Инженерная школа природных ресурсов

Направление подготовки Химическая технология

Отделение химической инженерии

**PYTHON ДЛЯ ЗАДАЧ ХИМИЧЕСКОЙ ТЕХНОЛОГИИ**

**Отчет по лабораторной работе № 4**

**Минимизация функций**

Выполнил студент гр. 2ДМ24 Баранова Д.И.

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Томск 2023 г.

**Задание 1**

Найдите минимум следующих функций, используя методы минимизации, доступные в функции scipy.optimize.minimize().

Начальное приближение: x0 = [0, 0].

1. Функция Экли:

2. Функция Била:

3. Функция Гольдшейна-Прайса:

Функция Матьяса:

**Программная реализация:**

from scipy.optimize import minimize

import numpy as np

def ackley\_function(xy):

x, y = xy

return -20 \* np.exp(-0.2 \* np.sqrt(0.5 \* (x\*\*2 + y\*\*2))) - np.exp(0.5 \* (np.cos(2 \* np.pi \* x) + np.cos(2 \* np.pi \* y))) + np.e + 20

def beales\_function(xy):

x, y = xy

return (1.5 - x + x \* y)\*\*2 + (2.25 - x + x \* y\*\*2)\*\*2 + (2.625 - x + x \* y\*\*3)\*\*2

def goldstein\_price\_function(xy):

x, y = xy

return (1 + (x + y + 1)\*\*2 \* (19 - 14 \* x + 3 \* x\*\*2 - 14 \* y + 6 \* x \* y + 3 \* y\*\*2)) \* (30 + (2 \* x - 3 \* y)\*\*2 \* (18 - 32 \* x + 12 \* x\*\*2 + 48 \* y - 36 \* x \* y + 27 \* y\*\*2))

def matyas\_function(xy):

x, y = xy

return 0.26 \* (x\*\*2 + y\*\*2) - 0.48 \* x \* y

x0 = [0, 0]

result\_ackley = minimize(ackley\_function, x0, method='Nelder-Mead')

result\_beales = minimize(beales\_function, x0, method='Nelder-Mead')

result\_goldstein\_price = minimize(goldstein\_price\_function, x0, method='Nelder-Mead')

result\_matyas = minimize(matyas\_function, x0, method='Nelder-Mead')

print("Ackley Function Minimum:", result\_ackley.x)

print("Beale's Function Minimum:", result\_beales.x)

print("Goldstein-Price Function Minimum:", result\_goldstein\_price.x)

print("Matyas Function Minimum:", result\_matyas.x)

**Ответ**:

Ackley Function Minimum: [0. 0.]

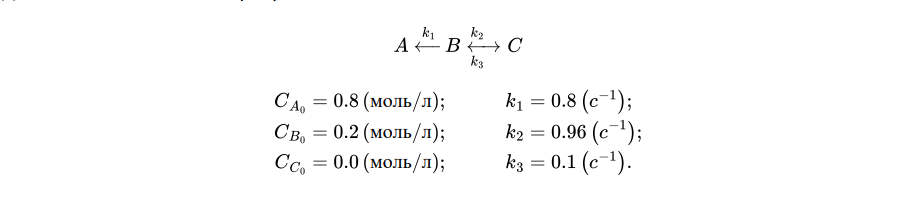
Beale's Function Minimum: [2,99994196 0,49998485]

Goldstein-Price Function Minimum: [-0,59995602 -0,40003256]

Matyas Function Minimum: [0. 0.]

**Задание 2**

Пусть дана схема химических превращений:



Необходимо определить с помощью генетического алгоритма и метода Нелдера-Мида (можно воспользоваться функцией scipy.optimize.mnimize() , указав соответствующее значение опционального аргумента method ) константы скоростей реакций: k1, k2 и k3, если известно, что к моменту времени t = 1(c) концентрации компонентов равны: CA = 0,1423; CB = 1,5243; CC = 0,5956 моль/л. Начальные условия: CA(0) = 1,0; CB = 0.0; CC = 0,5 моль/л. Область поиска для всех констант ограничьте интервалом [0; 2].

**Программная реализация:**

from scipy.integrate import solve\_ivp

from scipy.optimize import minimize

import numpy as np

def chemical\_system(t, concentrations, k1, k2, k3):

CA, CB, CC = concentrations

dCAdt = -k1 \* CA

dCBdt = 2 \* k1 \* CA - 2 \* k2 \* CB\*\*2 + 2 \* k3 \* CC

dCCdt = k2 \* CB\*\*2 - k3 \* CC

return [dCAdt, dCBdt, dCCdt]

def objective\_function(rate\_constants):

k1, k2, k3 = rate\_constants

result = solve\_ivp(chemical\_system, [0, 1], [1.0, 0.0, 0.5], args=(k1, k2, k3))

CA, CB, CC = result.y[:, -1]

error = (CA - 0.1423)\*\*2 + (CB - 1.5243)\*\*2 + (CC - 0.5956)\*\*2

return error

initial\_guess = [1, 1, 1]

opt\_result = minimize(objective\_function, initial\_guess, method='Nelder-Mead')

print("Optimized Rate Constants:", opt\_result.x)e)

plt.show()

**Ответ**:

Optimized Rate Constants: [1,95009837 0,33183658 0,83200744]