

多重网格-五点差分-共轭梯度法 ——求解二维泊松方程的一类边值问题： 电场与电荷相互推导的数值方法

欧纪阳 2019141220016

College of Physics, Sichuan University, Chengdu 610064, China

2021 年 5 月 13 日

摘要

本文报告了一种在二维空间中求解泊松方程的一类边值问题的方法——多重网格-五点差分-共轭梯度法，该算法为有限差分法的改进方法，利用五点差分法作为数值微分近似，构建对角线稀疏矩阵，以共轭梯度方法为线性系统求解器，采用 V 循环多重网格方法加速迭代。本文还展示了用 Python 求解在有限矩形区域内给定电荷分布，并已知电势的边值的条件下的，该区域内的电势和电场分布。相较于高斯-赛德尔、过松弛迭代等求解算法，其收敛速度更快，精度更高。在复杂的边值条件下，与普通网格下的共轭梯度迭代相比，可快速降低误差。

关键词：多重网格法 共轭梯度法 稀疏矩阵 泊松方程 边值问题 Python

目录

| | |
|------------------------|----------|
| 1 介绍 | 2 |
| 2 方法 | 2 |
| 2.1 五点差分方法 | 2 |
| 2.2 系数矩阵的组装 | 3 |
| 2.3 添加一类边值条件 | 4 |
| 2.4 共轭梯度方法 | 4 |
| 2.5 多重网格方法 | 4 |
| 2.6 可视化与误差分析 | 5 |
| 3 结果 | 5 |
| 3.1 二维正弦电荷分布 | 6 |
| 3.2 点电荷分布 | 6 |
| 3.3 连续体电荷分布 | 6 |
| 4 讨论 | 6 |

1 介绍

2 方法

报告的方法的最终目的是通过在二维空间中求解泊松方程的边值问题，

$$\nabla^2 \varphi = -\frac{\rho}{\epsilon} \Rightarrow \frac{\partial^2 \varphi(x, y)}{\partial x^2} + \frac{\partial^2 \varphi(x, y)}{\partial y^2} = f(x, y) \quad (1)$$

从而通过 $\vec{E} = -\nabla \varphi$ 求解电场强度矢量。(1) 式为一个二阶偏微分方程，计算机无法直接求解，报告的方法实质上是有限差分法的扩展，先通过数值微分将偏微分方程化为线性方程，再在合适的网格上施以合适的线性求解器进行求解。

2.1 五点差分方法

在说明二维问题前，先讨论一维数值微分，对于一个三阶可微函数 $f(x)$ ，根据泰勒定理有，

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f^{(3)}(c_1) \quad (2)$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f^{(3)}(c_2) \quad (3)$$

其中 $x-h < c_2 < x < c_1 < x+h$ ，最后一项为误差项，两式相减可以得到一阶导数的三点中心差分公式，

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{6}f^{(3)}(c) = \frac{f(x+h) - f(x-h)}{2h} + O(x^3) \quad (4)$$

用该公式在离散情况下对 $\vec{E} = -\nabla \varphi$ 做近似可得，

$$\vec{E}_{i,j} = \left(\frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2h_x}, \frac{\varphi_{i,j+1} - \varphi_{i,j-1}}{2h_y} \right) \quad (5)$$

其中 $\{\varphi_{i,j} | 0 \leq i \leq m-1, 0 \leq j \leq n-1\}$ ， h_x, h_y 分别为 x 方向和 y 方向上的网格精度。

类似地，该方法可以得到更高阶的差分公式，对四阶可微函数 $f(x)$ ，根据泰勒定理有，

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f^{(3)}(x) + \frac{h^4}{24}f^{(4)}(c_1) \quad (6)$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f^{(3)}(x) + \frac{h^4}{24}f^{(4)}(c_2) \quad (7)$$

其中 $x-h < c_2 < x < c_1 < x+h$ ，最后一项为误差项，两式相加可以得到二阶导数的三点中心差分公式，

$$f''(x) = \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} - \frac{h^2}{12}f^{(4)}(c) = \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} + O(x^4) \quad (8)$$

对泊松方程中的两个微分算子在离散情况下，分别应用二阶导数的三点中心差分公式，

$$\frac{\partial^2 \varphi_{i,j}}{\partial x^2} = \frac{\varphi_{i-1,j} - 2\varphi_{i,j} + \varphi_{i+1,j}}{h_x^2} \quad \frac{\partial^2 \varphi_{i,j}}{\partial y^2} = \frac{\varphi_{i,j-1} - 2\varphi_{i,j} + \varphi_{i,j+1}}{h_y^2} \quad (9)$$

由于每一个网格的偏微分需要五个网格进行计算，故称为**五点差分法**。由此便得到了应用五点差分法近似后得到的泊松方程，

$$\frac{\varphi_{i-1,j} + \varphi_{i+1,j}}{h_x^2} - 2\left(\frac{1}{h_x^2} + \frac{1}{h_y^2}\right)\varphi_{i,j} + \frac{\varphi_{i,j-1} + \varphi_{i,j+1}}{h_y^2} = f_{i,j} \quad (10)$$

2.2 系数矩阵的组装

由于求解的方程是二维的，需要将其重塑回到一维才能求解，

$$u_{i+jm} = a_{i,j}, \{a_{i,j} | 0 \leq i \leq m-1, 0 \leq j \leq n-1\} \quad (11)$$

例如对于一个 $A_{4 \times 4}$ 的矩阵将会变换为，

$$\begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix} \Rightarrow [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15].T \quad (12)$$

对于 (10) 式， φ 与 f 的大小为 (m, n) ，利用以上方法把 (10) 式改写为 $Ax = b$ 的形式，即 $\varphi \rightarrow x, f \rightarrow b, \nabla^2 \rightarrow A$ ，其中系数矩阵 A 为，

$$\left[\begin{array}{ccccccccc} -2(\frac{1}{h_x^2} + \frac{1}{h_y^2}) & \frac{1}{h_x^2} & \cdots & \frac{1}{h_y^2} & \frac{1}{h_x^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_x^2} \\ \frac{1}{h_x^2} & -2(\frac{1}{h_x^2} + \frac{1}{h_y^2}) & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_x^2} \\ \vdots & \vdots \\ \frac{1}{h_y^2} & \vdots \\ \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} \\ \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} \\ \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} \\ \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} \\ \vdots & \vdots \\ \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} & \cdots & \frac{1}{h_x^2} \\ \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} & \cdots & \frac{1}{h_y^2} \end{array} \right] \quad (13)$$

那么做变换后得到的 x, b 的大小为 $(1, mn)$ ，系数 A 矩阵的大小为 (mn, mn) ，矩阵中同一行同一列的 $\frac{1}{h_x^2}$ 与 $\frac{1}{h_y^2}$ 之间的间距为 $m-1$ ，即矩阵中的 \cdots 表示占据 $m-2$ 个元素，并且 1 级、-1 级副对角线上的元素，从第 m 个元素开始，每隔 $m-1$ 个元素应该为零 ((13) 式中没有体现出来)。例如对于一个 4×4 的网格，假设 $h_x = h_y = 1$ ，其系数矩阵 A 写作，

$$\left[\begin{array}{ccccccccc} -4 & 1 & & 1 & & & & & \\ 1 & -4 & 1 & & 1 & & & & \\ & 1 & -4 & 1 & & 1 & & & \\ & & 1 & -4 & 0 & & 1 & & \\ 1 & & 0 & -4 & 1 & & 1 & & \\ & 1 & & 1 & -4 & 1 & & 1 & \\ & & 1 & & 1 & -4 & 1 & & 1 \\ & & & 1 & & 1 & -4 & 0 & \\ & & & & 1 & & 0 & -4 & 1 \\ & & & & & 1 & & 1 & \\ & & & & & & 1 & -4 & 1 \\ & & & & & & & 1 & \\ & & & & & & & & 1 \\ & & & & & & & & & 1 \\ & & & & & & & & & & 1 \end{array} \right] \quad (14)$$

显然该矩阵为一个稀疏矩阵，只在主对角线、 $1, -1, m, -m$ 级副对角线五条对角线上有元素，因此在程序中构建此矩阵，并将其储存到内存中时，将应用对角稀疏矩阵方法，即内存器只储存非零对角线上的元素，在编写中使用 `scipy.sparse.dia_matrix` 库进行矩阵的存储与运算。

2.3 添加一类边值条件

对于(13)式中的矩阵，如果没有加边界条件的话，求解过程会非常缓慢，而且最终的求解结果显示，其等价的边界条件为四个边界均满足 $\varphi = 0$ ，因此在求解之前应施加一定的边界条件。在经改写后得到的线性方程组中添加一类边界，($\varphi = \phi(q)$ ，其中 q 为边界上的坐标)，的方法十分简单，只需在系数矩阵 A 中找到边界点对应的行，将对角线上的元素该写为 1，再将 b 中对应的元素改写为对应的 $\phi(q)$ 即可。

2.4 共轭梯度方法

在众多的线性方程求解器中，目前共轭梯度方法是求解稀疏矩阵最优化的方法，特别是在引入有效的预条件后，许多其他方法难以处理的问题使用该方法可以高效的解决，其收敛速度远高于高斯-赛德尔、过松弛迭代等方法，因此在求解线性近似的偏微分方程时，首选使用共轭梯度方法，可以在有限步骤获得线性系统 $Ax = b$ 的解，其伪代码如算法1。

```

Input:  $A$ : 系数矩阵
         $b$ : 常数项
         $x_0$ : 初始预测值
1  $d_0 = r_0 = b - Ax_0$ 
2 for  $k = 0, 1, 2, \dots, n-1$  do
3   if  $r_k = 0$  then
4     | break
5   end
6    $\alpha_k = \frac{r_k^T r_k}{d_k^T A d_k}$ 
7    $x_{k+1} = x_k + \alpha_k d_k$ 
8    $r_{k+1} = r_k - \alpha_k A d_k$  , 计算余项
9    $\beta = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$ 
10 end
```

Output: x_k : 线性方程组的解

Algorithm 1: 共轭梯度方法

算法1通过使用预条件方法，可以使迭代收敛速度加快，迭代方法的收敛率通常直接或间接依赖于系数矩阵 A 的条件数，预条件方法的思想便是降低问题中的条件数。在对称连续过松弛中，预条件子定义为，

$$M = (D + \omega L)D^{-1}(D + \omega U) \quad (15)$$

其中 $A = L + D + U$ 被分解为下三角部分、对角线、以及上三角部分， $0 < \omega < 2$ ，当 $\omega = 1$ 时被称为高斯-赛德尔与条件子。虽然预条件可以大大的加快迭代的速度，但是如果预条件子矩阵难以求逆，一般情况下反而会适得其反，故在本文中运用的是无预条件的共轭梯度方法。

2.5 多重网格方法

在数值分析中，多重网格法是一种使用层次离散化来求解微分方程的方法，是多分辨率方法的一个例子。由于直接在高分辨率（用于求解的间隔小）上进行求解时对于低频部分收敛较慢，与间隔的平方成反比，便想可以到先在低分辨率（间隔较大）上进行求解，然后再进行插值，提高其分辨率，再在更高分辨率进行计算，这样的方法就叫做多重网格方法，且迭代过程主要包含以下几个重要的组成部分：

- 平滑 (smoothing): 用线性求解器进行迭代，降低高频误差。

- 计算残差 (Residual Computation): 在平滑之后计算剩余误差 (残差) $r_i = b - Ax_i$ 。
- 限制 (Restriction): 降低采样率, 把残差放到更加粗糙的网格中。
- 延拓 (prolongation): 将粗网格计算的修正值插值到更精细的网格中
- 修正 (Correction): 将延拓的修正值添加到精细网格中。

多重网格方法有多种变体, 但均包含以上步骤, 在本文中运用的是最基础的 **V 循环多重网格方法**, 因其迭代过程为逐次预光滑降低到最粗糙的网格中, 再逐次求解插值到最精细的网格中, 所形成的迭代图像为一个“V”字形, 故得名 V 循环多重网格方法, 其重要步骤如算法2:

Input: φ : 初始预测值
 f : 常数项
 h : 最精细网格精度
 h_m : 最粗糙网格精度

```

1 Function VCycle( $\varphi, f, h$ ):
2    $\varphi = \text{smoothing}(\varphi, f, h)$  ' 预平滑 (pre-smoothing)
3    $res = \text{residual}(\varphi, f, h)$ 
4    $rhs = \text{restriction}(res)$ 
5    $eps = \text{zeros}(\text{size}(rhs))$ 
6   if  $h <= h_m$  then
7     | VCycle( $eps, rhs, 2h$ )
8   else
9     |  $\text{smoothing}(eps, rhs, 2h)$ 
10  end
11   $\varphi = \varphi + \text{prolongation}(eps)$ 
12   $\varphi = \text{smoothing}(\varphi, f, h)$  ' 后平滑 (post-smoothing) return  $\varphi$ 
13 end

```

Output: φ : 线性方程组的解

Algorithm 2: 多重网格方法

算法2中, 需要先对进入迭代的预测解 φ 进行一次预平滑, 即使用求解器对其进行较少次数的迭代, 使其较靠近精确解, 本文使用的求解器为共轭梯度方法, 在编写中一般要求在预平滑中迭代 3 ~ 5 次即可; 插值后进行的后平滑一般按照设定的精度进行, 每相邻两级网格之间的网格精度一般是翻一倍, 对于求解精度每一级网格设置为相同即可; 对于限制算子 (Restriction) 和延拓 (Prolongation) 均采用双线性插值方法, 其为一维两点线性插值的二维推广方法, 限于篇幅, 此处不再过多赘述。

2.6 可视化与误差分析

对于电势的可视化, 本文使用 matplotlib.pyplot 库中, 利用不同颜色来标注大小的 pcolor 函数来绘制电势在空间中的分布, 再使用 contour 函数绘制其等势线; 对于电场的可视化, 将在运用 pcolor 函数来绘制电场强度大小在空间中的分布之外, 利用 quiver 函数辅以 19×19 个箭头来表示电场强度矢量的方向。误差分析方面, 将对一组拥有解析解情况的泊松方程进行数值计算, 并与精确解进行对比, 计算均方误差 RMS。

3 结果

该部分展示了一些数值计算的例子, 为了证明该方法的有效性, 将会给予一组具有解析解的电荷分布及边值条件进行求解, 并与理论上的解析解的结果进行对比, 除此之外还将展示点电荷、电偶极子、带点

细棍等所产生的电场在有限空间中的分布。

3.1 二维正弦电荷分布

利用多重网格-五点差分-共轭梯度法和直接共轭梯度方法（只在一级网格中进行迭代）对以下边值问题进行求解，

$$\left\{ \begin{array}{l} \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = -\frac{\rho}{\varepsilon} = -2\pi^2 \sin(\pi x) \sin(\pi y) \\ \varphi = 0, x = \pm 1 \end{array} \right. \quad (16a)$$

$$\left\{ \begin{array}{l} \varphi = 0, y = \pm 1 \end{array} \right. \quad (16b)$$

$$\left\{ \begin{array}{l} \varphi = 0, y = \pm 1 \end{array} \right. \quad (16c)$$

显然该系统具有解析解，并可推导得到电场强度矢量的表达式，

$$\varphi = \sin(\pi x) \sin(\pi y) \quad E = \pi \cos(\pi x) \sin(\pi y) \vec{e}_x + \pi \sin(\pi x) \cos(\pi y) \vec{e}_y \quad (17)$$

在进行数值计算时，设置求解区域为 $S = \{(x, y) | -1 \leq x \leq 1, -1 \leq y \leq 1\}$ 的正方形区域，在两个维度上的网格精度均为 10^{-3} ，即求解区域的网格大小为 2000×2000 ，调节求解参数，多重网格等级为 3，预平滑迭代次数为 3，后平滑迭代求解精度为 10^{-10} 。最终求解结果如图1(a) 所示，实际计算误差 $RMS = 4.112377 \times 10^{-7}$ ，数值的后平滑迭代过程在三级网格中经历了 1 次，在二级网格中经历了 7 次，在一级网格中经历了 43 次，包括预平滑、限制、插值、后平滑等所有过程在内的 10 次平均总耗时约为 6.89692s，与直接共轭梯度方法的 55 次迭代，10 次平均总耗时 10.2639s 相比，误差降低的速度要高出许多，误差下降曲线如图1(b) 所示，蓝色曲线从最高一级粗网格后迭代开始记录误差，从第九个数据点开始回到一级精细网格，虽然此处误差较大，但是可以以很快的速度降低误差，因为前 8 次迭代是在粗化的网格下进行，所以所报告的算法总体效率要高于直接共轭梯度法。

3.2 点电荷分布

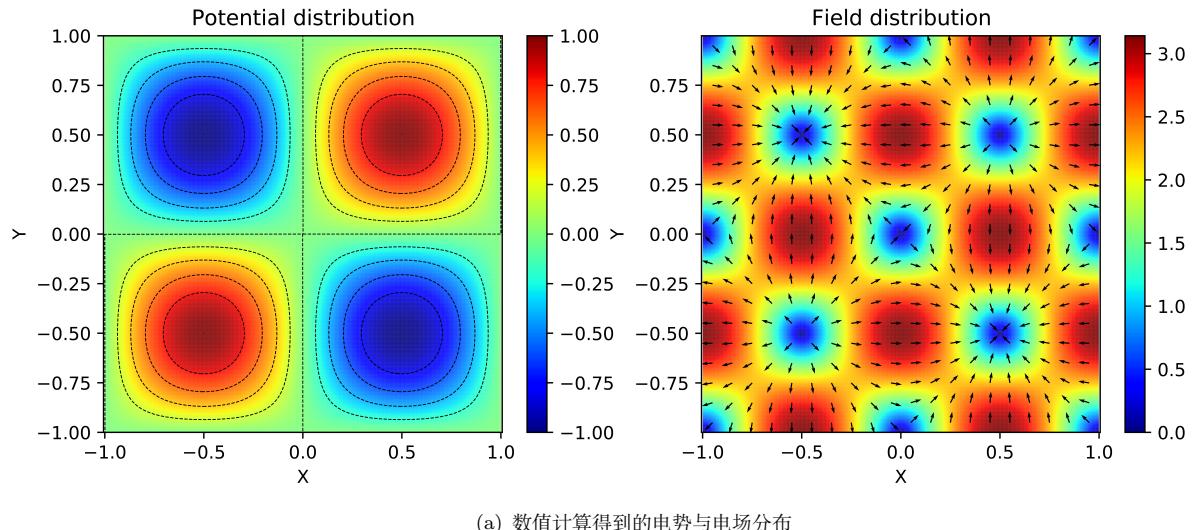
设置求解区域为 $S = \{(x, y) | -1 \leq x \leq 1, -1 \leq y \leq 1\}$ 的正方形区域，在两个维度上的网格精度均为 2×10^{-3} ，即求解区域的网格大小为 1000×1000 ，调整多重网格等级为 4，预平滑迭代次数为 4，后平滑迭代求解精度为 10^{-5} 。在坐标原点放置一个 $Q/\varepsilon = 1/h_x h_y = 0.25 \times 10^6$ 的点电荷，边界条件电势为零，即 (16b)(16c) 式，计算其电场分布，迭代结果如图2(a) 所示。在相同的网格和求解参数下，另取两个绝对值相等的正负电荷组成电偶极子，分别放置在 $(-0.2, 0), (0.2, 0)$ 的位置进行求解，其结果如图2(b) 所示。

由于该问题难以给出解析解，在此处只给出定性的法分析，可以看到在正电荷处，电场方向背离电荷，在负电荷处电场方向指向电荷，在越靠近电荷的位置电场越强；在靠近边界处，电场方向均垂直于边界，符合所给定的边界条件，边界附近电场方向变化大的位置，电场强度大小较小。

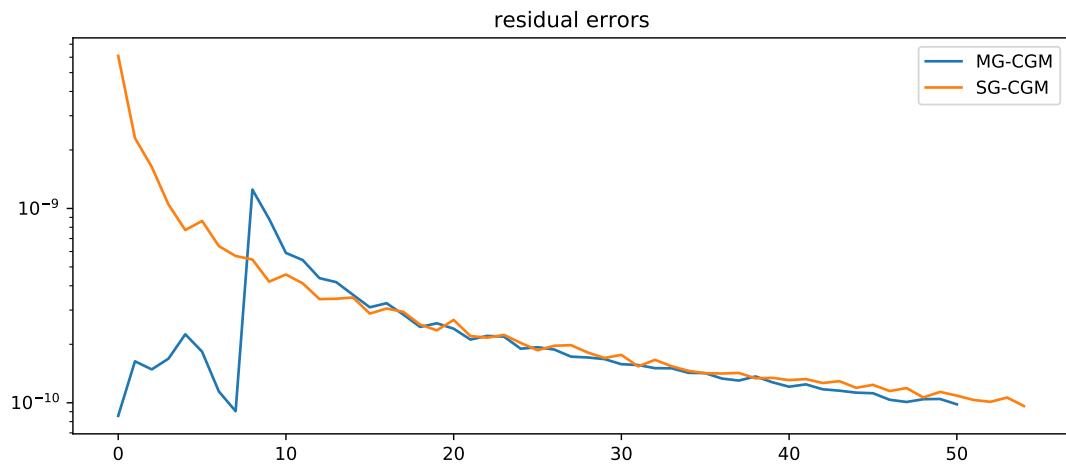
3.3 连续体电荷分布

求解区域以及迭代参数与点电荷分布的数值实验设置相同，放置一根长 0.8，中心位于原点的沿 x 方向的带电细棒，电荷密度为 $\rho/(\varepsilon h_x h_y) = 1$ ，边界条件设置为电势为零，即 (16b)(16c) 式，电场的计算结果如图2(3) 所示。在相同的求解参数下，求解三根带电细棒在有限空间中的电场分布，计算结果如图2(4) 所示，第一、二根细棒长 0.6，沿 y 轴方向，电荷密度为 $\rho/(\varepsilon h_x h_y) = 1$ ，中心分别位于 $(-0.4, 0.3), (0.4, 0.3)$ 处，第三根细棒长 0.8，沿 x 轴方向，电荷密度为 $\rho/(\varepsilon h_x h_y) = -2$ ，中心位于 $(0, -0.6)$ 。数值实验的结果符合静电场规律，在边界处电场方向垂直与边界，在带电棒附近，电场方向几乎垂直与细棒。

4 讨论



(a) 数值计算得到的电势与电场分布



(b) 误差下降曲线

(a) 中左图展示的是电势在求解空间中的分布情况，黑色虚线表示等势面；右图展示的是电场强度在求解空间中的分布情况，颜色的不同表示电场强度大小，箭头表示箭头根部所在的位置的电场强度矢量的方向。(b) 为多重网格-共轭梯度法 (MG-CGM) 后迭代过程 (蓝色)、直接共轭梯度法 (SG-CGM) (橙色) 的误差下降曲线。

图 1: 二维正弦电荷分布的数值计算结果

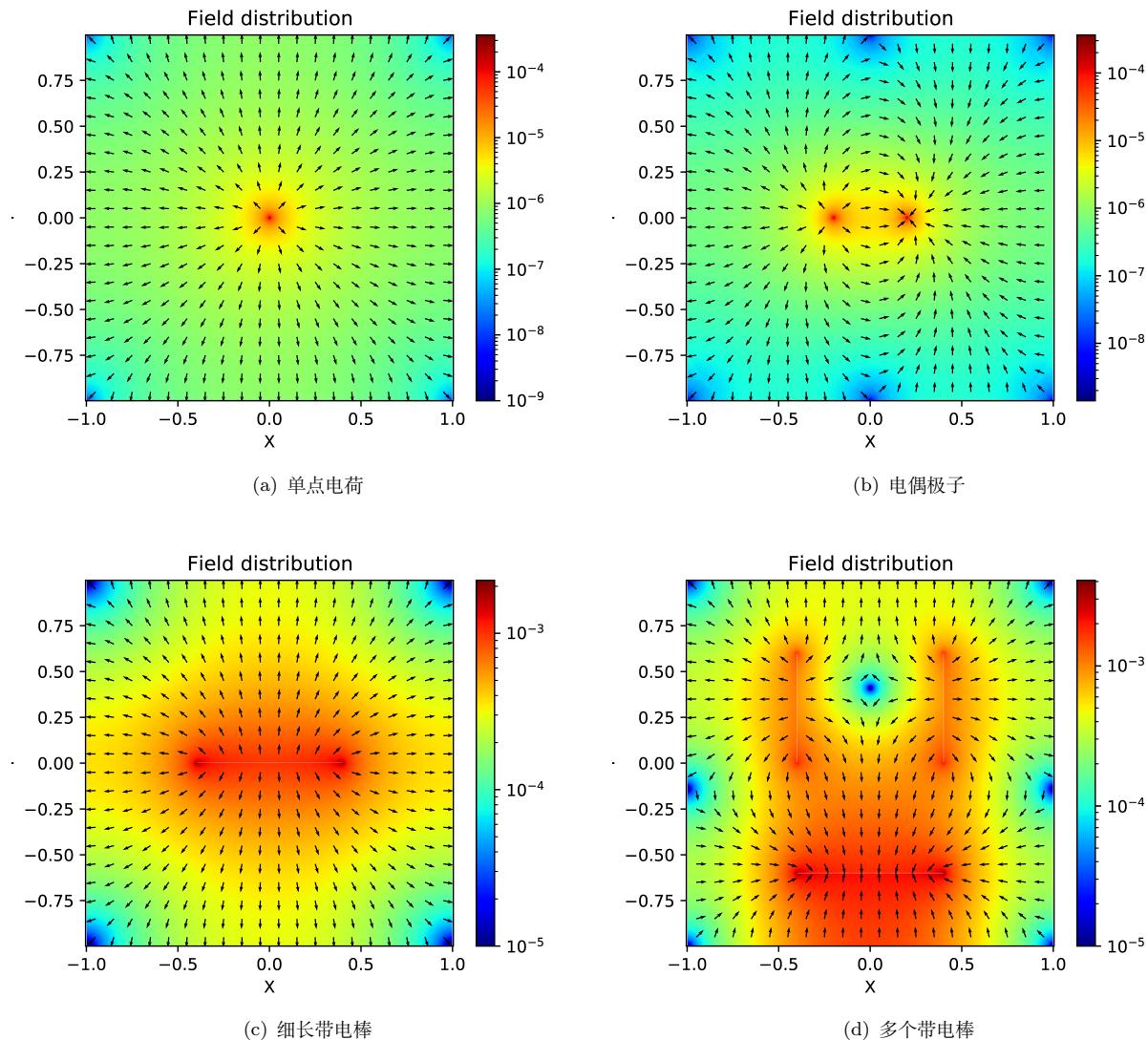


图 2: 二维点电荷分布的数值计算电场可视化