Курсовая работа

по дисциплине «Математические модели»

Выполнил

студент гр.3530904/80004 < > Мащенко Б.Б.

Преподаватель <> Устинов С.М.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2020 г.

**Оглавление**

[Задача 3](#_Toc38636055)

[Выполнение работы 4](#_Toc38636056)

[Код программы 6](#_Toc38636057)

[Результат работы программы 8](#_Toc38636058)

[Проверка 12](#_Toc38636059)

[Вывод 13](#_Toc38636060)

[Список литературы 14](#_Toc38636061)

# Задача

71. Для **МОДЕЛИ 14** в плоскости параметров (**p4, p1**) построить бифуркационные диаграммы точек бифуркации Андронова-Хопфа

(**p1, p4** > 0). Априорно известно, что p1, p4, p5, x1, x2 > 0; p2, p3 > 1.  
(p2 = 1.5, p3 = 3, p5 = 0.01).

# Выполнение работы

Матрица Якоби для исходных уравнений:

Получаем 3-е уравнение:

Получаем систему из 3-х уравнений:

Так как неизвестных 4 а уравнений 3, одну из неизвестных будем варьировать. Варьировать будем переменную , так как относительно нее сложнее всего решать данную систему. Получается система из 3-х уравнений с 3-мя неизвестными , , .

*Решаем систему*:

Из уравнения (2) выразим .

Подставим в уравнение (1) и выразим из него .

Подставим полученные и в уравнение (3).

Разрешим полученное уравнение относительно P4.

Теперь видны коэффициенты квадратного уравнения, которые далее мы запишем в программу, которая будет их вычислять и находить , затем подставлять его в предыдущие уравнения и получать и .

Варьируя в промежутке [0.1, 1.3] с шагом 0.01, будем подставлять ее в ранее полученные уравнение и находить , , . Таким образом получим множество точек бифуркации по которым будем строить диаграмму.

Нам нужно построить бифуркационную диаграмму точек бифуркации Андронова-Хопфа. Но условие: «Сумма собственных значений равна 0» является необходимым условием, но не достаточным, поэтому для проверки будем находить собственные числа, и если собственные значения были чисто мнимой парой корней с нулевой или почти нулевой вещественной частью – эти нужные точки. На графике эти точки будем различать разным цветом.

# Код программы

Программа написана на языке Python

from math import sqrt  
import matplotlib.pyplot as plt  
  
  
def frange*(*froom, to, step*)*:  
 while froom < to:  
 yield froom  
 froom += step  
  
  
def solve\_system*(*p2, p3, p5*)*:  
 res = *{*'x1': *[]*, 'x2': *[]*, 'p1': *[]*, 'p4': *[]}* for x1 in frange*(*0.1, 1.3, 0.01*)*:  
 a = -1  
 b = -1 + 2 \* x1 + *((*1 + x1 \*\* p3*)* \* p3 \* x1 \*\* p3 \* *((*1 + x1 \*\* p3*)* - *(*p5 + x1 \*\* p3*)))* / \  
 *((*p5 + x1 \*\* p3*)* \* *((*1 + x1 \*\* p3*)* \*\* 2*))* c = x1 \* *(*1 - p2 - x1*)* + *(*p3 \* x1 \*\* *(*p3 + 1*)* \* *(*p2 - 1*)* \* *(*1 + x1 \*\* p3*)* \* *((*1 + x1 \*\* p3*)* - *(*p5 + x1 \*\* p3*)))* / \  
 *((*p5 + x1 \*\* p3*)* \* *((*1 + x1 \*\* p3*)* \*\* 2*))* discriminant = *(*b \*\* 2 - 4 \* a \* c*)* if discriminant >= 0:  
 lambda\_1 = *(*- b + sqrt*(*discriminant*))* / *(*2 \* a*)* lambda\_2 = *(*- b - sqrt*(*discriminant*))* / *(*2 \* a*)* if lambda\_1 > 0:  
 x2 = x1 \* p2 / *(*lambda\_1 - x1*)* if x2 > 0:  
 p1 = x1 \* *(*1 + x2*)* \* *(*1 + x1 \*\* p3*)* / *(*p5 + x1 \*\* p3*)* if p1 > 0:  
 if not *(*x1 == lambda\_1 and x1 \* x2 == 0*)*:  
 res*[*'x1'*]*.append*(*x1*)* res*[*'x2'*]*.append*(*x2*)* res*[*'p1'*]*.append*(*p1*)* res*[*'p4'*]*.append*(*lambda\_1*)* if lambda\_2 >= 0:  
 x2 = x1 \* p2 / *(*lambda\_2 - x1*)* if x2 > 0:  
 p1 = x1 \* *(*1 + x2*)* \* *(*1 + x1 \*\* p3*)* / *(*p5 + x1 \*\* p3*)* if p1 > 0:  
 if not *(*x1 == lambda\_2 and x1 \* x2 == 0*)*:  
 res*[*'x1'*]*.append*(*x1*)* res*[*'x2'*]*.append*(*x2*)* res*[*'p1'*]*.append*(*p1*)* res*[*'p4'*]*.append*(*lambda\_2*)* print 'x1\t\t\tx2\t\t\t\tp1\t\t\t\t\tp4\t\t\t\tcheck\t\t\tlambda1\t\t\t\tlambda2'  
 for i in range*(*0, len*(*res*[*'x1'*])*, 1*)*:  
 aa = 1  
 bb = 0 - *(((*res*[*'p1'*][*i*]* \* p3 \* res*[*'x1'*][*i*]* \*\* *(*p3 - 1*)* \* *((*1 + res*[*'x1'*][*i*]* \*\* p3*)* - *(*p5 + res*[*'x1'*][*i*]* \*\* p3*)))* / *((*1 + res*[*'x1'*][*i*]* \*\* p3*)* \*\* 2*))* - *(*1 + res*[*'x2'*][*i*]))* \  
 - *(*res*[*'x1'*][*i*]* - res*[*'p4'*][*i*])* cc = *(((*res*[*'p1'*][*i*]* \* p3 \* res*[*'x1'*][*i*]* \*\* *(*p3 - 1*)* \* *((*1 + res*[*'x1'*][*i*]* \*\* p3*)* - *(*p5 + res*[*'x1'*][*i*]* \*\* p3*)))* / *((*1 + res*[*'x1'*][*i*]* \*\* p3*)* \*\* 2*))* - *(*1 + res*[*'x2'*][*i*]))* \  
 \* *(*res*[*'x1'*][*i*]* - res*[*'p4'*][*i*])* + res*[*'x1'*][*i*]* \* *(*p2 + res*[*'x2'*][*i*])* discriminant = *(*bb \*\* 2 - 4 \* aa \* cc*)* if discriminant >= 0:  
 lambda\_1 = *(*- bb + sqrt*(*discriminant*))* / *(*2 \* aa*)* lambda\_2 = *(*- bb - sqrt*(*discriminant*))* / *(*2 \* aa*)* print 'd >= 0 ', '%0.2f' % res*[*'x1'*][*i*]*, ' ', '%0.10f' % lambda\_1, ' ', '%0.10f' % lambda\_2, ' ', '%0.10f' % discriminant  
 else:  
 print 'd < 0 ', '%0.2f' % res*[*'x1'*][*i*]*, ' ', '%0.10f' % *((*- bb*)* / *(*2 \* aa*))*, ' ', '%0.10f' % sqrt*(*-discriminant*)* , ' ', '%0.10f' % discriminant  
  
 print '%0.2f' % res*[*'x1'*][*i*]*, ' ', '%0.10f' % res*[*'x2'*][*i*]*, ' ', \  
 '%0.10f' % res*[*'p1'*][*i*]*, ' ', '%0.10f' % res*[*'p4'*][*i*]*, ' ', \  
 *(*res*[*'p1'*][*i*]* \* p3 \* res*[*'x1'*][*i*]* \*\* *(*p3 - 1*)* \* *(*1 + res*[*'x1'*][*i*]* \*\* p3*)* - res*[*'p1'*][*i*]* \* *(* p5 + res*[*'x1'*][*i*]* \*\* p3*)* \* p3 \* res*[*'x1'*][*i*]* \*\* *(*p3 - 1*))* / \  
 *((*1 + res*[*'x1'*][*i*]* \*\* p3*)* \*\* 2*)* - *(*1 + res*[*'x2'*][*i*])* + res*[*'x1'*][*i*]* - res*[*'p4'*][*i*]* return res  
  
  
result = solve\_system*(*1.5, 3, 0.01*)*plt.title*(***u'Bifurcation diagram'***)*plt.xlabel*(*'p4'*)*plt.ylabel*(*'p1'*)*plt.plot*(*result*[*'p4'*]*, result*[*'p1'*]*, '.'*)*plt.xlim*(*0, 5*)*plt.ylim*(*0, 25*)*plt.show*()*

# Результат работы программы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.01 |  |  |  |  |  |
| 0.02 |  |  |  |  |  |
| 0.03 |  |  |  |  |  |
| 0.04 |  |  |  |  |  |
| 0.05 |  |  |  |  |  |
| 0.06 |  |  |  |  |  |
| 0.07 |  |  |  |  |  |
| 0.08 |  |  |  |  |  |
| 0.09 |  |  |  |  |  |
| 0.10 |  |  |  |  |  |
| 0.11 |  |  |  |  |  |
| 0.12 |  |  |  |  |  |
| 0.13 |  |  |  |  |  |
| 0.14 |  |  |  |  |  |
| 0.15 |  |  |  |  |  |
| 0.16 |  |  |  |  |  |
| 0.17 |  |  |  |  |  |
| 0.18 | 1.32437942 | 26.58087009 | 0.38386907 | 0.000000 + 1.36649i | 0.000000 - 1.36649i |
| 0.19 | 0.79394770 | 20.35636478 | 0.54896571 | -0.000000 + 1.10814i | -0.000000 - 1.10814i |
| 0.20 | 0.60418783 | 17.96690372 | 0.69653433 | -0.000000 + 0.83496i | -0.000000 - 0.83496i |
| 0.21 | 0.50458038 | 16.55614991 | 0.83428111 | -0.000000 + 0.35347i | -0.000000 - 0.35347i |
| 0.22 | 0.44350186 | 15.54397030 | 0.96407805 | 0.3550799081 | -0.3550799081 |
| 0.23 | 0.40279647 | 14.73220496 | 1.08651198 | 0.5440308656 | -0.5440308656 |
| 0.24 | 0.37429357 | 14.03584761 | 1.20181187 | 0.6893849555 | -0.6893849555 |
| 0.25 | 0.35373135 | 13.41349657 | 1.31012657 | 0.8126718379 | -0.8126718379 |
| 0.26 | 0.33865408 | 12.84331983 | 1.41161762 | 0.9209631251 | -0.9209631251 |
| 0.27 | 0.32754094 | 12.31314728 | 1.50648665 | 1.0175771133 | -1.0175771133 |
| 0.28 | 0.31939711 | 11.81587831 | 1.59497746 | 1.1044159277 | -1.104415927 |
| 0.29 | 0.31354326 | 11.34717464 | 1.67736834 | 1.1827355472 | -1.1827355472 |
| 0.30 | 0.30949925 | 10.90423568 | 1.75396150 | 1.2534569267 | -1.2534569267 |
| 0.31 | 0.30691598 | 10.48512473 | 1.82507261 | 1.3173082662 | -1.3173082662 |
| 0.32 | 0.30553365 | 10.08839038 | 1.89102173 | 1.3748958182 | -1.3748958182 |
| 0.33 | 0.30515510 | 9.71285219 | 1.95212594 | 1.4267415219 | -1.4267415219 |
| 0.34 | 0.30562827 | 9.35747998 | 2.00869379 | 1.4733042307 | -1.4733042307 |
| 0.35 | 0.30683427 | 9.02132724 | 2.06102137 | 1.5149924503 | -1.5149924503 |
| 0.36 | 0.30867909 | 8.70349578 | 2.10938962 | 1.5521725287 | -1.5521725287 |
| 0.37 | 0.31108770 | 8.40311846 | 2.15406281 | 1.5851743364 | -1.5851743364 |
| 0.38 | 0.31399979 | 8.11935200 | 2.19528784 | 1.6142955145 | -1.6142955145 |
| 0.39 | 0.31736662 | 7.85137524 | 2.23329406 | 1.6398048715 | -1.6398048715 |
| 0.40 | 0.32114865 | 7.59839005 | 2.26829373 | 1.6619452423 | -1.6619452423 |
| 0.41 | 0.32531376 | 7.35962344 | 2.30048259 | 1.6809359824 | -1.6809359824 |
| 0.42 | 0.32983589 | 7.13432977 | 2.33004080 | 1.6969751909 | -1.6969751909 |
| 0.43 | 0.33469392 | 6.92179287 | 2.35713391 | 1.7102417151 | -1.7102417151 |
| 0.44 | 0.33987089 | 6.72132751 | 2.38191389 | 1.7208969674 | -1.7208969674 |
| 0.45 | 0.34535329 | 6.53228060 | 2.40452024 | 1.7290865728 | -1.7290865728 |
| 0.46 | 0.35113056 | 6.35403161 | 2.42508100 | 1.7349418606 | -1.7349418606 |
| 0.47 | 0.35719465 | 6.18599276 | 2.44371379 | 1.7385812112 | -1.7385812112 |
| 0.48 | 0.36353965 | 6.02760869 | 2.46052676 | 1.7401112650 | -1.7401112650 |
| 0.49 | 0.37016156 | 5.87835588 | 2.47561949 | 1.7396280038 | -1.7396280038 |
| 0.50 | 0.37705802 | 5.73774175 | 2.48908381 | 1.7372177113 | -1.7372177113 |
| 0.51 | 0.38422815 | 5.60530367 | 2.50100456 | 1.7329578201 | -1.7329578201 |
| 0.52 | 0.39167238 | 5.48060778 | 2.51146032 | 1.7269176529 | -1.7269176529 |
| 0.53 | 0.39939232 | 5.36324776 | 2.52052400 | 1.7191590633 | -1.7191590633 |
| 0.54 | 0.40739068 | 5.25284351 | 2.52826344 | 1.7097369830 | -1.7097369830 |
| 0.55 | 0.41567118 | 5.14903987 | 2.53474190 | 1.6986998794 | -1.6986998794 |
| 0.56 | 0.42423845 | 5.05150529 | 2.54001855 | 1.6860901280 | -1.6860901280 |
| 0.57 | 0.43309804 | 4.95993058 | 2.54414884 | 1.6719443024 | -1.6719443024 |
| 0.58 | 0.44225634 | 4.87402765 | 2.54718490 | 1.6562933833 | -1.6562933833 |
| 0.59 | 0.45172055 | 4.79352835 | 2.54917587 | 1.6391628878 | -1.6391628878 |
| 0.60 | 0.46149867 | 4.71818331 | 2.55016814 | 1.6205729157 | -1.6205729157 |
| 0.61 | 0.47159949 | 4.64776087 | 2.55020564 | 1.6005381116 | -1.6005381116 |
| 0.62 | 0.48203261 | 4.58204606 | 2.54933006 | 1.5790675355 | -1.5790675355 |
| 0.63 | 0.49280838 | 4.52083969 | 2.54758103 | 1.5561644340 | -1.5561644340 |
| 0.64 | 0.50393799 | 4.46395745 | 2.54499630 | 1.5318258998 | -1.5318258998 |
| 0.65 | 0.51543343 | 4.41122906 | 2.54161187 | 1.5060424023 | -1.5060424023 |
| 0.66 | 0.52730757 | 4.36249757 | 2.53746213 | 1.4787971651 | -1.4787971651 |
| 0.67 | 0.53957414 | 4.31761865 | 2.53257999 | 1.4500653606 | -1.4500653606 |
| 0.68 | 0.55224780 | 4.27645992 | 2.52699694 | 1.4198130767 | -1.4198130767 |
| 0.69 | 0.56534420 | 4.23890047 | 2.52074312 | 1.3879959971 | -1.3879959971 |
| 0.70 | 0.57887999 | 4.20483025 | 2.51384746 | 1.3545577149 | -1.3545577149 |
| 0.71 | 0.59287295 | 4.17414970 | 2.50633763 | 1.3194275653 | ­-1.3194275653 |
| 0.72 | 0.60734202 | 4.14676929 | 2.49824020 | 1.2825178163 | - 1.2825178163 |
| 0.73 | 0.62230739 | 4.12260921 | 2.48958058 | 1.2437199879 | -1.2437199879 |
| 0.74 | 0.63779061 | 4.10159908 | 2.48038311 | 1.2028999580 | -1.2028999580 |
| 0.75 | 0.65381468 | 4.08367770 | 2.47067107 | 1.1598913424 | -1.1598913424 |
| 0.76 | 0.67040419 | 4.06879291 | 2.46046671 | 1.1144863558 | -1.1144863558 |
| 0.77 | 0.68758545 | 4.05690144 | 2.44979122 | 1.0664228827 | -1.0664228827 |
| 0.78 | 0.70538664 | 4.04796884 | 2.43866482 | 1.0153656510 | -1.0153656510 |
| 0.79 | 0.72383799 | 4.04196952 | 2.42710667 | 0.9608778483 | -0.9608778483 |
| 0.80 | 0.74297197 | 4.03888678 | 2.41513495 | 0.9023764864 | -0.9023764864 |
| 0.81 | 0.76282353 | 4.03871296 | 2.40276681 | 0.8390584338 | -0.8390584338 |
| 0.82 | 0.78343031 | 4.04144965 | 2.39001840 | 0.7697693921 | -0.7697693921 |
| 0.83 | 0.80483297 | 4.04710794 | 2.37690483 | 0.6927504447 | -0.6927504447 |
| 0.84 | 0.82707547 | 4.05570885 | 2.36344017 | 0.6050839334 | -0.6050839334 |
| 0.85 | 0.85020548 | 4.06728378 | 2.34963747 | 0.5012363533 | -0.5012363533 |
| 0.86 | 0.87427476 | 4.08187508 | 2.33550868 | 0.3677629177 | -0.3677629177 |
| 0.87 | 0.89933965 | 4.09953680 | 2.32106467 | 0.1347708459 | -0.1347708459 |
| 0.88 | 0.92546163 | 4.12033552 | 2.30631521 | -0.000000 + 0.63255i | -0.000000 - 0.63255i |
| 0.89 | 0.95270793 | 4.14435139 | 2.29126891 | 0.000000 + 0.93671i | 0.000000 - 0.93671i |
| 0.90 | 0.98115226 | 4.17167935 | 2.27593323 | 0.000000 + 1.16592i | 0.000000 - 1.16592i |
| 0.91 | 1.01087571 | 4.20243059 | 2.26031438 | -0.000000 + 1.35875i | -0.000000 - 1.35875i |
| 0.92 | 1.04196764 | 4.23673427 | 2.24441733 | -0.000000 + 1.52909i | -0.000000 - 1.52909i |
| 0.93 | 1.07452693 | 4.27473956 | 2.22824573 | 0.000000 + 1.68389i | 0.000000 - 1.68389i |
| 0.94 | 1.10866326 | 4.31661810 | 2.21180187 | -0.000000 + 1.82720i | -0.000000 - 1.82720i |
| 0.95 | 1.14449876 | 4.36256689 | 2.19508654 | -0.000000 + 1.96167i | -0.000000 - 1.96167i |
| 0.96 | 1.18216987 | 4.41281177 | 2.17809905 | 0.000000 + 2.08913i | 0.000000 - 2.08913i |
| 0.97 | 1.22182966 | 4.46761160 | 2.16083703 | -0.000000 + 2.21096i | -0.000000 - 2.21096i |
| 0.98 | 1.26365049 | 4.52726329 | 2.14329635 | -0.000000 + 2.32819i | -0.000000 - 2.32819i |
| 0.99 | 1.30782735 | 4.59210793 | 2.12547098 | -0.000000 + 2.44168i | -0.000000 - 2.44168i |
| 1.00 | 1.35458181 | 4.66253823 | 2.10735283 | -0.000000 + 2.55214i | 0.000000 - 2.55214i |
| 1.01 | 1.40416698 | 4.73900771 | 2.08893151 | 0.000000 + 2.66016i | 0.000000 - 2.66016i |
| 1.02 | 1.45687355 | 4.82204200 | 2.07019409 | -0.000000 + 2.76630i | -0.000000 - 2.76630i |
| 1.03 | 1.51303730 | 4.91225303 | 2.05112486 | -0.000000 + 2.87105i | -0.000000 - 2.87105i |
| 1.04 | 1.57304864 | 5.01035681 | 2.03170487 | 0.000000 + 2.97489i | 0.000000 - 2.97489i |
| 1.05 | 1.63736465 | 5.11719611 | 2.01191157 | 0.000000 + 3.07828i | 0.000000 - 3.07828i |
| 1.06 | 1.70652455 | 5.23376950 | 1.99171821 | 0.000000 + 3.18171i | 0.000000 - 3.18171i |
| 1.07 | 1.78116984 | 5.36126925 | 1.97109318 | -0.000000 + 3.28566i | -0.000000 - 3.28566i |
| 1.08 | 1.86207082 | 5.50113115 | 1.94999913 | -0.000000 + 3.39066i | -0.000000 - 3.39066i |
| 1.09 | 1.95016202 | 5.65510119 | 1.92839188 | 0.000000 + 3.49730i | 0.000000 - 3.49730i |
| 1.10 | 2.04659038 | 5.82532617 | 1.90621898 | -0.000000 + 3.60625i | -0.000000 - 3.60625i |
| 1.11 | 2.15278196 | 6.01447919 | 1.88341785 | -0.000000 + 3.71829i | -0.000000 - 3.71829i |
| 1.12 | 2.27053615 | 6.22593690 | 1.85991335 | -0.000000 + 3.83433i | -0.000000 - 3.83433i |
| 1.13 | 2.40216210 | 6.46403612 | 1.83561433 | -0.000000 + 3.95553i | -0.000000 - 3.95553i |
| 1.14 | 2.55068148 | 6.73445533 | 1.81040907 | 0.000000 + 4.08330i | 0.000000 - 4.08330i |
| 1.15 | 2.72013973 | 7.04480009 | 1.78415860 | 0.000000 + 4.21948i | 0.000000 - 4.21948i |
| 1.16 | 2.91610161 | 7.40553517 | 1.75668703 | 0.000000 + 4.36653i | 0.000000 - 4.36653i |
| 1.17 | 3.14647596 | 7.83153594 | 1.72776686 | 0.000000 + 4.52781i | 0.000000 - 4.52781i |
| 1.18 | 3.42296420 | 8.34481267 | 1.69709568 | -0.000000 + 4.70817i | -0.000000 - 4.70817i |
| 1.19 | 3.76378002 | 8.97962556 | 1.66425726 | 0.000000 + 4.91487i | 0.000000 - 4.91487i |
| 1.20 | 4.19921248 | 9.79294705 | 1.62865180 | -0.000000 + 5.15958i | -0.000000 - 5.15958i |
| 1.21 | 4.78438373 | 10.88845375 | 1.58935920 | -0.000000 + 5.46267i | -0.000000 - 5.46267i |
| 1.22 | 5.63364308 | 12.48120508 | 1.54483421 | 0.000000 + 5.86431i | 0.000000 - 5.86431i |
| 1.23 | 7.03982023 | 15.12189448 | 1.49208056 | 0.000000 + 6.46074i | 0.000000 - 6.46074i |
| 1.24 | 10.13785496 | 20.94474978 | 1.42347076 | -0.000000 + 7.58875i | -0.000000 - 7.58875i |
| 1.25 | 55.81564546 | 106.83457688 | 1.28359273 | 0.000000 + 16.92849i | 0.000000 - 16.92849i |
| 1.26 |  |  |  |  |  |
| 1.27 |  |  |  |  |  |
| 1.28 |  |  |  |  |  |
| 1.29 |  |  |  |  |  |
| 1.30 |  |  |  |  |  |

* Красный цвет – точки, в которых дискриминант меньше 0 и в которых, следовательно, нет решения для
* Жёлтый – точки, в которых пара собственных значений равны по модулю, но разные по знаку и которые не подходят нам
* Зелёный – точки, в которых пара собственных значений – чисто мнимая пара корней с нулевой вещественной частью и которые подходят нам

*Построенная по полученным точкам бифуркационная диаграмма*



# Проверка

*Проверка результатов системы из 3-х уравнений на примере точек:*

Погрешность обусловлена погрешностью вычислений языка Python

*Проверка для определения точек, для которых пара собственных значений матрицы – чисто мнимая пара корней с нулевой вещественной частью:*

Квадратное уравнение относительно :

Вычисления будем производить программно

Для

– чисто мнимая пара корней с нулевой вещественной частью – значит точки подходят нам

Для

– равны по модулю, но разные по знаку и вещественная часть не нулевая – точки нам не подходят

# Вывод

В данной работе были найдены точки бифуркации Андронова-Хопфа и по ним построена бифуркационная диаграмма.

Так же была проведена проверка. Так как условие: «Сумма собственных значений равна 0» является необходимым условием, но не достаточным, поэтому были найдены собственные значения матрицы в каждой полученной точке, и если собственные значения были чисто мнимой парой корней с нулевой или почти нулевой вещественной частью – эти нужные точки.

# Список литературы

1. С.М.Устинов, В.А.Зимницкий. Вычислительная математика. – СПб.: БХВ-Петербург, 2009. – 336с. – (Учебное пособие.);

2. Форсайт Дж., Малькольм М., Моулер К. Машинные методы математических вычислений. – М.: Мир, 1980. – 280с.