

計算物理学および演習

常微分方程式の数値解法 2

畝山多加志

1 数値計算における誤差

前回、Euler 法のような常微分方程式の誤差として局所誤差と大域誤差というものを考えた。これらの誤差はどちらも時間刻み h に依存しており、 $O(h^n)$ (n は指数) のような形となっていた。そのため、 h を十分に小さくすることで誤差を小さくできることになる。一見、高精度な解法を考えずとも h さえ小さくしてやれば十分な精度を得られるように思える。

この考え方自体は原理的には間違っていないのだが、実際に数値計算を行う際には h を小さくしたからといってうまくいくわけではない。これは計算機の扱えるデータが常に有限であることに関係する。計算機上では実数は常に有限桁の精度しか持たない。計算機上で扱う実数の有効数字は約 7 桁あるいは 16 桁である¹。数値の演算の結果生じる後ろのほうの桁のデータは取り扱えないため適当なところで四捨五入などを行って捨てられてしまう。機械イプシロンと呼ばれる量がこの性質を端的に表している。機械イプシロン ϵ_m は次式を満たす最小の ϵ として定義される。

$$1 + \epsilon > 1 \quad (1)$$

これは数学的には任意の正の実数 ϵ に対して成立するが、計算機では有限精度しかデータを持たないため、 ϵ_m 以下の ϵ に対しては $1 + \epsilon = 1$ となってしまう。例えば、実数 1 が 7 桁の精度しかないとすれば、 1.0×10^{-10} のような小さい数を足しても

$$1.000000 + 1.0 \times 10^{-10} = 1.000000 \quad (2)$$

となり、左辺と右辺は計算精度の範囲内で同じ 1.000000 となってしまうわけである。

機械イプシロンを考慮すると、時間刻み h をあまりに小さくすると不具合が生じることがわかる。例えば、Euler 法で h を ϵ_m より十分に小さくしてしまうと

$$x_{i+1} = x_i + hf(x_i, h_i) = x_i \quad (h \ll \epsilon_m) \quad (3)$$

となってしまう、Euler 法を適用してもまったく変化が生じなくなってしまう。これではどれだけ時間発展を繰り返そうとも x は全く変化しないことになる。これは実数の精度の問題によって誤差が生じたためと解釈でき、このような誤差は「丸め誤差」と呼ばれる。 h が小さすぎると丸め誤差が増え、 h が大きいと局所誤差が増える。つまり、時間刻みを際限なく小さくしても意味はなく、どこかに誤差を最小化できる h の最適値が存在することになる。ある数値スキームで達成できる最小誤差は丸め誤差と数値スキームの誤差の競合で決まり、最適値における精度を上げるには数値スキームの局所誤差 (または大域誤差) 自身を小さくおさえるしかない。

¹精度は型によって変わる。Fortran の real と C の float は約 7 桁、Fortran の double precision と C の double は約 16 桁。より精度の高い型もあるが、いずれにせよ実数を使う際には有限の桁しか表現できない。

2 多変数・高階の常微分方程式の数値解法

前回考えた常微分方程式の数値解法は 1 変数 1 階の常微分方程式に対するものであった。しかし、カオスを生成することで知られる Lorenz の方程式は

$$\frac{dx(t)}{dt} = -\sigma x + \sigma y \quad (4)$$

$$\frac{dy(t)}{dt} = -xz + rx - y \quad (5)$$

$$\frac{dz(t)}{dt} = xy - bz \quad (6)$$

(σ, r, b は定数) は x, y, z の 3 つの変数について連立した常微分方程式である。さらに、相互作用する N 個の古典粒子系の運動を記述するには $3N$ 個の変数に対する 2 階の常微分方程式を考える必要がある。このような常微分方程式には前回のスキームは直接適用できないように見えるかもしれない。ここでは多変数および高次の常微分方程式の数値解法を考える。

まず、高階の微分方程式は適当な変数を導入することで 1 階の常微分方程式の組に分解できる。すなわち、 $x(t)$ に対する n 階の常微分方程式があり、

$$\frac{d^n x(t)}{dt^n} = f\left(x(t), \frac{dx(t)}{dt}, \frac{d^2 x(t)}{dt^2}, \dots, \frac{d^{n-1} x(t)}{dt^{n-1}}, t\right) \quad (7)$$

のように表現されているとする。 f は x, t だけでなく、 x の高階の微分にも依存した関数である。新しい変数

$$y_k(t) = \frac{d^{k-1} x(t)}{dt^{k-1}} \quad (k = 1, 2, \dots, n-1) \quad (8)$$

を導入すると、 n 階の微分方程式を次のような 1 階の微分方程式に書き直すことができる。

$$\frac{dy_n(t)}{dt} = f(y_1(t), y_2(t), \dots, y_{n-1}(t), t) \quad (9)$$

さらに、 $y_1(t), y_2(t)$ などに対しては

$$\begin{aligned} \frac{dy_1(t)}{dt} &= y_2(t) \\ \frac{dy_2(t)}{dt} &= y_3(t) \\ &\vdots \\ \frac{dy_{n-1}(t)}{dt} &= y_n(t) \end{aligned} \quad (10)$$

と表現でき、やはり 1 階の微分方程式で書き直すことができる。さらに、初期条件は $y_k(0)$ に対する条件になる。このようにしてやれば、結局 n 階の常微分方程式は n 個の 1 階の連立常微分方程式、つまり多変数の常微分方程式に帰着する。変数が多くても同様にやれば多変数の 1 階の連立常微分方程式に帰着できる。

以上より、多変数の常微分方程式のスキームを考えれば十分であることがわかったので、最も簡単なスキームである Euler 法が多変数の場合にどうなるか考えてみる。変数が N 次元ベクトル $\mathbf{x}(t)$ (その α 成分 $x_\alpha(t)$ を用いた表記も併用する) として、以下の常微分方程式を考える。

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t), t) \quad (11)$$

ここで \mathbf{f} は N 次元ベクトル関数とする。Euler 法を導出した際と同様に、時間と変数を離散化する。

$$t_i = ih \quad (i = 0, 1, 2, \dots) \quad (12)$$

$$\mathbf{x}_i = \mathbf{x}(t_i) \quad (i = 0, 1, 2, \dots) \quad (13)$$

\mathbf{x}_{i+1} に対して Taylor 展開を行うと

$$\begin{aligned}\mathbf{x}_{i+1} &= \mathbf{x}_i + \left. \frac{d\mathbf{x}(t)}{dt} \right|_{t=t_i} h + O(h^2) \\ &= \mathbf{x}_i + h\mathbf{f}(\mathbf{x}_i, t_i) + O(h^2)\end{aligned}\tag{14}$$

$O(h^2)$ の項を無視すれば多変数の場合の Euler 法が得られる。

$$\mathbf{x}_{i+1} = \mathbf{x}_i + h\mathbf{f}(\mathbf{x}_i, t_i)\tag{15}$$

これは 1 変数の場合とほぼ同じ形であり、多変数の場合でも 1 変数の場合とほぼ同様のスキームが利用できることがわかる。

プログラム上で注意すべき点としては、関数 f は一般に \mathbf{x} 全ての値に依存しているということである。つまり、 x_1 の値を i から $i+1$ に進めるのに $f_{1,i}(\mathbf{x}_i, t_i)$ を求めて、すぐに x_1 を計算して x_1 を上書きしてしまうと、 $f_{2,i}(\mathbf{x}_i, t_i)$ が計算できない ($x_{1,i}$ が $x_{1,i+1}$ で上書きされ消えてしまったため) ことになる。従って、まず先に f_i の全成分をまとめて計算し、それを使って各成分を新しくするよう計算する、という手順を踏むようにせねばならない。

次に Heun 法の場合と同様により高次の項まで残して考えてみる。

$$\begin{aligned}\mathbf{x}_{i+1} &= \mathbf{x}_i + \left. \frac{d\mathbf{x}(t)}{dt} \right|_{t=t_i} h + \frac{1}{2} \left. \frac{d^2\mathbf{x}(t)}{dt^2} \right|_{t=t_i} h^2 + O(h^3) \\ &= \mathbf{x}_i + h\mathbf{f}(\mathbf{x}_i, t_i) + \frac{h^2}{2} \left[\left. \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial t} \right|_{t=t_i} + \mathbf{f}(\mathbf{x}_i, t_i) \cdot \left. \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial \mathbf{x}} \right|_{t=t_i} \right] + O(h^2)\end{aligned}\tag{16}$$

括弧内の第 2 項は

$$\mathbf{f}(\mathbf{x}_i, t_i) \cdot \left. \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial \mathbf{x}} \right|_{t=t_i} = \sum_{\beta} f_{\beta}(\mathbf{x}_i, t_i) \left. \frac{\partial f_{\alpha}(\mathbf{x}, t)}{\partial x_{\beta}} \right|_{t=t_i}\tag{17}$$

であり、 $\partial f_{\alpha} / \partial x_{\beta}$ は $N \times N$ 次元の行列となっている。Heun 法の導出の際に微分の評価を必要としないことが望ましいとしたのはこのためである。(より高次まで展開をすすめると行列よりさらに複雑な量が出てきてしまう。) また、陰的なスキームも 1 変数の場合より複雑になり好ましくない。幸い、Heun 法のような陽的で微分を計算しないでいいスキームであれば特に問題なく多変数の常微分方程式に適用することができる。

多変数版の Heun 法は次のような形となる。

$$\mathbf{x}_i^* = \mathbf{x}_i + h\mathbf{f}(\mathbf{x}_i, t_i)\tag{18}$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \frac{h}{2}[\mathbf{f}(\mathbf{x}_i, t_i) + \mathbf{f}(\mathbf{x}_i^*, t_{i+1})]\tag{19}$$

3 高精度な数値解法

3.1 Runge-Kutta 法

Euler 法より Heun 法のほうが Taylor 展開の高次項まで取り込んでいるので精度が高くなっていた。原理的には、より高次の項まで取り込めるようにすれば精度を上げられるはずである。しかしながら、どのような数値スキームにすれば Taylor 展開の高次項を残せるかという問題は単純ではない。

よく知られている高精度なスキームに Runge-Kutta 法 (正確には 4 次の Runge-Kutta 法) がある。ここでは導出はせずにスキームだけを示す。前回と同様の 1 変数常微分方程式

$$\frac{dx(t)}{dt} = f(x(t), t)\tag{20}$$

に対して、離散化 $t_i = ih, x_i = x(t_i)$ ($i = 1, 2, 3, \dots$) を行う。Runge-Kutta 法では x_{i+1} を次のように 4 段階の式を使って求める。

$$x_i^* = x_i + \frac{h}{2} f(x_i, t_i) \quad (21)$$

$$x_i^{**} = x_i + \frac{h}{2} f\left(x_i^*, t_i + \frac{h}{2}\right) \quad (22)$$

$$x_i^{***} = x_i + hf\left(x_i^{**}, t_i + \frac{h}{2}\right) \quad (23)$$

$$x_{i+1} = x_i + \frac{h}{6} \left[f(x_i, t_i) + 2f\left(x_i^*, t_i + \frac{h}{2}\right) + 2f\left(x_i^{**}, t_i + \frac{h}{2}\right) + f(x_i^{***}, t_i + h) \right] \quad (24)$$

まず x_i から x_i^* を求める。さらに求めた x_i^* を使って x_i^{**} を求め、そこから x_i^{***} を求める。最後にこれらの値全てを使って陽的に x_{i+1} を計算する。計算回数は Heun 法より増えるが、Runge-Kutta 法では局所誤差は $O(h^5)$ まで減少している。(局所誤差の導出は Taylor 展開を用いれば行えるが、あまりに煩雑なので省略する。)

直感的には、Heun 法が $t = t_i$ まわりと $t = t_{i+1}$ まわりの展開を平均することで精度を上げていたように、Runge-Kutta 法は $t = (t_i + t_{i+1})/2$ の点も使うことで精度を上げているものと考えられる。もちろん、陰解法にしないですむように $t = (t_i + t_{i+1})/2$ の点の値は x_i^*, x_i^{**} で近似している。Runge-Kutta 法は単純な形であり、Euler 法等と同様に多変数の場合にもそのまま拡張して使えるため、常微分方程式の数値解法として広く使われている。

なお、一般的な Runge-Kutta 法は 4 段階ではなく段階の数を任意に変更できる。あまり使われないが 3 段階の Runge-Kutta 法もある。Heun 法は 2 段階の Runge-Kutta 法と解釈することもできる。

3.2 適応刻み幅

常微分の形によってはこれまで紹介した陽解法では刻み幅を非常に小さくしないと安定して解けないことがある。そのような場合には陰解法を使って解くという方法がある。陰解法は陽解法と比べて大きな刻み幅に対しても安定であることが多いためである。しかしながら、陰解法はプログラムの実装が複雑になりがちであり、特に変数が多かったり関数が複雑な場合には陽解法のままで安定性を上げることが望ましい。

刻み幅が大きいと不安定になるのであれば、刻み幅自体を自動的に調節してやり常に安定な刻み幅にすれば安定に計算できるはずである。このような考え方に基づいたスキームは適応刻み幅と呼ばれる。

適応刻み幅スキームでは時間 t は等間隔の点に離散化されない。従って $t_{i+1} - t_i$ も一定ではない。そこで、各時点での刻み幅を

$$h_i = t_{i+1} - t_i \quad (i = 1, 2, 3, \dots) \quad (25)$$

と表現する。この刻み幅を用いれば Euler 法は以下のように書き換えられる。

$$x_{i+1} = x_i + h_i f(x_i, t_i) \quad (26)$$

刻み幅一定の場合と比べてこの式自体はほとんど同じであり、 h_i の値さえ決まれば簡単にプログラムを組むことができる。問題は h_i をどのように選べばよいか、という点である。 h_i は十分に小さければいいわけであるが、「十分に」という漠然とした表現では具体的な値を決められない。

適応刻み幅スキームでは、局所誤差として許容できる限界値 ϵ を決め、局所誤差が常に限界未満になるように刻み幅を調節する。つまり、Euler 法等とは異なり、刻み幅で局所誤差が決まるのではなく局所誤差が ϵ となるように刻み幅を決めるわけである。Euler 法では局所誤差は $O(h^2)$ であったから、確かに h を小さくすれば与えられた限界値以下にすることができる。ただし、これだけでは $O(h^2)$ は h^2 以上の高次項という以外の情報がないため、まだ具体的な h の値を決められない。

そこで、局所誤差のうち主要な項は h^2 に比例し、かつ比例係数は定数であると仮定する。すなわち、適当な定数 C を用いて

$$x_{i+1} = x_i + h_i f(x_i, t_i) + Ch_i^2 \quad (27)$$

であると仮定する。さて、ある刻み幅 h を用いて x_i から $x(t_i + h)$ を計算したとする。この値を x_{i+1}^* とすると、

$$x_{i+1}^* = x_i + hf(x_i, t_i) \quad (28)$$

である。真の値を x_{i+1} とすれば局所誤差は

$$x_{i+1} - x_{i+1}^* = 2Ch^2 \quad (29)$$

となる。一方、2 倍の刻み幅を半分にして 2 ステップ計算すれば誤差は小さくなるはずである。このように計算した値を $x_{i+1/2}^{**}, x_{i+1}^*$ とすると

$$x_{i+1/2}^{**} = x_i + \frac{h}{2}f(x_i, t_i), \quad x_{i+1}^* = x_{i+1/2}^{**} + \frac{h}{2}f(x_{i+1/2}^{**}, t_i + h/2) \quad (30)$$

であり、誤差は 1 ステップあたり $C(h/2)^2$ の局所誤差が 2 ステップ分積算されるので

$$x_{i+1} - x_{i+1}^{**} = 2C(h/2)^2 = Ch^2/2 \quad (31)$$

となっているはずである。従って、刻み幅が異なると同じ時間まで進めても誤差も異なるわけであるが、これらを組み合わせれば

$$x_{i+1}^* - x_{i+1}^{**} = Ch^2/2 \quad (32)$$

と書くことができる。ここから、定数 C は

$$C = \frac{2(x_{i+1}^* - x_{i+1}^{**})}{h^2} \quad (33)$$

とできる。

ここで、刻み幅 h の代わりに刻み幅 h_i を使うとすると、誤差は Ch_i^2 となるわけであるから、

$$Ch_i^2 = \frac{2(x_{i+1}^* - x_{i+1}^{**})}{h^2}h_i^2 \quad (34)$$

と書け、この誤差 (の絶対値) が ϵ より小さくなっているためには

$$\left| \frac{2(x_{i+1}^* - x_{i+1}^{**})}{h^2}h_i^2 \right| < \epsilon \quad (35)$$

すなわち

$$h_i < \sqrt{\frac{\epsilon h^2}{2|x_{i+1}^* - x_{i+1}^{**}|}} \quad (36)$$

であればよい。これでようやく具体的な刻み幅の値を決めることができるようになった。

適応刻み幅スキームとしては、結局以下のような形となる。刻み h として 1 つ前のステップでの値の h_{i-1} を使うとすれば、刻み幅 h_{i-1} および $h_{i-1}/2$ でそれぞれ

$$x_{i+1/2}^* = x_i + \frac{h_{i-1}}{2}f(x_i, t_i), \quad x_{i+1}^* = x_{i+1/2}^* + \frac{h_{i-1}}{2}f(x_{i+1/2}^*, t_i + h_{i-1}/2) \quad (37)$$

$$x_{i+1}^{**} = x_i + h_{i-1}f(x_i, t_i) \quad (38)$$

を求め、これらから適切な刻み幅を

$$h_i = \sqrt{\frac{\epsilon}{2|x_{i+1}^* - x_{i+1}^{**}|}}h_{i-1} \quad (39)$$

で求めて

$$x_{i+1} = x_i + h_i f(x_i, t_i), \quad t_{i+1} = t_i + h_i \quad (40)$$

とすればよい。実際にはこれだけではうまく動かないこともあり、例えば刻み幅が大きくなりすぎるときには刻み幅の最大値を設定して最大値を越えないようにしたりといった工夫が必要となる。しかしながら大枠としてはこれで刻み幅を自動的に決めて安定に計算することが可能となると言える。また、Euler 法よりも高精度な方法と適応刻み幅を組み合わせることも可能である。(対象のスキームの局所誤差を考え、同じように刻み幅を計算する式を導出すればよい。)