

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧЕРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Санкт-Петербургский политехнический университет Петра Великого»

ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ  
Высшая школа программной инженерии

## Отчет по курсовой работе по дисциплине «Математические модели»

Выполнила студентка гр. 3530904/80001

Прохорова А. И.

Руководитель

Устинов С. М.

Санкт-Петербург  
2020

## 1 Задание

Для **МОДЕЛИ 3** в плоскости параметров (**p4, p6**) построить бифуркационные диаграммы точек поворота (**p4, p6** > 0). При построении диаграммы целесообразно использовать логарифмический масштаб ( $\log(\mathbf{p4}, \mathbf{p6})$ ) по обеим осям. Убедиться, что это точки поворота, а не ветвления. Проиллюстрировать количество решений в каждой области.  
( $p1 = 8.4\text{e-}6$ ,  $p2 = 6.6667\text{e-}4$ ,  $p3 = 1.7778\text{e-}5$ ,  $p5 = 2$ ).

$$\begin{aligned}\frac{dx_1}{dt} &= (p_1x_2 - x_1x_2 + x_1 - x_1^2)/p_2 - p_4x_1; \\ \frac{dx_2}{dt} &= (-p_1x_2 - x_1x_2 + p_5x_3)/p_3 + p_4(p_6 - x_2); \\ \frac{dx_3}{dt} &= x_1 - x_3 - p_4x_3;\end{aligned}$$

## 2 Блок аналитических преобразований

Будем решать систему, в качестве  $\varepsilon$  варьируя  $p_4$

$$\begin{cases} \det(J_N) = 0 \\ f(x, \varepsilon) = 0 \end{cases} \quad (1)$$

Посчитаем матрицу Якоби:

$$J_N = \begin{pmatrix} -\frac{x_2}{p_2} + \frac{1}{p_2} - \frac{2x_1}{p_2 - p_4} & \frac{p_1}{p_2} - \frac{x_1}{p_2} & 0 \\ -\frac{x_2}{p_3} & -\frac{p_1}{p_3} - \frac{x_1}{p_3 - p_4} & \frac{p_5}{p_3} \\ 1 & 0 & -1 - p_4 \end{pmatrix}$$

Запишем систему  $f(x, p_4) = 0$ :

$$\begin{cases} (p_1x_2 - x_1x_2 + x_1 - x_1^2)/p_2 - p_4x_1 = 0; \\ (-p_1x_2 - x_1x_2 + p_5x_3)/p_3 + p_4(p_6 - x_2) = 0; \\ x_1 - x_3 - p_4x_3 = 0; \end{cases} \quad (2)$$

Из первого уравнения выразим  $x_2$  через  $x_3$ , из третьего уравнения выразим  $x_3$  через  $x_1$ :

$$\begin{aligned}x_2 &= \frac{x_1^2 + p_2p_4x_1 - x_1}{p_1 - x_1} \\ x_3 &= \frac{x_1}{1 + p_4}\end{aligned}$$

Подставим их в уравнение  $\det(J_N) = 0$  и получим кубическое уравнение относительно  $x_1$ . Находим корни, подставляем их в выражения для  $x_2$  и  $x_3$ .

$p_6$  можем найти из второго уравнения:

$$p_6 = \frac{p_1x_2 + x_1x_2 - p_5x_3}{p_3p_4}$$

## 3 Результаты работы программы

```
p4 = 10.000000
x11 = -0.000036 x12 = 0.000053 x13 = 0.405660
x22 = 1.179783 x32 = 0.000005 p6 = 36.582517
p6check = 36.582517
x23 = 0.587685 x33 = 0.036878 p6 = 92614.526489
p6check = 92614.526489
```

## 4 Выводы

Для всех вариантов матриц значение обусловленности очень большое ( $> 50000$ ). Следовательно, доверять полученным результатам мы не можем.

## 5 Приложение

### 5.1 Исходный код

```
1 syms x1 x2 x3 p1 p2 p3 p4 p5 p6
2
3 det11 = -x2/p2 + 1/p2 - 2*x1/p2 - p4;
4 det12 = p1 / p2 - x1 / p2;
5 det13 = 0;
6 det21 = -x2/p3;
7 det22 = - p1 / p3 - x1 / p3 - p4;
8 det23 = p5/p3;
9 det31 = 1;
10 det32 = 0;
11 det33 = - 1 - p4;
12
13 % det11 = - (x1 * x1 + p2 * p4 * x1 - x1) / p2 * (p1 - x1) + 1 / p2 - 2 * x1 / p2 -
    p4;
14 % det12 = p1 / p2 - x1 / p2;
15 % det13 = 0;
16 % det21 = - (x1 * x1 + p2 * p4 * x1 - x1) / p3 * (p1 - x1);
17 % det22 = - p1 / p3 - x1 / p3 - p4;
18 % det23 = p5/p3;
19 % det31 = 1;
20 % det32 = 0;
21 % det33 = - 1 - p4;
22
23 x2val = (x1 * x1 + p2 * p4 * x1 - x1) / (p1 - x1);
24 x3val = x1 / (1 + p4);
25 p6 = (p1 * x2 + x1 * x2 - p5 * x3) / p3 * p4 + x2;
26
27 det11 = subs(det11, x2, x2val);
28 det21 = subs(det21, x2, x2val);
29
30 res = det ( [det11 det12 det13; det21 det22 det23; det31 det32 det33] );
31 eq = res == 0;
32
33 x2val = subs(x2val, p1, 8.4e-6);
34 p6 = subs(p6, p1, 8.4e-6);
35 eq = subs(eq, p1, 8.4e-6);
36
37
38 eq = subs(eq, p2, 6.6667e-4);
39 x2val = subs(x2val, p2, 6.6667e-4);
40 p6 = subs(p6, p2, 6.6667e-4);
41
42 eq = subs(eq, p3, 1.7778e-5);
43 p6 = subs(p6, p3, 1.7778e-5);
44
45 eq = subs(eq, p5, 2);
46 p6 = subs(p6, p5, 2);
47
48 fileID = fopen('result.txt', 'w');
49
50 %for i = 1.0: 0.1: 2.0
51 for i = 10.0: 1.0: 10.0
```

```

52 inner = subs(eq, p4, i);
53 x2tmp = subs(x2val, p4, i);
54 x3tmp = subs(x3val, p4, i);
55 p6tmp = subs(p6, p4, i);
56
57 fprintf(fileID, 'p4 = %f \n', i);
58
59 solution = solve(inner, x1, 'IgnoreProperties', true);
60 if (isempty(solution)==false)
61     x11 = solution(1, 1);
62     x12 = solution(2, 1);
63     x13 = solution(3, 1);
64     fprintf(fileID, 'x11 = %f x12 = %f x13 = %f \n', x11, x12, x13);
65     if (isAlways(x11 > 0) == true)
66         x21 = subs(x2tmp, x1, x11);
67         x31 = subs(x3tmp, x1, x11);
68         p6res = subs(p6tmp, x1, x11);
69         p6res = subs(p6res, x2, x21);
70         p6res = subs(p6res, x3, x31);
71         fprintf(fileID, 'x21 = %f x31 = %f p6 = %f \n', x21, x31, p6res);
72         p6check = (8.4e-6 * x21 + x11 * x21 - 2 * x31) / 1.7778e-5 * i + x21;
73         fprintf(fileID, 'p6check = %f \n', p6check);
74     end
75     if (isAlways(x12 > 0))
76         x22 = subs(x2tmp, x1, x12);
77         x32 = subs(x3tmp, x1, x12);
78         p6res = subs(p6tmp, x1, x12);
79         p6res = subs(p6res, x2, x22);
80         p6res = subs(p6res, x3, x32);
81         fprintf(fileID, 'x22 = %f x32 = %f p6 = %f \n', x22, x32, p6res);
82         p6check = (8.4e-6 * x22 + x12 * x22 - 2 * x32) / 1.7778e-5 * i + x22;
83         fprintf(fileID, 'p6check = %f \n', p6check);
84     end
85     if (isAlways(x13 > 0) == true)
86         x23 = subs(x2tmp, x1, x13);
87         x33 = subs(x3tmp, x1, x13);
88         p6res = subs(p6tmp, x1, x13);
89         p6res = subs(p6res, x2, x23);
90         p6res = subs(p6res, x3, x33);
91         fprintf(fileID, 'x23 = %f x33 = %f p6 = %f \n', x23, x33, p6res);
92         p6check = (8.4e-6 * x23 + x13 * x23 - 2 * x33) / 1.7778e-5 * i + x23;
93         fprintf(fileID, 'p6check = %f \n', p6check);
94     end
95     fprintf(fileID, '\n');
96 end
97
98
99 end
100 fclose(fileID);

```