МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Э. БАУМАНА

Факультет информатики и систем управления

Кафедра теоретической информатики и компьютерных технологий

Численные методы поиска условного экстремума

Исполнитель: Ю.А. Волкова

Группа: ИУ9-111

25 Декабря, 2016

1 Методы последовательной безусловной оптимизации. Метод штрафов

1.1 Постановка задачи

Даны дважды непрерывно дифференцируемые целевая функция $f(x) = f(x_1 \ldots x_n)$ и функции ограничений $g_j(x) = 0, j = 1 \ldots m$ и $g_j(x) \leq 0, j = m + 1 \ldots p$, определяющие множество допустимых решений X.

Требуется найти локальный минимум целевой функции на множестве X, т.е. такую точку $x^e \in X$, что

$$f(x^e) = \min_{x \in X} f(x)$$

$$\begin{cases} g_j(x) = 0, j = 1 \ldots m; m < n \\ g_j(x) \leq 0, j = m + 1 \ldots p \end{cases}$$

Где $f(x), g(x)$:

$$f(x) = x_1^2 + x_2^2 \to min$$

$$g_1(x) = -x_1 + 1 \leq 0$$

$$g_2(x) = x_1 + x_2 - 2 \leq 0$$

1.2 Код программы

```
from lsgnonlin import lsgnonlin


r = 0.01
eps = 0
C = 4


def f( x ):
    return x[0]**2 + x[1]**2
```

```python
def F( x ):
    return f( x ) + P( x )


def P( x ):
    return r * ( max(0, -x[0] + 1)**2 + max(0, x[0] + x[1] - 2)**2 ) / 2


def gradF( x ):
    g1  = 0 if x[0] >= 1 else r * x[0] - r
    g21 = 0 if x[0] + x[1] <= 2 else r * ( x[0] - 2 + x[1] )
    g22 = 0 if x[0] + x[1] <= 2 else r * ( x[1] - 2 + x[0] )

    return [ 2 * x[0] + g1 + g21, 2 * x[1] + g22 ]


def H( x ):
    g1 = 0 if x[0] >= 1 else 2 * r
    g2 = 0 if x[0] + x[1] <= 2 else r

    return [[2 + g1, g2], [g2, 2 + g2]]


def penalty( x ):
    global r

    xk, Fxk = lsgnonlin( F, gradF, H, x, 10**4, 0 )

    if P( xk ) <= eps:
        return xk, f( xk )

    r *= C
    return penalty( xk )


def main():
```

```
    x = [1, 1]

    print penalty( x )

if __name__ == '__main__':
    main()
```

1.3 Результат работы программы

Для поиска безусловного минимума используется программа, написанная для лабораторной работы 2.2.

В результате работы программы получена точка [1.0, 0.0]. Значение функции в этой точке: 1.0.

2 Метод модифицированных функций Лагранжа

2.1 Постановка задачи

Даны дважды непрерывно дифференцируемые целевая функция $f(x) = f(x_1 \dots x_n)$ и функции ограничений $g_j(x) = 0, j = 1 \dots m$ и $g_j(x) \leq 0, j = m+1 \dots p$, определяющие множество допустимых решений X.

Требуется найти локальный минимум целевой функции на множестве X, т.е. такую точку $x^e \in X$, что

$$f(x^e) = \min_{x \in X} f(x)$$

$$\begin{cases} g_j(x) = 0, j = 1 \dots m; m < n \\ g_j(x) \leq 0, j = m+1 \dots p \end{cases}$$

Где $f(x), g(x)$:

$$f(x) = x_1^2 + x_2^2 \to min$$

$$g_1(x) = -x_1 + 1 \leq 0$$

$$g_2(x) = x_1 + x_2 - 2 \leq 0$$

2.2 Код программы

```
from lsgnonlin import lsgnonlin


r = 0.01
eps = 0
C = 4
mu = [0, 0]
n = 2


def g1( x ):
    return -x[0] + 1
```

```
def g2( x ):
    return x[0] + x[1] - 2

g = [ g1, g2 ]

def f( x ):
    return x[0]**2 + x[1]**2

def L( x ):
    return f( x ) + P( x )

def P( x ):
    return sum([ max( 0, mi + r * gi(x) )**2 - mi**2 for mi, gi in zip( mu, g )

def gradL( x ):
    xg1  = 0 if mu[0] + r * g[0](x) <= 0 else - mu[0] + r * x[0] - r
    xg21 = 0 if mu[1] + r * g[1](x) <= 0 else mu[1] + r * x[0] + r * x[1] / 2 -
    xg22 = 0 if mu[1] + r * g[1](x) <= 0 else mu[1] + r * x[0] / 2 + r * x[1] -

    return [ 2 * x[0] + xg1 + xg21, 2 * x[1] + xg22 ]

def H( x ):
    xg1 = 0 if mu[0] + r * g[0](x) <= 0 else r
    xg21 = 0 if mu[1] + r * g[1](x) <= 0 else r / 2
    xg22 = 0 if mu[1] + r * g[1](x) <= 0 else r

    return [[2 + xg1 + xg22, xg21], [xg21, 2 + xg22]]

def penalty( x ):
    global r, mu
```

```
        xk, Lxk = lsgnonlin( L, gradL, H, x, 10**4, 0 )

        if abs( P( xk ) ) <= eps:
            return xk, f( xk )

        r *= C
        mu = [ max( 0, mi + r * gi( xk ) ) for mi, gi in zip( mu, g ) ]
    return penalty( xk )

def main():
    x = [1, 1]

    print penalty( x )

if __name__ == '__main__':
    main()
```

2.3 Результат работы программы

Для поиска безусловного минимума используется программа, написанная для лабораторной работы 2.2.

В результате работы программы получена точка [1.0, 0.0]. Значение функции в этой точке: 1.0.

3 Вывод

В качестве начальной точки бралась точка $x = [1, 1]$. Итоговый условный минимум, полученный методами: точка $x = [1.0, 0.0]$. Значение функции в этой точке $f(x) = 1.0$.