Homework No.8

Ruqi Feng

December 2022

1 Electrostatics

1.1 Description

Consider the Poisson equation

$$\nabla^2\psi = -\frac{\rho(x,y)}{\varepsilon_0} \tag{1}$$

from electrostatics on a rectangular geometry with $x \in [0, L_x]$ and $y \in [0, L_y]$. Write a program that solves this equation using the relaxation method. Test your program with:

- $\rho(x,y) = 0$, $\phi(0,y) = \phi(L_x,y) = \phi(x,0) = 0$, $\phi(\mathrm{x}, L_y) = 1$ V, $L_x = 1$ m, and $L_y = 1.5$ m;
- $\frac{\rho(x,y)}{\varepsilon_0} = 1$ V/m^2, $\phi(0,y) = \phi(L_x,y) = \phi(x,0) = \phi(x,L_y) = 0$, and $L_x = L_y = 1$ m.

1.2 Solution

为了数值解，我们首先离散化 Poisson 方程1

$$h^2\rho_{ij} = u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i-1,j} - 4u_{i,j} \tag{2}$$

其中 h^2/ε_0 是网格面积。

以 2 维为例，如果网格是 $n \times n$ 的，可以按照行优先的方式将 u_{ij} 写成列向量

$$\begin{pmatrix} u_{0,0} & u_{0,1} & \dots & u_{0,n-1} & u_{1,0} & \dots & u_{1,n-1} & \dots & u_{n-1,n-1} \end{pmatrix}^T \tag{3}$$

这意味着方程2的系数矩阵（大小为 $n^2 \times n^2$）是一个分块对角矩阵，分块矩阵的大小约为 $3n \times 3n$。它十分稀疏，直接用 Gauss 消元法求解的时间复杂度达到了 $O(n^6)$，分块使用 Gauss 消元法求解的时间复杂度也高达 $O(3n \times (3n)^3) = O(n^4)$。所以我们采用 Relaxation method 迭代求解上述方程。

根据 Jacobi 法，解的迭代公式

$$u_k^{n+1} = \frac{1}{A_k}\left(b_k - \sum_{k \neq j} A_{ij} u_j^n\right) \tag{4}$$

又由于求和中只有 4 项非零，一次迭代的时间复杂度为 $O(n^2)$。为了获得更快的收敛速率，我使用了 Gauss-Seidel 方法，只需用已经算出的 u_k^{n+1} 代替 u_k^n 来计算 $u_l^{n+1}(l > k)$。

整理后的迭代公式为

$$u_{i,j}^{n+1} = \frac{1}{4}(u_{i+1,j}^n + u_{i-1,j}^{n+1} + u_{i,j+1}^n + u_{i,j-1}^{n+1} + h^2 \rho_{i,j}) \tag{5}$$

所以只需从边界开始，向 i, j 增加的方向逐点求解即可。

1.3 Input and Outputs

按照题目所给的边界条件和 $\rho(x, y)$ 求解 Poisson 方程，取 $\varepsilon_0 = 1$，使用 100×100 的格点和误差 $< 1e - 5$ 的收敛判据，结果如图1和2所示

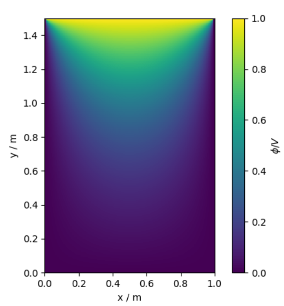

图 1: 仅上边界高电势

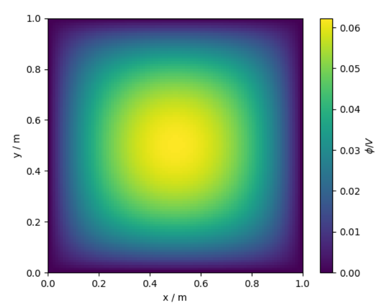

图 2: 均匀分布的面电荷

程序使用了 numpy, matplotlib 库。在安装了依赖库的环境中运行 q1_electrostatics.py 即可。

1.4 Pseudocode

Algorithm 1 Electrosatics

Input: $\rho(x, y)$, boundary conditions $\phi(0, y), \phi(L_x, y), \phi(x, 0), \phi(x, L_y)$, grid number N, *tol*

Output: $\phi_{i,j}$

$\phi_{i,j} = 0$ ▷ Initial guess
Set $\phi_{i,0}, \phi_{i,N}, \phi_{0,j}, \phi_{N,j}$ according to boundary conditions
$\phi_{ij}^{last} \leftarrow \phi_{ij}$
while max(abs($\phi_{ij} - \phi_{ij}^{last}$))> *tol* **do**
    $\phi_{ij}^{last} \leftarrow \phi_{ij}$
    for i in range(N) **do**
        for j in range(N) **do**
            $\phi_{ij} \leftarrow \frac{1}{4}(\phi_{i+1,j}^{last} + \phi_{i-1,j} + \phi_{i,j+1}^{last} + \phi_{i,j-1} + h^2\rho_{i,j})$
        end for
    end for
end while
return $\phi_{i,j}$

2 Time-dependent Schrodinger Equation

2.1 Description

Solve the time-dependent Schrodinger equation using the Crank–Nicolson method and stable explicit scheme. Consider the one-dimensional case and test it by applying it to the problem of a square well with a Gaussian initial state coming in from the left.

2.2 Solution

我们分别使用 Stable Explicit method and Crank-Nocolson 方法求解含时 Schrodinger 方程

2.2.1 Crank-Nicolson method

使用 Forward method 和 Backward method 的线性组合来近似 $\frac{d\psi}{dt}$ 得到

$$\frac{\beta}{\Delta x^2}(u_{i-1,j}-2u_{i,j}+u_{i+1,j})+\frac{1-\beta}{\Delta x^2}(u_{i+1,j-1}-2u_{i,j-1}+u_{i-1,j-1})=\frac{1}{\Delta t}(u_{i,j}-u_{i,j-1}) \tag{6}$$

其中 j 是时间，i 为空间网格坐标。当 $beta=\frac{1}{2}$ 时，$u_{i,j}$ 的迭代关系就可以表达为

$$\vec{u}_j=(2I+\alpha B)^{-1}(2I-\alpha B)\vec{u}_{j-1} \tag{7}$$

其中 $B_{ij}=2\delta_{ij}-\delta_{\|i-j\|,1}$。它总是收敛的，而且有 $O(\Delta x^2+\Delta t^2)$ 的精度。

2.2.2 Stable explicit method

Askar 等人在 1978 年提出 [1]，只需要在 Explicit 方法中的合适位置以上一步的 ψ 替代如下

$$\psi_j^{n+1}=\psi_j^{n-1}+\frac{i\Delta t}{\Delta x^2}(\psi_{j+1}^n+\psi_{j-1}^n-2\psi_j^n)-2i\Delta t V_j\psi_j^n \tag{8}$$

此方法就无条件收敛。**但是**，Rubin 等人在 1979 年指出 [2]，前者的证明含有'trivial' 的错误，所以它事实上是有条件收敛的。

2.3 Input and Outputs

取 $\frac{\hbar^2}{m} = 1$，分别在 3e−2 和 1e−3 的时间步长以及 6e−2 的空间步长下使用 C-N 方法和 Stable[1] explicit 方法求解入射方势阱的高斯波包的演化。

当在 Stable explicit 方法中取时间步长较大时，我们观察到方程的解立即发散了。这和 [2] 中的结论相符合：在没有感受到势能时，收敛条件为 $\frac{\Delta t}{\Delta x^2} < \frac{1}{2}$，和普通的 explicit method 无异。

从图4中可以看出两种方法收敛后给出一致的结果。在时间较大时，波函数有振荡现象，我认为这是反射波 (边界条件相当于完美反射壁) 干涉导致的。

程序使用了 numpy 和 matplotlib 和 scipy 库。在安装了依赖库的环境下运行 `q2_schrodinger_explicit.py` 或 `q2_schrodinger_CN.py` 即可。

2.4 Pseudocode

Algorithm 2 Time Dependent Schrodinger: CN

Input: potential $V(r)$, initial wave function ψ, simulation time t_m

Output: $\psi(x,t)$

$\psi_{i,0} \leftarrow$ input init ψ

$t \leftarrow 0$

$\alpha \leftarrow \frac{dt}{dx^2}$

while $t < t_m$ **do**

  $t \leftarrow t + dt$

  Construct matrix B ▷ according to 7

  $\psi_{i,t} \leftarrow (2I + \alpha B)^{-1}(2I - \alpha B)\psi_{i,t-dt}$

end while

Return: $\psi_{i,t}$

[1]事实上应该不是 stable 的，参见 [2]

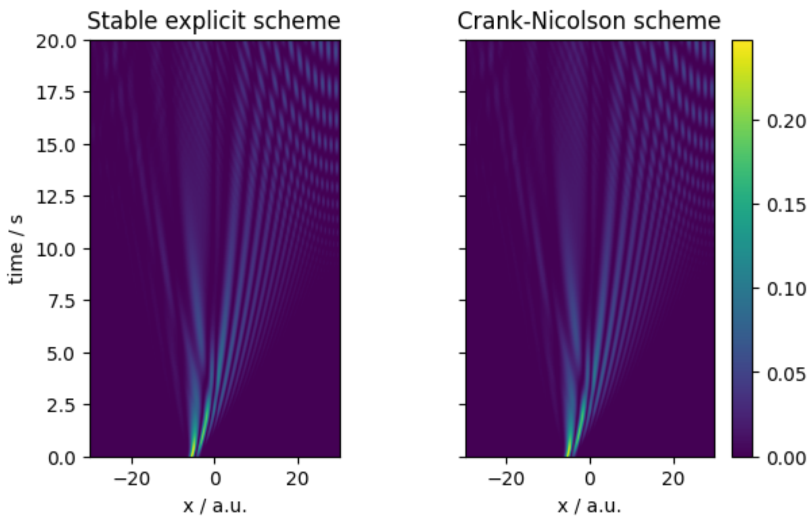

图 3: C-N method and stable explicit method 的结果

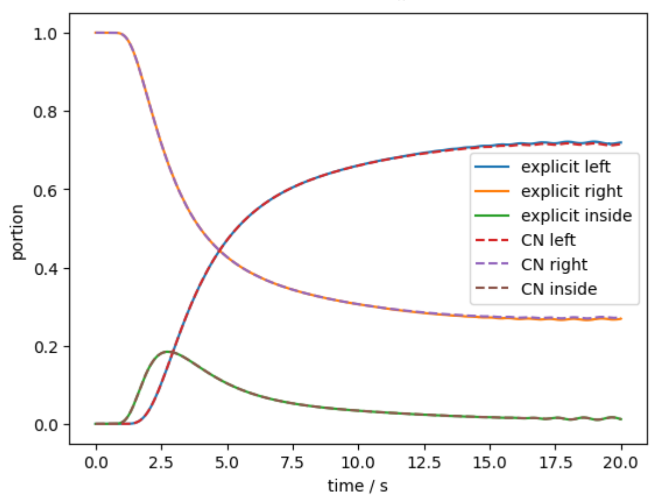

图 4: 两种方法中粒子密度随时间的变化。可以观察到在势阱左侧的粒子逐渐减少，右侧逐渐增加。

2.5 Pseudocode

Algorithm 3 Time Dependent Schrodinger: stable explicit

Input: potential $V(r)$, initial wave function ψ, simulation time t_m

Output: $\psi(x,t)$

$\psi_{i,0}, \psi_{i,-dt} \leftarrow$ input init ψ

$t \leftarrow 0$

$\alpha \leftarrow \frac{dt}{dx^2}$

while $t < t_m$ **do**

  $t \leftarrow t + dt$

  $\psi_{j,t} \leftarrow \psi_{j,t-dt} + \frac{idt}{dx^2}(\psi_{j+1,t-dt} + \psi_{j-1,t-2dt}\psi_{j,t-dt}) - 2idtV_j\psi_{j,t-dt}$

end while

Return: $\psi_{i,t}$

3 Stability of 1D explicit method

As for 1D wave equations, insert the ansatz $u_{x_i,t_j} = \xi^{t_j} e^{iKx_i\Delta x}$ into

$$u_{x_i,t_{j+1}} = 2u_{x_i,t_j} - u_{x_i,t_{j-1}} + \frac{c^2\Delta t^2}{\Delta x^2}(u_{x_{i+1},t_j} - 2u_{x_i,t_j} + u_{x_{i-1},t_j}) \tag{9}$$

which is

$$\xi^{t_{j+1}} e^{iKx_i\Delta x} = 2\xi^{t_j} e^{iKx_i\Delta x} - \xi^{t_{j-1}} e^{iKx_i\Delta x} \tag{10}$$

$$+\alpha^2(\xi^{t_j} e^{iKx_{i+1}\Delta x} - 2\xi^{t_j} e^{iKx_i\Delta x} + \xi^{t_j} e^{iKx_{i-1}\Delta x}) \tag{11}$$

where $\alpha^2 = \frac{c^2\Delta t^2}{\Delta x^2}$. Noting $x_{i+1} = x_i + 1$ and $t_{i+1} = t + 1$, it further simplifies to

$$-\xi + 2 - \xi^{-1} + \alpha^2(e^{iK\Delta x} - 2 + e^{-iK\Delta x}) = 0 \tag{12}$$

Denoting $1 + \alpha^2(e^{iK\Delta x/2} - 1 + e^{-iK\Delta x/2}) = 1 - 2\alpha^2 \sin^2 \frac{K\Delta x}{2}$ as η and the solution yields

$$\|\xi\| = \|\eta \pm \sqrt{\eta^2 - 1}\| \tag{13}$$

When $\frac{c\Delta t}{\Delta x} = \|\alpha\| \leq 1$, $-1 \leq \eta \leq 1$, $\|\xi\|^2 = \eta^2 + 1 - \eta^2 = 1$ and thus convergent. On the other hand, when $\|\alpha\| > 1$, $\|\eta + \sqrt{\eta^2 - 1}\|$ certainly exceeds the boundary.

Therefore, the convergence condition is $\|\alpha\| = \frac{cdt}{dx} \leq 1$

References

[1] Attila Askar and Ahmet S Cakmak. “Explicit integration method for the time-dependent Schrodinger equation for collision problems”. In: *The Journal of Chemical Physics* 68.6 (1978), pp. 2794–2798.

[2] Robert J. Rubin. “Comment on explicit integration method for the time-dependent Schrödinger equation”. In: *The Journal of Chemical Physics* 70.10 (1979), pp. 4811–4811. DOI: 10.1063/1.437245. URL: https://doi.org/10.1063/1.437245.