拡散方程式に対する数値解法

(中村祐介, 2015/6)

概要

拡散方程式を数値的に解くための幾つかの手法を紹介する

1　問題設定

$$\frac{\partial}{\partial t}f(x,t) = D\frac{\partial^2}{\partial x^2}f(x,t)\,, \qquad f(\pm\infty,t)=0 \tag{1}$$

を解くための数値計算手法について考えたい。色々な方法があるが、ここでは少なくとも x 方向には離散化するものとする：$f(x,t)\to f_n(t)$

2　数値計算手法

そもそも優れた数値計算手法というのはなんだろうか？「速く解ける」「やプログラミングが容易」の前に、「誤差が少ない」ことが重要である。しかしそもそも解けてもいない問題の誤差を云々することは出来ない。そこで多くの場合、方程式の対称性より保存量を求め、その保存量が保存されているかを確認することで、誤差を評価する。Schördinger 方程式ならばエネルギーや波動関数のノルムが保存量である。拡散方程式 (1) の場合 $\int \mathrm{d}x\left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)^2$ などが保存量である。一般に、**方程式の持っている保存量**を壊さない計算手法が優れていると言われる。

また安定性という概念も重要である。例えば時間微分も差分へと近似するとしよう。微分を差分に近似する為には、時間の刻み幅は十分小さくなくてはいけない。刻み幅と誤差の関係は密接に関わっており、一般には刻み幅がある値より大きいと、1 ステップ毎に誤差が指数関数的に増大する様になる。この要な場合、その数値計算は「不安定」であるという。逆に逆に刻み幅を十分細かく取れば、安定になる（つまり誤差の増大しなくなる）場合が多い。例えば、ロジスティック方程式は陽差分法では、どんなに小さな時間刻み幅に対しても不安定であることが知られている。

2.1　Runge-Kutta 法

gsl の課題で行ったように、$\frac{\partial^2}{\partial x^2}f(x,t)$ という 2 階微分も差分化するのは素朴なアイデアである。すると例の三重対角対称行列 K を用いて、拡散方程式 (1) は

$$\frac{\mathrm{d}}{\mathrm{d}t}f_n(t) = D\sum_m K_{nm}f_m(t) \tag{2}$$

となる。これは連立常微分方程式なので Runge-Kutta 法が適応できる。ただしこの方法では $\int \mathrm{d}x\left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)^2 \to (\Delta x)^2\sum_{nm}f_nK_{nm}f_m$ が保存されない。また時間の刻み幅に対する安定性も良くない。つまり空間刻み幅に対して、十分細かい時間刻み幅を選ばないと安定性を失ってしまう。空間刻み幅を 2 分の一にした場合、安定性のためには時間刻み幅は 4 分の一にする必要がある。

2.2　Crank-Nicolson 法

Runge-Kutta 法がこの方程式に対してイマイチだったのは、空間と時間の微分をそれぞれバラバラに差分化した点である。一般に偏微分方程式に対して、このようにバラバラに差分化するのは良くない。

以下のように空間と時間の微分を同時に差分化する方法を Crank-Nicolson 法という。

$$\frac{\partial}{\partial t} f(x,t) \to \frac{f_n^{t+1} - f_n^t}{\Delta t} \tag{3}$$

$$\frac{\partial^2}{\partial x^2} f(x,t) \to \frac{1}{2}\left[\frac{f_{n+1}^{t+1} - 2f_n^{t+1} + f_{n-1}^{t+1}}{(\Delta x)^2} + \frac{f_{n+1}^t - 2f_n^t + f_{n-1}^t}{(\Delta x)^2}\right] \tag{4}$$

拡散方程式の右辺をただの中央 2 階差分にするのではなく、二つの時刻における中央 2 階差分の平均とするところがポイントである。拡散方程式をこのように差分化して、それをまとめると

$$\sum_m A^{nm} f_m^{t+1} = \sum_m B^{nm} f_m^t \tag{5}$$

という構造にまとまる。ここで A, B はともに時間に依存しない三重対角対称行列である。もし f_n^t が分かっているのならば、右辺は計算可能である。すると、問題は行列 A に関する連立方程式に帰着する。この連立方程式を解けば f_n^{j+1} が求まる。Crank-Nicolson 法では $(\Delta x)^2 \sum_{nm} f_n K_{nm} f_m$ が保存する。また、任意の Δt に対して安定である。その意味で Runge-Kutta 法を用いる方法より、Crank-Nicolson 法の方が優れている。1 ステップ当たりにかかる計算時間は Crank-Nicolson 法の方が多いが、Δt を大きく取れるので、トータルでは計算速度も Runge-Kutta 法より速い。

このように A が対角行列にならず、連立方程式に帰着するようなアルゴリズムを陰解法という。いっぽう Runge-Kutta 法の様に連立方程式を解かなくても良い方法を陽解法という。一般には陽解法より陰解法の方が安定性が良い。

なお「任意の Δt に対して安定」というのはあくまで、Δt をどんなに大きくしても「誤差がステップ毎に指数関数的に増大する」という悲惨なことにはならないというだけである。もちろん Δt を大きくすればするほど、誤差自体は大きくなる。

2.3　FFT を用いた方法

FFT を使って空間微分を計算すると、拡散方程式は常微分方程式に帰着する。そしてこの常微分方程式は解析的に解けてしまう。この解法ではそもそも近似をしていないので、$\int \mathrm{d}x \left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)^2$ に限らず全ての保存量が保存する。時間の差分化を行っていないのでそもそも「安定性」という概念すら無い。初期条件さえ与えれば、任意の時刻の解が誤差なく求められる。しかも計算速度も Crank-Nicolson 法よりも高し、メモリの使用量も少ない。