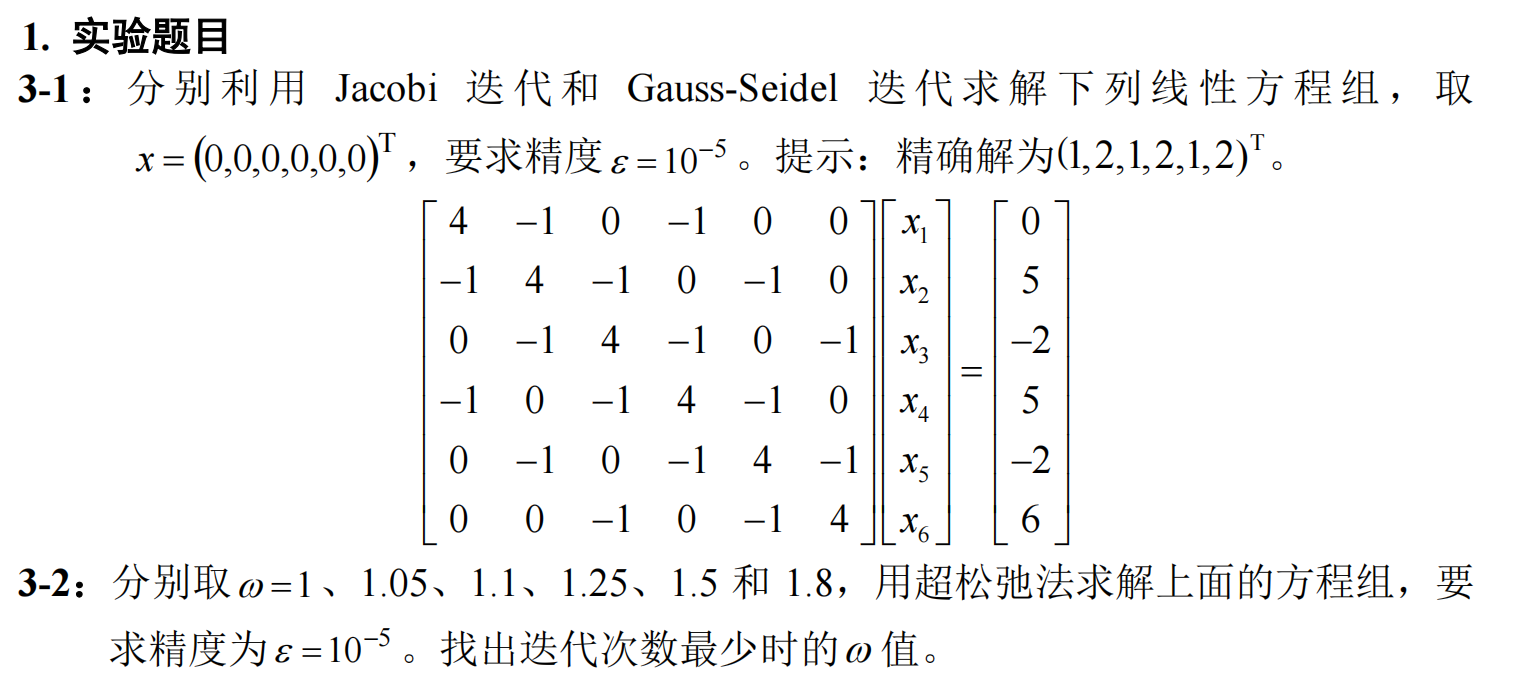
**实验五 线性方程组的迭代法实验**



**3-1：**

1. Jacobi迭代法求解方程组

设计思想：核心在于将一般形式的线性方程组的求解归结为对角方程组求解过程的重复。主要步骤如下：

1. 分解系数矩阵：将系数矩阵 A 分解成对角元素和非对角元素的和，即 A = D + (L + U)，其中 D 是对角矩阵，L 是下三角矩阵（包括主对角线上的元素为零），U 是上三角矩阵（包括主对角线上的元素为零）。

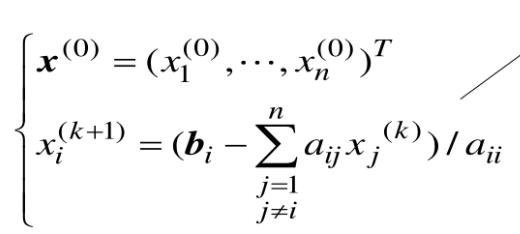
2. 迭代过程：根据分解后的系数矩阵，可以将原方程组转化为迭代形式，即 Dx = - (L + U)x + b。接下来，我们使用迭代方法逐步逼近解向量 x。

3. 迭代步骤：根据 Jacobi 迭代法的思想，每次迭代的更新公式如下：x(k+1) = D^(-1) \* [-(L + U)x(k) + b]其中，x(k) 表示第 k 次迭代得到的解向量，x(k+1) 表示第 k+1 次迭代的解向量，D^(-1) 是对角矩阵 D 的逆矩阵。

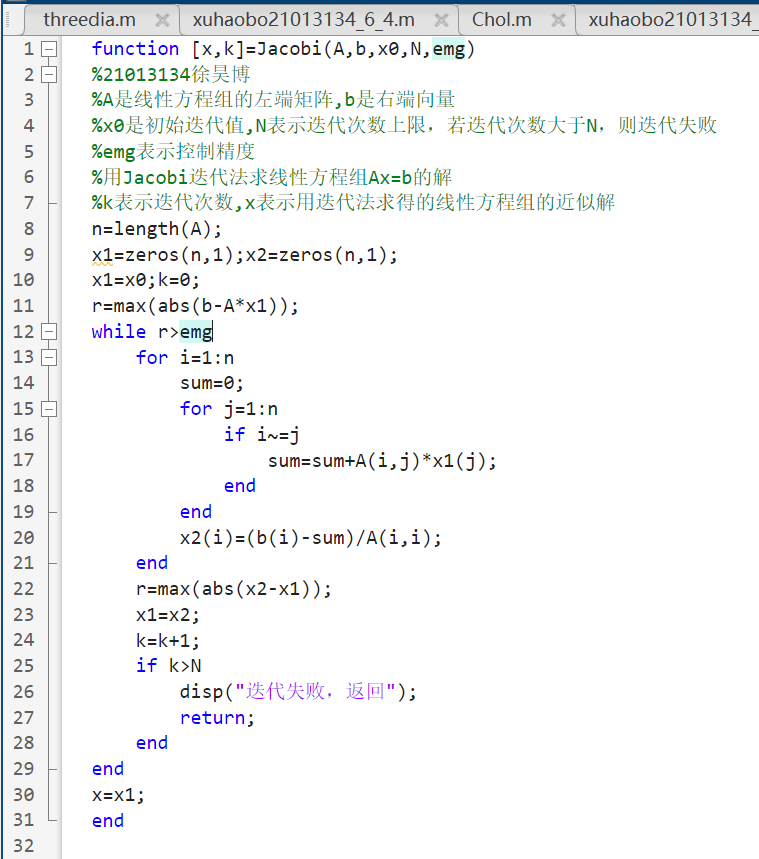
4. 迭代终止条件：通过多次迭代，不断更新解向量 x，直到满足预设的收敛条件。一般情况下，可以设置一个迭代次数上限，或者设定一个误差限制，当解向量的更新变化小于该误差限制时，迭代过程停止。

5. 解的计算：最终，当迭代终止时，得到的解向量 x 就是线性方程组的近似解。

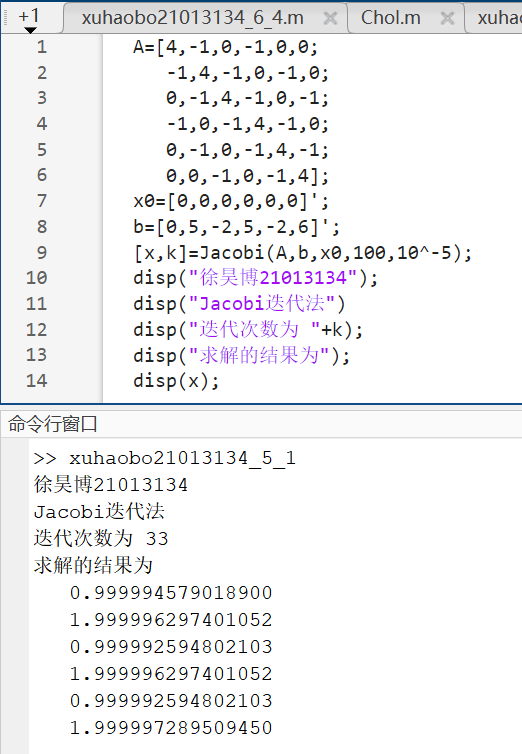
具体公式如下：



代码如下所示：



将题目中的方程组的左端矩阵和右端向量输入MATLAB中进行计算得到结果：



可见迭代次数为33次，次数较多。

1. Guass-Seidel迭代法求解方程组

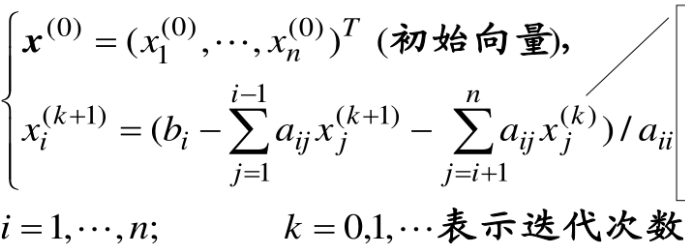
设计思想：相较于Jacobi方法，其设计思想是在每次迭代中尽可能使用最新的解来更新方程组中的未知量。主要步骤如下：

1. 分解系数矩阵：将系数矩阵 A 分解成下三角部分 L、对角矩阵 D 和上三角部分 U，即 A = L + D + U，其中 L 包含了 A 的下三角元素（包括主对角线上的元素为零），D 是 A 的对角矩阵，U 包含了 A 的上三角元素（包括主对角线上的元素为零）。

2. 迭代过程：根据分解后的系数矩阵，可以将原方程组转化为迭代形式，即 (L + D + U)x = b。接下来，我们使用迭代方法逐步逼近解向量 x。

3. 迭代步骤：根据 Gauss-Seidel 迭代法的思想，每次迭代的更新公式如下：x(k+1) = (D + L)^(-1) \* (-Ux(k) + b)其中，x(k) 表示第 k 次迭代得到的解向量，x(k+1) 表示第 k+1 次迭代的解向量，D + L 是下三角矩阵的和。关键的思想在于，更新解向量 x(k+1) 时，对于方程组中的每个未知量，我们使用上一次迭代中已经更新的未知量来计算，即使用最新的解值。

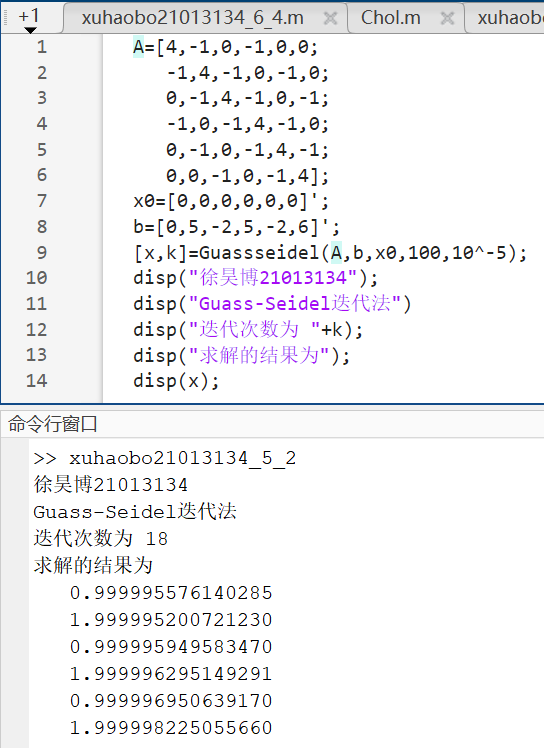
迭代终止条件与解的计算和Jacobi相同。具体公式如下所示：



代码如下所示：



同上一个方法将方程组输入得到迭代结果：



可见Guass-Seidel方法迭代次数明显更少，收敛速度更快。

**3-2：**超松弛迭代法求解方程组

超松弛迭代法（Successive Over-Relaxation，简称SOR）是一种用于求解线性方程组的迭代算法，其设计思想是在每次迭代中结合Jacobi迭代法和Gauss-Seidel迭代法，并引入一个松弛因子来加速收敛。下面是SOR迭代法的设计思想：

1. 分解系数矩阵：将系数矩阵 A 分解成下三角部分 L、对角矩阵 D 和上三角部分 U，即 A = L + D + U，其中 L 包含了 A 的下三角元素（包括主对角线上的元素为零），D 是 A 的对角矩阵，U 包含了 A 的上三角元素（包括主对角线上的元素为零）。

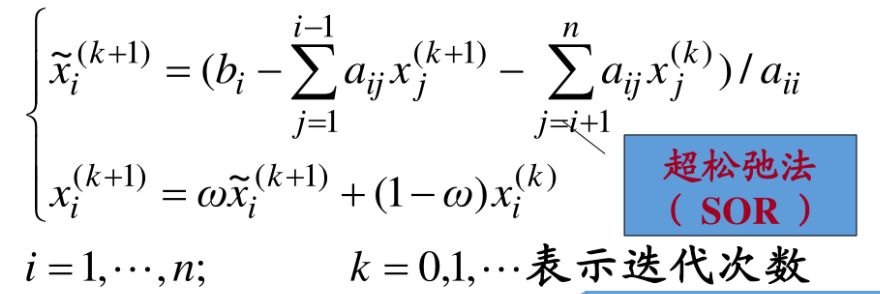
2. 迭代过程：根据分解后的系数矩阵，可以将原方程组转化为迭代形式，即 (D + wL)x = -wUx(k) + (1-w)b。

其中，x(k) 表示第 k 次迭代得到的解向量，x 表示第 k+1 次迭代的解向量，w 是松弛因子，取值范围为 0 < w < 2。

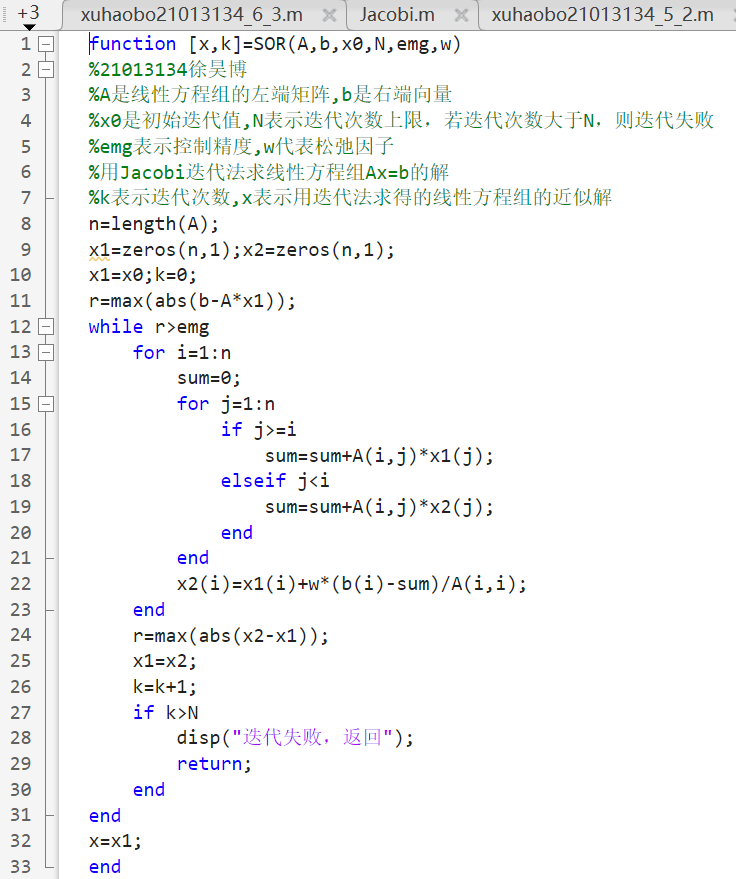
3. 迭代步骤：根据 SOR 迭代法的思想，每次迭代的更新公式如下：x(k+1) = (D + wL)^(-1) \* [-wUx(k) + (1-w)b]关键的思想在于，更新解向量 x(k+1) 时，对于方程组中的每个未知量，我们使用上一次迭代中已经更新的未知量来计算，并引入松弛因子 w 来调节更新的幅度。

当 w = 1 时，SOR迭代法退化为Gauss-Seidel迭代法；当 w = 0 时，SOR迭代法退化为Jacobi迭代法。

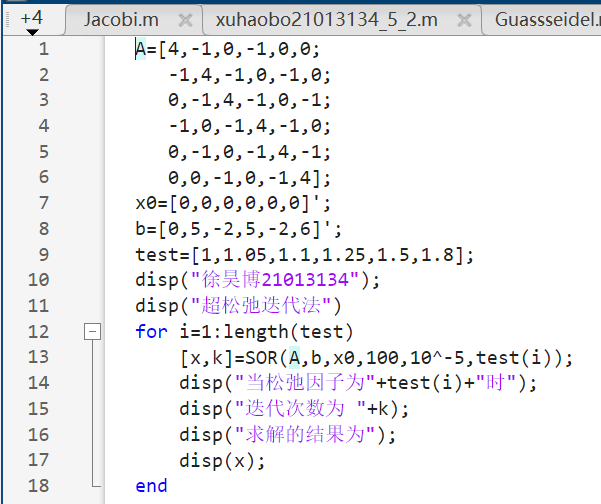
迭代终止条件与解的计算和Jacobi相同。具体公式如下所示：



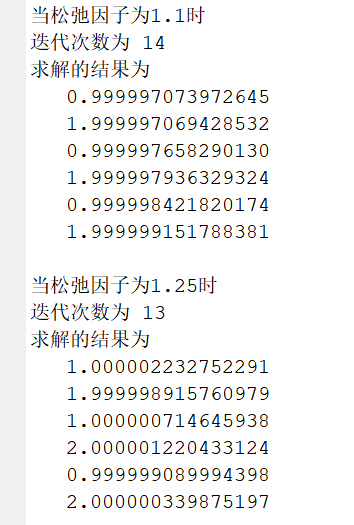
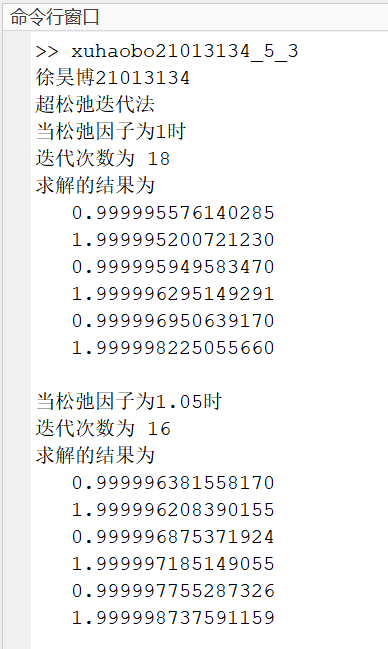
代码如下所示：



同上一个方法将方程组输入得到迭代结果：



运行结果：





由结果可以看出当松弛因子为1.25时，迭代次数最少，为13次。

**实验体会**

就本次实验而言，三个迭代方法设计要点和注意事项总结如下：

1、Jacobi迭代法：

-将系数矩阵分解为对角和非对角部分，利用迭代的方式逐步逼近解向量。

-每次迭代更新解向量时，使用前一次迭代的解向量来计算下一次迭代的解向量。

-设置适当的迭代终止条件，如迭代次数上限或解的更新变化小于预设误差限制。

-Jacobi迭代法通常收敛速度较慢，特别是对于病态的线性方程组，可能会出现收敛困难的情况。

Gauss-Seidel迭代法：

-将系数矩阵分解为下三角、对角和上三角部分，利用迭代的方式逐步逼近解向量。

-每次迭代更新解向量时，利用最新的解值来计算下一次迭代的解值，即使用局部更新策略。

-设置适当的迭代终止条件，如迭代次数上限或解的更新变化小于预设误差限制。

- Gauss-Seidel迭代法通常比Jacobi迭代法收敛速度更快，但对于病态的线性方程组仍可能出现收敛困难的情况。

超松弛迭代法（SOR）：

-结合了Jacobi迭代法和Gauss-Seidel迭代法的思想，引入松弛因子来加速收敛。

-选择合适的松弛因子 w 值，通常取 0 < w < 2。

-设置适当的迭代终止条件，如迭代次数上限或解的更新变化小于预设误差限制。

- SOR迭代法在选择合适的松弛因子时可以加速收敛，但过大或过小的 w 值可能导致算法不稳定或收敛性下降。

总体而言，迭代方法在求解线性方程组时需要注意选择合适的终止条件，确保解的精度和计算效率之间的平衡。此外，对于病态的线性方程组，可能需要使用预处理技术或其他更高级的迭代方法来提高求解效果。

对于上述几种方法在我的理解看来，具有一定的通性，核心在于将一般形式的线性方程组的求解归结为对角方程组求解过程的重复，不同的是Guass相比Jacobi使用了更多的新信息，而SOR方法比Guass法多了松弛因子的加入更加精准。需要注意的是，这些方法的收敛性和效率受到线性方程组本身性质的影响，对于病态的线性方程组可能需要使用其他更高级的方法或预处理技术来提高求解效果。