Dominik Ciesiołkiewicz 44289

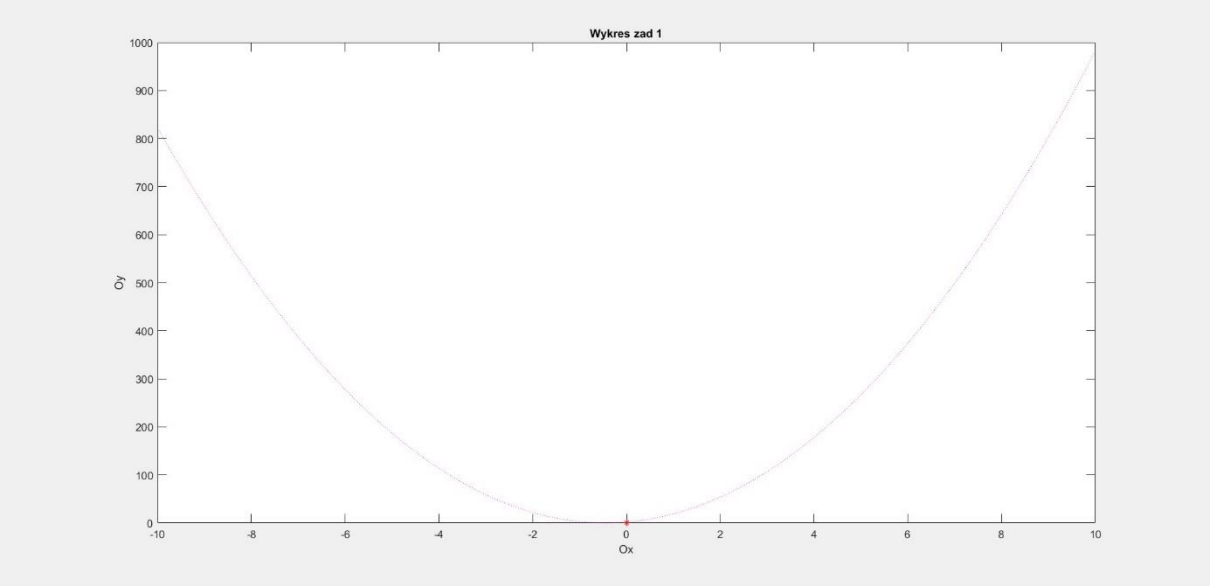
Transmisja danych Lab 1

Zad 1

Kod źródłowy:

```
#include <iostream>
#include <fstream>

 using namespace std;

int main()
{
    float a = 9;
    float b = 8;
    float c = 2;

    float det = b * b - 4 * a * c;
    cout << "Wyznacznik: " << det << "." << endl;
    if (det < 0)
    {
        cout << "Brak miejsc zerowych." << endl;
    }
    else if(det == 0)
    {
        float t = -b / 2 * a;

        cout << "Istnieje jedno miejsce zerowe t=" << t << "." << endl;
    }
    else
    {
        float t1 = (-b + sqrt(det)) / 2 * a;
        float t2 = (-b - sqrt(det)) / 2 * a;

        cout << "Istnieją dwa miejsca zerowe t1=" << t1 << ", t2=" << t2 << "." << endl;
    }

    //Do wykresu******************
    ofstream save("data.txt");

    for (double i = -10; i < 10; i = i + 0.01)
    {
        //cout << i << endl;
        float fun = a * i * i + b * i + c;
        //cout << fun << endl;
        save << fun << endl;
    }

    save.close();
}
```

Wykres:



Zad 2

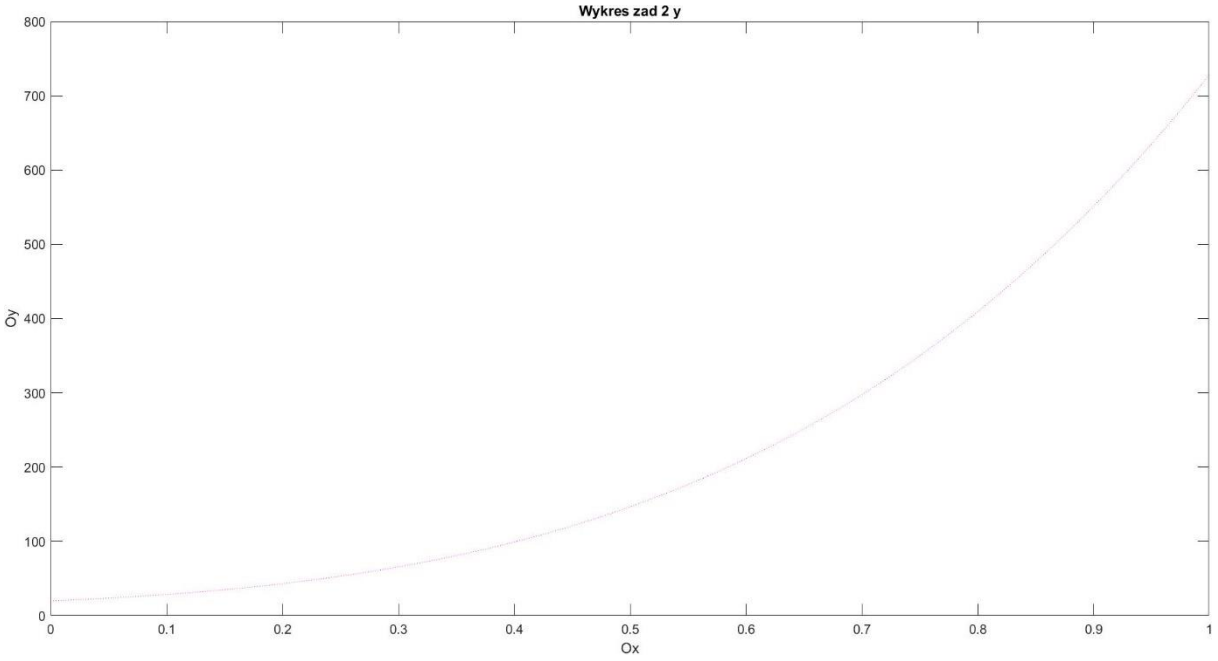
Kod źródłowy:

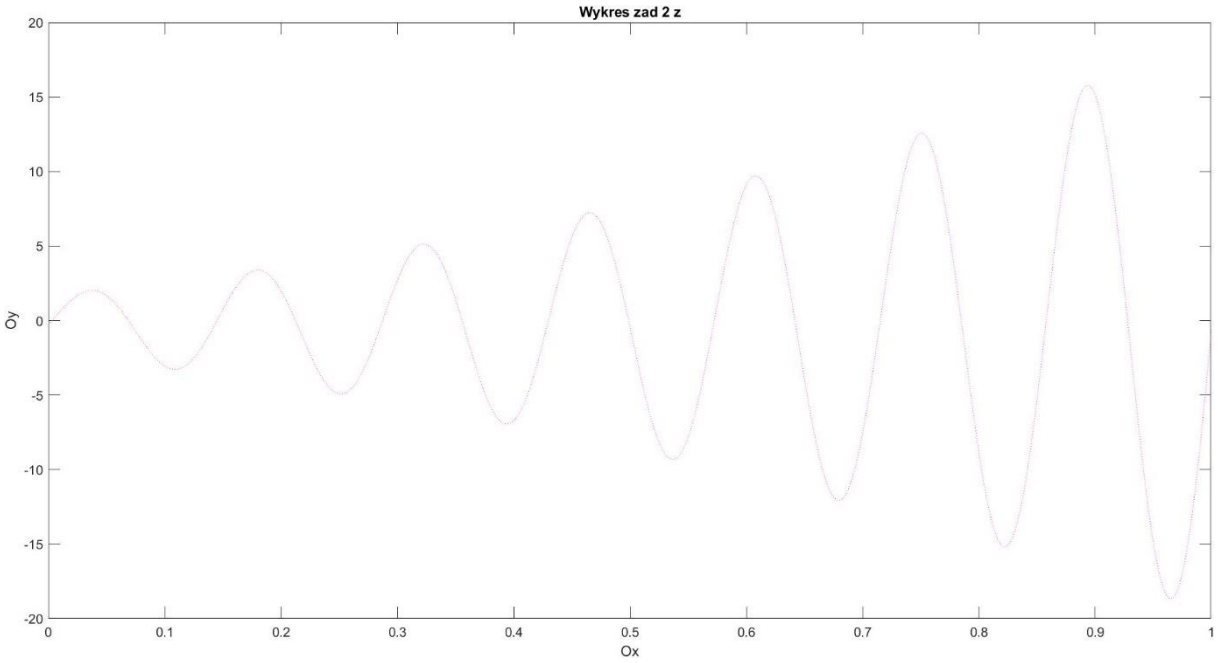
```cpp
#include <iostream>
#include <fstream>
#include <cmath>

using namespace std;

int main()
{
    int a = 9;
    int b = 8;
    int c = 2;

    float pi = 3.14159265359;

    //Funkcje y, z, u

    ofstream savey("data_y.txt");
    ofstream savez("data_z.txt");
    ofstream saveu("data_u.txt");
    ofstream saveOX("data_OX.txt");
    for (double i = 0; i <= 1; i = i + 1./22050)
    {
        //cout << i << endl;
        float x = a * i * i + b * i + c;
        float y = 2 * x * x + 12 * cos(i);
        savey << y << endl;
        float z = sin(2 * pi * 7 * i) * x - 0.2 * log10(abs(y) + pi);
        savez << z << endl;
        float u = sqrt(abs(y * y * z)) - 1.8 * sin(0.4 * i * z * x);
        saveu << u << endl;
        saveOX << i << endl;
    }

    savey.close();
    savez.close();
    saveu.close();
```

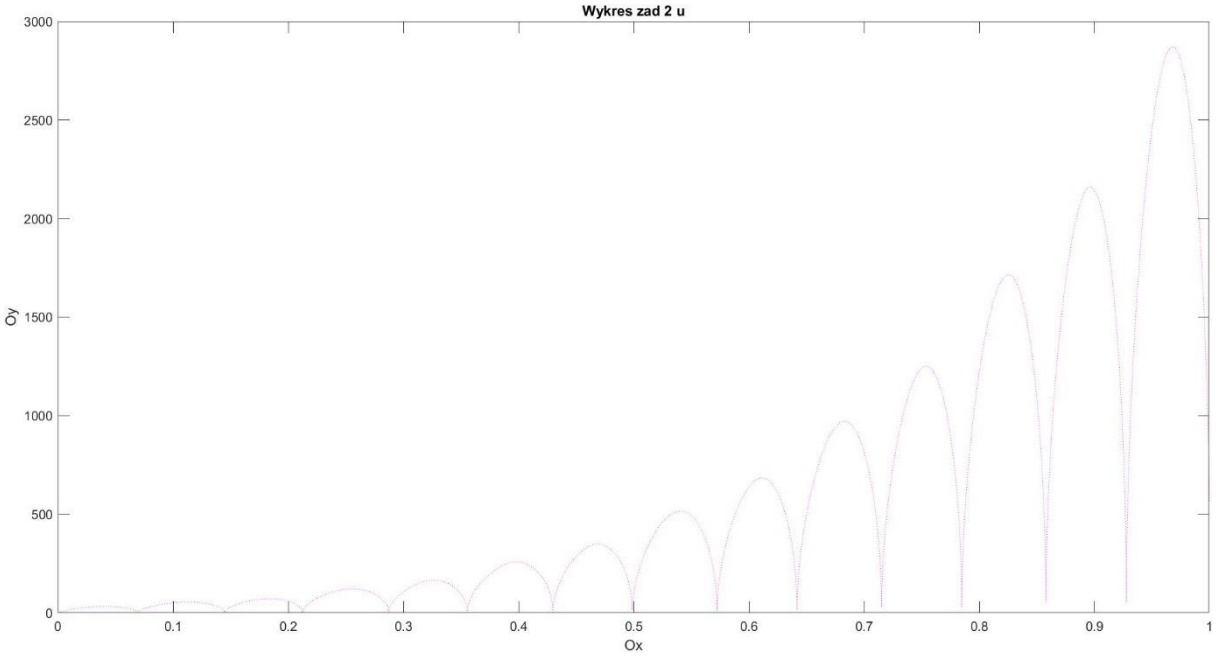
```
36      saveu.close();
37      saveOX.close();
38
39      //Funkcja v
40
41      ofstream savev("data_v.txt");
42
43      for (double i = 0; i <= 1; i = i + 1./22050)
44      {
45          float v;
46          if (i < 0.22)
47              v = (1 - 7 * i) * sin((2 * pi * i * 10) / (i + 0.04));
48          else if (i < 0.7)
49              v = 0.63 * i * sin(125 * i);
50          else
51              v = pow(i, -0.662) + 0.77 * sin(8 * i);
52
53          savev << v << endl;
54      }
55
56      savev.close();
57
58      //Funkcja p
59
60      ofstream savep("data_p.txt");
61
62      //int N = 2;
63      //int N = 4;
64      int N = 98;
65
66      for (double i = 0; i <= 1; i = i + 1. / 22050)
67      {
68          float p = 0;
69          for (int n = 1; n < N; n++)
70          {
71              p +=(cos(12 * i * n * n) + cos(16 * i * n)) / (n * n);
```

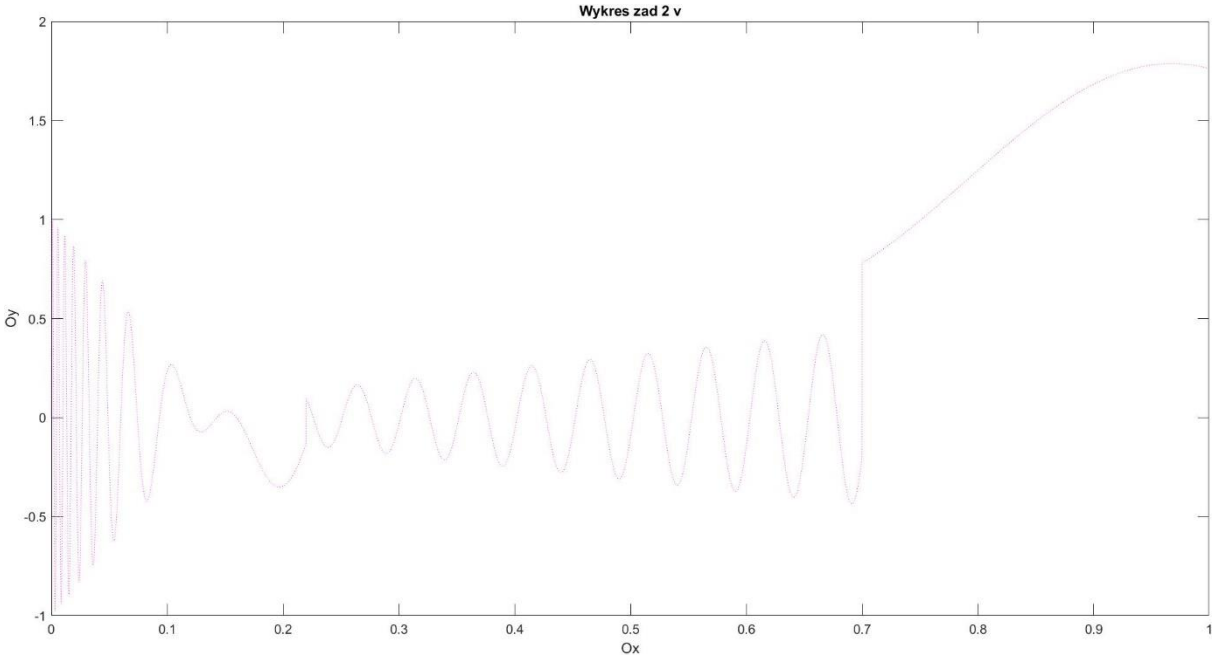
```
71              p +=(cos(12 * i * n * n) + cos(16 * i * n)) / (n * n);
72          }
73          savep << p << endl;
74      }
75
76      savep.close();
77  }
78
```

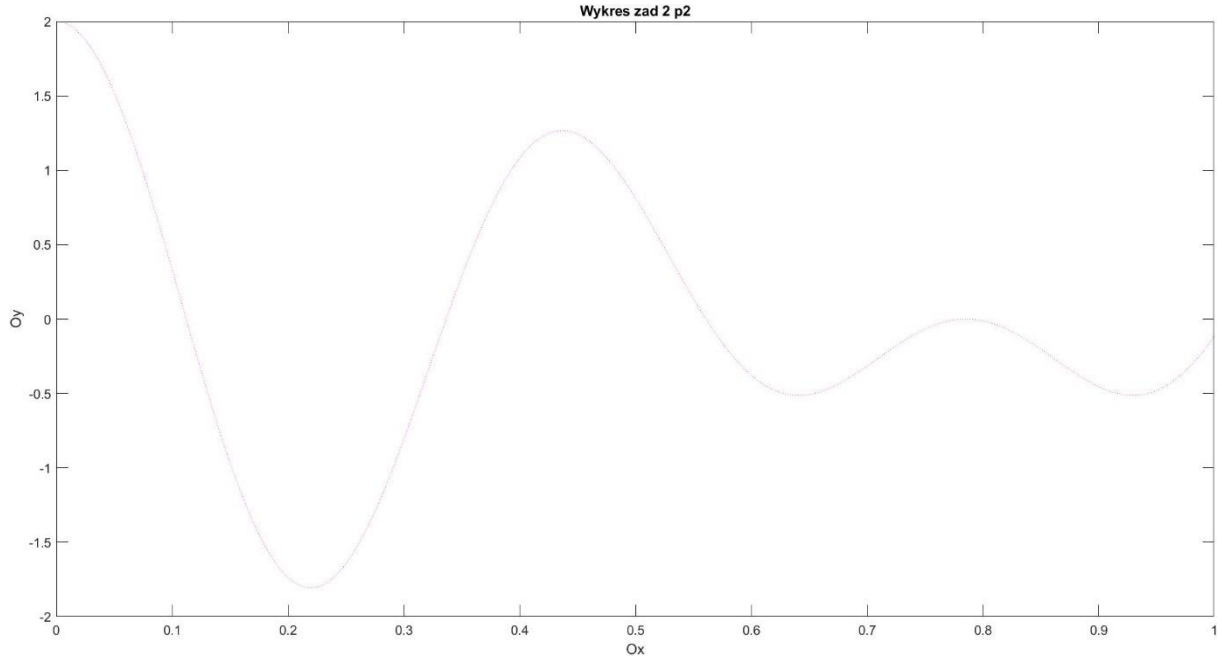
Wykresy:

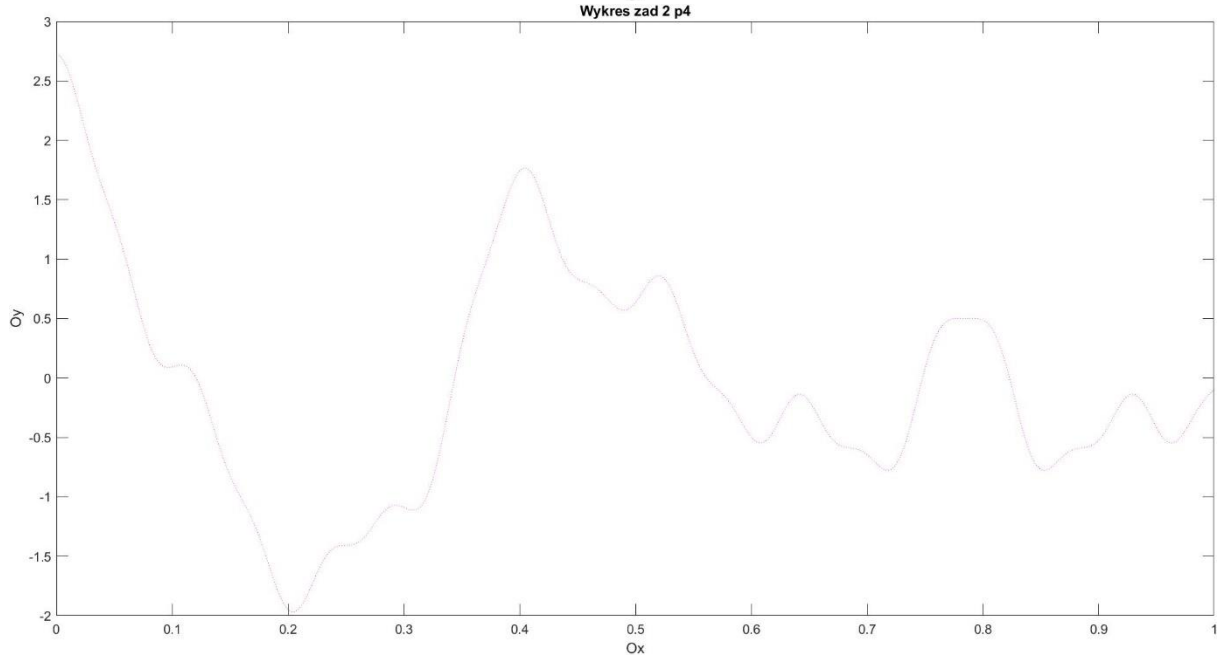

Wykres zad 2 y
800
700
600
500
400
300
200
100
0
Oy
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Ox

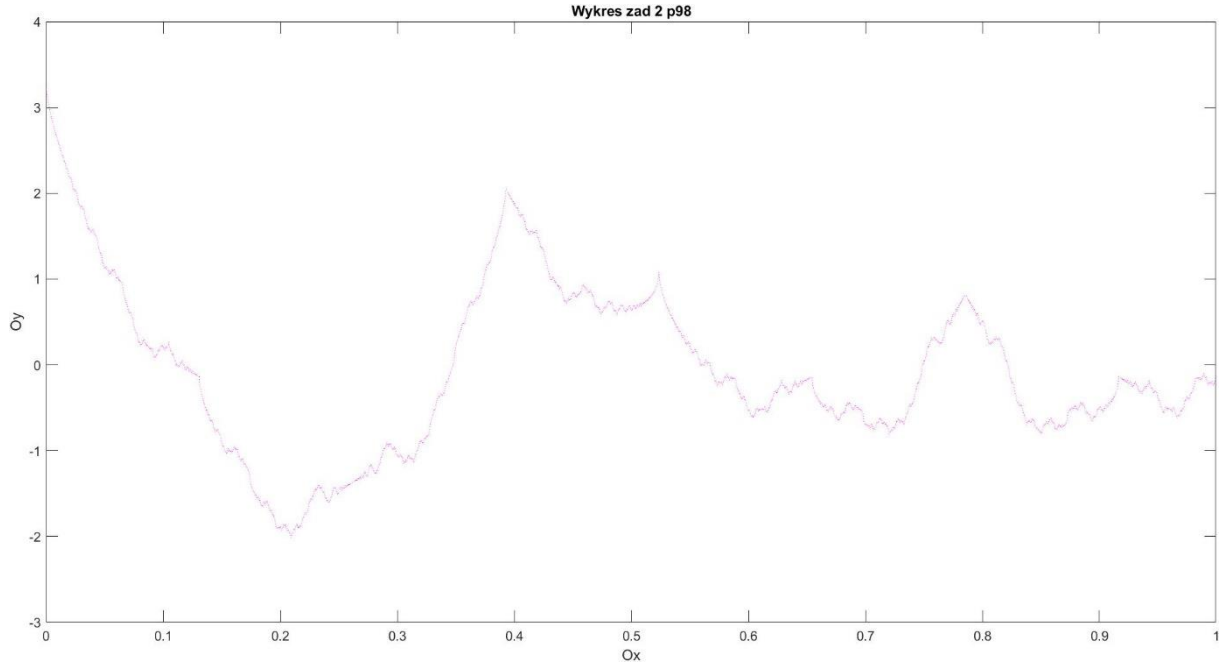

Wykres zad 2 z
20
15
10
5
0
-5
-10
-15
-20
Oy
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Ox


Wykres zad 2 u
3000
2500
2000
1500
1000
500
0
Oy
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Ox


Wykres zad 2 v
2
1.5
1
0.5
0
-0.5
-1
Oy
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Ox


Wykres zad 2 p2
Oy
Ox
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Wykres zad 2 p4
Oy
Ox
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Wykres zad 2 p98
4
3
2
1
0
-1
-2
-3
Oy
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Ox