

# Лабораторная работа

## Аналитическое построение уравнения границы сложной области

Волкова Юлия

16 Января, 2016

# 1 Постановка задачи

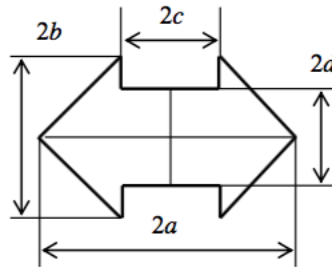
- Для области, изображённой на схеме, построить нормализованное уравнение границы в неявной форме:

$$F(x, y) = 0$$

Визуализировать линии уровня функции  $z = f(x, y)$ .

- Построить «функцию склейки»  $U(x, y)$ , принимающую значения -1 и 1 на участках границы  $-$  и  $+$ , расположенных слева и справа от вертикальной оси симметрии области соответственно. Визуализировать линии уровня функции  $z = U(x, y)$ .

Вариант 1, условия задачи:



**1**

Рис. 1 – Схема области

Параметры  $a, b, c, d$  определены следующим образом:

$$a = 8$$

$$b = 5$$

$$c = 3$$

$$d = 4$$

## 2 Решение задачи

Разобьём функцию  $F$  на несколько независимых функций, которые впоследствии объединим в единую посредством следующих правил:

$$x \vee y = x + y + \sqrt{x^2 + y^2}$$

$$x \wedge y = x + y - \sqrt{x^2 + y^2}$$

Будем рассматривать функции, которые внутри области  $F$  принимают положительное значение, и нулевое на границе и за границами области.

Для образования двух границ будущей прямоугольной границы, выберем следующие функции:

$$w_{11} = c - x + |x - c|$$

$$w_{12} = c + x + |x + c|$$

Применим операцию пересечения функций  $w_{11}$  и  $w_{12}$ , получим следующую трёхмерную поверхность:

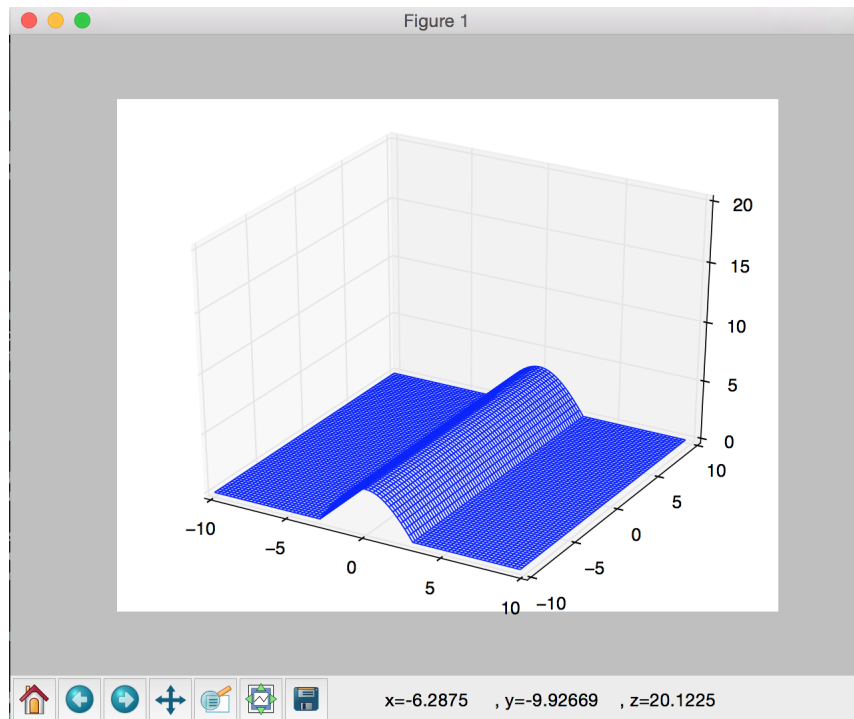


Рис. 2 – Результат пересечения функций  $w_{11}$  и  $w_{12}$

Для образования двух других границ будущей прямоугольной границы, выберем сле-

дующие функции:

$$w_{21} = d - y + |y - d|$$

$$w_{22} = d + y + |y + d|$$

Применим операцию пересечения функций  $w_{21}$  и  $w_{22}$ , получим следующую трёхмерную поверхность:

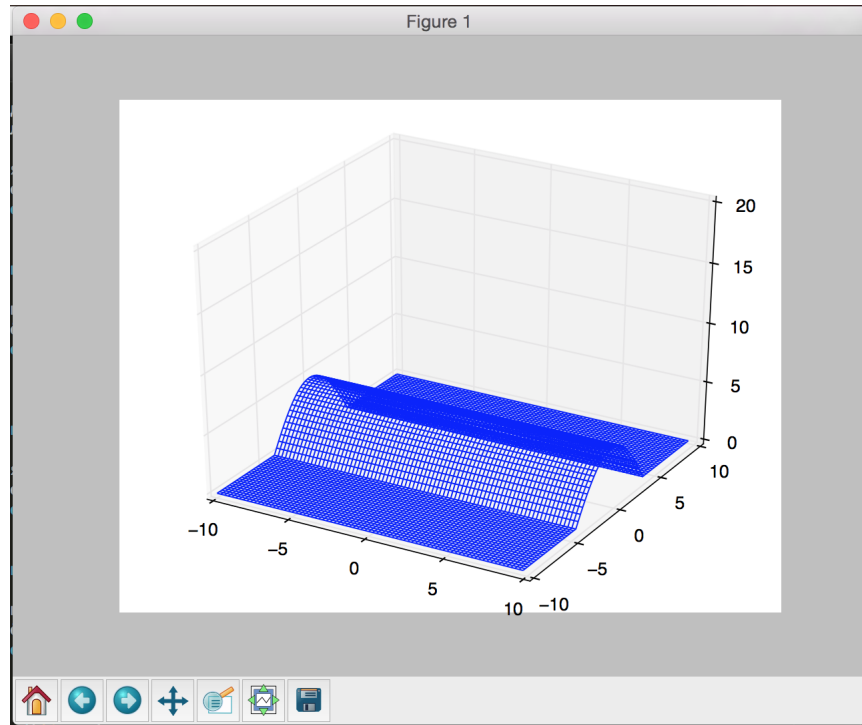


Рис. 3 – Результат пересечения функций  $w_{21}$  и  $w_{22}$

Для построения треугольных областей выберем следующие функции:

$$w_{31} = -y + \frac{b(a - |x|)}{a - c} + \left| y - \frac{b(a - |x|)}{a - c} \right|$$

$$w_{32} = +y + \frac{b(a - |x|)}{a - c} + \left| y - \frac{b(a + |x|)}{a - c} \right|$$

Применим операцию пересечения функций  $w_{31}$  и  $w_{32}$ , получим следующую трёхмерную поверхность:

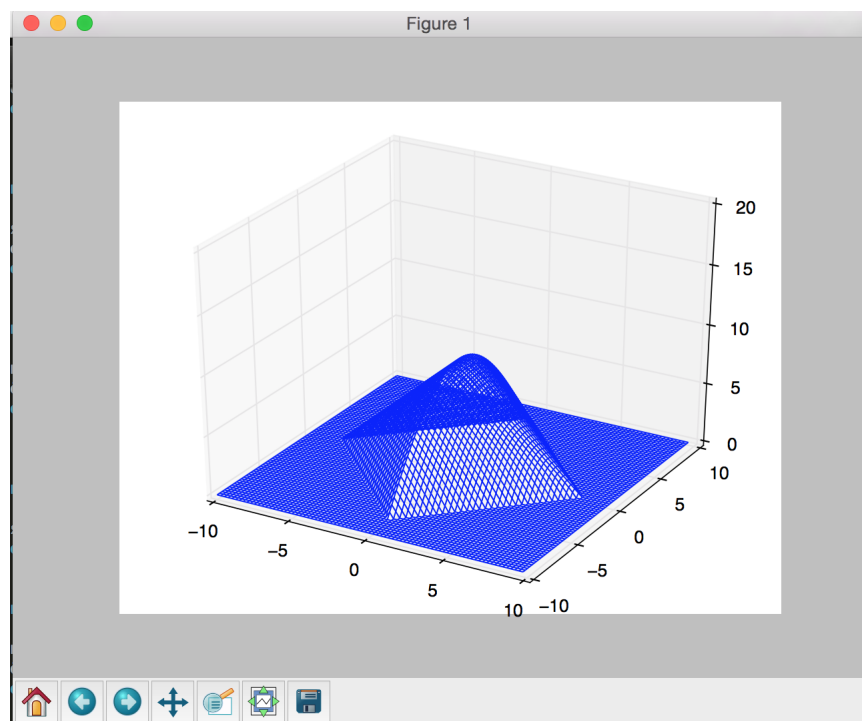


Рис. 4 – Результат пересечения функций  $w_{31}$  и  $w_{32}$

Для ограничения треугольников выберем следующие функции:

$$w_{41} = a - x + |x - c|$$

$$w_{42} = a + x + |x + c|$$

$$w_{43} = |x| - c + ||x| - c|$$

$$w_{44} = |x| + c + ||x| + c|$$

Применим операцию пересечения функций  $w_{41}, w_{42}, w_{43}, w_{44}$ , получим следующую трёхмерную поверхность:

Формула функции  $F$  будет иметь следующий вид:

$$F = (w_1 \wedge w_2) \vee (w_3 \wedge w_4)$$

Получим следующую трёхмерную поверхность:

Формула функции  $U$  будет иметь следующий вид:

$$U = 0, w(x, y) \leq 0 \quad U = \text{sign}(x), w(x, y) > 0$$

Получим следующую трёхмерную поверхность:

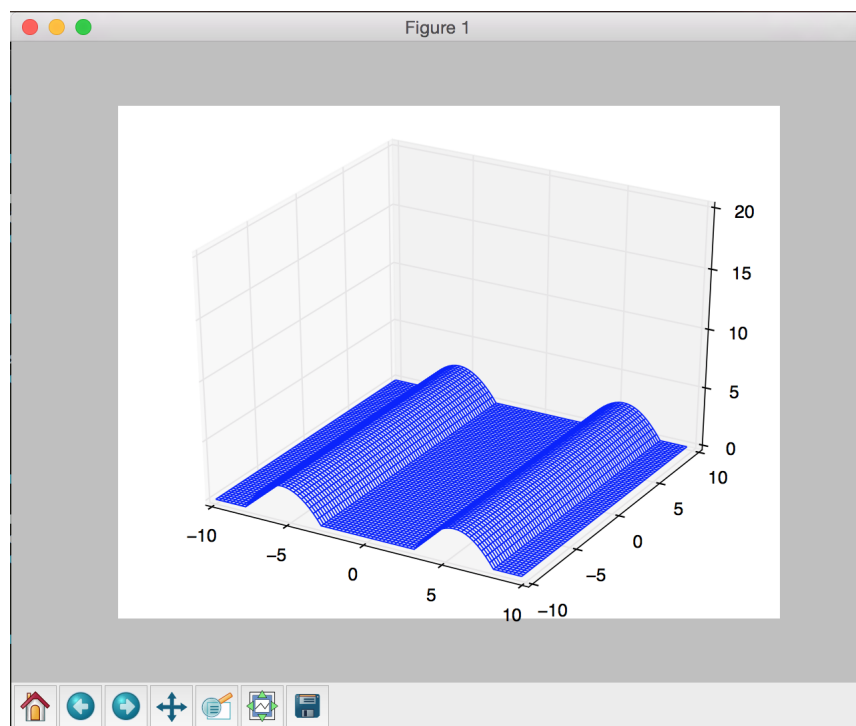


Рис. 5 – Результат пересечения функций  $w_{41}, w_{42}, w_{43}, w_{44}$

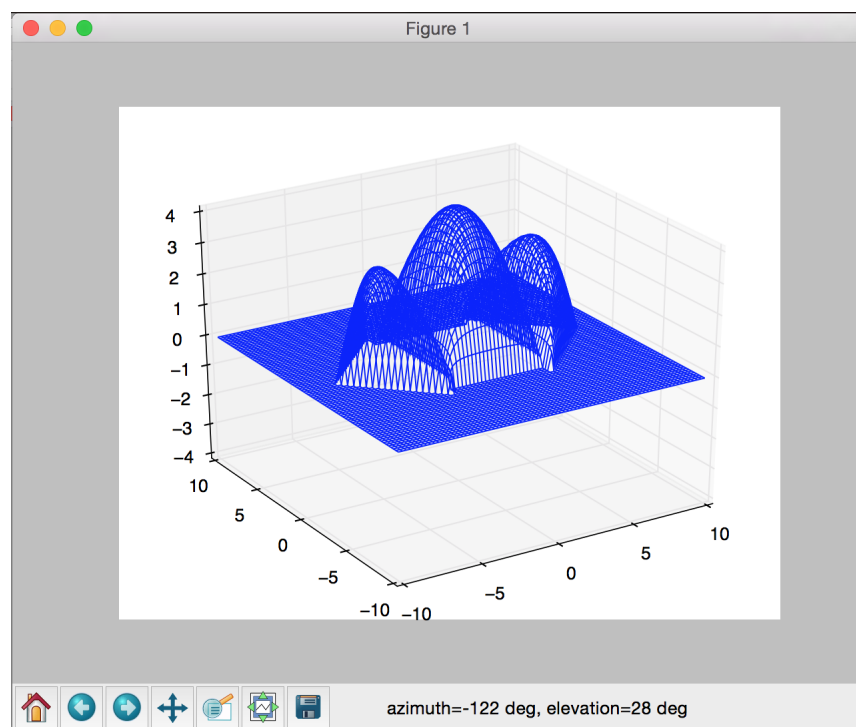


Рис. 6 – Функция  $F$

### 3 Код программы

```
import math, numpy, matplotlib.pyplot as plt
```

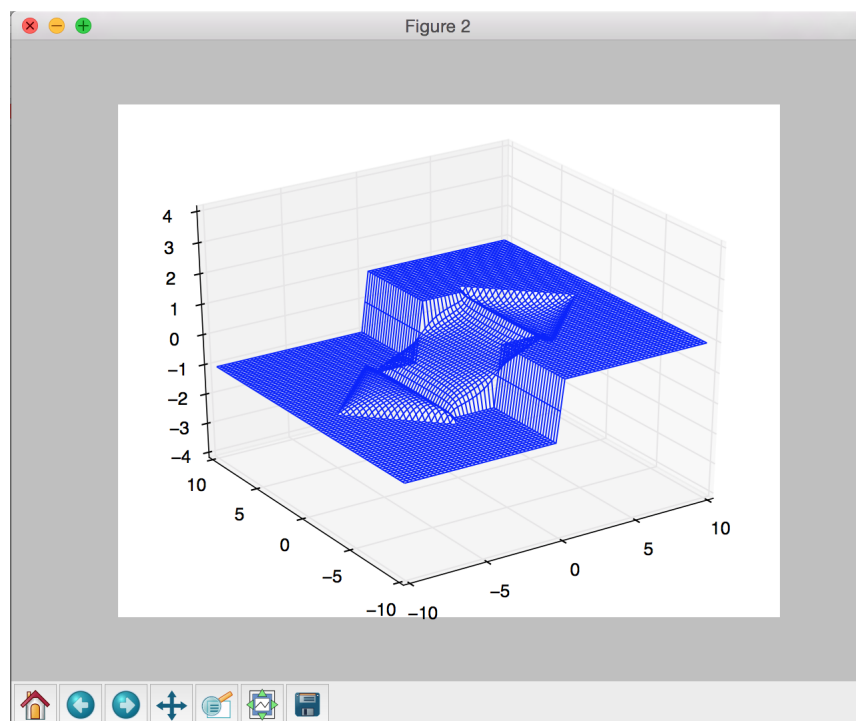


Рис. 7 – Функция  $U$

```
from mpl_toolkits.mplot3d import Axes3D
```

```
from matplotlib import cm
```

```
v = lambda w1, w2: lambda x, y: w1(x, y) + w2(x, y) + \
    math.sqrt(w1(x, y)**2 + w2(x, y)**2)
```

```
n = lambda w1, w2: lambda x, y: w1(x, y) + w2(x, y) - \
    math.sqrt(w1(x, y)**2 + w2(x, y)**2)
```

```
o = lambda w: lambda x, y: 0 if w(x, y) != 0 else 1
```

```
a = 8
```

```
b = 5
```

```
c = 3
```

```
d = 4
```

```
w11 = lambda x, y: -x + (c + 0.1) + abs(x - (c + 0.1))
```

```
w12 = lambda x, y: +x + (c + 0.1) + abs(x + (c + 0.1))
```

```
w1 = n(w11, w12)
```

w21 = **lambda** x, y: -y + d + **abs**(y - d)

w22 = **lambda** x, y: +y + d + **abs**(y + d)

w2 = n(w21, w22)

w31 = **lambda** x, y: -y + b \* (a - **abs**(x)) / (a - c) + \
**abs**(y - b \* (a - **abs**(x)) / (a - c))

w32 = **lambda** x, y: +y + b \* (a - **abs**(x)) / (a - c) + \
**abs**(y + b \* (a - **abs**(x)) / (a - c))

w3 = n(w31, w32)

w41 = **lambda** x, y: -x + a + **abs**(x - a)

w42 = **lambda** x, y: +x + a + **abs**(x + a)

w43 = **lambda** x, y: -c + **abs**(x) + **abs**(**abs**(x) - c)

w44 = **lambda** x, y: +c + **abs**(x) + **abs**(**abs**(x) + c)

w4 = n( n(w41, w42), n(w43, w44) )

w = v( n(w1, w2), n(w3, w4) )

Fp = **lambda** F: v( (**lambda** x, y: -x), F )

Fm = **lambda** F: v( (**lambda** x, y: +x), F )

U = **lambda** x, y: 0 **if** ( Fp(w)(x, y) + Fm(w)(x, y) ) == 0 \
**else** ( -Fp(w)(x, y) + Fm(w)(x, y) ) / \
( Fp(w)(x, y) + Fm(w)(x, y) )

**def** getFigure( left , right , dx ):

    X = numpy.arange( left , right , dx )

    Y = numpy.arange( left , right , dx )

    X, Y = numpy.meshgrid(X, Y)



```

Z = [[ w(xi , yi) for xi , yi in zip(x, y) ] for x, y in zip(X, Y) ]
return X, Y, Z

```

```

def getU( left , right , dx ):
    X = numpy.arange( left , right , dx )
    Y = numpy.arange( left , right , dx )
    X, Y = numpy.meshgrid(X, Y)

```

```

Z = [[ U(xi , yi) for xi , yi in zip(x, y) ] for x, y in zip(X, Y) ]
return X, Y, Z

```

```

def draw(x, y, z):
    fig = plt.figure()
    ax = fig.gca(projection='3d')
    ax.set_zlim(-4.01, 4.01)

    surf = ax.plot_wireframe(x, y, z, rstride=1, cstride=1)

```

```

if __name__ == '__main__':
    x, y, z = getFigure( -10, 10, 0.25 )
    draw(x, y, z)

    x, y, z = getU( -10, 10, 0.25 )
    draw(x, y, z)

    plt.show()

```