

数值代数实验报告

PB21010483 郭忠炜

2023 年 11 月 18 日

一. 问题描述

Exercise1

对于给定的二阶微分方程边值问题，将问题离散化为线性方程组后，使用 Jacobi 迭代法、Gauss-Seidel 迭代法和 SOR 迭代法求解该方程组。比较这些迭代方法在不同参数 (ϵ) 设置下的性能，并分析数值解与精确解之间的误差。

对于给定的 Dirichlet 问题，将其转化为网络规格为 n 的差分方程之后，在一个正方形区域内使用共轭梯度法求解对应 $(n-1)^2$ 维线性方程组。需要输出迭代次数、计算所用时间以及求解向量。需要特别注意确保边界条件与线性方程系统的正确关联以保证准确性。

Exercise2

针对不同规模的问题 ($n = 20, 40, 60, 80$)，构建相应规模的 Hilbert 矩阵和右端向量，并利用共轭梯度法求解线性方程组。在每次求解中，设定迭代停止条件，输出迭代次数、求解所用时间以及解向量。

Exercise3

针对给定的线性方程组，使用 Jacobi 迭代法、Gauss-Seidel 迭代法和共轭梯度法进行求解。输出各个迭代法的迭代次数、求解时间和解向量。通过比较这三种方法的表现，我们可以评估它们在解决这个特定线性方程组时的效果和性能差异。

二. 程序介绍

向量内积

- **函数描述:** `dotProduct` 函数用于计算两个向量的点积。
- **使用方式:** 调用 `dotProduct(v1, v2)` 函数, 传入两个向量 $v1$ 和 $v2$, 函数会返回它们的点积。

共轭梯度法求解差分方程

- **函数描述:** `ConjugateGradient1` 函数用于执行共轭梯度法求解线性方程组。
- **使用方式:** 调用 `ConjugateGradient1(A, x, b)` 函数, 传入系数矩阵 A 、初始解向量 x 和右端向量 b , 函数会进行共轭梯度法的迭代计算并输出解向量和运行时间。

共轭梯度法求解线性方程组

- **函数描述:** `ConjugateGradient2` 函数也用于执行共轭梯度法求解线性方程组。
- **使用方式:** 调用 `ConjugateGradient2(A, x, b)` 函数, 传入系数矩阵 A 、初始解向量 x 和右端向量 b , 函数会进行共轭梯度法的迭代计算并输出解向量和运行时间。

生成矩阵

- **函数描述:** `generateMatrixA` 函数用于生成特定形式的矩阵 A 。
- **使用方式:** 调用 `generateMatrixA(n)` 函数, 传入整数参数 n , 函数会生成一个 $(n-1) \times (n-1)$ 维的矩阵 A 。

三. 实验结果

Exercise1

解向量																					迭代次数: 457		运行时间: 1.7580 seconds.	
0.0000	0.0025	0.0100	0.0225	0.0400	0.0625	0.0900	0.1225	0.1600	0.2025	0.2500	0.3025	0.3600	0.4225	0.4900	0.5625	0.6400	0.7225	0.8100	0.9025	1.0000				
0.0025	-0.0000	-0.0001	-0.0001	-0.0001	0.0002	0.0003	0.0002	0.0002	0.0001	0.0001	0.0000	-0.0000	0.0001	0.0002	0.0004	0.0003	0.0002	0.0002	0.0001	1.0025				
0.0100	0.0000	0.0001	0.0001	0.0001	-0.0001	-0.0002	-0.0001	-0.0001	-0.0000	-0.0000	0.0000	0.0001	-0.0000	-0.0001	-0.0002	-0.0002	-0.0000	-0.0001	-0.0001	1.0100				
0.0225	0.0000	0.0001	0.0001	-0.0000	-0.0001	-0.0001	-0.0002	-0.0001	-0.0001	-0.0000	0.0000	0.0000	-0.0000	-0.0002	-0.0002	-0.0002	-0.0001	-0.0001	-0.0000	1.0225				
0.0400	-0.0000	-0.0001	-0.0001	-0.0001	0.0002	0.0003	0.0002	0.0001	0.0001	0.0001	0.0000	-0.0000	0.0001	0.0001	0.0003	0.0004	0.0002	0.0002	0.0002	1.0400				
0.0625	-0.0000	-0.0000	0.0000	0.0001	-0.0000	-0.0000	0.0001	0.0001	0.0001	0.0000	0.0001	0.0001	0.0001	0.0001	0.0000	0.0001	0.0002	0.0001	0.0000	1.0625				
0.0900	0.0001	0.0001	0.0002	0.0000	-0.0001	-0.0002	-0.0002	-0.0001	-0.0001	-0.0000	0.0000	0.0000	-0.0000	-0.0002	-0.0003	-0.0003	-0.0002	-0.0001	-0.0000	1.0900				
0.1225	-0.0000	-0.0000	-0.0001	-0.0001	0.0001	0.0002	0.0001	0.0000	0.0001	0.0001	0.0000	-0.0000	-0.0000	0.0002	0.0003	0.0002	0.0001	0.0001	0.0001	1.1225				
0.1600	-0.0001	-0.0001	-0.0000	0.0001	0.0001	0.0001	0.0002	0.0002	0.0001	0.0001	0.0001	0.0001	0.0002	0.0003	0.0003	0.0003	0.0004	0.0003	0.0001	1.1600				
0.2025	0.0001	0.0001	0.0002	0.0000	-0.0001	-0.0002	-0.0002	-0.0001	-0.0000	0.0000	0.0001	0.0001	0.0001	-0.0001	-0.0002	-0.0002	-0.0001	-0.0000	0.0000	1.2025				
0.2500	0.0001	0.0000	-0.0001	-0.0000	0.0000	0.0000	-0.0000	-0.0001	-0.0000	0.0001	0.0000	-0.0001	-0.0001	0.0000	0.0000	-0.0000	-0.0001	-0.0000	0.0001	1.2500				
0.3025	-0.0001	-0.0002	-0.0000	0.0001	0.0001	0.0002	0.0003	0.0003	0.0001	0.0001	0.0001	0.0001	0.0002	0.0003	0.0004	0.0004	0.0004	0.0003	0.0001	1.3025				
0.3600	0.0000	0.0001	0.0002	0.0001	-0.0000	-0.0000	-0.0000	0.0000	0.0001	0.0001	0.0002	0.0002	0.0002	0.0000	0.0000	0.0001	0.0001	0.0001	0.0001	1.3600				
0.4225	0.0001	0.0001	-0.0000	-0.0000	-0.0000	-0.0001	-0.0001	-0.0002	-0.0000	0.0001	0.0001	-0.0000	-0.0001	-0.0000	-0.0001	-0.0002	-0.0002	-0.0001	0.0001	1.4225				
0.4900	-0.0001	-0.0002	-0.0001	0.0000	0.0001	0.0002	0.0003	0.0003	0.0001	0.0000	-0.0000	0.0000	0.0002	0.0002	0.0003	0.0004	0.0003	0.0003	0.0001	1.4900				
0.5625	-0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0002	0.0002	0.0001	0.0002	0.0003	0.0003	0.0002	0.0002	0.0003	0.0003	0.0003	0.0003	1.5625				
0.6400	0.0002	0.0002	0.0001	0.0000	-0.0000	-0.0001	-0.0002	-0.0002	-0.0000	0.0002	0.0002	0.0001	-0.0000	-0.0000	-0.0001	-0.0002	-0.0002	-0.0000	0.0001	1.6400				
0.7225	-0.0001	-0.0001	-0.0000	-0.0000	0.0001	0.0001	0.0001	0.0001	0.0000	-0.0000	-0.0000	-0.0000	0.0001	0.0001	0.0002	0.0002	0.0002	0.0001	0.0000	1.7225				
0.8100	-0.0001	-0.0000	0.0000	0.0001	0.0001	0.0002	0.0003	0.0003	0.0003	0.0001	0.0001	0.0003	0.0003	0.0003	0.0004	0.0005	0.0005	0.0004	0.0003	1.8100				
0.9025	0.0002	0.0002	0.0001	0.0001	0.0000	-0.0000	-0.0001	-0.0001	0.0001	0.0003	0.0003	0.0002	0.0001	0.0001	0.0001	-0.0000	-0.0000	0.0001	0.0002	1.9025				
1.0000	1.0025	1.0100	1.0225	1.0400	1.0625	1.0900	1.1225	1.1600	1.2025	1.2500	1.3025	1.3600	1.4225	1.4900	1.5625	1.6400	1.7225	1.8100	1.9025	2.0000				

共轭梯度法求解 Dirichlet 问题的差分方程

Exercise2

矩阵规模	解向量									
20	0.3335	0.3310	0.3377	0.3349	0.3317	0.3304	0.3304	0.3314	0.3327	0.3340
	0.3351	0.3358	0.3362	0.3361	0.3357	0.3348	0.3337	0.3322	0.3304	0.3285
	迭代次数: 5				运行时间: 0.0040s					
40	0.3332	0.3352	0.3295	0.3325	0.3350	0.3359	0.3356	0.3346	0.3336	0.3326
	0.3319	0.3315	0.3312	0.3313	0.3314	0.3317	0.3321	0.3326	0.3331	0.3335
	0.3340	0.3344	0.3347	0.3350	0.3352	0.3354	0.3354	0.3354	0.3353	0.3351
	0.3348	0.3345	0.3341	0.3336	0.3331	0.3324	0.3318	0.3310	0.3302	0.3294
	迭代次数: 7				运行时间: 0.0090s					
60	0.3330	0.3373	0.3274	0.3299	0.3339	0.3364	0.3373	0.3371	0.3362	0.3351
	0.3339	0.3328	0.3318	0.3311	0.3306	0.3303	0.3301	0.3301	0.3303	0.3305
	0.3309	0.3313	0.3317	0.3322	0.3326	0.3331	0.3336	0.3340	0.3344	0.3348
	0.3352	0.3355	0.3358	0.3360	0.3362	0.3363	0.3364	0.3364	0.3364	0.3364
	0.3363	0.3361	0.3359	0.3357	0.3354	0.3351	0.3348	0.3344	0.3340	0.3335
	0.3330	0.3325	0.3319	0.3313	0.3307	0.3301	0.3294	0.3287	0.3280	0.3273
	迭代次数: 7				运行时间: 0.0120s					
80	0.3335	0.3317	0.3369	0.3338	0.3315	0.3309	0.3315	0.3324	0.3334	0.3342
	0.3348	0.3351	0.3352	0.3351	0.3349	0.3346	0.3342	0.3339	0.3335	0.3331
	0.3328	0.3325	0.3323	0.3321	0.3319	0.3318	0.3318	0.3317	0.3318	0.3318
	0.3319	0.3320	0.3321	0.3322	0.3324	0.3326	0.3327	0.3329	0.3331	0.3333
	0.3335	0.3336	0.3338	0.3340	0.3341	0.3343	0.3344	0.3345	0.3346	0.3347
	0.3348	0.3349	0.3349	0.3349	0.3350	0.3350	0.3349	0.3349	0.3348	0.3348
	0.3347	0.3346	0.3345	0.3343	0.3342	0.3340	0.3338	0.3336	0.3334	0.3332
	0.3329	0.3326	0.3324	0.3321	0.3318	0.3315	0.3311	0.3308	0.3304	0.3301
	迭代次数: 7				运行时间: 0.0190s					

共轭梯度法求解 Hilbert 矩阵

Exercise3

方法	解向量	迭代次数	运行时间 (seconds)
Jacobi 迭代法	1.0000 -2.0000 3.0000 -2.0000 1.0000	65	0.0010
G-S 迭代法	1.0000 -2.0000 3.0000 -2.0000 1.0000	37	0.0010
共轭梯度法	1.0000 -2.0000 3.0000 -2.0000 1.0000	5	0.0010

迭代法比较

四. 结果分析

Exercise1

对于网格规模 $n = 20$, 共轭梯度法要求解的线性方程组规模为 $(n - 1)^2 = 361$, 在如此大的规模下, 共轭梯度法表现出了相当优秀的计算性能。

Exercise2

对于求解系数矩阵为 Hilbert 矩阵的线性方程组, 随着矩阵规模的增加 ($n = 20, 40, 60, 80$), 共轭梯度法的在求解结果基本稳定前提下, 其迭代次数和运行时间并未有大幅提高, 表现了共轭梯度法对于特定结构的对称正定矩阵有较好的收敛性能。

Exercise3

从计算结果上看, 对于给定线性方程组, Jacobi 迭代法、Gauss-Seidel 迭代法和共轭梯度法都完成了计算任务, 但从计算性能上 (迭代次数和运行时间), Gauss-Seidel 迭代法优于 Jacobi 迭代法, 而共轭梯度法明显优于 Gauss-Seidel 迭代法。