

Содержание

Введение	2
1 Исходные данные	2
2 Дискретизация модели и проверка управляемости	3
3 Синтез модального регулятора по состоянию	4
3.1 Выбор желаемых полюсов	5
3.2 Расчёт коэффициентов K	5
4 Определение масштабирующего коэффициента k_0	6
5 Моделирование системы и переходные процессы	7
Выводы	16
Приложение А. Листинг программы в MATLAB	16

Введение

Целью лабораторной работы является исследование метода модального управления дискретными системами автоматического управления и влияния размещения полюсов замкнутой системы на качество переходного процесса.

В ходе работы требуется:

- задать модель объекта управления в пространстве состояний;
- выполнить её дискретизацию для нескольких значений периода дискретизации T_0 ;
- проверить управляемость дискретной системы;
- синтезировать регулятор по состоянию методом модального управления для двух вариантов размещения полюсов: с перерегулированием порядка 30% и с апериодическим переходным процессом;
- подобрать масштабирующий коэффициент k_0 , обеспечивающий нулевую установившуюся ошибку на ступенчатом воздействии;
- реализовать структуру замкнутой системы в среде Simulink без применения блока State-Space и исследовать переходные процессы.

1 Исходные данные

Объект управления задан в виде

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \quad (1)$$

где $x(t) \in \mathbb{R}^3$ — вектор состояния, $u(t)$ — управляющее воздействие, $y(t)$ — выход системы.

Для варианта 1 матрицы имеют вид

$$A = \begin{bmatrix} -39 & -71 & -23 \\ 23 & 40 & 13 \\ -6,6 & -10,7 & -3,3 \end{bmatrix}, \quad B = \begin{bmatrix} 0,228 \\ -0,127 \\ 0,038 \end{bmatrix}, \quad C = \begin{bmatrix} 28000 & 84000 & 112000 \end{bmatrix}. \quad (2)$$

Периоды дискретизации, используемые в работе:

$$T_0 \in \{0,1; 1; 10\} \text{ с.}$$

2 Дискретизация модели и проверка управляемости

При дискретизации по нулевому порядку удержания (*ZOH*) непрерывная модель переходит в дискретную:

$$x_{k+1} = A_d x_k + B_d u_k, \quad y_k = C_d x_k, \quad (3)$$

где

$$A_d = e^{AT_0}, \quad B_d = \int_0^{T_0} e^{A\tau} B d\tau, \quad C_d = C. \quad (4)$$

Расчёты выполнены в среде MATLAB. Получены следующие матрицы.

Период дискретизации $T_0 = 0,1$ с

$$A_d^{(0,1)} = \begin{bmatrix} -2,3833 & -5,7090 & -1,8796 \\ 1,8225 & 3,8935 & 0,9575 \\ -0,4563 & -0,6412 & 0,8052 \end{bmatrix}, \quad B_d^{(0,1)} = \begin{bmatrix} 0,017508 \\ -0,0087103 \\ 0,0023217 \end{bmatrix}.$$

Период дискретизации $T_0 = 1$ с

$$A_d^{(1)} = \begin{bmatrix} -1,2509 & -2,5863 & -0,9697 \\ 0,7203 & 1,2968 & 0,0901 \\ 0,0844 & 0,5675 & 1,4380 \end{bmatrix}, \quad B_d^{(1)} = \begin{bmatrix} 0,0048087 \\ -0,00085652 \\ 0,0041248 \end{bmatrix}.$$

Период дискретизации $T_0 = 10$ с

$$A_d^{(10)} = \begin{bmatrix} -0,39861 & -1,1924 & -1,7848 \\ -0,85955 & -2,5711 & -3,8485 \\ 3,3336 & 9,9717 & 14,9260 \end{bmatrix}, \quad B_d^{(10)} = \begin{bmatrix} -0,029755 \\ -0,052071 \\ 0,22143 \end{bmatrix}.$$

Проверка управляемости

Матрица управляемости дискретной системы:

$$W_c = [B_d, A_d B_d, A_d^2 B_d].$$

Для всех трёх значений периода дискретизации ранги матриц управляемости равны порядку системы:

$$W_c^{(0,1)} = W_c^{(1)} = W_c^{(10)} = 3.$$

Следовательно, дискретные модели объекта полностью управляемы, что позволяет использовать метод модального управления.

3 Синтез модального регулятора по состоянию

Примем закон управления по состоянию

$$u_k = -Kx_k + k_0 g_k, \tag{5}$$

где $K = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix}$ — вектор обратной связи по состоянию, k_0 — масштабирующий коэффициент, g_k — задающее воздействие (ступенька единичной амплитуды).

Матрица замкнутой системы:

$$A_{cl} = A_d - B_d K. \quad (6)$$

Собственные значения матрицы A_{cl} задаются заранее согласно требуемому характеру переходного процесса.

3.1 Выбор желаемых полюсов

Для варианта с перерегулированием $\approx 30\%$ используется связь между перерегулированием M_p и коэффициентом затухания ζ :

$$M_p = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \approx 0,3, \quad (7)$$

откуда $\zeta \approx 0,36$.

В непрерывной области выбирается доминирующая комплексно-сопряжённая пара полюсов

$$s_{1,2} = -\sigma \pm j\omega_d,$$

а также дополнительный действительный полюс $s_3 = -5$. Для малых и средних значений периода дискретизации ($T_0 = 0,1$ и $T_0 = 1$) желаемые дискретные полюса получены экспоненциальным отображением

$$z_i = e^{s_i T_0}.$$

Для крупного шага $T_0 = 10$ устойчивые дискретные полюса заданы непосредственно в z -области:

$$z_{30} = \{0,7 \pm 0,3j; 0,4\}, \quad z_0 = \{0,7; 0,5; 0,3\}.$$

3.2 Расчёт коэффициентов K

Вектор обратной связи K находится методом Аккерманна (place в MATLAB) для каждого периода дискретизации и каждого варианта размещения полюсов.

Результаты сведены в таблицы 1 и 2.

Таблица 1: Коэффициенты K для варианта с перерегулированием $\approx 30\%$

$T_0, \text{с}$	k_1	k_2	k_3
0,1	209,9	592,6	868,0
1	72,65	208,0	289,4
10	598900,0	390900,0	172500,0

Таблица 2: Коэффициенты K для апериодического варианта

$T_0, \text{с}$	k_1	k_2	k_3
0,1	-54,92	-108,0	45,88
1	69,46	195,9	270,1
10	271100,0	176900,0	78080,0

Собственные значения матриц A_{cl} во всех случаях совпадают с заданными полюсами, что подтверждает корректность синтеза регулятора.

4 Определение масштабирующего коэффициента k_0

Для замкнутой системы со ступенчатым входом $g_k = 1$ установившееся значение выхода определяется выражением

$$y_\infty = C_d(I - A_{cl})^{-1}B_d k_0, \quad A_{cl} = A_d - B_d K. \quad (8)$$

Требуя $y_\infty = 1$, получаем формулу для масштабирующего коэффициента:

$$k_0 = \frac{1}{C_d(I - A_{cl})^{-1}B_d}. \quad (9)$$

Численные значения k_0 для всех режимов приведены в таблице 3.

Таблица 3: Масштабирующий коэффициент k_0

$T_0, \text{ с}$	k_0 (30% перерегулир.)	k_0 (апериодич.)
0,1	$7,547 \cdot 10^{-3}$	$7,313 \cdot 10^{-4}$
1	$2,066 \cdot 10^{-3}$	$1,88 \cdot 10^{-3}$
10	$5,669 \cdot 10^{-6}$	$5,511 \cdot 10^{-6}$

Подбор k_0 обеспечивает нулевую установившуюся ошибку по ступенчатому задающему воздействию во всех рассмотренных случаях.

5 Моделирование системы и переходные процессы

Моделирование выполнено двумя способами:

- прямым программированием дискретной модели в MATLAB по уравнениям

$$x_{k+1} = A_d x_k + B_d u_k, \quad u_k = -K x_k + k_0 g_k, \quad y_k = C_d x_k;$$

- реализацией аналогичной структуры в среде Simulink с использованием сумматоров, блоков усиления и задержки по одному шагу (Unit Delay), без блока State-Space.

На рисунке 1 приведён вид структурной схемы замкнутой системы в Simulink.

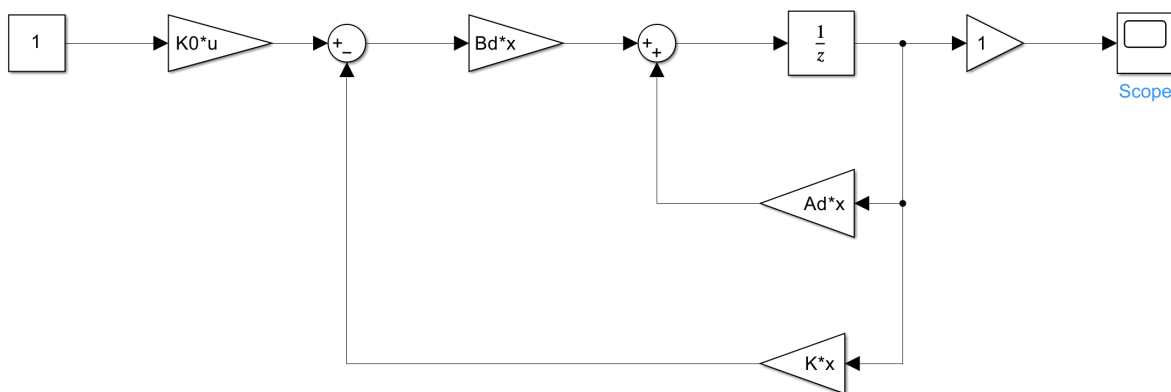


Рис. 1: Структурная схема замкнутой системы в среде Simulink

Шаг моделирования для каждого варианта равен соответствующему периоду дискретизации T_0 , число шагов выбрано одинаковым ($N = 80$), что позволяет сравнивать длительность переходных процессов по реальному времени $t = NT_0$.

Переходные процессы и управляющие воздействия

На рисунках приведены переходные характеристики $y(k)$ и графики управляющего воздействия $u(k)$ для обоих типов регуляторов и всех значений периода дискретизации.

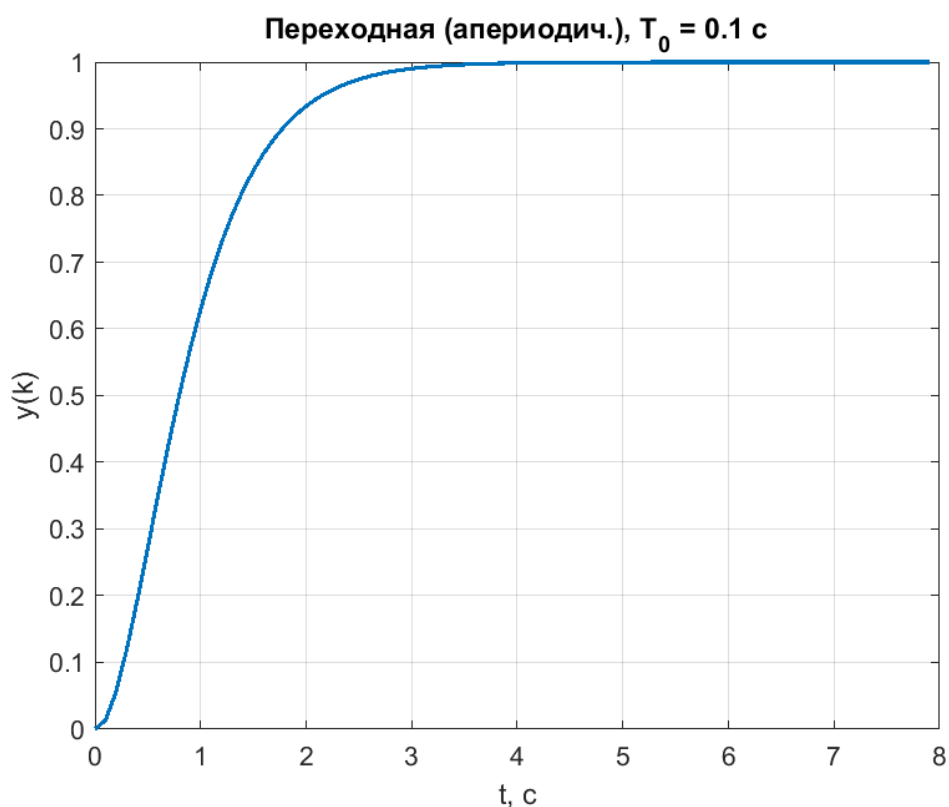


Рис. 2: Переходная характеристика (апериодическая настройка), $T_0 = 0,1$ с

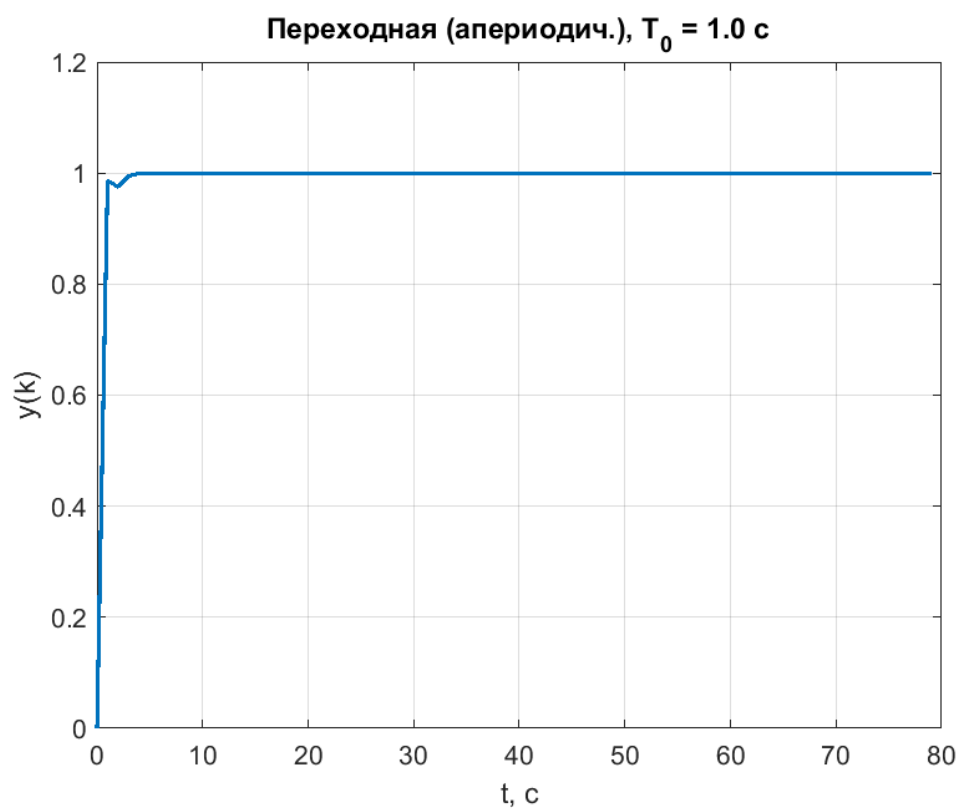


Рис. 3: Переходная характеристика (апериодическая настройка), $T_0 = 1$ с

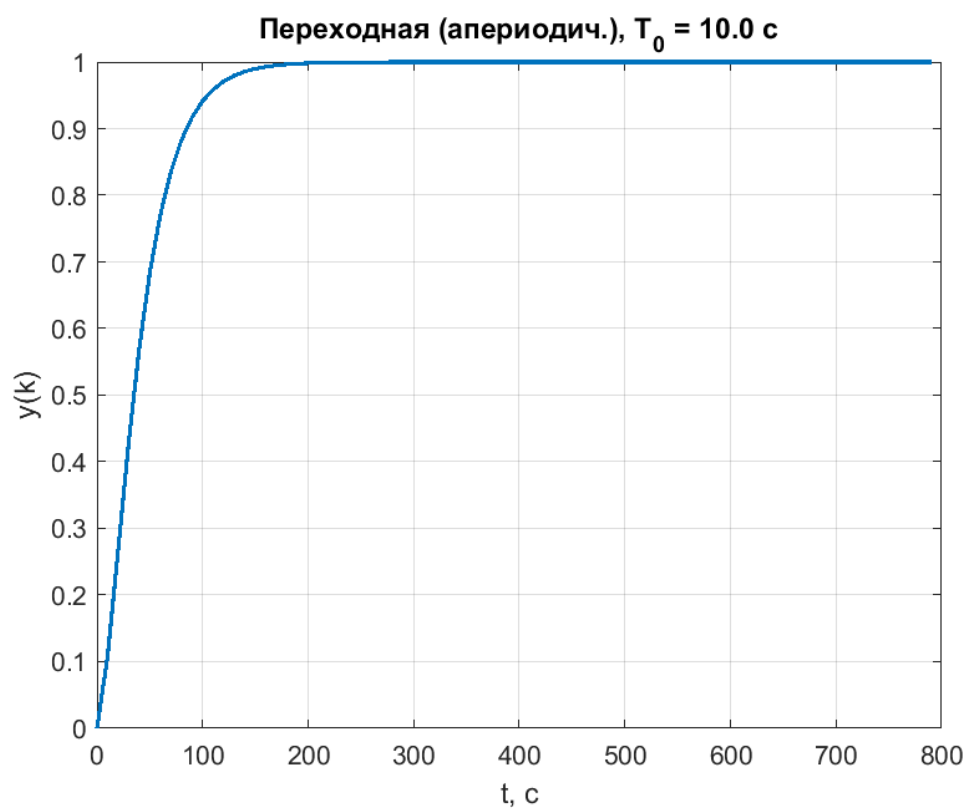


Рис. 4: Переходная характеристика (апериодическая настройка), $T_0 = 10$ с

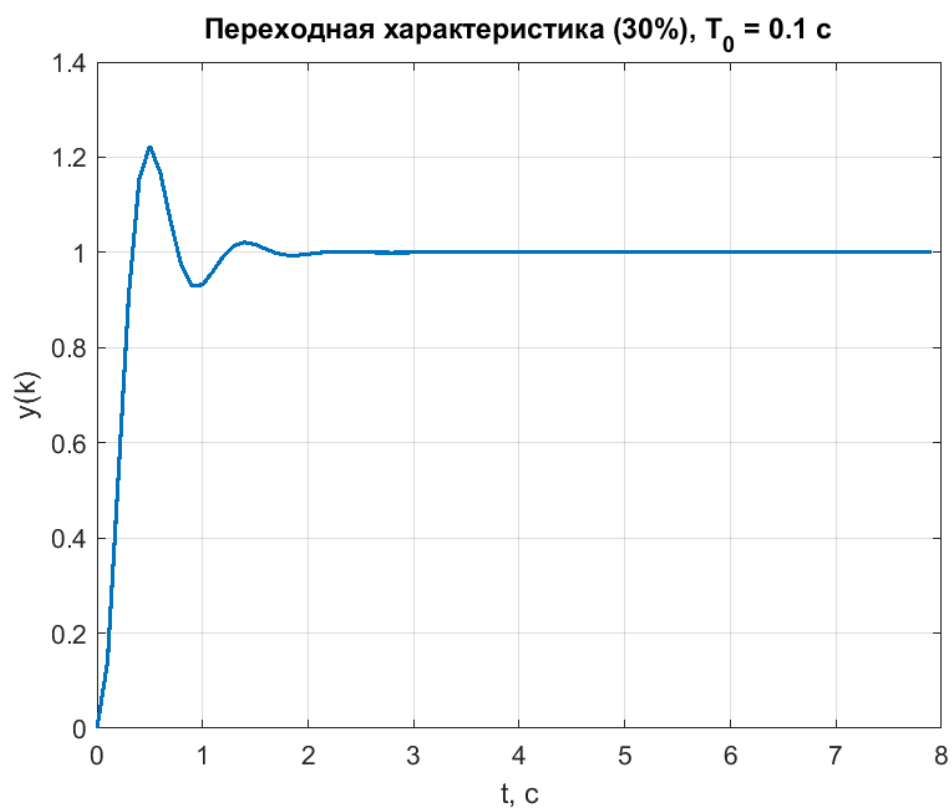


Рис. 5: Переходная характеристика (перерегулирование $\approx 30\%$), $T_0 = 0,1$ с

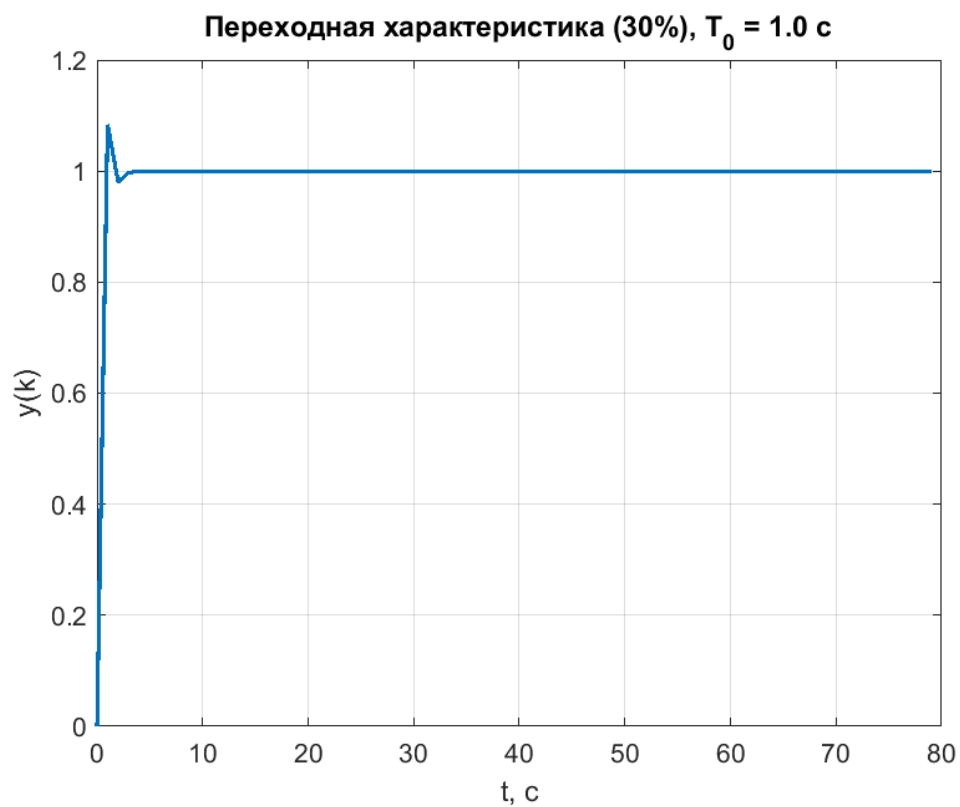


Рис. 6: Переходная характеристика (перерегулирование $\approx 30\%$), $T_0 = 1$ с

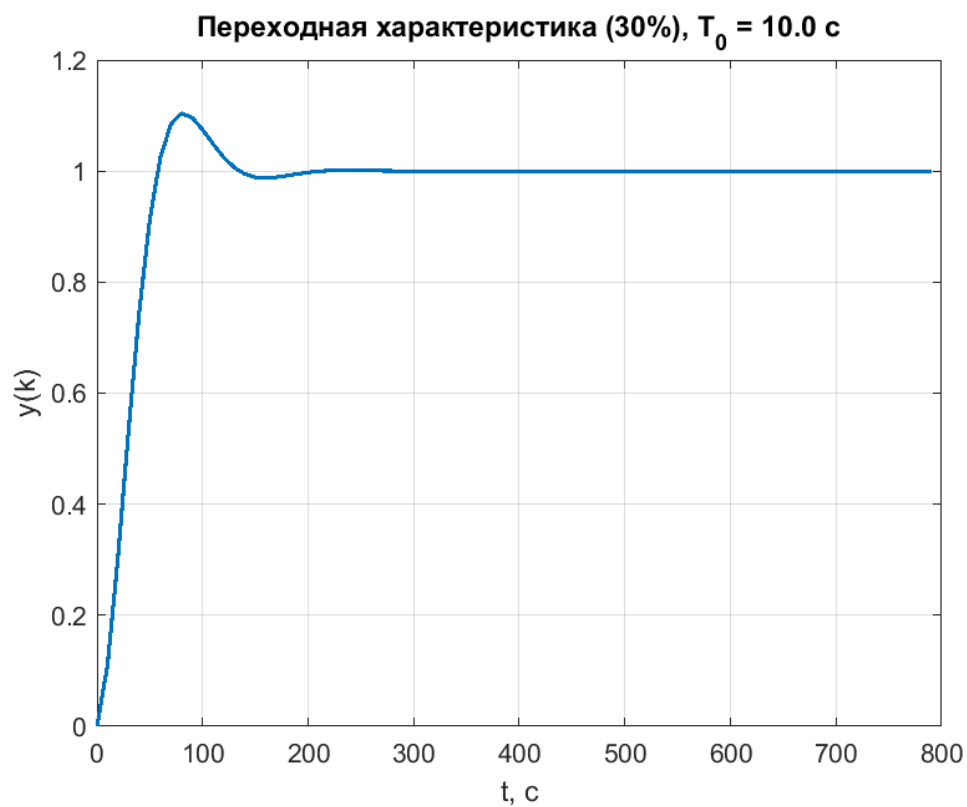


Рис. 7: Переходная характеристика (перерегулирование $\approx 30\%$), $T_0 = 10$ с

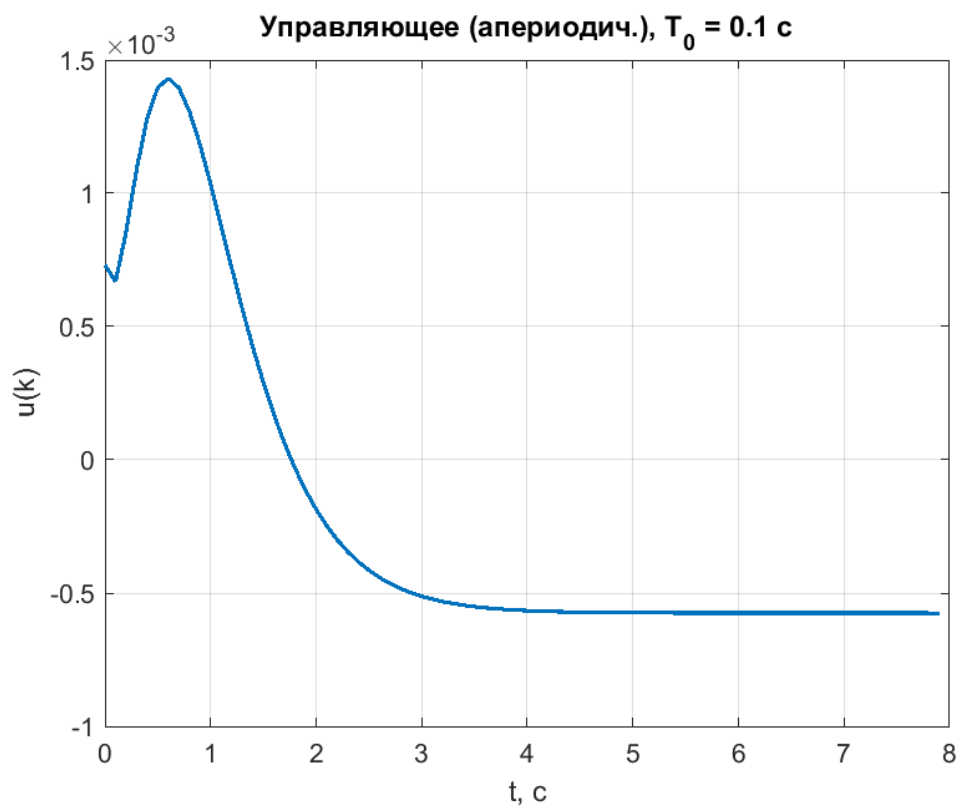


Рис. 8: Управляющее воздействие (апериодическая настройка), $T_0 = 0,1$ с

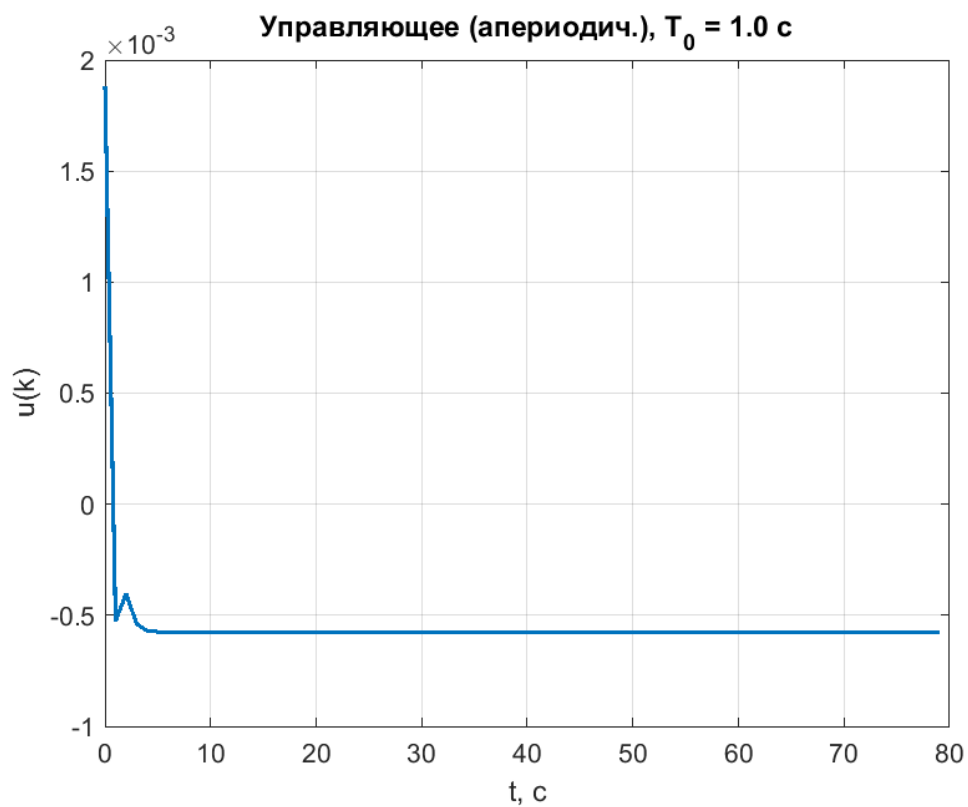


Рис. 9: Управляющее воздействие (апериодическая настройка), $T_0 = 1$ с

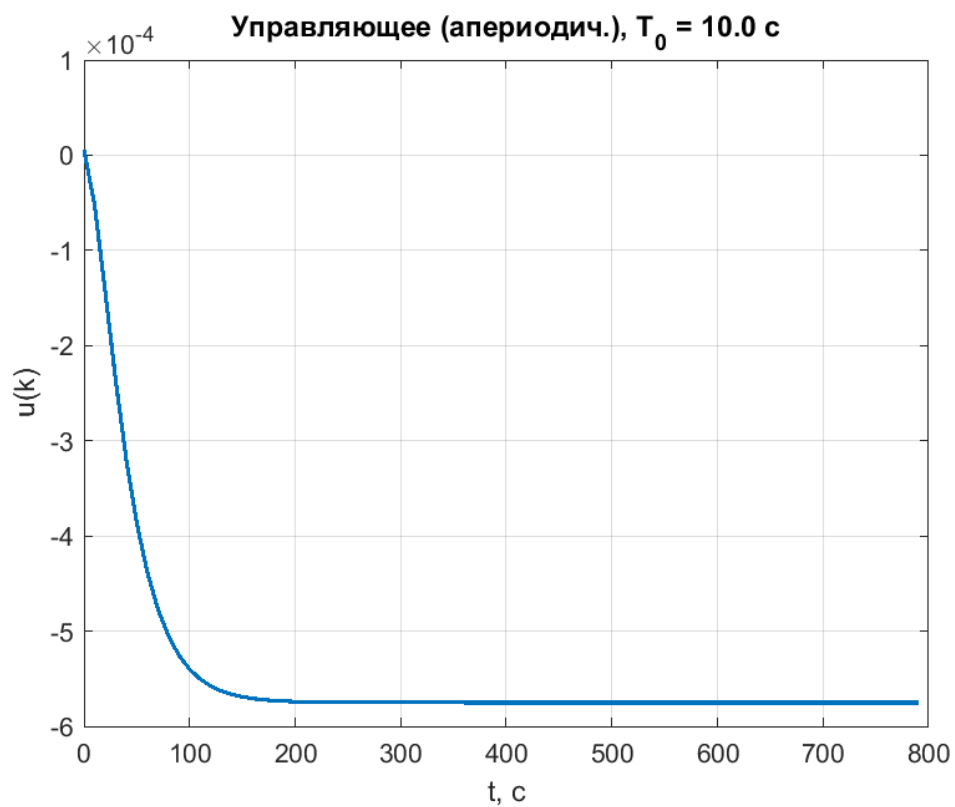


Рис. 10: Управляющее воздействие (апериодическая настройка), $T_0 = 10$ с

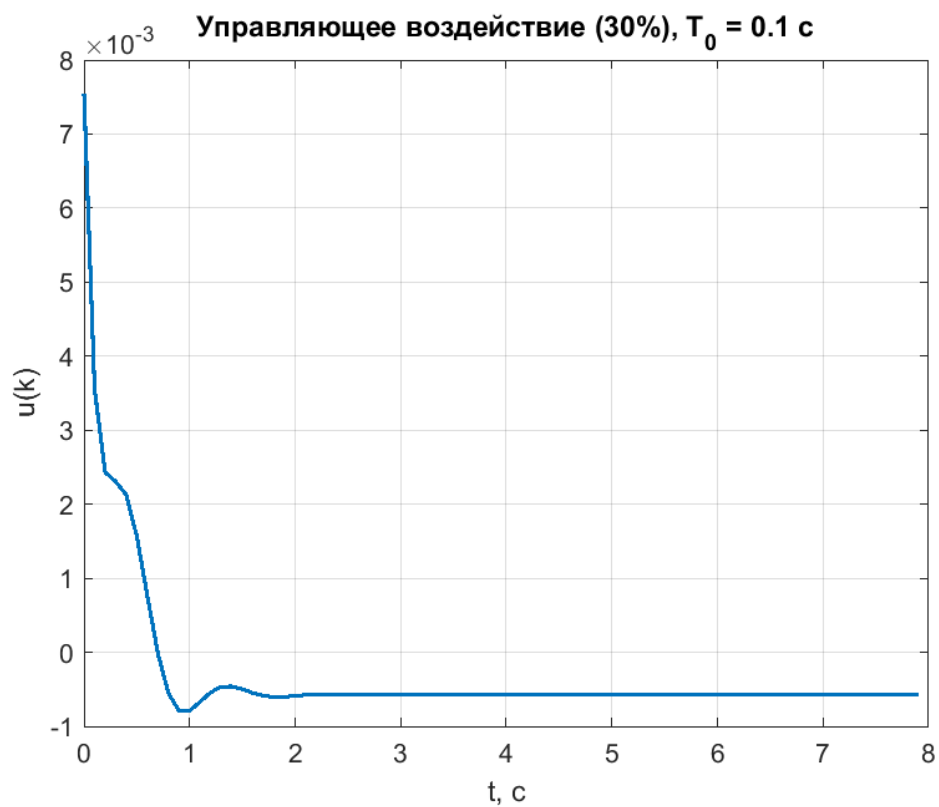


Рис. 11: Управляющее воздействие (перерегулирование $\approx 30\%$), $T_0 = 0,1$ с

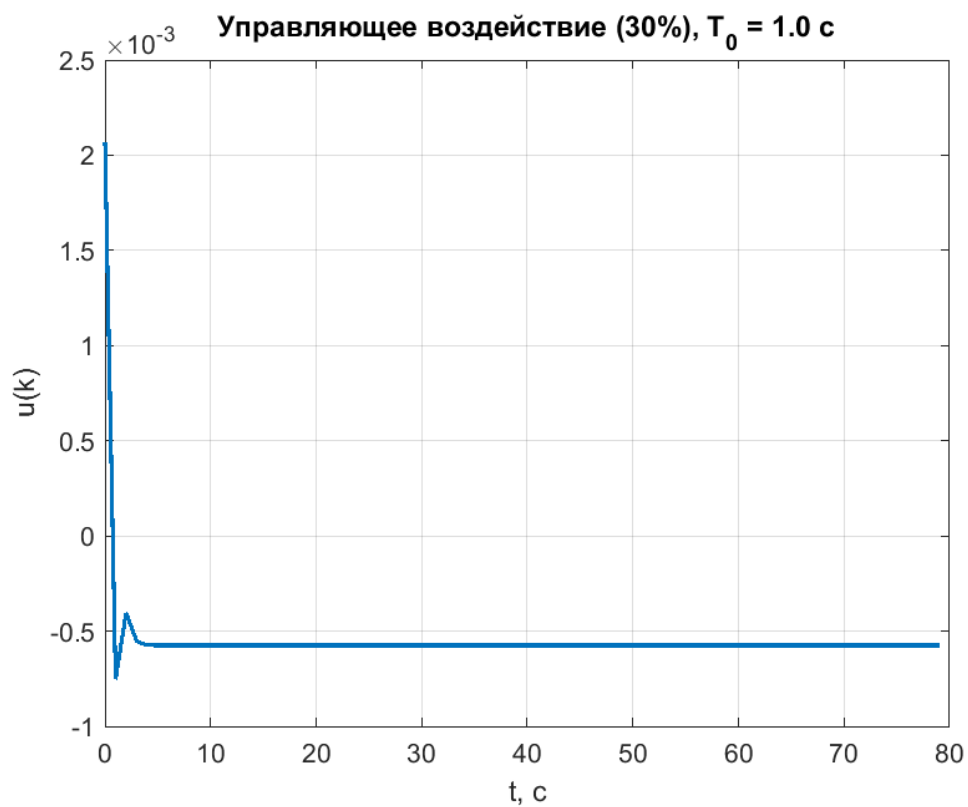


Рис. 12: Управляющее воздействие (перерегулирование $\approx 30\%$), $T_0 = 1$ с

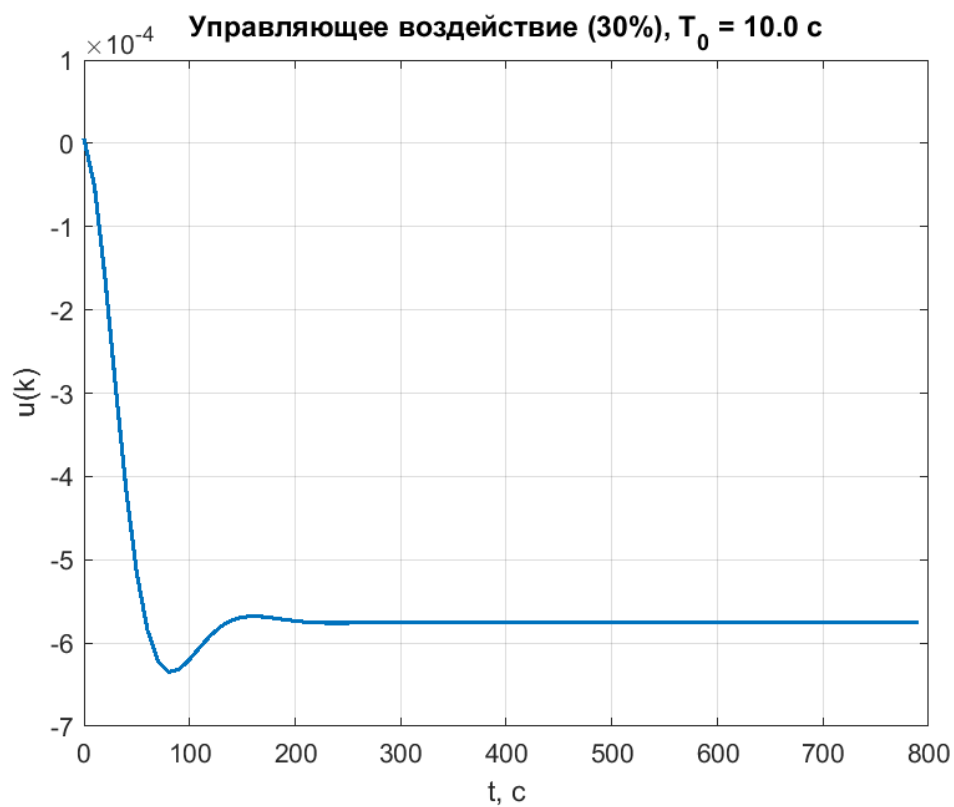


Рис. 13: Управляющее воздействие (перерегулирование $\approx 30\%$), $T_0 = 10$ с

Выводы

В результате выполнения лабораторной работы №3:

1. Для заданного непрерывного объекта управления получены дискретные модели при периодах дискретизации $T_0 = 0,1; 1; 10$ с. Во всех случаях система оказалась полностью управляемой.
2. Синтезированы регуляторы по состоянию методом модального управления для двух вариантов размещения полюсов: с комплексно-сопряжённой парой (перерегулирование порядка 30%) и с вещественными полюсами (апериодический переходный процесс).
3. Подбор масштабирующего коэффициента k_0 обеспечил нулевую установившуюся ошибку по ступенчатому задающему воздействию.
4. Показано, что выбор положения полюсов позволяет управлять качеством переходного процесса: при комплексных полюсах достигается более быстрое, но колебательное регулирование, при вещественных полюсах переходный процесс становится апериодическим, но более медленным.
5. Увеличение периода дискретизации приводит к росту коэффициентов обратной связи и растяжению переходных процессов во времени, что подтверждает необходимость разумного выбора шага дискретизации при проектировании дискретных систем автоматического управления.

Приложение А

Листинг программы в MATLAB

```
1 clc; clear; close all;  
2  
3 A = [ -39   -71   -23;
```



```

4      23   40   13;
5      -6.6 -10.7 -3.3 ];
6
7      B = [ 0.228;
8            -0.127;
9            0.038 ];
10
11     C = [ 28000 84000 112000 ];
12     D = 0;
13
14     sys_c = ss(A,B,C,D);
15
16     Tvec = [0.1 1 10];
17
18     outDir = 'img';
19     if ~exist(outDir,'dir'), mkdir(outDir); end
20
21     fprintf('==== \n');
22
23     for k = 1:numel(Tvec)
24         T0 = Tvec(k);
25         sys_d = c2d(sys_c,T0,'zoh');
26         Ad = sys_d.A;
27         Bd = sys_d.B;
28         Cd = sys_d.C;
29         Dd = sys_d.D;
30
31         disc(k).T0 = T0;
32         disc(k).Ad = Ad;
33         disc(k).Bd = Bd;
34         disc(k).Cd = Cd;
35         disc(k).Dd = Dd;
36
37         Wc = ctrb(Ad,Bd);
38         r = rank(Wc);
39
40         fprintf('\nT0 = %.1f c\n',T0);
41         fprintf('Ad = \n'); disp(Ad);

```

```

42     fprintf('Bd = \n'); disp(Bd);
43     fprintf('rank(ctrb) = %d\n',r);
44 end
45
46 Mp   = 0.3;
47 zeta = 0.36;
48 Ts   = 1.5;
49 wn   = 4/(zeta*Td);
50 sigma = zeta*wn;
51 wd    = wn*sqrt(1 - zeta^2);
52
53 s1_30 = -sigma + 1i*wd;
54 s2_30 = -sigma - 1i*wd;
55 s3_30 = -5;
56
57 s1_0 = -2;
58 s2_0 = -3;
59 s3_0 = -4;
60
61 fprintf('\n====   ===\n');
62
63 for k = 1:numel(Tvec)
64     T0 = disc(k).T0;
65     Ad = disc(k).Ad;
66     Bd = disc(k).Bd;
67     Cd = disc(k).Cd;
68
69     if T0 < 2
70         p30 = exp([s1_30 s2_30 s3_30]*T0);
71         p0  = exp([s1_0  s2_0  s3_0 ]*T0);
72     else
73         p30 = [0.7+0.3j  0.7-0.3j  0.4];
74         p0  = [0.7      0.5      0.3];
75     end
76
77     K30 = place(Ad,Bd,p30);
78     Acl30 = Ad - Bd*K30;
79     k0_30 = 1 / (Cd * ((eye(size(Ad)) - Acl30) \ Bd));

```

```

80
81 K0 = place(Ad,Bd,p0);
82 Acl0 = Ad - Bd*K0;
83 k0_0 = 1 / (Cd * ((eye(size(Ad)) - Acl0) \ Bd));
84
85 disc(k).K30 = K30;
86 disc(k).K0 = K0;
87 disc(k).k0_30 = k0_30;
88 disc(k).k0_0 = k0_0;
89
90 fprintf('\nT0 = %.1f c\n',T0);
91 fprintf(' poles 30%% : %s\n',mat2str(p30,4));
92 fprintf(' K30 = [%s]\n',num2str(K30,' %.4g'));
93 fprintf(' k0_30 = %.4g\n',k0_30);
94 fprintf(' poles 0%% : %s\n',mat2str(p0,4));
95 fprintf(' K0 = [%s]\n',num2str(K0,' %.4g'));
96 fprintf(' k0_0 = %.4g\n',k0_0);
97 end
98
99 fprintf('\n=== ==\n');
100
101 N = 80;
102
103 for k = 1:numel(Tvec)
104     T0 = disc(k).T0;
105     Ad = disc(k).Ad;
106     Bd = disc(k).Bd;
107     Cd = disc(k).Cd;
108
109     t = (0:N-1)' * T0;
110     g = ones(N,1);
111
112     % 30%
113     K = disc(k).K30;
114     k0 = disc(k).k0_30;
115
116     x = zeros(3,1);
117     y = zeros(N,1);

```

```

118     u = zeros(N,1);
119
120     for n = 1:N
121         u(n) = -K*x + k0*g(n);
122         y(n) = Cd*x;
123         x     = Ad*x + Bd*u(n);
124     end
125
126     figure('Visible','off');
127     plot(t,y,'LineWidth',1.5); grid on;
128     xlabel('t, c'); ylabel('y(k)');
129     title(sprintf(' (30%%), T_0 = %.1f c',T0));
130     saveas(gcf,fullfile(outDir,sprintf('resp_y_30_T%.1f.png',T0)));
131
132     figure('Visible','off');
133     plot(t,u,'LineWidth',1.5); grid on;
134     xlabel('t, c'); ylabel('u(k)');
135     title(sprintf(' (30%%), T_0 = %.1f c',T0));
136     saveas(gcf,fullfile(outDir,sprintf('ctrl_u_30_T%.1f.png',T0)));
137
138     %
139     K = disc(k).K0;
140     k0 = disc(k).k0_0;
141
142     x = zeros(3,1);
143     y = zeros(N,1);
144     u = zeros(N,1);
145
146     for n = 1:N
147         u(n) = -K*x + k0*g(n);
148         y(n) = Cd*x;
149         x     = Ad*x + Bd*u(n);
150     end
151
152     figure('Visible','off');
153     plot(t,y,'LineWidth',1.5); grid on;
154     xlabel('t, c'); ylabel('y(k)');
155     title(sprintf(' (.), T_0 = %.1f c',T0));

```

```

156     saveas(gcf,fullfile(outDir,sprintf('resp_y_0_T%.1f.png',T0)));
157
158     figure('Visible','off');
159     plot(t,u,'LineWidth',1.5); grid on;
160     xlabel('t, c'); ylabel('u(k)');
161     title(sprintf(' (.), T_0 = %.1f c',T0));
162     saveas(gcf,fullfile(outDir,sprintf('ctrl_u_0_T%.1f.png',T0)));
163 end

```

Листинг 1: Скрипт моделирования, вариант 1