文章编号： 1672-8785(2010)03-0042-05

通过有限差分和MATLAB矩阵运算直接求解一维薛定谔方程

王忆锋　唐利斌

(昆明物理研究所，云南昆明 650223)

摘　要：根据有限差分法原理，将求解范围划分为一系列等间距的离散节点后，一维薛定谔方程转化为可以用一个矩阵方程表示的节点线性方程组。利用 MATLAB 提供的矩阵左除命令，即可得到各未知节点的函数近似值。该方法概念简单，使用方便，不需要花费较多精力编程即可求解大型线性方程组。

关键词：半导体；量子力学；薛定谔方程；有限差分法； MATLAB

中图分类号： O413.1　　**文献标识码**： A　　**DOI**： 10.3969/j.issn.1672-8785.2010.03.009

Direct Solution of One-dimensional Schrödinger Equation through Finite Difference and MATLAB Matrix Computation

WANG Yi-feng ,　TANG Li-bin

(Kunming Institute of Physics, Kunming Yunnan 650223, China)

Abstract: According to the finite difference principle, a one-dimensional Schrödinger equation can be converted into a set of nodal linear equations expressed in a matrix equation after the space is divided into a series of discrete nodes with an equal interval. The matrix left division command offered in the MATLAB software can be used to derive the function approximation of each unknown nodal function. This method is simple in concept, convenient in operation and can solve large linear equations without more efforts in programming.

Key words: semiconductor; quantum mechanics; Schrödinger equation; finite difference method; MATLAB

1 引言

薛定谔方程是量子力学中最基本的方程，其地位有如经典力学中的牛顿第三定律、电磁学中的麦克斯韦方程。碲镉汞等半导体能带结构的计算就需要求解薛定谔方程。一般情况下，薛定谔方程没有解析解，需要对其做数值计算，其求解也一直作为一个典型的数学物理问题被反复分析讨论，且形成了各种解决方案。这些方案无一例外地要解决两个基本问题，一个是算法设计，另一个是通过编写程序来实现算法。

MATLAB 应用的简捷性不仅体现于特殊函数的计算 [1,2]，也反映在部分数学物理方程的求解过程中 [3]。利用 MATLAB 强大的矩阵计算功能，可以大幅减少求解一维薛定谔方程所需的工作量。

2 一维薛定谔方程的求解过程分析

一维定态薛定谔波动方程如下：

$$\frac{d^2\varphi}{dx^2}+f(x)\varphi=0 \tag{1}$$

收稿日期：2009–10–13

作者简介：王忆锋 (1963-)，男，湖南零陵人，工学士，高级工程师，主要从事器件仿真研究。 E-mail: wangyifeng63@sina.com

式中，$\varphi(x)$ 是波函数 $\varphi(x,t)$ 的空间部分。另有

$$f(x)=\frac{2m}{\hbar^2}[E-V(x)] \tag{2}$$

式中，$V(x)$ 为势能函数，$\hbar$ 为约化普朗克常数，m 为粒子质量，E 为能量。

式 (1) 为泛定方程，利用 MATLAB 的 dsolve( ) 命令，可以求出其通解为

$$\varphi(x)=C_1\sin\left(x\sqrt{f(x)}\right)+C_2\cos\left(x\sqrt{f(x)}\right) \tag{3}$$

泛定方程附加一些条件 (如已知开始运动时的情况或边界上受到的外界约束) 后，就能完全确定运动状态，这样的条件被称为定解条件。利用 MATLAB 的 dsolve( ) 命令容易看出薛定谔方程的定解问题是否有解析解。事实上，能做解析计算的薛定谔方程并不多，它在大多数情况下只能进行数值计算。

薛定谔方程一般是在定解条件下讨论的。表示开始情况的附加条件称为初始条件，相应的定解问题称为初值问题；表示在边界上受到的约束条件称为边界条件，相应的问题称为边值问题。MATLAB 提供了若干个求解一阶微分方程初值问题的函数，其中最常用的是基于变步长四阶五级 Runge-Kutta-Felhberg 算法的 ode45( ) 函数。边值问题无法用 ode45( ) 类函数来直接求解，但是可以借助于打靶法将边值问题转化为初值问题，随后调用 ode45( ) 函数求解。

3 利用 MATLAB 矩阵除法求解一维薛定谔方程

一维薛定谔方程的求解不仅需要给出 $\varphi(x)$ 的边界条件，还需要确定 E 和 $V(x)$。其中，E 为本征值或本征函数。这里讨论 E 和 $V(x)$ 均为已知的情况。

如图 1 所示，以步长 Δx 将区间 $[a,b]$ 等间距离散化为 N 个小区间，并有

$$\Delta x=\frac{b-a}{N} \tag{4}$$

式中，a 和 b 分别为两个边界点的坐标。节点数为 $N+1$，各节点的坐标分别为 $x_1=a$，$x_2=a+\Delta x$，$x_3=a+2\Delta x$，$\cdots$，$x_N=a+(N-1)\Delta x$，$x_{N+1}=b$。去除两端边界上已知的函数值 $\varphi(a)=V_a$ 和 $\varphi(b)=V_b$，于是问题转化为求出在其中 $N-1$ 个节点上的近似值 $\varphi(x_i)$。显然，节点数越多，近似值的精度就越高。

对于节点 i，有

$$\left.\frac{d^2\varphi}{dx^2}\right|_i+f(x_i)\varphi(i)=0,\quad i=2,3,\cdots,N \tag{5}$$

有限差分法的基础是泰勒级数理论。在节点 $(i-1)$ 和节点 $(i+1)$ 上分别作泰勒级数展开，可以写出

$$\varphi(i-1)=\varphi(i)-\Delta x\left.\frac{d\varphi}{dx}\right|_i+\frac{(\Delta x)^2}{2!}\left.\frac{d^2\varphi}{dx^2}\right|_i -\frac{(\Delta x)^3}{3!}\left.\frac{d^3\varphi}{dx^3}\right|_i+\frac{(\Delta x)^4}{4!}\left.\frac{d^4\varphi}{dx^4}\right|_i-\cdots \tag{6}$$

$$\varphi(i+1)=\varphi(i)+\Delta x\left.\frac{d\varphi}{dx}\right|_i+\frac{(\Delta x)^2}{2!}\left.\frac{d^2\varphi}{dx^2}\right|_i +\frac{(\Delta x)^3}{3!}\left.\frac{d^3\varphi}{dx^3}\right|_i+\frac{(\Delta x)^4}{4!}\left.\frac{d^4\varphi}{dx^4}\right|_i+\cdots \tag{7}$$

两式相加，并略去高次项，有

$$\left.\frac{d^2\varphi}{dx^2}\right|_i=\frac{\varphi(i+1)+\varphi(i-1)-2\varphi(i)}{(\Delta x)^2} \tag{8}$$

将式 (8) 代入式 (5)，可得

$$\varphi(i-1)-[2-(\Delta x)^2f(x_i)]\varphi(i)+\varphi(i+1)=0 \tag{9}$$

上述方程可以用矩阵形式记为

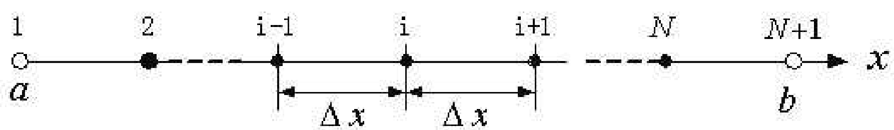

图 1 按顺序排列的等间距网格节点

$$K\Phi = B \tag{10}$$

式中,

$$K = \begin{bmatrix} -[2-(\Delta x)^2 f(x_2)] & 1 & & & \\ 1 & -[2-(\Delta x)^2 f(x_3)] & 1 & & \\ & 1 & -[2-(\Delta x)^2 f(x_4)] & 1 & \\ & & \cdots & \cdots & \cdots \\ & & & 1 & -[2-(\Delta x)^2 f(x_N)] \end{bmatrix} \tag{11}$$

$$\Phi = \begin{bmatrix} \varphi(2) \\ \varphi(3) \\ \vdots \\ \varphi(N-1) \\ \varphi(N) \end{bmatrix}, \quad B = \begin{bmatrix} -\varphi(a) \\ 0 \\ \vdots \\ 0 \\ -\varphi(b) \end{bmatrix} \tag{12}$$

至此, 式 (5) 所示的一维薛定谔方程转化为求解式 (10) 所构成的线性方程组。线性方程组的具体解法可参见各种文献, 例如文献 [4] 等, 本文不再涉及。这些解法的一个共同点是要根据相应算法来编写和调试程序, 因而有一定的工作量。而利用 MATLAB 强大的矩阵计算功能, 先分别定义好矩阵 K 和 B, 再采用如下表达式

$$\Phi = K \backslash B \tag{13}$$

即可直接计算式 (10)。其中的反斜线代表矩阵除法, 它在 MATLAB 中称为矩阵的左除, 比通过计算逆矩阵来求解式 (10) 具有更好的数值稳定性 [5]。

式 (11) 所示的 K 为稀疏矩阵, 只在三条对角线上有非零元素。借助于 MATLAB 提供的用带状向量生成稀疏矩阵的命令 spdiags( ), 容易构造 N×N 阶的三对角线矩阵。例如, 构造一个 4×4 阶的三对角线矩阵

$$SF = \begin{bmatrix} -4 & 1 & & \\ 1 & -4 & 1 & \\ & 1 & -4 & 1 \\ & & 1 & -4 \end{bmatrix} \tag{14}$$

只需要输入下列语句即可:

```
N=4;
p=ones(N,1);
M=spdiags([p,-4*p,p],[-1,0,1],N,N);
% [p,-4*p,p] 为三条对角线上的线元素; [-1,0,1] 为三条对角线上元素的位置
SF=full(M) % 将稀疏矩阵转换为满矩阵,
```

结果为

$$SF = \begin{matrix} -4 & 1 & 0 & 0 \\ 1 & -4 & 1 & 0 \\ 0 & 1 & -4 & 1 \\ 0 & 0 & 1 & -4 \end{matrix} \tag{15}$$

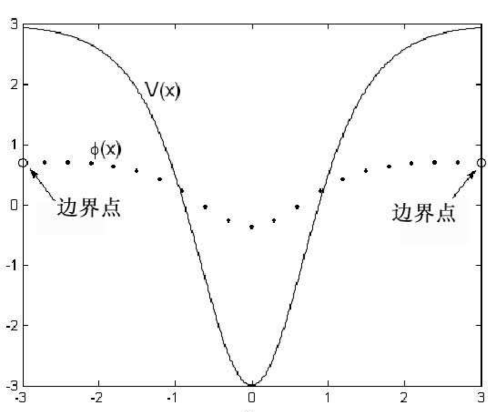

图 2 求解 19 元一次方程组时画出的 $\varphi(x)$ 曲线

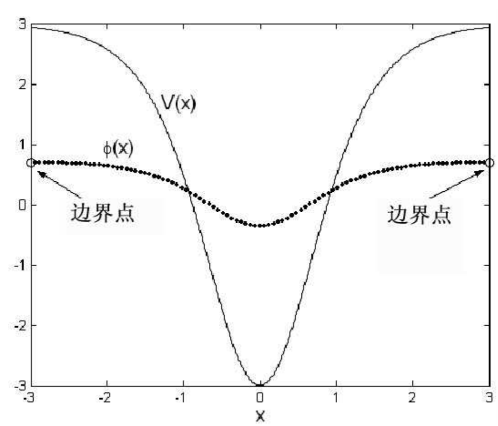

图 3 求解 99 元一次方程组时画出的 $\varphi(x)$ 曲线

作为一个算例，在式 (1) 中考虑下面的势阱情况 [6]

$$V(x)=\frac{\hbar^2}{2m}\alpha^2\lambda(\lambda-1)\left[\frac{1}{2}-\frac{1}{\cos h^2(\alpha x)}\right] \tag{16}$$

$$E_n=\frac{\hbar^2}{2m}\alpha^2\left[\frac{\lambda(\lambda-1)}{2}-(\lambda-1-n)^2\right] \tag{17}$$

```
close all; % 关闭所有窗口
clear all; % 清除所有内存变量
N=100; % 定义差分区间数
a=–3;b=3; % 定义边界坐标


deltax=(b–a)/N; % 计算区间步长
phia=0.7; phib=0.7; % 边界条件
alpha=1;h=1;lambda=4;m=1;n=3; % 势阱参数
En=h^2/2/m*alpha^2*(lambda*(lambda-1)/2-(lambda–1-n)^2);
Cv=h^2/2/m*alpha^2*lambda*(lambda-1);

p=ones(N–1,1); % 创建元素为 1 的 (N–1)×1 阶矩阵
K=spdiags([p,–2*p,p],[–1,0,1],N–1,N–1); % 生成三对角线稀疏矩阵
for i=1:N–1
    y1(i)=a+deltax*i; % 计算各节点的坐标值
    K(i,i)=–(2–(deltax)^2*(En–Cv*(1/2–1/(cosh(alpha*(a+deltax*i))^2)))); % 主对角线元素赋值
end
B=zeros(N–1,1); % 创建元素为 0 的 (N–1)×1 阶矩阵
B(1)=B(1)–phia; B(N–1)=B(N–1)–phib; % 设定矩阵两端的元素值
phi=K\B; % 矩阵左除，计算各节点的函数近似值.
x2=a:0.01:b;
V2=h^2/2/m*alpha^2*lambda*(lambda–1)*(1/2–1/(cosh(x2))^2);
plot(x2,V2) % 绘制函数 V(x) 的曲线
hold on;
plot (y1,phi,'.') % 绘制函数 phi(x) 的曲线
plot(a,phia,'ro',b,phib,'ro') % 标示出边界值所在位置
```

取 $\alpha=1$ 、 $\lambda=4$ 、 $\hbar=1$ 、 $m=1$ 、 $n=0$ 。执行下述程序，可以分别画出如图 2 和图 3 所示的曲线。现有计算机的运算速度很快，如在本例中取 200 个、 500 个甚至更多的节点，计算均几乎是瞬间完成。只是各节点的分布已经密不可分，并呈现为一条连续的曲线。

4 有限差分法的计算结果与波函数系数之间的关系

在两个相邻的差分节点 (例如节点 $(j-1)$ 和节点 j) 构成的区域 $[x_{j-1},x_j]$ ，将势能函数 $V(x)$ 的平均值记为

$$\tilde{V}_j=\frac{V(x_{j-1})+V(x_j)}{2} \tag{18}$$

当差分区间的划分步长 Δx 的值取得很小时，$\tilde{V}_j$ 将趋于一个常数，这时式 (1) 的解可写为

$$\varphi_j(x)=A_j\exp(ik_jx)+B_j\exp(-ik_jx) \tag{19}$$

式中，第一项代表入射波， A_j 为粒子的透射振幅；第二项代表反射波， B_j 为反射振幅； k_j 为波数，并有

$$k_j=\frac{\sqrt{2m(E-\tilde{V}_j)}}{\hbar} \tag{20}$$

由于已计算出 $\varphi_j(x_{j-1})$ 、 $\varphi_j(x_j)$ ，故有

$$\begin{cases}\varphi_j(x_{j-1}) = A_j\exp(ik_jx_{j-1}) + B_j\exp(-ik_jx_{j-1}) \\ \varphi_j(x_j) = A_j\exp(ik_jx_j) + B_j\exp(-ik_jx_j)\end{cases} \tag{21}$$

利用 MATLAB 提供的 solve( ) 命令可算出

$$\begin{cases}A_j = \dfrac{\varphi_j(x_{j-1})\exp(ik_jx_{j-1}) - \varphi_j(x_j)\exp(ik_jx_j)}{\exp(i2k_jx_{j-1}) - \exp(i2k_jx_j)} \\ B_j = -\dfrac{\varphi_j(x_{j-1})\exp[ik_j(x_{j-1}+2x_j)]}{\exp(i2k_jx_{j-1})} \rightarrow \\ \quad \leftarrow \dfrac{-\varphi_j(x_j)\exp[ik_j(x_j+2x_{j-1})]}{-\exp(i2k_jx_j)}\end{cases} \tag{22}$$

以此为基础可进一步展开后续计算。

5 结束语

差分法是常用的数值解法之一。通过选取网格化的节点，无论是常微分方程还是偏微分方程，初值问题还是边值问题，线性方程还是非线性方程，都可以用差分法将其转化为代数方程组，继而求出其近似数值解。作为实际应用，节点必须达到一定的数量，才能获得较为理想的精度。但是当节点较多时，手工备好各节点方程的工作量太大。事实上，若在上述程序中插入相应的计时函数命令，如 clock 或 cputime 等，可以发现程序的运行时间主要耗费在矩阵准备上，而用在矩阵除法上的时间很短。例如当 N=256×256 时，完成矩阵除法所耗时间显示为 0.125s(该值可能会因为软硬件环境的不同而变化) 。换言之，由于求解线性方程组的算法已经趋于成熟和稳定，在某种程度上可以认为，用有限差分法求解一维薛定谔方程的重点不在计算，而在于写出由众多节点构成的线性方程组。这一过程的工作量可以利用本文介绍的方法大为简化。

参考文献

[1] 王忆锋，毛京湘．用 MATLAB 实现普朗克函数积分的快捷计算 [J]. **红外**, 2008, **29**(4): 12–14.

[2] 王忆锋，毛京湘．用 MATLAB 和数值逼近方法实现费米函数的简捷计算 [J]. **红外**, 2008, **29**(8): 34–36.

[3] 王忆锋，毛京湘．用 MATLAB 和打靶法实现平面 PN 结一维泊松方程的简捷计算 [J]. **红外**, 2010, **31**(2):44–46.

[4] 任玉杰． **数值分析及其 MATLAB 实现** [M]. 北京：高等教育出版社， 2007.

[5] E B Magrab, S Azarm, B Balachandran 等著．高会生，李新叶，胡智奇，等译． **MATLAB 原理与工程应用** [M]. 北京：电子工业出版社， 2002.

[6] Tao Pang. *An Introduction to Computational Physics* [M]. Cambridge University Press, 1997.

新闻动态 News

土耳其实施 4.5 亿美元战略侦察计划

据 http://defense-update.com 网站报道，根据土耳其政府武器采购局最近批准的一项 1.41 亿美元的计划，土耳其空军已选择以色列一家工业团队为其提供战略型混合机载图像情报 (IMINT) 系统。另外，土耳其还选择了意大利卫星制造商 Telespazio 公司作为其价值 2.5 亿欧元的 Göktürk 电光卫星计划的最佳投标者。该卫星计划于三年内发射，它将装载一台由法国 Alcatel 公司制造的能够传递分辨率为 0.8m 的图像的空间相机。

作为该工程项目的一部分，意大利卫星制造商 Telespazio 公司将同土耳其的一个合作伙伴建立一个合资企业，以开展基于该新卫星的商业应用服务。

机载 IMINT 系统由两个不同的系统组成，它具有在全天候和各种能见度条件下昼夜进行远距离侦察的能力。该系统包括一个由总部位于以色列海法的 Elbit 系统有限公司研制的远距离电光成像侦察吊舱和一个由以色列航空航天工业有限公司 (IAI) 的子公司 Elta 系统公司提供的合成孔径雷达侦察吊舱。 Elta 系统公司还将提供地面处理与控制中心。

□ 高国龙