1. 算法描述

Krylov子空间法解决具有如下何下形式的大规模线性规划问题：

其中，为目标函数系数，为非奇异约束矩阵，x为n维解相量。既然是基于投影的迭代方法，必须先构造投影用的子空间和并根据k的关系分别为正投影和斜投影，在子空间K上找到近似解，使得残余向量垂直于子空间。给定初始点，则可用仿射空间代替子空间k，则近似过程变成：找到近似解，满足。在投影方法基础上应用krylov子空间：





其中，子空间为m维。

对于满足充要条件是



具体的krylov算法由不同的预处理过程和不同的子空间决定。在众多krylov子空间算法中，我们选用比较成熟的GMRES方法。

GMRES算法要求系数矩阵A是非奇异、非对称的ｎ维方阵。其krylov子空间如上式，GMRES算法利益Arnoldi变换构造一个正交基来表示krylov子空间。Full GMRES算法允许krylov子空间维数增加到ｎ而且总是在最大迭代次数ｎ内终止运算；另一种restart GMRES算法则严格要求子空间维数为一个定值ｍ，在进行了ｍ次迭代后，以得到的最后迭代结果作为初始点重新进行Arnoldi变换，当残余向量ｒ＝Ａ－ｂｘ满足时，终止计算。综合考虑时间和空间复杂度，restart GMRES更适合我们的要求。

算法大致过程：

第一步 由原问题（1）构造线性方程；

第二步 求初始值；

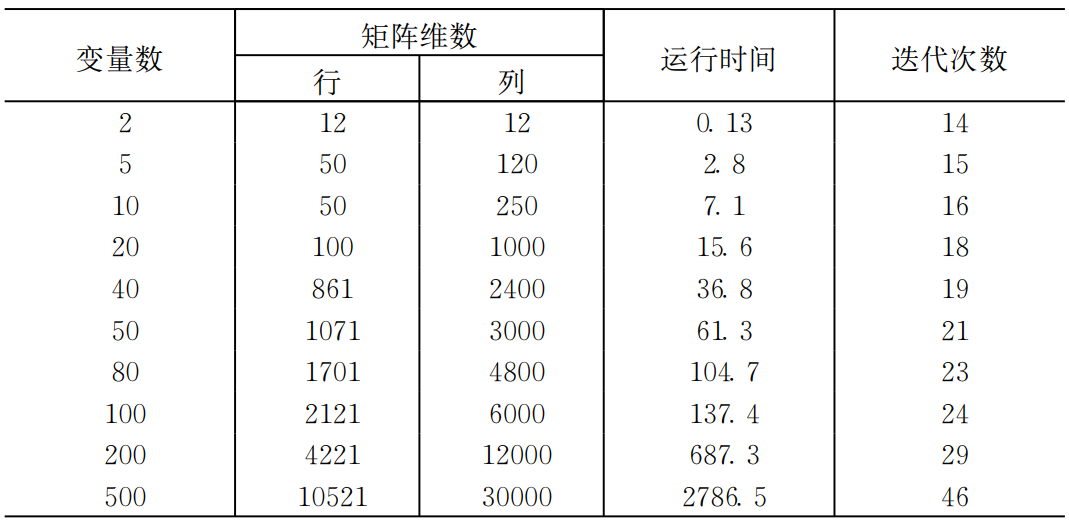
第三步 通过初始值 ｖ0得到雅可比矩阵；

第四步 调用MATLAB工具箱函数luinc对Jacobi矩阵进行不完全LU分解采用的是，得到预处理矩阵LU；

第五步 调用MATLAB工具箱函数gmres进行内部迭代近似计算得到，从而计算出，继续重复步骤第三，四，五步，直到收敛。

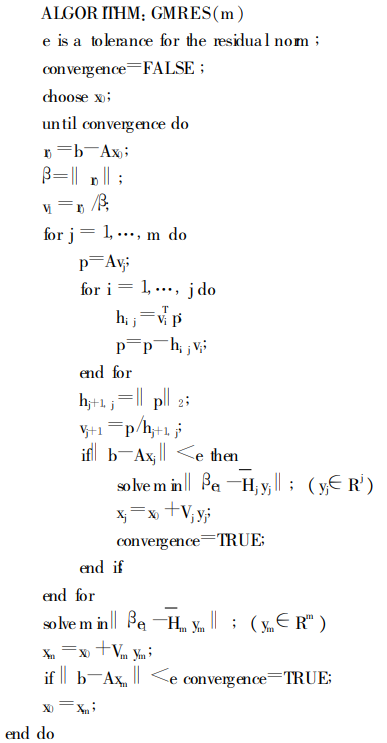
通过以上计算得到实验结果如表1所示，可以看出GMRES方法有效的求解了大规模线性优化问题。

表1 GMRES方法求解结果



1. 伪代码

对于问题 （5）有如下restart GMRES算法：



其中对于 1≤ｋ≤ｍ，矩阵是的上Hessenberg矩阵，由以上算法可以得到基本迭代关系：，GMRES算法通过计算得到可行解，其中的通过计算最小二乘问题得到,通常对矩阵采用QR分解来运算。

1. 收敛性分析

GMRES算法的收敛性完全取决于系数矩阵Ａ的特征值的性质，为了改善这些性质，时常会对线性方程组进行适当的变换，这个过程就称为预处理过程。不同的迭代法，其预处理的目标不尽相同。对于GMRES方法，预处理后的目标是使得预处理后系数矩阵Ａ的特征值分布在复平面内尽可能小的区域来加快收敛速度，通过构造预处理矩阵Ｍ对问题进行预处理，分以下三种情况：

左预处理：。

右预处理：

分解预处理：；



。

常用的预处理方法有对角块预处理法、快速解耦法、不完全LU分解法、稀疏近似逆方法以及压缩预处理方法等。这里考虑应用比较广泛的不完全LU分解法。

采用上述的分解预处理，对系数矩阵进行LU分解。若Ａ是大型稀疏矩阵则其因子L和U的下、上三角部分一般来说都是满秩矩阵。若直接采用预处理矩阵则理论上最好，但实际上完全行不通。一种不完全LU分解(ILU)把系数矩阵Ａ分解成和,它们对应的元素与Ｌ、Ｕ相同，但两个因子分别与矩阵 Ａ的下、上三角部分具有完全相同的非零结构，这就是无填充的不完全LU分解,记为，这里的＂0＂代表因子分解时产生的新的非零元素的个数为零。

通常的不完全LU分解记为，其中ε为大于零的常数（通常很小）则不完全LU分解时会根据ε的大小产生相应的一些填充量。这样计算量自然会增加，矩阵Ｌ和Ｕ的稀疏性也难以保证，但预处理效果会比好些。这就是说，不完全LU分解需要寻找一种平衡：既不能产生太多的填充量，又要达到预期的预处理效果。所以很有必要在不完全LU分解中设定阀值ε并采取一定的策略来选择和控制填充量。