偏微分方程式入門 (1) 1 次元熱伝導方程式

桂田 祐史

1992 年 6 月 10 日

1 はじめに

今回から 3 回（予定）に渡って偏微分方程式の数値シミュレーションを体験してもらいます。

前回までに扱った常微分方程式とは、独立変数が t 唯一つのみの未知関数 $u = u(t)$ に関する、導関数を含む関係式、例えば

$$\frac{du}{dt} = F(t, u)$$

のような形の方程式のことでした[1]。こうした常微分方程式でモデル化される問題は相当に多く、重要なものも少なくないのですが[2]、実は偏微分方程式と呼ばれる、はるかに広範な現象をモデル化出来るものがあります。それは一言で説明すると、独立変数が二個以上の未知関数に関する、導関数（多変数なのですから当然偏導関数です）を含んだ方程式のことです。この情報処理 II では、簡単で現象のイメージのつかみやすい熱伝導方程式と波動方程式を取り上げることにします。

2 熱伝導方程式

t, x という 2 つの独立変数についての関数 $u = u(x, t)$ についての方程式

(1) $$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \qquad (0 \le x \le 1,\ t \ge 0)$$

は一次元熱伝導方程式と呼ばれます。これは（物理的には）長さが 1 の一様な針金における熱の伝わり具合を表わす方程式と解釈することが出来るからです[3]。この場合、t は時刻、x は針金上の位置を表わす座標、$u(x, t)$ は位置 x、時刻 t における針金の温度を表わしています。

この方程式は、時刻 $t = 0$ での針金の各部分の温度を指定することに相当する初期条件

(2) $$u(x, 0) = f(x) \qquad (0 \le x \le 1)$$

---

[1] 前回までは未知関数を x で表わしましたが、今回は後の偏微分方程式と対比しやすいように u を用います。

[2] 佐藤總夫「自然の数理と社会の数理 I,II」日本評論社、という本に色々楽しい例が載っています (もっと前に紹介すべきでした)。

[3] x が 2 次元や 3 次元のベクトル、すなわち u が 3 あるいは 4 つの独立変数 t, x_1, x_2 (,x_3) の関数である場合にも似たような方程式が考えられます。これは熱伝導現象以外にも、拡散現象を表わしているという解釈も可能です。伝染病の流行のシミュレーションに使われたりもしますし、証券会社の人が勉強するような経済学の本に出て来たりします（私はびっくりしました）。このように、数学の上では一つの方程式が、複数の現象に対応しているということがあります。一つのイメージを固定しないほうがいいのかもしれません。

以外に、各時刻での針金の両端の状態を指定する「境界条件」を課すことにより、解 u が決定される問題となります。境界条件としては、両端の温度が常に 0 であるという

$$(3D) \qquad u(0,t) = u(1,t) = 0 \qquad (t \geq 0),$$

あるいは、両端で熱の出入りがない（断熱）という

$$(3N) \qquad \frac{\partial u}{\partial x}(0,t) = \frac{\partial u}{\partial x}(1,t) \qquad (t \geq 0)$$

を考えることにしましょう。((3D) のように u の値を指定するものを Dirichlet(ディリクレ) 境界条件、 (3N) のように $\partial u/\partial x$ の値を指定するものを Neumann(ノイマン) 境界条件と呼びます。)

これらの問題は、結果がどうなるかは直感的にもある程度理解できます。熱は温度の高いところから低いところに流れるので、時間の経過とともに針金の温度分布は場所によらない定数に近付いて行くでしょう。そして (3D) の場合は冷え切って温度 0 に近付き、 (3N) の場合は（熱量が保存されるので）初期条件 f で決定される温度に近付くでしょう。

3　差分法によるシミュレーション

常微分方程式の初期値問題とは異なり、偏微分方程式を解く手段にはかなりのバリエーションがありますが、ここでは差分法と呼ばれる方法を用いてみましょう。これは常微分方程式を解くための Euler 法や Runge-Kutta 法などと同様に、 (1) 独立変数の範囲を分割して、とびとびの点の上での値を求めることを目標とする、 (2) 方程式に現われる微分を差分商で置き換える、という二つのアイディアを用いたものです。常微分方程式の時と異なるのは、独立変数の個数が 2 以上のため、変数の領域の分割の仕方が少し難しくなることです。問題 (1)-(3) の場合は、考えている変数 (x,t) は適当な長方形領域に属すると考えればいいので、長方形格子に区切るのが自然で実際それで OK なのですが、後で述べるような、明らかでない注意が必要になります（例えば正方形に区切るだけではうまく行きません）。

「空間変数」 x については、区間 $[0,1]$ を N 等分するのが、自然でしょう。

$$h = 1/N, \quad x_i = ih \quad (i = 0,1,2,\cdots,N).$$

時間変数 t については、刻み幅（間隔）を τ としましょう。 t_j を

$$t_j = j\tau \quad (j = 0,1,\cdots)$$

で定めます。

t に関する一階偏微分 $\dfrac{\partial u}{\partial t}(x,t)$ については「前進差分商」 $\dfrac{u(x,t+\tau)-u(x,t)}{\tau}$ で近似し、 x に関する二階偏微分 $\dfrac{\partial^2 u}{\partial x^2}(x,t)$ については、「二階中心差分商」 $\dfrac{u(x+h,t)-2u(x,t)+u(x-h,t)}{h^2}$ で近似することにより、 (1) の近似方程式として

$$\frac{u(x,t+\tau)-u(x,t)}{\tau} = \frac{u(x+h,t)-2u(x,t)+u(x-h,t)}{h^2}$$

が得られます。そこで格子点上 (x_i,t_j) での u の近似値 $U_{i,j}$ を考えると次の方程式の採用が有力となります。

$$(4) \qquad \frac{U_{i,j+1}-U_{i,j}}{\tau} = \frac{U_{i+1,j}-2U_{i,j}+U_{i-1,j}}{h^2} \quad (i = 1,2,\cdots,N-1;\ j = 0,1,2,\cdots),$$

一方 (2) からごく自然に

$$U_{i,0} = f(x_i) \quad (i = 0, 1, 2, \cdots, N) \tag{5}$$

が得られます。(3D) からは

$$U_{0,j} = U_{N,j} = 0 \quad (j = 0, 1, 2, \cdots), \tag{6D}$$

(3N) からは（ちょっと分かりにくいかもしれませんが）

$$U_{0,j} = U_{1,j}, \quad U_{N,j} = U_{N-1,j} \quad (j = 0, 1, 2, \cdots) \tag{6N}$$

が考えられます。

数列 $\{U_{i,j}\}$ に関する方程式 (4),(5),(6) は二つの添字 i, j を含んでいて一見複雑ですが、時刻に関する方の添字 j の小さい方から順に計算していくことが可能です。それを見るには、 $\lambda = \tau/h^2$ と置いて、 (4) を

$$U_{i,j+1} = (1 - 2\lambda)U_{i,j} + \lambda(U_{i+1,j} + U_{i-1,j}) \tag{7}$$

と変形すると分かりやすいでしょう。（例えば、 $j = 0$ の場合、 $U_{i,j} = U_{i,0}$ $(0 \le i \le N)$ は (5) から分かるので、 $U_{i,j+1} = U_{i,1}$ が計算できます。これをすべての i について実行してから、次の j に移行すれば OK です。）

さて、ここで一つだけ注意すべき問題があります。常微分方程式の時は、刻み幅が一種類しかなかったのですが、ここでは h, τ と性質の違うものが 2 つあります。刻み幅を小さくする（0 に近付ける）時、どうやればうまく行くのでしょうか？結論を天下りに述べると、 λ を $0 < \lambda \le 1/2$ と選んで固定したまま、 h,τ を小さくする必要があります。 $\tau = \lambda h^2$ ということですから、 τ は h よりもずっと速く小さくしないといけないことになります。このことは少し難しいので、これ以上詳しい解説はしません。

今回例題プログラムとして用意した `reidai7-1.f` は、 (4),(5),(6D) をプログラミングしたものです（つまり (1),(2),(3D) のシミュレーションをします）。初期値は

$$f(x) = \begin{cases} x & 0 \le x \le 1/2 \\ 1 - x & 1/2 \le x \le 1 \end{cases}$$

という、中央に山のある関数です。 `f77x` でコンパイル、実行すると空間方向の分割数 N と、上で解説したパラメーター λ の入力を促してきますので、試してみたい値を入力して下さい。以下の例では、 $N = 20$, $\lambda = 2/5$ としています（λ は上で注意した条件を満足するように選びました）。

```
waltz11% f77x reidai7-1.f
waltz11% reidai7-1
 空間の分割数 (<=   100), λ = 時間刻み幅 / 空間刻み幅の 2 乗を入力してください。
20 0.4                                    ← N=40, λ =2/5 とした
 時間刻み幅 =     1.00000E-03 になりました。
```

するとウィンドウが現われて、針金の温度の時間変化を見ることが出来ます。

時間に関する添字 j に関しては、とりあえず 1 万ステップ動かしますが、それでも足りないという場合に備えて

計算を続けますか？続けたいなら 1 を入力して下さい。

と尋ねて来るようになっています。止めたいなら、例えば 0 を入力して下さい。

問題 7-1　次のことを実験しなさい。(1) 初期値を変えてみる。(2) λ を変えてみる。特に条件 $0 < \lambda \le 1/2$ を満足しない場合を試す。

問題 7-2　プログラムを書き換えて、Neumann 問題 (4),(5),(6N) のシミュレーションをしなさい。