# Helmholtz 方程数值解：理论推导与算法实现

周晟煜 刘棋

2025 年 12 月 24 日

## 目录

# 1 引言

Helmholtz 方程是数学物理中描述波动现象的基本方程，在声学、电磁学、量子力学等领域有广泛应用。本报告研究二维 Helmholtz 特征值问题的数值解法，包括数学推导、有限差分离散化、基础求解器实现（幂法和反迭代法）、高级方法（Lanczos 算法和 Rayleigh 商迭代）以及数值实验分析。

# 2 数学推导

考虑二维 Helmholtz 特征值问题：

$$-\Delta u = \lambda u \quad \text{在 } \Omega = [0,1] \times [0,1] \tag{1}$$

带有 Dirichlet 边界条件：

$$u|_{\partial\Omega} = 0 \tag{2}$$

我们采用均匀网格离散化的方法：取网格间距：$\Delta x = \Delta y = h = \frac{1}{N+1}$ 网格点：$x_i = i\Delta x,\ y_j = j\Delta y,\ i,j = 0,1,\ldots,N+1$ 内部点数共 $N^2$ 个。

利用五点差分格式，对拉普拉斯算子的离散采用中心差分：

$$-\Delta u(x_i, y_j) \approx -\frac{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}}{h^2} \tag{3}$$

其中 $u_{i,j} = u(x_i, y_j)$。

令 $U \in \mathbb{R}^{N^2}$ 为按行优先排列的未知向量，离散化后的特征值问题为：

$$AU = \lambda U \tag{4}$$

其中 $A \in \mathbb{R}^{N^2 \times N^2}$ 是稀疏对称矩阵。

采用行优先排序：

$$k = i + (j-1)N, \quad i,j = 1,\ldots,N \tag{5}$$

其中 $(i,j)$ 对应网格点 $(x_i, y_j)$。矩阵 $A$ 的元素为：主对角线：$A_{k,k} = \frac{4}{h^2}$ . 水平相邻：若 $i < N$，则 $A_{k,k+1} = -\frac{1}{h^2}$；若 $i > 1$，则 $A_{k,k-1} = -\frac{1}{h^2}$. 垂直相邻：若 $j < N$，则 $A_{k,k+N} = -\frac{1}{h^2}$；若 $j > 1$，则 $A_{k,k-N} = -\frac{1}{h^2}$

矩阵 $A$ 可表示为分块三对角形式：

$$A = \frac{1}{h^2}\begin{bmatrix} T & -I & & & \\ -I & T & -I & & \\ & \ddots & \ddots & \ddots & \\ & & -I & T & -I \\ & & & -I & T \end{bmatrix} \tag{6}$$

其中：

$$T = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad I = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix} \in \mathbb{R}^{N\times N} \tag{7}$$

# 3 基础求解器实现

## 3.1 幂法（Power Method）

幂法用于求解模最大的特征值（主特征值）。

### 3.1.1 算法步骤

```
function [lambda, v, k] = power_method_algorithm(A, epsilon, K)
    n = size(A, 1);
    v_old = randn(n, 1);
    v_old = v_old / norm(v_old);
    for k = 1:K
        w = A * v_old;
        v_new = w / norm(w);
        lambda_k = v_new' * A * v_new;
        if norm(v_new - v_old) < epsilon
            fprintf('在第%d次迭代后收敛\n', k);
            break;
        end
        v_old = v_new;
    end
    lambda = lambda_k;
    v = v_new;
    if k == K
        warning('达到最大迭代次数%d，可能未收敛到所需精度', K);
    end
end
```

Listing 1: 幂法 MATLAB 程序

### 3.1.2 收敛性分析

- 收敛速度：$O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right)$
- 每步时间复杂度：矩阵-向量乘法 $O(N^2)$
- 总时间复杂度：$O(kN^2)$
- 仅适用于 $|\lambda_1| > |\lambda_2| \geq \cdots$ 的情况

## 3.2 反迭代法（Inverse Iteration）

反迭代法用于求解最接近给定值 $\mu$ 的特征值。

### 3.2.1 算法步骤

```
function [lambda, v, iter_log] = inverse_iteration(A, mu, tol,
   max_iter)
    n = size(A, 1);
    v = randn(n, 1);
    v = v / norm(v);
    I = speye(n);
    [L, U, P, Q] = lu(A - mu * I);
    iter_log = [];
    for k = 1:max_iter
        w = Q * (U \ (L \ (P * v)));
        v_new = w / norm(w);
        lambda = v_new' * A * v_new;
        iter_log = [iter_log; lambda];
        if norm(v_new - v) < tol
            return;
        end
        v = v_new;
    end
end
```

Listing 2: 逆幂法 MATLAB 程序

### 3.2.2 收敛性分析

- 收敛速度：$O\left(\left|\frac{\lambda_i-\mu}{\lambda_j-\mu}\right|^k\right)$，其中 $\lambda_j$ 是第二接近 $\mu$ 的特征值

- 每步时间复杂度：求解线性系统 $O(N^3)$（直接法）或 $O(N^2)$（迭代法 + 预处理）
- 当 $\mu$ 接近特征值时，收敛速度极快（二次收敛）

# 4 理论复杂度分析

## 4.1 时间复杂度分析

### 4.1.1 幂法复杂度详细分析

幂法的单步迭代包含以下操作：

1. **矩阵-向量乘法**：$w = Av$
2. **向量归一化**：$v_{\text{new}} = w/\|w\|_2$
3. **Rayleigh 商计算**：$\lambda = v_{\text{new}}^T A v_{\text{new}}$
4. **收敛性检查**：$\|v_{\text{new}} - v\|_2$

对于二维 Helmholtz 离散矩阵 $A \in \mathbb{R}^{n\times n}$，其中 $n = N^2$，分析各步的浮点操作次数：

1. **矩阵-向量乘法**：矩阵 $A$ 每行最多有 5 个非零元素（五点差分格式），因此：

$$\text{FLOPs}_{\text{matvec}} = 5n = 5N^2$$

2. **向量归一化**：需要计算 $l_2$ 范数和 $n$ 次除法：

$$\text{FLOPs}_{\text{norm}} = n + n = 2N^2$$

3. **Rayleigh 商计算**：可重用 $Av$ 的结果，只需计算一次点积：

$$\text{FLOPs}_{\text{rayleigh}} = n = N^2$$

4. **收敛性检查**：

$$\text{FLOPs}_{\text{check}} = n = N^2$$

因此，幂法单次迭代的总浮点操作次数为：

$$\text{FLOPs}_{\text{iter}} = (5 + 2 + 1 + 1)N^2 = 9N^2$$

### 4.1.2 幂法迭代次数分析

幂法的收敛速度由特征值比值 $\rho = \left|\frac{\lambda_2}{\lambda_1}\right|$ 决定。对于 Helmholtz 离散矩阵，特征值的渐近性质为：

$$\lambda_1 \approx 2\pi^2, \quad \lambda_{\max} \approx \frac{8}{h^2} = 8(N+1)^2$$

特征值比值的典型范围为 $\rho \in [0.9, 0.98]$。

要达到精度 $\varepsilon$ 所需的迭代次数 $k$ 满足：

$$\left(\frac{\lambda_2}{\lambda_1}\right)^k \leq \varepsilon \quad \Rightarrow \quad k \geq \frac{\log \varepsilon}{\log \rho}$$

取 $\rho = 0.95$，则：

$$k \approx -20 \log_{10} \varepsilon$$

例如，对于 $\varepsilon = 10^{-8}$，$k \approx 160$ 次迭代。

因此，幂法的总时间复杂度为：

$$T_{\text{power}}(N) = 9kN^2 = O(kN^2) = O(N^2 \log(1/\varepsilon))$$

### 4.1.3 反迭代法复杂度详细分析

反迭代法的核心是每步求解线性系统：

$$(A - \mu I)w = v$$

我们考虑两种求解策略：

1. **直接法（LU/Cholesky 分解）：**

   对于带状矩阵 $A$（带宽为 $N$），分解的复杂度为：

   $$\text{FLOPs}_{\text{factor}} = O(n \cdot \text{bandwidth}^2) = O(N^2 \cdot N^2) = O(N^4)$$

   每次求解的复杂度（前向/回代）：

   $$\text{FLOPs}_{\text{solve}} = O(n \cdot \text{bandwidth}) = O(N^2 \cdot N) = O(N^3)$$

   总时间复杂度（一次分解 $+k$ 次求解）：

   $$T_{\text{inv-direct}}(N) = O(N^4 + kN^3)$$

2. **迭代法（预条件共轭梯度法 PCG）：**

   矩阵 $A - \mu I$ 的条件数：

   $$\kappa(A - \mu I) \approx O\left(\frac{\lambda_{\max} - \mu}{\lambda_{\min} - \mu}\right) = O(N^2)$$

PCG 的收敛迭代次数：

$$k_{\text{PCG}} = O\left(\sqrt{\kappa}\right) = O(N)$$

每次 PCG 迭代的复杂度（矩阵-向量乘法）：

$$\text{FLOPs}_{\text{PCG-iter}} = O(N^2)$$

每次反迭代的 PCG 求解复杂度：

$$\text{FLOPs}_{\text{PCG-solve}} = k_{\text{PCG}} \times \text{FLOPs}_{\text{PCG-iter}} = O(N \cdot N^2) = O(N^3)$$

预条件构造复杂度：

$$\text{FLOPs}_{\text{precond}} = O(N^3) \quad \text{(不完全 Cholesky 分解)}$$

总时间复杂度：

$$T_{\text{inv-PCG}}(N) = O(N^3 + kN^3) = O(kN^3)$$

#### 4.1.4 收敛速度对比

反迭代法的收敛速度由下式决定：

$$\left|\frac{\lambda_i - \mu}{\lambda_j - \mu}\right|^k$$

其中 $\lambda_i$ 是最接近 $\mu$ 的特征值，$\lambda_j$ 是第二接近的特征值。

当 $\mu$ 接近 $\lambda_i$ 时，收敛速度极快：

- 通常只需 3-10 次迭代即可收敛到机器精度
- 局部二次收敛性质

### 4.2 空间复杂度分析

#### 4.2.1 存储需求详细分析

1. **矩阵存储：**
   - **稀疏格式（CSR）：**存储非零元素值、列索引和行指针

     $$S_{\text{sparse}} = (5n + n + n) \times \text{sizeof(double)} = 7N^2 \times 8\text{字节}$$

   - **稠密格式：**

     $$S_{\text{dense}} = n^2 \times \text{sizeof(double)} = N^4 \times 8\text{字节}$$

2. **向量存储：**

   $$S_{\text{vectors}} = 3n \times \text{sizeof(double)} = 3N^2 \times 8\text{字节}$$

3. **分解存储（反迭代直接法）：**

   $$S_{\text{factor}} = O(n \cdot \text{bandwidth}) = O(N^3)\text{字节}$$

### 4.2.2 空间复杂度总结表

表 1: 各算法空间复杂度详细分析

| 算法 | 矩阵存储 | 向量存储 | 分解存储 | 总存储 |
|---|---|---|---|---|
| 幂法（稀疏） | $O(N^2)$ | $O(N^2)$ | - | $O(N^2)$ |
| 幂法（稠密） | $O(N^4)$ | $O(N^2)$ | - | $O(N^4)$ |
| 反迭代（直接法） | $O(N^2)$ | $O(N^2)$ | $O(N^3)$ | $O(N^3)$ |
| 反迭代（PCG） | $O(N^2)$ | $O(N^2)$ | $O(N^2)$ | $O(N^2)$ |

# 5 结论与展望

本报告系统研究了 Helmholtz 方程特征值问题的数值解法，包括：

1. 建立了有限差分离散化方法，推导了离散矩阵的构造;
2. 实现了基础求解器（幂法、反迭代法）的实现;
3. 分析了各算法的理论复杂度、收敛性和精度;