side of equation (29) shows that such a root is always present in the domain of real parameters and is the only one. Moreover, the graph of this function has a great slope at the extremity $[0,\ (I_0 + I_c)c/J]$ and satisfies the condition $f(\varepsilon_2)\cdot f''(\varepsilon_2) > 0$. So it is reasonable to solve the equation (20) applying the tangent method with the initial approximation of the root equal to $(I_0 + I_c)c/J$.

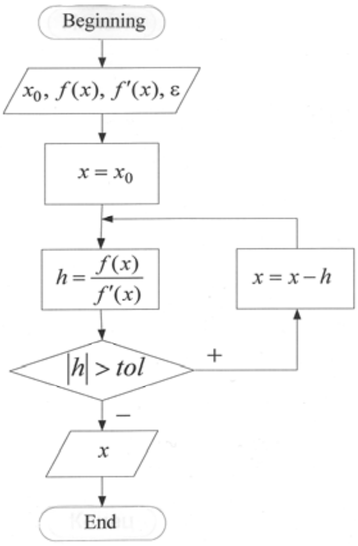
In the library *Math* of the application *Simulink* there is the unit *Algebraic Constraint* that is used to find the roots of the equation $f(x) = 0$ by means of creating an algebraic loop, but such a technique is impossible in *Real Time WorkShop*.

Figure 6 presents the algorithm of updating the given initial approximation x_0 of the root of the equation $f(x)=0$ by means of the tangent method (also called Newton method). Figure 7 provides a *Simulink*-model of this iteration algorithm. The model involves *Enabled Subsystem* for loop organization. *Enabled Subsystem* (*j2_kasat*) is working only if the inequality $|h|>tol$ is true (tol is the equation solution accuracy assigned by the variable *e* in the unit *Const)*. In the unit *Memory* (which implements one step delay of numerical integration) the *Initial Condition* is assumed to be equal to 1 as the iteration process should start without the delay of the first step. A comprehensive model of this subsystem is given in Figure 8. In the subsystem *j2_kasat* the initial approximation of the root $x_0 = (I_0 + I_c)c/J$ is formed and the iteration body is implemented.

The values of the function, which makes up the left side of equation (20)
$f(x) = x^4\left(6\omega_0 t_0 \varepsilon_1 - 3\omega_0^2 - 6\varphi_3 \varepsilon_1\right) - x^3\left(4\omega_0^2 \varepsilon_1\right) - \varepsilon_1^4 \omega_0^2$ and its derivative
$f'(x) = 4x^3\left(6\omega_0 t_0 \varepsilon_1 - 3\omega_0^2 - 6\varphi_3 \varepsilon_1\right) - 3x^2\left(4\omega_0^2 \varepsilon_1\right)$
are calculated in subsystems $f(x)$ and $df(x)$.

If the difference between two successive approximations $h = \dfrac{f(x)}{df(x)}$ does not ensure the accuracy tol, the next iteration is performed. When the given accuracy is achieved the iteration process stops. The parameter *Threshold* of the unit *Switch* is equal to 0.5. The vector of the signals (which are necessary for calculating the coefficients of polynomial functions $f(x)$ and $f'(x)$) comes to the input *In*1 (*data*)

Control system examination has shown that three iterations are required to solve equation (29) with the accuracy
$tol = 0.001$ by means of the tangent method



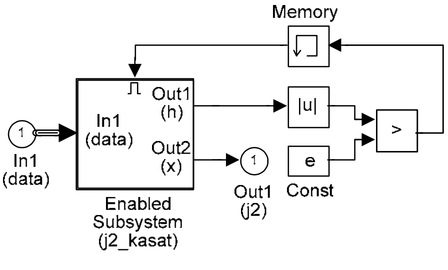


Figure 6. The algorithm for specifying the real root of an algebraic equation by tangent method

Figure 7. A *Simulink*-model of solving the equation (29) in real time

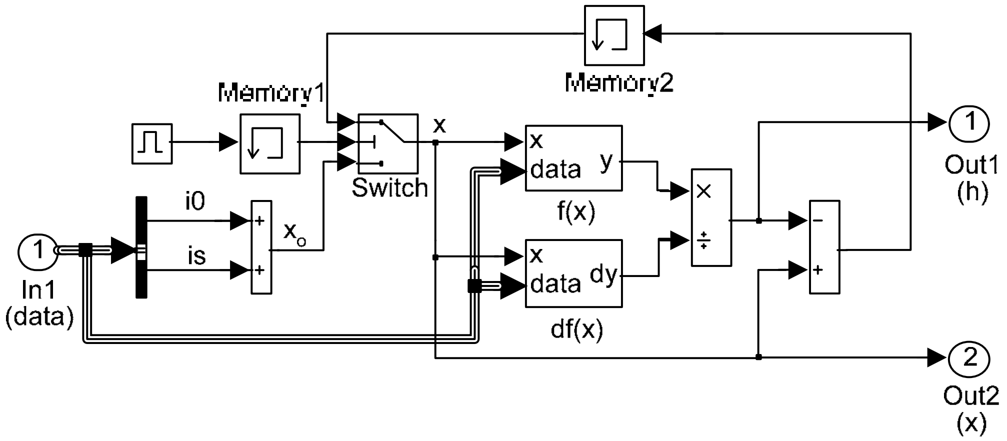


Figure 8. A comprehensive model of *Enabled Subsystem* (*j2_kasat*) in case of equation (20) solution by tangent method

The suggested models have been successfully applied in the control of positional electric drives of the experimental facility in real time operational system QNX.

So the technique of creating optimal diagrams for real time implementation comprises the following principal stages:

1) We should calculate the node points of the acceleration task diagram proceeding from the value of motion $\varphi_з$ and the desired time of motion t_0 taking into account speed constraint ω_0 and acceleration constraint ε_0 according to the accepted optimization criterion (discreteness in time is ignored);

2) Then we must correct the abscissas of the node points of the diagram $\varepsilon_{зп}(t)$ in order to make them multiple of the sampling period T. This procedure leads to the correction of ordinates, otherwise the value of motion and the diagrams will be distorted.

3) The obtained signal $\varepsilon_{зп}(t)$ is first shifted to the right by half of the sampling period, and then it is sampled and extrapolated:

$$\varepsilon_{зпэ}(nT) = \varepsilon_{зп}(t - T/2)\big|_{t=nT};$$

4) If there is no jerk constraint we use *Backward Euler* (21) for the first digital integrator (DI) which forms the discrete extrapolated signal of acceleration task $\omega_{зпэ}(nT)$, and for the second integrator which forms the signal $\Delta\varphi_{зпэ}(nT)$ we use Simpson method.

Conclusion

1. Setting devices that form motion diagrams (optimal or quasi-optimal by heat losses) can be successfully implemented in digital form only. It requires the correction of the algorithms of the digital implementation of motion diagrams in real time (quantization and extrapolation effects should be taken into account).

2. At the first stage of correcting the optimal diagrams of positional electric drive control the effect of time quantization should be taken into account, because the time of the breaks of standard signal graphs should be multiple of the sampling period. The formulas from Table 2 allow recalculating the maximum accelerations in case of speed up and in case of braking after the time intervals had been corrected.

3. The use of NI methods with constant step reduces the accuracy of representing the pieces of diagrams with infinite gain factor. In case of triangular tachogram implementation it leads to one NI step increase in the time of signal formation, and in case of parabolic tachogram implementation it leads to the output signal error.

4. The parabolic tachogram can be successfully implemented only if the initial continuous acceleration task diagram had been corrected (shifted to the right by half-sampling period).

5. The agreement of the standard analog diagram and the discrete extrapolated one (at the moments of time multiple of the interrupt period) is achieved through acceleration task signal integration by left-side rectangles methods if jerk constraint is absent and by trapezoid method if this constraint exists.

6. In the process of integrating the parabolic signal of acceleration task by Euler method or by trapezoid method the signal of position task is formed with steady-state error calculated by formula (28). If sampling period is small the error will be insignificant.

7. The numerical solution to simple transcedental and algebraic equations can be successfully performed in real time in the environment *MatLab* → *Simulink* → *Real Time WorkShop*.

References:

1. Кожевников К.И. Динамика тока двигателей вспомогательных механизмов прокатных станов / К.И. Кожевников // Электричество. – 1956. – №6. – С. 15-21.
2. Петров Ю.П. Оптимальное управление электроприводом / Ю.П. Петров. – М-Л.: Госэнергоиздат, 1961. – 187 с.
3. Чистов В.П. Оптимальное управление электрическими приводами / В.П. Чистов, В.И. Бондаренко, В.А. Святославский. – М.: Энергия, 1968. – 231 с.
4. Толочко О.И. Формирование оптимальных по нагреву диаграмм управления позиционным приводом с учетом ограничений на скорость и ускорение / О.И. Толочко, П.И. Розкаряка // Збірник наукових праць ДонДТУ. Серія: «Електротехніка і енергетика». – 2002. – Вип. – 41 – С. 159-163.
5. Костенко В.И. Формирование оптимальных по нагреву диаграмм отработки заданных перемещений при наличии постоянного статического момента на валу двигателя / В.И. Костенко, П.Х. Коцегуб, П.И. Розкаряка, О.И. Толочко // Вісник Національного Технічного Університету. – 2002. – Т.2. — №12. — С. 350-354.
6. Карнюшин Л.В. Области существования оптимального управления электроприводами / Л.В. Карнюшин, В.Д. Пышкало, А.И. Рогачев // Электромашиностроение и электрооборудование. – 1972. – Вып. 15. – С. 3-8.
7. Чермалых А.В. Универсальная многоканальная задающая модель для систем управления позиционным электроприводом / А.В. Чермалых // Вісник Кременчуцького державного політехнічного університету. – 2004. – №2 (25). – С. 139-143.
8. Толочко О.И. Особенности цифровой реализации оптимальных алгоритмов управления позиционным электроприводом / О.И. Толочко, П.Х. Коцегуб, П.И. Розкаряка // Вісник Кременчуцького державного політехнічного університету. – 2006. – №3 (38). Ч. 1. – С. 8-11.
9. Коцегуб П.Х. Практическая реализация цифровых САУ в среде пакета МАТЛАБ с использованием платформы реального времени «QNX TARGET» / П.Х. Коцегуб, О.И. Толочко, Р.В. Федоряк // Вісник Національного Технічного Університету «Харківський політехнічний інститут». Зб. наук. праць «Проблеми автоматизованого електроприводу. Теорія і практика». – 2002. – Т.1, №12. — С. 98-101.
10. Демидович Б.П. Основы вычислительной математики / Б.П. Демидович, И.А. Марон. – М.: Наука, 1970. – 664 с.
11. Толочко О.И. Повышение точности цифрового интегрирования при реализации оптимальных алгоритмов управления позиционным приводом / О.И. Толочко, П.Х. Коцегуб, П.И. Розкаряка // Електромашинобудування та електрообладнання. Міжвідомчий науково-технічний збірник. – 2006. – Вип. 66. – С. 66-68.

Received on 02.03.2010