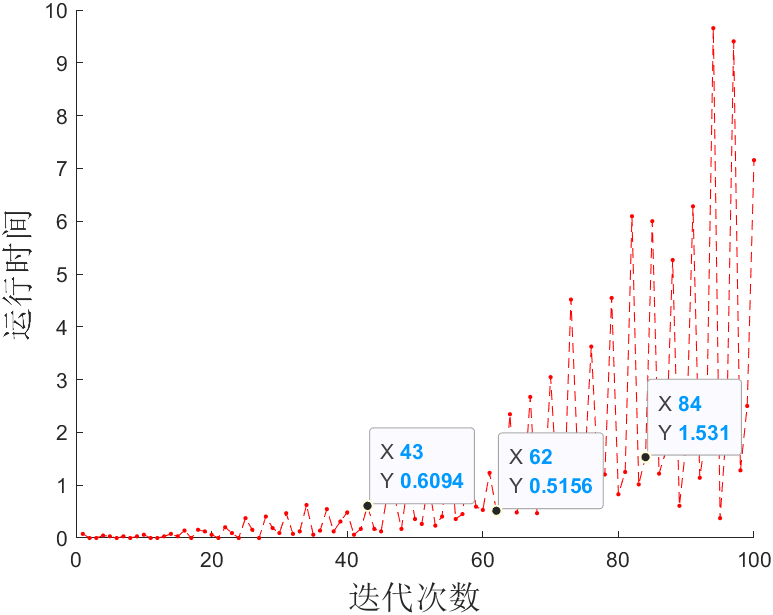
**单纯型法结果展示**

**Test 1**

原代码段：

for n=1:100

f=-ones(1,n);

A=sparse(n,0.1);

b=rand(n,1);

if mod(n,3) == 0

[x1,fval1]=MyLPSolver(f,A,b);

end

if mod(n,3) == 1

随机生成Aeq,beq

[x2,fval2]=MyLPSolver(f,A,b,Aeq,beq);

end

if mod(n,3) == 2

随机生成Aeq,beq,lb,ub

[x3,fval3]=MyLPSolver(f,A,b,Aeq,beq,lb,ub);

end

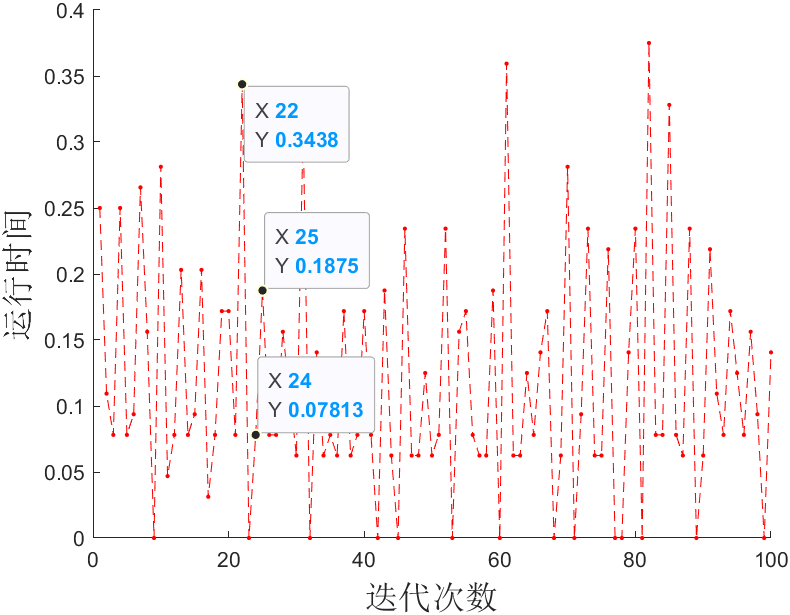
end

注意：其中A矩阵的稀疏度为0.1。x2的等式约束有m个（0<m<n+1），x3的等式约束只有1个。

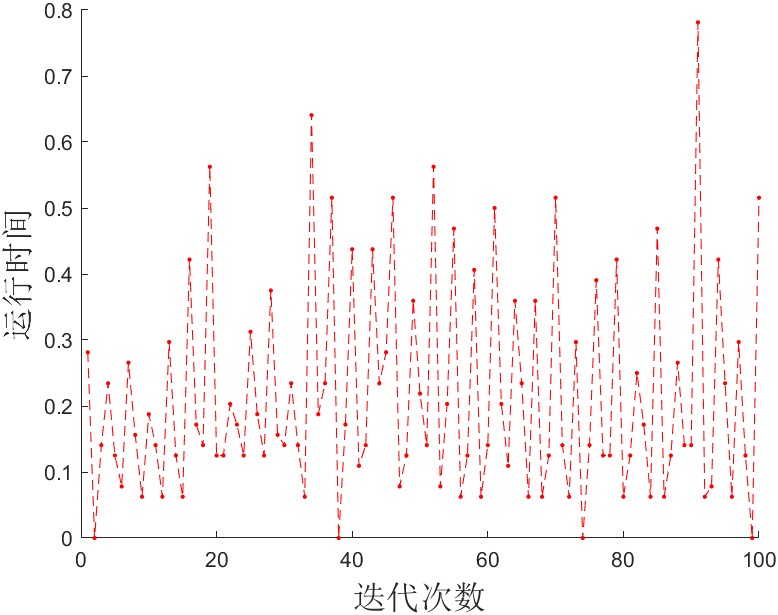
发现：三种方法各自消耗的时间不同，在目前代码段中，[x2,fval2]的计算时间最大，[x1,fval1]的计算时间最少，但都大体随A矩阵大小的增加而增加。而且除求解x2时间增长幅度较大外，其余两种方法时间复杂程度增长的速率仍然较慢。可以认为是等式约束的处理使得运行时间变缓的程度增大。

可对比命令行log：随矩阵规模增大的问题求解情况.txt

**Test 2**



**Another same test**



关键代码段：

A=sparse(25,n/100);

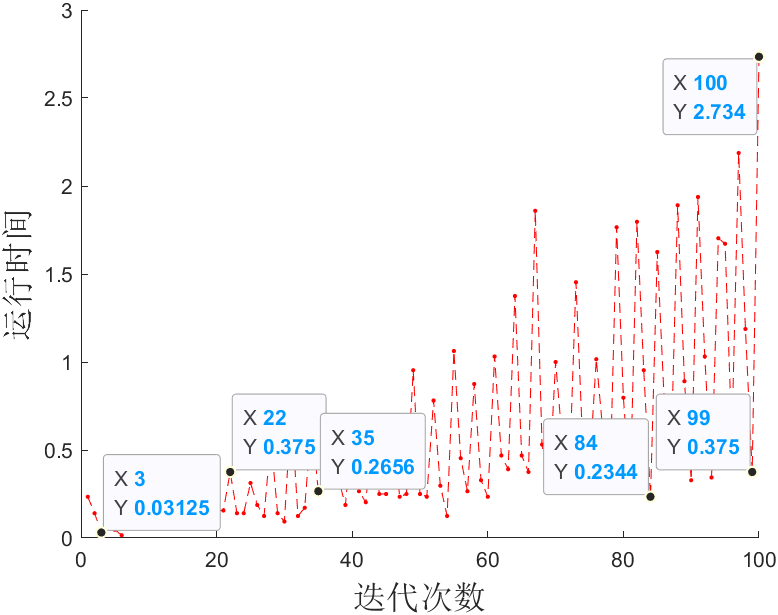
迭代次数与稀疏度的关系：

Sparsity=n/100;

发现与反思：矩阵大小控制在25\*25，随稀疏度增加，A中含0元素增加。但三种方法各自的运行时间基本在同一水平上波动。x3的计算时间多次出现接近于0的情况，推测可能是各个变量的约束条件增多增加了当前矩阵不可行的情况，除了这些情况外，x3的时间复杂度表现和x1类似。

可对比命令行log：随稀疏度变化问题求解的情况.txt

**Test 3**



关键代码段：

A=rand(n,25);

迭代次数决定了A矩阵的行数。

发现与反思：随着A矩阵的变长，约束增多，时间复杂度会增加，不过时间复杂度远远小于同量级方阵的时间复杂度。对x3，我们发现35与84两点，一个接近方阵，另一个，行数是变量个数的3倍多，但时间复杂度的变化不大。不过，在大部分问题无界的情况下，仅有x3能够出现最优解，需要反思的是——是否在A矩阵的设计中忽略了负系数的加入，或者系数都过小（0~1之间），导致大部分随机问题均无界。

可对比命令行log：A矩阵逐渐变长时的求解情况.txt

**Test 4**

规范型表格的变化见example.txt

原问题的代码：

f=[-1,2,0,2];

A=[3,-1,2,-2;4,3,2,-5;-2,3,-1,4;2,-1,5,3];

b=[7;-1;-4;0];

Aeq=[-2,2,-3,4];

beq=3\*ones(1);

lb=[0;0;-2;1];

ub=[2;inf;5;inf];

[x,fval]=MyLPSolver(f,A,b,Aeq,beq,lb,ub)