

**МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра САПР**

ЗАДАНИЕ
по лабораторной работе №2
по дисциплине «Методы оптимизации»
Тема: «Методы одномерной оптимизации на основе поиска
стационарной точки»

Студенты:

Литвинов К.Л.

Гарцев Е.А.

Бурков М.П.

Преподаватель:

Каримов А.И.

Цель работы

Изучение среды MATLAB, создание программы для реализации одного из методов одномерного поиска на основе поиска стационарной точки:

- Метод секущих;

Основные теоретические положения

Критические и стационарные точки функции определяются следующим образом.

Критические точки функции $f(x)$ – точки, в которых производная $f'(x)$ не существует или обращается в нуль.

Стационарные точки функции $f(x)$ – точки, в которых производная $f'(x)$ обращается в нуль.

При этом стационарные точки подразделяются на:

- экстремумы – точки минимума или максимума;
- седловые точки – точки, в которых производная нулевая, но минимум или максимум не достигается.

Лемма Ферма утверждает: производная $f'(x)$ дифференцируемой функции в точке экстремума равна нулю. В соответствии с этой леммой, возможно использования метода нахождения нуля производной в качестве метода оптимизации. Для этого осуществляются следующие шаги:

- 1) Поиск $x_i^* : f''(x^*) = 0$
- 2) Осуществляется проверка: x_i^* – экстремум, если

$$f'''(x^*) \neq 0 \quad (1)$$

или

$$f''(x^* - \epsilon)f''(x^* + \epsilon) < 0, \quad (2)$$

где $\epsilon > 0$ – малое число (взаимозаменяемые условия).

Метод секущих

Метод секущих предлагает заменить вторую производную $f''(x_k)$ в ньютоновской формуле её линейной аппроксимацией $(f'(x_k) - f'(x_{k-1})) / (x_k - x_{k-1})$. Тем самым,

$$x_{k+1} = x_k - \frac{f'(x_k)(x_k - x_{k-1})}{f'(x_k) - f'(x_{k-1})}.$$

Легко видеть, что x_{k+1} – точка пересечения с осью абсцисс секущей прямой, проходящей через точки x_k и x_{k-1} .

Псевдокод

Цикл

$$x_{k+1} = b_k - f'(b_k)(b_k - a_k) / (f'(b_k) - f'(a_k));$$

Если

$$|f'(x_{k+1})| \leq \epsilon, // \text{КОП}$$

то

остановиться

иначе // Уменьшить интервал поиска минимума

Если

$f'(x_{k+1}) > 0,$
то
 $a_{k+1} = a_k, b_{k+1} = x_{k+1},$
иначе
 $a_{k+1} = x_{k+1}, b_{k+1} = b_k;$
 $k = k + 1;$
Пока не выполнен КОП

Код программы

Листинг 1: Метод Больцано

```

1
2 function [xk, k] = Bolcano(dfunction, funct , a, b, tol)
3
4     % VISUALIZATION
5
6
7     format long g
8     t = a:0.1:b; k = 1;
9     epsilon = tol; delta = tol;
10
11    %Visualization
12    ctr = 1;
13    deltaX = (b-a)/100;
14    figure(3); hold on
15    [miny, maxy] = drawplot(funct,a,b,a,b);
16    deltaY = abs(maxy - miny)/100;
17    placelabel(a,0,deltaX,deltaY,ctr);
18    placelabel(b,0,deltaX,deltaY,ctr);
19    k = 0;
20    input("");
21    print('-djpg',[num2str(k), ' Bolcano itter'])
22    %Main algorithm
23    xk = (a + b) / 2;
24    while (abs(dfunction(xk)) >= epsilon) & (abs(b - a) >= delta)
25        if (dfunction(xk)) > 0
26            b = xk;
27        else
28            a = xk;
29        end
30        k = k + 1;
31        xk = (a + b) / 2;
32        drawplot(funct,a,b,xk,xk);
33        placelabel(xk,funct(xk),deltaX,deltaY, k);
34        print('-djpg',[num2str(k), ' Bolcano itter'])
35        input("");
36    end
37    hold off
38 end
39
40 function [miny maxy] = drawplot(f,a,b,x1,x2)

```

```

41     figure(3);
42     h = (b-a)/100;
43     x = a:h:b;
44     y = feval(f,x);
45
46     miny = min(y);
47     maxy = max(y);
48
49     colp = hsv2rgb([rand(), 1, 0.5+0.5*rand()]);
50     plot(x,y,'LineWidth',1,'Color',colp);
51     scatter([a b],[feval(f,a), feval(f,b)],'Marker','o','
        MarkerFaceColor',colp,'MarkerEdgeColor',colp);
52     xlabel('\itx');
53     ylabel('\ity');
54     line([a b],[0 0],'Color','k','LineWidth',1); %axis x
55     col = hsv2rgb([rand(), 1, 0.5+0.5*rand()]);
56     y1 = feval(f, x1);
57     line([x1 x1],[0 y1],'Marker','s','Color',col,'LineWidth',1,'
        MarkerSize',4);
58     %y2 = feval(f,x2);
59     %line([x2 x2],[0 y2],'Marker','s','Color',col,'LineWidth',1,'
        MarkerSize',4);
60 end
61
62 function placelabel(x,y,deltaX,deltaY,iternumber)
63     if iternumber <=10
64         text(x - deltaX/2, y + 4*deltaY, num2str(iternumber));
65     end
66 end

```

Графики, демонстрирующие работу метода Больцано

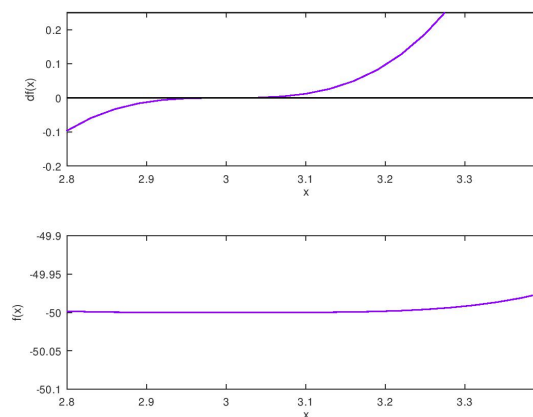


Рис. 1: инициализация

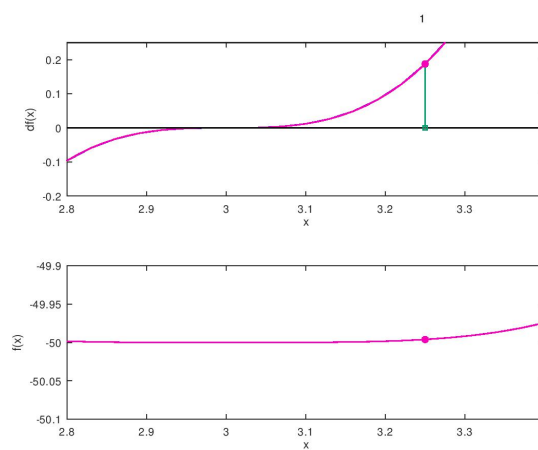


Рис. 2: Первая итерация

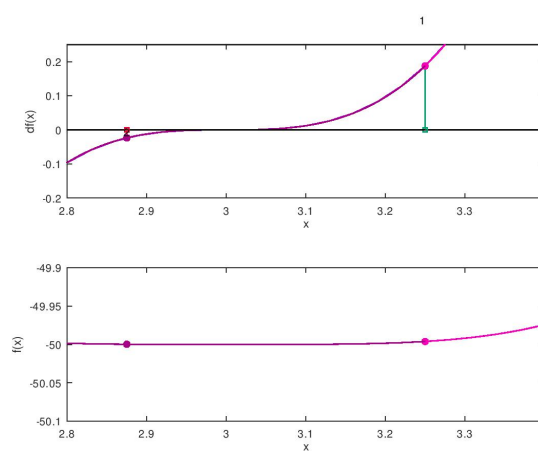


Рис. 3: Вторая итерация

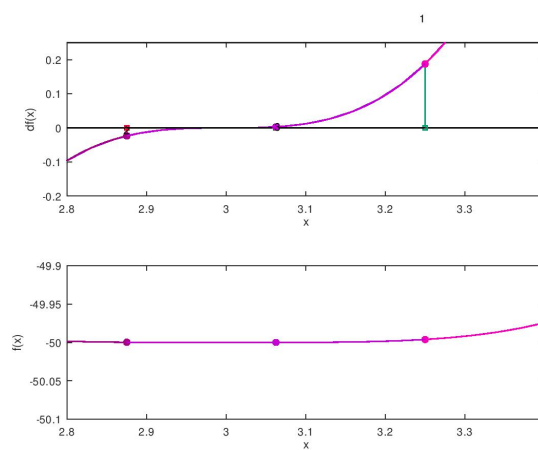


Рис. 4: Третья итерация

Код метода секущих

Листинг 2: Метод секущих

```
1
2 function [xk, k] = secant(dfunction, funct, a, b, tol)
3     epsilon = tol;
4
5     %Visualization
6     ctr = 1;
7     deltaX = (b-a) / 100;
8     figure(3); hold on
9     [miny, maxy] = drawplot(funct,a,b,a,b);
10    deltaY = abs(maxy - miny)/100;
11    placelabel(a,0,deltaX,deltaY,ctr);
12    placelabel(b,0,deltaX,deltaY,ctr);
13
14    %Main algorithm
15    while 1
16        xk = b - dfunction(b)*(b - a) / (dfunction(b) - dfunction(a)
17        ))
18        if dfunction(xk) <= epsilon
19            break
20        else
21            if dfunction(xk) > 0
22                b = xk
23            else
24                a = xk
25            end
26        end
27        drawplot(funct,a,b,xk,xk);
28        k = k + 1
29    end
30
31 function [miny maxy] = drawplot(f,a,b,x1,x2)
32     figure(3);
33     h = (b-a)/100;
34     x = a:h:b;
35     y = feval(f,x);
36
37     miny = min(y);
38     maxy = max(y);
39
40     colp = hsv2rgb([rand(), 1, 0.5+0.5*rand()]);
41     plot(x,y,'LineWidth',1,'Color',colp);
42     scatter([a b],[feval(f,a), feval(f,b)], 'Marker','o', '
43         MarkerFaceColor',colp, 'MarkerEdgeColor',colp);
44     xlabel('\itx');
45     ylabel('\ity');
46     line([a b],[0 0], 'Color','k','LineWidth',1); %axis x
47     col = hsv2rgb([rand(), 1, 0.5+0.5*rand()]);
```

```

47     y1 = feval(f, x1);
48     line([x1 x1],[0 y1], 'Marker','s','Color',col,'LineWidth',1, '
        MarkerSize',4);
49     y2 = feval(f,x2);
50     line([x2 x2],[0 y2], 'Marker','s','Color',col,'LineWidth',1, '
        MarkerSize',4);
51     input("");
52 end
53
54 function placelabel(x,y,deltaX,deltaY,iternumber)
55     if iternumber <=10
56         text(x - deltaX/2, y + 4*deltaY, num2str(iternumber));
57     end
58 end

```

Графики, демонстрирующие работу метода секущих

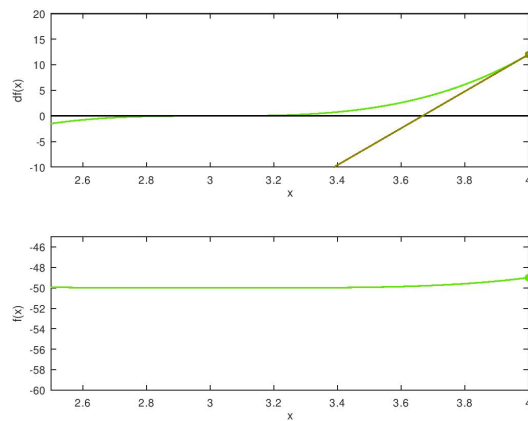


Рис. 5: инициализация

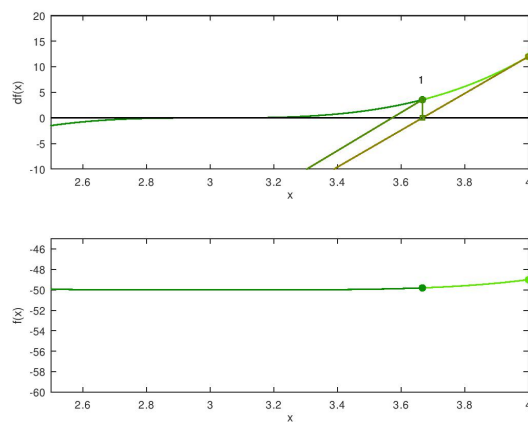


Рис. 6: Первая итерация

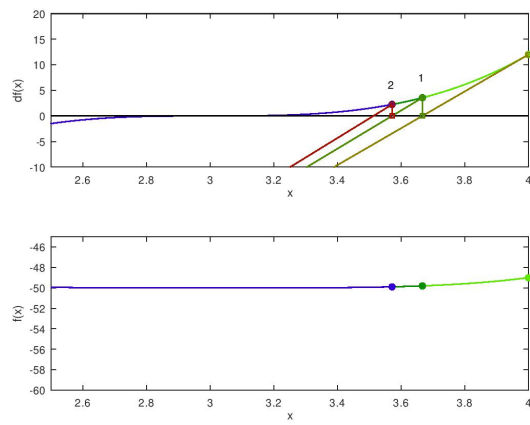


Рис. 7: Вторая итерация

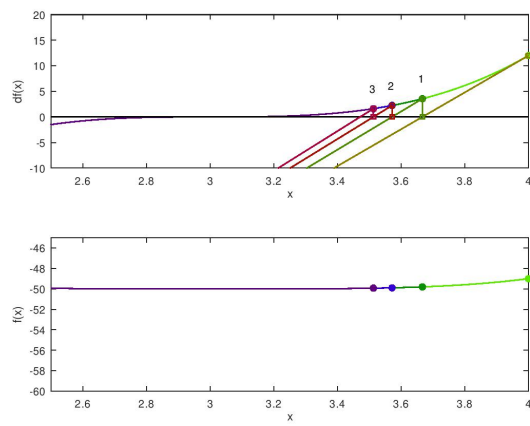


Рис. 8: Третья итерация

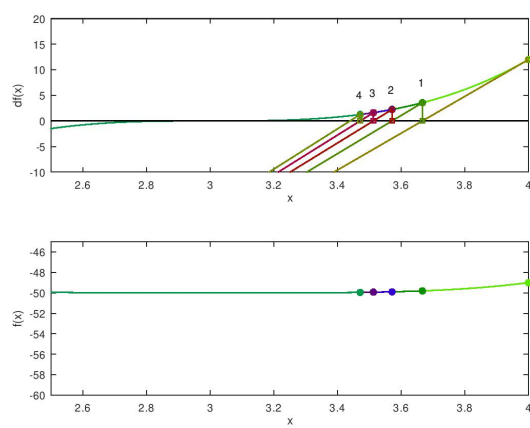


Рис. 9: Четвёртая итерация

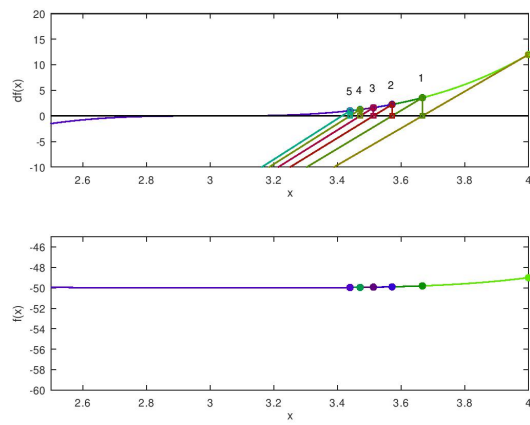


Рис. 10: Пятая итерация

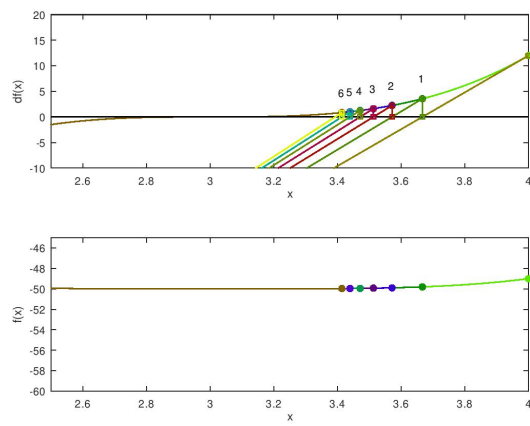


Рис. 11: Шестая итерация

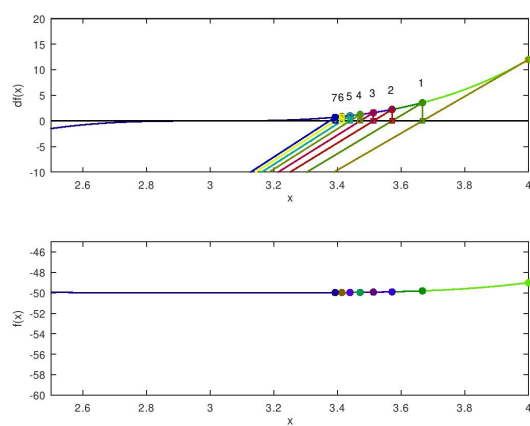


Рис. 12: Седьмая итерация

Вывод