

第四次作业报告 PB23000141 刘彦宏

编译环境

作业使用cmake组织项目, 编译器使用msvc1942. 开发使用的IDE是Clion, VS2022当然也可以正常构建并运行.
代码运行方式与之前的作业一致, 按先前作业的批改反馈此处不再赘述.

问题描述

根据课本给出的算法, 利用迭代法求解线性方程组.

问题分析

因为只是根据课本算法直接求解线性方程组, 略掉问题分析.

不过值得一提的是我手头的这个扫描版的电子课本, 他第二题的五点差分格式写错了, 有个负号写成了正号.

结果展示

第一题

```
D:\Code\N_A\cmake-build-debug-visual-studio\hw4\hw4_1.exe

eps: 1
Jacobi Method Error: 0.0193186
G-S Method Error: 0.0143577
SOR Method Error: 0.00497328

eps: 0.1
Jacobi Method Error: 0.0153039
G-S Method Error: 0.0139553
SOR Method Error: 0.0105531

eps: 0.01
Jacobi Method Error: 0.0664286
G-S Method Error: 0.0663977
SOR Method Error: 0.06621

eps: 0.0001
Jacobi Method Error: 0.00506316
G-S Method Error: 0.00496649
SOR Method Error: 0.0049652
```

进程已结束, 退出代码为 0

第二题

```
D:\Code\N_A\cmake-build-debug-visual-studio\hw4\hw4_2.exe
-----Sparse-----
N = 20
iterations: 580
Function took 16 milliseconds to execute.

-----Sparse-----
N = 40
iterations: 2215
Function took 279 milliseconds to execute.

-----Sparse-----
N = 80
iterations: 8438
Function took 3816 milliseconds to execute.

-----Dense-----
N = 20
iterations: 580
Function took 826 milliseconds to execute.

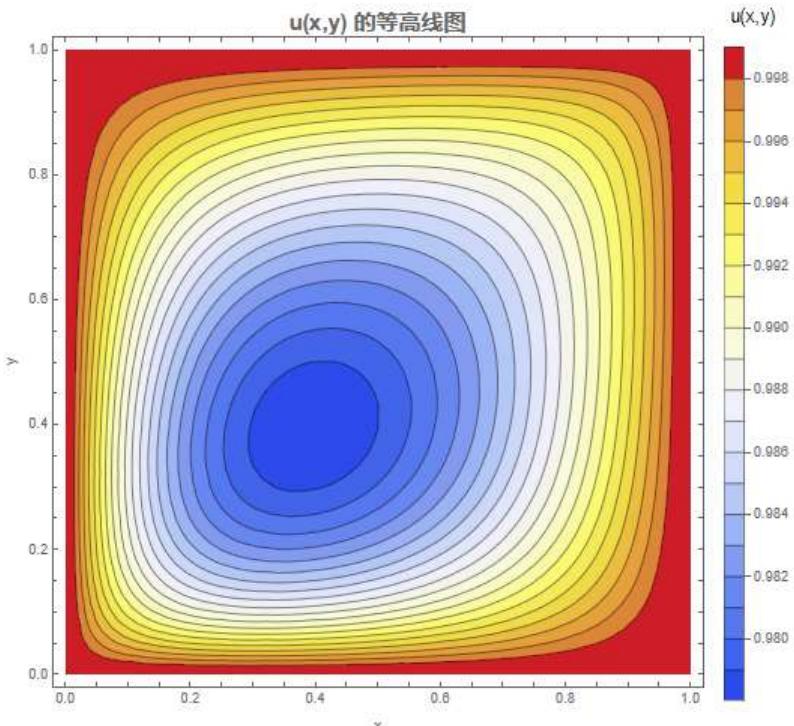
-----Dense-----
N = 40
iterations: 2215
Function took 44564 milliseconds to execute.
```

讨论总结

第一题我不是很懂如何控制四位有效数字, 所以停止迭代的标准实际上用的是 $\|x_{k+1} - x_k\|_2 < 10^{-4}$, 这样与第二题的停止迭代条件是一样的, 比较省事().

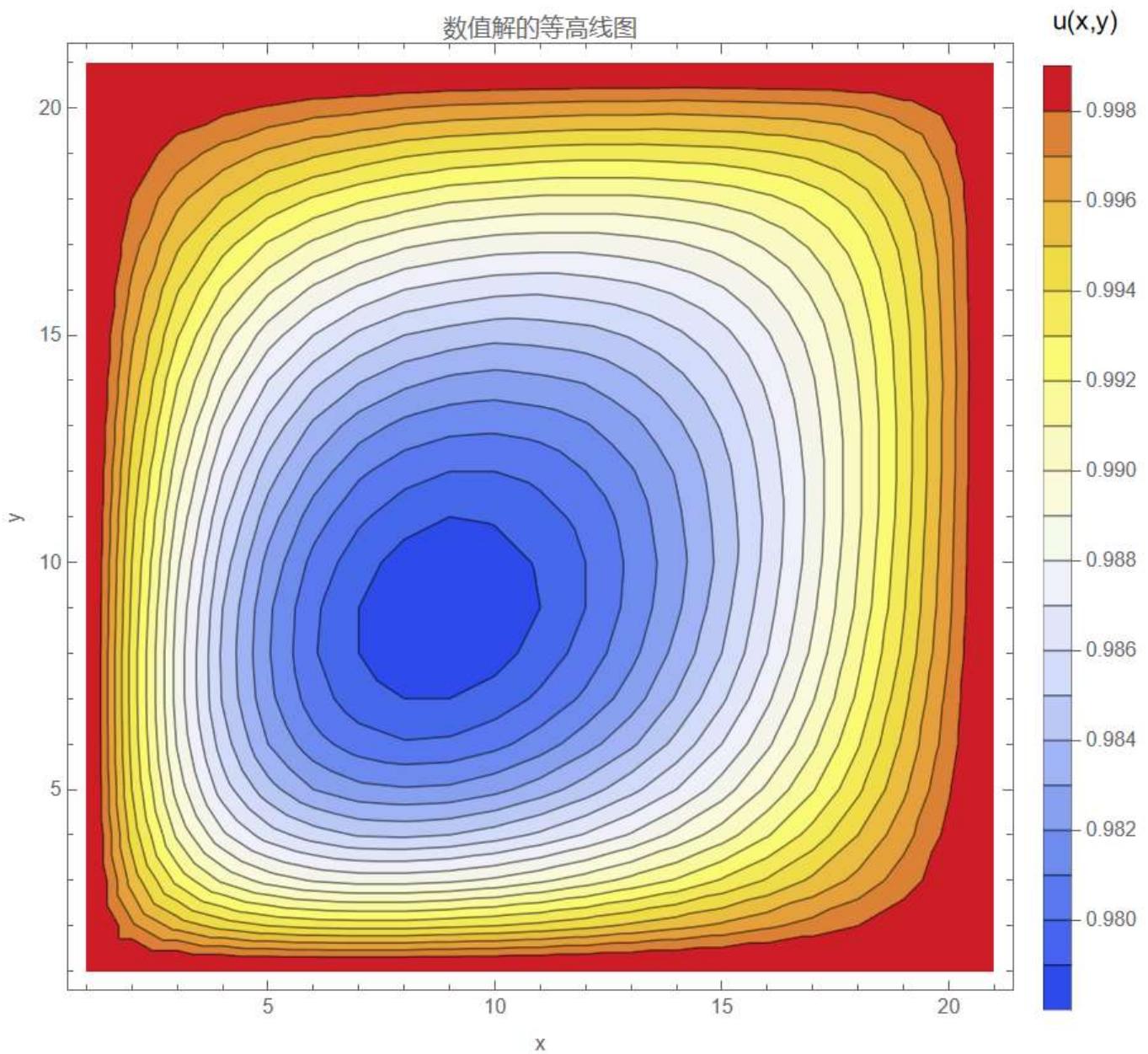
一开始我都是按照稠密矩阵的想法来写的求解器(其实是想偷懒了), 但是在算第二题的 $N = 80$ 的时候直接炸缸了, 算半天算不出来(倒也合理, 矩阵太稀疏了当稠密矩阵来做效率就是会很差), 所以简单实现了一点点稀疏矩阵的内容, 让第二题能够正常计算.

为了保证结果的正确性, 我用Mathematica求解了一下第二题的方程, 如图所示.



$N=20$ 算出来的结果如图, 已然是初现端倪

数值解的等高线图



$N=80$ 算出来的结果如图, 简直就是一模一样, 非常的酷.

数值解的等高线图

