

《面向领域的并行数值方法》作业2

陳日康 信息与计算科学 22336049

一、题目描述

已知下列非齐次两点边值问题：

$$\begin{cases} Lu = -\frac{d}{dx}\left(p\frac{du}{dx}\right) + qu = f, & a < x < b \\ u(a) = \alpha, & u'(b) = \beta, \end{cases}$$

设 $[a, b] = [-1, 1]$, $p(x) \equiv -(-\pi^2 - 1)^{-1}$, $q(x) \equiv 1$, $\alpha = 0$, $\beta = -\pi e$, 以及 $f(x) = \frac{2\pi}{\pi^2 - 1} \cdot \cos(\pi x) \cdot e^x$ 。

任务1：

分别取 $h = 0.20, 0.10, 0.05, 0.02$ 时，将求解域 $[a, b]$ 等分成长度为 h 的单元或子区间。根据上述划分，记剖分节点为 $x_0 = a < x_1 < \cdots < x_n = b$ (即 $x_i = a + ih$)，按照要求构造与节点 x_i 对应的基函数 $\varphi_i(x)$ ：

(i). 当 x_i 为内部点 (即 $1 \leq i \leq n-1$) 时，在区间 $[x_{i-1}, x_{i+1}]$ 上， $\varphi_i(x)$ 是一个二次多项式函数，且满足 $\varphi_i(x_i) = 1$ 、 $\varphi_i(x_{i-1}) = \varphi_i(x_{i+1}) = 0$ ，在其他区间上 $\varphi_i(x) = 0$ 。

(ii). 当 $x_n = b$ (即 $i = n$) 时，在区间 $[x_{n-1}, x_n]$ 上， $\varphi_n(x)$ 是线性函数，满足 $\varphi_n(x_n) = 1$ 与 $\varphi_n(x_{n-1}) = 0$ ，在其他区间上 $\varphi_n(x) \equiv 0$ 。

(iii). 当 $x_0 = a$ (即 $i = 0$) 时，在区间 $[x_0, x_1]$ 上， $\varphi_0(x)$ 是线性函数，且满足 $\varphi_0(x_0) = 1$ 与 $\varphi_0(x_1) = 0$ ，在其他区间上 $\varphi_0(x) \equiv 0$ 。

将解函数表示成 $u_h(x) = \sum_{i=0}^n u_i \varphi_i(x)$ ，参考教材习题 1.2.3 的结论以及 1.4 章“Ritz-Galerkin 方法”等内容，为边值问题构建有限元 (即未知量 u_i , $1 \leq i \leq n$ 所满足的线性方程组)。然后分别使用高斯消去法和雅可比迭代法 (迭代 30 次)，求解上述有限元方程，获得数值解 u_h ，绘制数值解 u_h 的函数图像。

任务2：

已知 $u(x) = \sin(\pi x) \cdot e^x$ 是上述边值问题的解析解，针对不同的步长 h 和不同计算方法得到的数值解 u_h ，绘制误差函数 $u_h - u$ 的函数图像，且进行观察分析。

二、实现过程

2.1 推导过程

为求边值问题的近似解，使用 Ritz-Galerkin 方法，将问题转化为等价的变分问题：

$$J(u) = \frac{1}{2}a(u, u) - (f, u) - p(b)\beta u(b), \quad \text{最小化 } J(u),$$

其中双线性形式：

$$a(u, v) = \int_a^b (p(x)u'(x)v'(x) + q(x)u(x)v(x))dx, \quad (f, v) = \int_a^b f(x)v(x)dx.$$

设网格节点为 $x_i = a + ih$, $i = 0, 1, \dots, n$ ，将解 $u(x)$ 近似表示为： $u_h(x) = \sum_{j=0}^n u_j \varphi_j(x)$ ，其中 $\varphi_i(x)$ 为节点对应的基函数，满足：(i). 对内部节点 $i = 1, \dots, n-1$ ， $\varphi_i(x)$ 为二次函数，定义在 $[x_{i-1}, x_{i+1}]$ 上；(ii). 对边界点 $i = 0$ 或 $i = n$ ， $\varphi_i(x)$ 为线性函数，分别定义在 $[x_0, x_1]$ 、 $[x_{n-1}, x_n]$ 上。

将 $u_h(x)$ 代入变分式 $J(u)$ ，利用 Ritz-Galerkin 原理，对每一个测试函数 $\varphi_i(x)$ （取与基函数相同）分别作用，得到如下关于系数 u_j 的线性代数方程组：

$$\sum_{j=0}^n u_j a(\varphi_j, \varphi_i) = (f, \varphi_i) + p(b)\beta \varphi_i(b), \quad i = 0, 1, \dots, n.$$

将上述线性系统记为矩阵形式： $Au = b$ ，其中刚度矩阵 $A_{ij} = a(\varphi_j, \varphi_i)$ ，右端项 $b_i = (f, \varphi_i) + p(b)\beta \varphi_i(b)$ 。该线性系统可通过高斯消去法或雅可比迭代法进行求解，从而得到有限元近似解 $u_h(x)$ 。

2.2 关键代码解析

1. `f(x)` 与 `exact_u(x)`：

```
def f(x):
    return (2 * pi / (pi**2 - 1)) * np.cos(pi * x) * np.exp(x)

def exact_u(x):
    return np.sin(pi * x) * np.exp(x)
```

`f(x)` 是边值问题右端项函数 $f(x) = \frac{2\pi}{\pi^2 - 1} \cdot \cos(\pi x) \cdot e^x$ ，对应于有限元线性方程组右侧向量 b 中每一项的积分近似值。`exact_u(x)` 给出了该问题的解析解 $u(x) = \sin(\pi x) \cdot e^x$ ，在任务 2 中用于与数值解做误差比较，评估高斯消元法与雅可比迭代法的数值精度与收敛行为。

2. `gauss_solve(A, b)` :

```
def gauss_solve(A, b):  
    return np.linalg.solve(A, b)
```

函数内部调用 NumPy 提供的 `np.linalg.solve()` 接口实现直接解法。

3. `jacobi(A, b, iterations=30)` :

```
def jacobi(A, b, iterations=30):  
    x = np.zeros_like(b)  
    D = np.diag(A)  
    R = A - np.diagflat(D)  
    for _ in range(iterations):  
        x = (b - np.dot(R, x)) / D  
    return x
```

函数实现雅可比迭代法，用于求解线性系统。初始解向量取全零，通过迭代公式：
 $x^{(k+1)} = D^{-1}(b - Rx^{(k)})$ ，其中 D 为对角矩阵， $R = A - D$ 。默认迭代次数为 30。

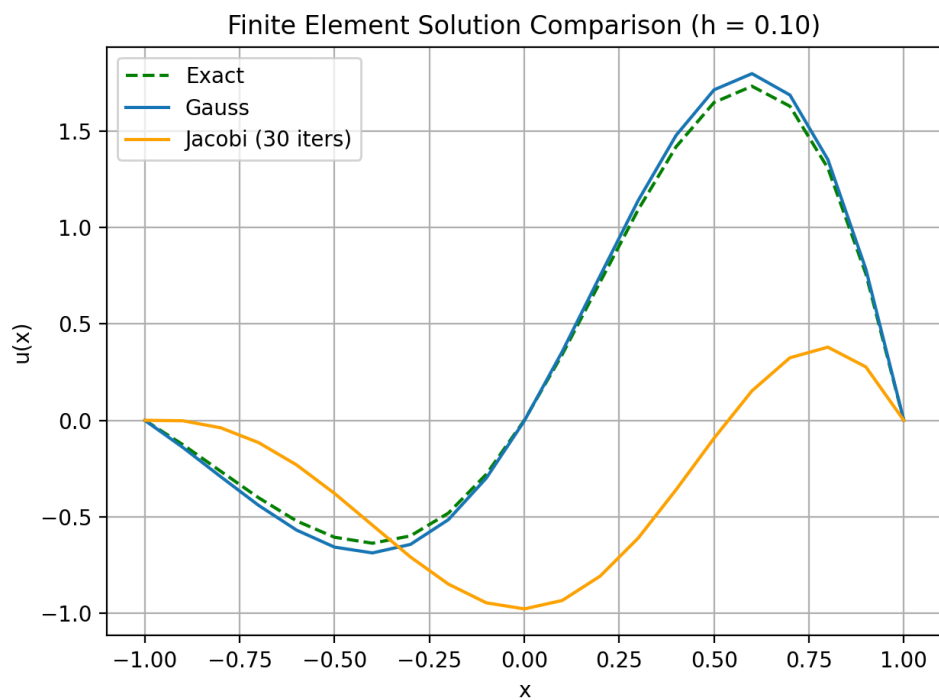
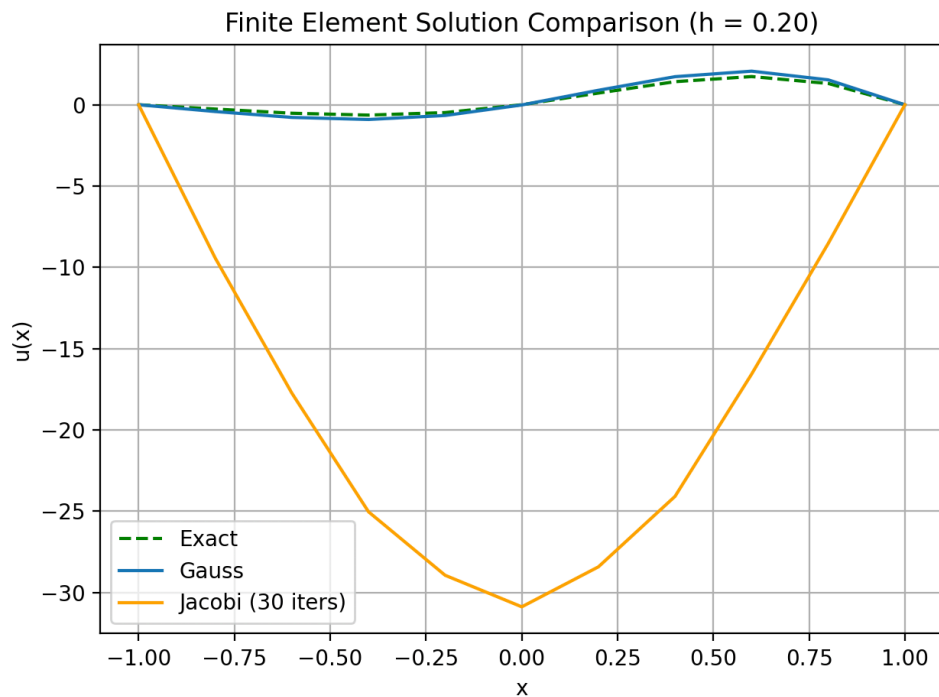
4. `assemble(h)`

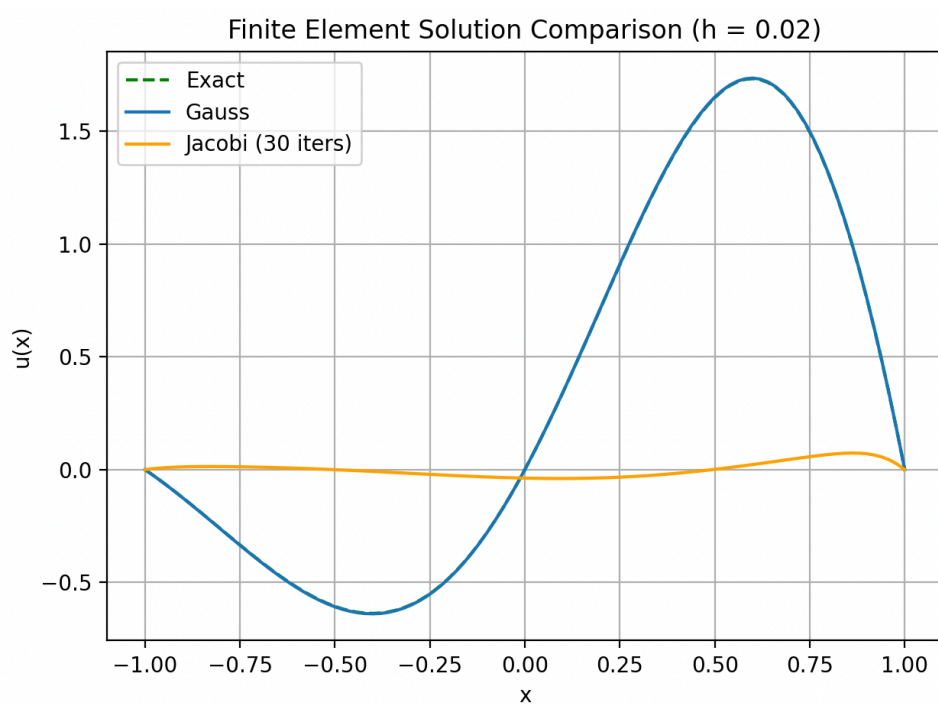
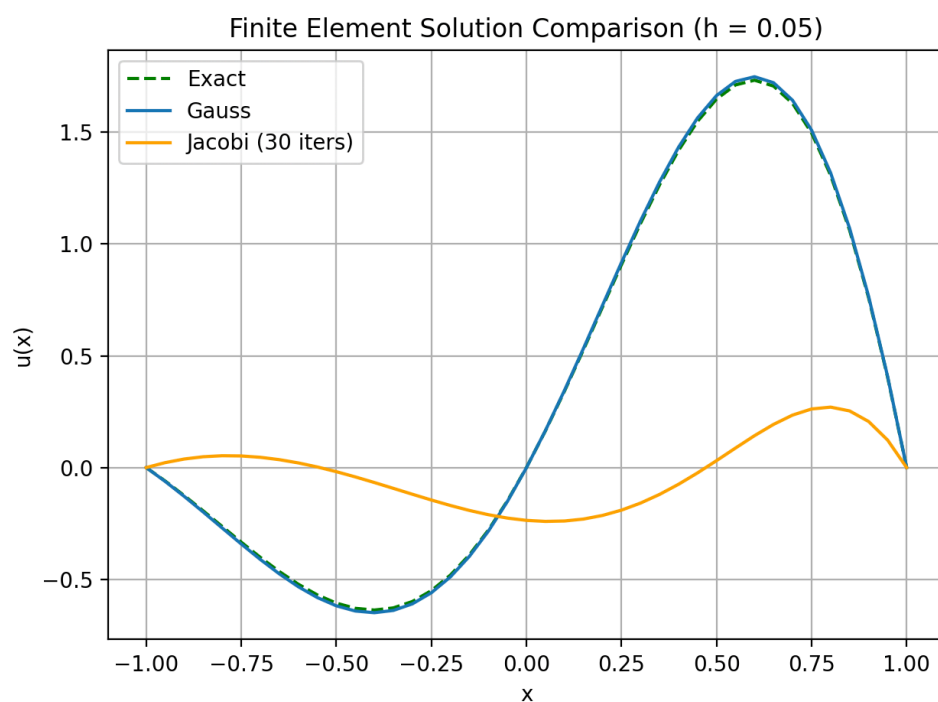
```
def assemble(h):  
    N = int((b - a) / h)  
    x = np.linspace(a, b, N + 1)  
    A = np.zeros((N + 1, N + 1))  
    rhs = np.zeros(N + 1)  
  
    for i in range(1, N):  
        A[i, i-1] = p_val / h  
        A[i, i] = -2 * p_val / h + q_val * h  
        A[i, i+1] = p_val / h  
        rhs[i] = h * f(x[i])  
  
    # 边界条件  
    A[0, 0] = 1  
    rhs[0] = alpha  
    A[N, N] = 1  
    rhs[N] = exact_u(b)  
  
    return A, rhs, x
```

函数根据步长 h 构造有限元离散系统。利用线性基函数构建刚度矩阵 A 和右端向量 b ，采用差分近似构造三对角矩阵结构。边界条件通过修改矩阵首尾项实现 Dirichlet 强制赋值。主对角线： $-\frac{2p}{h} + qh$ ，上/下对角线： $\frac{p}{h}$ ，右端项中 $b_i = h \cdot f(x_i)$ ，函数返回构造完成的系数矩阵 A 、右端向量 rhs ，以及节点向量 x 。

三、结果分析

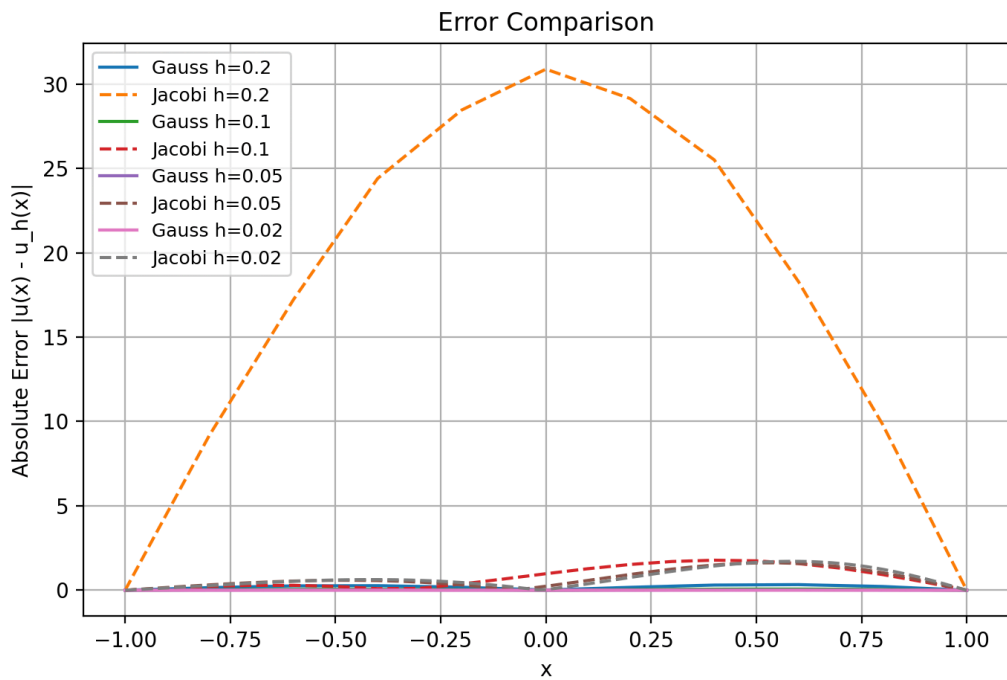
任务1：





在不同步长 h 下，通过高斯消去法和雅可比迭代法计算出的有限元数值解呈现出一致趋势：随着步长减小，解的精度显著提升。高斯消去法的数值解在各步长下表现稳定，能够较好地逼近精确解的整体曲线形状，误差随之收敛，并在 $h = 0.02$ 时几乎与精确解重合。相比之下，雅可比迭代法在 30 次迭代限制下，仅在步长较小时解的形状有所改善，但整体仍存在明显偏离，尤其在区间内部误差较大，数值解明显偏低，收敛速度不足。

任务2：



从误差图像可清晰观察到，高斯消去法在不同步长下均保持高度精度，误差整体平滑且逐步下降，最大误差集中在中间区间，且随 h 减小时迅速趋近于零，验证了有限元法结合直接求解在线性系统上的高阶精度与全局收敛性。特别是在 $h = 0.02$ 时，高斯消去法的误差曲线几乎贴近于零基线。

雅可比法在迭代步数固定为 30 的情况下则存在显著局限。在较大步长时，误差曲线整体偏离严重，峰值超过 30，说明其解尚未充分收敛。尽管在 $h = 0.05$ 与 $h = 0.02$ 时误差有所下降，但误差仍显著高于高斯消去法。这说明雅可比迭代法在有限迭代步内难以确保数值精度，尤其对高精度计算任务不具备竞争力。

此外，雅可比迭代法的误差分布明显集中于中段，边界误差较小，反映出其松弛迭代在两端较快收敛、中心区域收敛缓慢的特征。这一现象可从其迭代矩阵的谱半径分析得到解释。由于该问题对应的刚度矩阵对称正定但谱半径可能较大，导致雅可比迭代法在迭代次数不足的情况下收敛速度极慢，甚至接近发散，呈现出低效迭代特征。

综上，实验图像与误差对比结果充分说明高斯消去法在解精度与稳定性方面表现更优，而雅可比迭代法受限于谱半径与迭代次数，对问题规模及收敛需求更为敏感，不适合作为高精度近似手段。

四、总结

本次实验围绕非齐次边值问题的有限元求解展开，采用高斯消去法与雅可比迭代法分别实现 Ritz-Galerkin 离散后线性方程组的求解，并通过多组网格步长对比误差表现。从运行结果看出高斯消去法精度更高、收敛更快，而雅可比迭代法在步长较小时误差明显增大。在实验过程中，我不仅掌握了有限元框架的实现细节，也了解到不同解法在数值稳定性与迭代效率上的权衡，进一步提升了对并行数值方法实际应用的理解。