电子科技大学

课程实验报告

课程名称：最优化理论与应用

实验题目：约束优化

姓　　名：徐宇健

学　　号：202422280813

专　　业：085400 电子信息工程

学　　院：（深圳）高等研究院

完成日期：2024 年 11 月 11 日

实验目的：

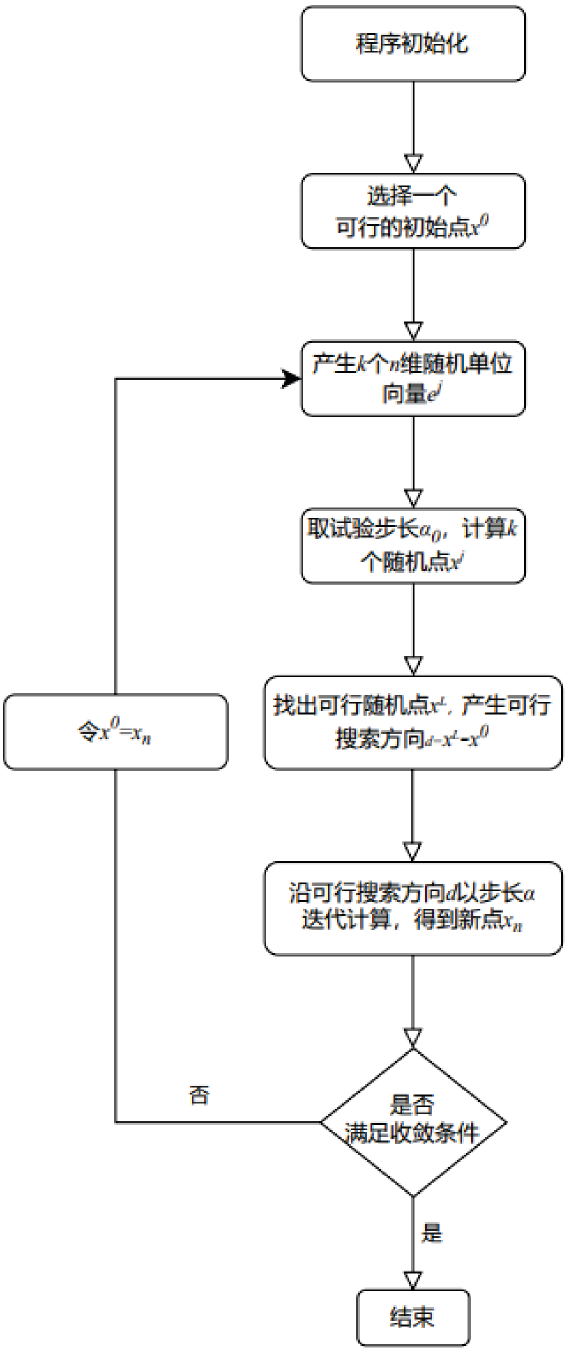
1. 以本次实验为案例，了解算法复现流程；

2. 掌握 matlab 的基本操作，学习编写 matlab 程序，提高编程能力和技巧；

3. 学习用已学的知识解决实际问题，将理论应用于实际。

算法步骤：

随机方向法：



Matlab 程序：

随机方向法（求解程序）：

```
%随机方向法
clc,clear
tic  %计时开始
%% 初始化
n=3;
a0=0.1; %试验步长
epsilon=1e-6;   %收敛精度
k=8;    %搜索方向个数
a=0;   %下限值
b=42;    %上限值
%% 选择初始点
while 1
    x0=a+(b-a)*rand(n,1);  %随机产生初始点
    g=gcon(x0);
    if all(g<=0)
        break;
    end
end
f0=fun(x0);
%%
while 1
    %% 产生可行搜索方向
    a0=0.1;
    r=zeros(n,k);   %随机方向
    e=zeros(n,k);   %单位化随机方向
    while 1
        j=1;    %可行点数目的计数
        for i=1:k
          r(:,i)=rands(n,1);
          e(:,i)=r(:,i)/norm(r(:,i));   %单位化随机方向
          xs=x0+a0*e(:,i);  %沿随机方向移动后的点
          gz=gcon(xs);
          if all(gz<=0)
```

```
            fs=fun(xs); %计算移动后可行点的目标函数值
            x(:,j)=xs;  %记录沿随机方向移动后的可行点
            fz(:,j)=fs; %记录沿随机方向移动后的可行点目标函数值
            j=j+1;
          else
             continue;
          end
        end
        xf=[x;fz];
        [B,ind]=sort(xf(n+1,:));    %升序排列移动后可行点的目标函数值
        fl=B(1);
        if fl<f0
            break;
        else
            a0=0.9*a0;
        end
    end
    xl=x(:,ind(1)); %找到最小的可行点
    d=xl-x0;    %找到可行搜索方向
    f0=fl;  %将初始点移动到试探可行点
    x0=xl;
    %% 沿可行方向采用加速步长进行搜索
    while 1
        a0=1.3*a0;  %步长加速
        xl=x0+a0*d; %计算沿可行方向移动的下一点
        fl=fun(xl);
        g=gcon(xl);
        if(all(g<=0) && fl<f0)
            x0=xl;
            f0=fl;
        else
            break;
        end
    end
    xn=x0-a0*d/1.3;
    fl=fun(xn);
```

```
    %%
    if abs((f0-fl)/f0)<epsilon  %判断是否达到精度
        break;
    end
end
xe=x0;
fe=f0;
% 输出最终结果
fprintf('经过随机方向法运算后得到结果如下: \n D2=%.4f\n d=%.4f\n n=%.4f\n 
f(x)=%.4f\n', xe(1), xe(2), xe(3),fe);
toc  %计时结束
```

随机方向法（优化函数程序）：

```
function f=fun(x)
f=x(1)*(x(3)+2)*x(2)^2/cos(atan(0.4/pi));
```

随机方向法（约束条件程序）：

```
function g=gcon(x)
g=[
    22-x(1)+x(2)
    x(1)+x(2)-42
    110-0.4*x(1)*x(3)-2*x(2)
    0.4*x(1)*x(3)+2*x(2)-130
    (5600*x(1)^0.84)/(pi*x(2)^2.84)-785
];
```

实验结果：

对题目的合理数学推导结果（参照论文）：

$$P = 0.4x_1 \qquad k = \frac{4c-1}{4c-4} + \frac{0.615}{c} \approx 1.66\left(\frac{x_2}{x_1}\right)^{0.16}$$

$$\min f(x) = x_1 \cdot (x_3 + 2) \cdot x_2^2 / \cos\left[\arctan\left(\frac{0.4}{\pi}\right)\right]$$

$$g_1(x) = x_1 - x_2 - 22 \geq 0$$

$$g_2(x) = 42 - x_1 - x_2 \geq 0$$

$$g_3(x) = 0.4x_1x_3 + 2x_2 - 110 \geq 0$$

$$g_4(x) = 130 - 0.4x_1x_3 - 2x_2 \geq 0$$

$$g_5(x) = 785 - \frac{5600}{\pi} x_1^{0.84} \cdot x_2^{-2.84}$$

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} D_2 \\ d \\ n \end{pmatrix}$$

题目：

初始题目：

优化设计数学模型为

$$\begin{cases}
\min f(\boldsymbol{x}) = \dfrac{x_1(n+2)}{\cos\alpha} x_2^2 \\
g_1(\boldsymbol{x}) = x_1 - x_2 - 22 \geq 0 \\
g_2(\boldsymbol{x}) = 42 - x_1 - x_2 \geq 0 \\
g_3(\boldsymbol{x}) = np + 2x_2 - 110 \geq 0 \\
g_4(\boldsymbol{x}) = 130 - np - 2x_2 \geq 0 \\
g_5(\boldsymbol{x}) = \arctan\dfrac{p}{\pi x_1} - 5^\circ \geq 0 \\
g_6(\boldsymbol{x}) = 10^\circ - \arctan\dfrac{p}{\pi x_1} \geq 0 \\
g_7(\boldsymbol{x}) = [\tau] - \left(\dfrac{4c-1}{4c-4} + \dfrac{0.615}{c}\right)\dfrac{8F_{\max}x_1}{\pi x_2^3} \geq 0 \\
\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} D_2 \\ d \end{pmatrix}
\end{cases}$$

求解结果：

命令行窗口

经过随机方向法运算后得到结果如下:
D2=25.4788
d=3.4780
n=10.1117
f(x)=3762.9025
时间已过 0.020141 秒。
fx >>

实验收获：

最优化算法中的**随机方向法**是一种在处理**多维约束最优化问题**时非常有效的方法。在这个实验中，我们通过不断调整初始点，利用随机方向和步长来寻找满足所有约束条件的可行解。在实验的开始，我们需要选择一个可行的初始点。这一步虽然看似简单，但实际上在约束条件复杂的情况下，找到一个**既满足所有约束条件又能作为起始点的解**并不容易。这一步的选择会直接影响到整个优化过程的效率和最终结果的质量，有时候甚至需要对题目进行相应的重构。

随后，我们生成多个随机单位向量作为搜索方向。这一步的目的在于探索更大的解空间，避免陷入局部最优解。在生成随机方向时，要确保这些向量的维度和问题的维度一致，并且要保持它们的随机性，从而在不同方向上进行充分的搜索。通过试验步长，我们计算出多个随机点。在这些随机点中，我们寻找一个满足约束条件的点，并以此产生可行的搜索方向。在这一步中，选择合适的步长尤为重要，因为过大的步长可能会跳过最优解，而过小的步长则可能导致搜索过程过于缓慢。在沿着可行搜索方向迭代计算的过程中，我们需要不断检验新的点是否满足所有约束条件，并观察目标函数值的变化。当找到一个新的点满足所有约束条件且目标函数值不再下降时，我们可以认为找到了一个局部最优解。如果在多次迭代后仍未找到合适的解，需要重新调整步长或方向，继续搜索。

综上所述，最优化算法中的**随机方向法**在解决复杂约束优化问题时具有很大的应用潜力。通过多次随机试验和迭代搜索，我们能够找到满足所有约束条件并符合精度要求的最优解。通过这个实验，我深刻体会到了随机方向算法的精妙之处，以及在实际工程应用中灵活调整策略的重要性。