偏微分方程式入門 (2) 1 次元波動方程式

桂田 祐史

1992 年 6 月 17 日

1　波動方程式

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

は一次元波動方程式 (wave equation) と呼ばれます[1]（ここで c は与えられた正の定数です）。それは、この方程式が，一様な弦の振動や，細い管の中の空気の振動などの、一次元的な振動・波動現象を表わすものであると解釈出来るからです。弦の振動の場合は $u(x,t)$ は時刻 t における、弦上の点 x の釣り合いの位置からの変位を表わすものと考えます。

実は定数 c は波の伝播の速さになりますが、以下では時刻の単位を適当に取り替える（数学的には ct を新たに t とする変数変換を行う）ことによって $c = 1$ であるとして扱います[2]。

この方程式は、時刻 $t = 0$ での各部分の変位と「速度」を指定することに相当する初期条件

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

(3) $$\frac{\partial u}{\partial t}(x,0) = g(x) \qquad (a \le x \le b)$$

と、各時刻での弦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、両端が固定されていて（管の中の空気の振動の場合では「端が閉じられていて」）変位が常に 0 であるという

$(4D)$ $$u(a,t) = u(b,t) = 0 \qquad (t \ge 0),$$

あるいは、両端で自由に動ける（管の中の空気の振動の場合では「端が開放されている」）という

$(4N)$ $$\frac{\partial u}{\partial x}(a,t) = \frac{\partial u}{\partial x}(b,t) \qquad (t \ge 0)$$

を考えることにしましょう。熱伝導方程式の場合と同様に (4D) を Dirichlet 境界条件、(4N) を Neumann 境界条件と呼びます。

---

[1]熱伝導方程式の場合と同様に、波動方程式にも空間 2 次元や 3 次元のものが考えられます。

[2]このような x, t の単位の変換（数学的には変数の一次変換）は「スケーリング」と呼ばれ、かなり重要な話題なのですが、ここでは注意するだけに留めておきます。

2　差分法によるシミュレーション

今回も前回と同様に差分法を用いてシミュレートしてみます。

「空間変数」 x については、区間 $[a,b]$ を N 等分します：

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

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。

方程式 (1) に現れる二つの微分、 t に関する二階偏微分 $\dfrac{\partial u}{\partial t}(x,t)$ と x に関する二階偏微分 $\dfrac{\partial^2 u}{\partial x^2}(x,t)$ の双方を、ともに「二階中心差分商」で近似すると次の近似方程式が得られます：

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

そこで格子点上 (x_i,t_j) での u の近似値 $U_{i,j}$ を決定する方程式としては次のものが考えられます。

$$\frac{U_{i,j+1}-2U_{i,j}+U_{i,j-1}}{\tau^2}=\frac{U_{i+1,j}-2U_{i,j}+U_{i-1,j}}{h^2}\quad (i=1,2,\cdots,N-1;\ j=1,2,3,\cdots),$$

これを変形すると

(5)　$U_{i,j+1}=2(1-\lambda^2)U_{i,j}+\lambda^2(U_{i-1,j}+U_{i+1,j})-U_{i,j-1}\quad (i=1,2,\cdots,N-1;\ j=1,2,3,\cdots)$

となります。ただし $\lambda=\tau/h$ と置きました。

一方 (2) からは、ごく自然に

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

が得られます。 (3) からは、例えば

(7)　$$U_{i,1}=(1-\lambda^2)f(x_i)+\frac{\lambda^2}{2}(f(x_{i-1})+f(x_{i+1}))+\tau g(x_i)\quad (1\le i\le N)$$

が得られます[3]。 (4D) からは

(8D)　$$U_{0,j}=U_{N,j}=0\quad (j=0,1,2,\cdots),$$

また (4N) からは

(8N)　$$U_{0,j}=U_{1,j},\quad U_{N,j}=U_{N-1,j}\quad (j=0,1,2,\cdots)$$

が考えられます。

---

[3]（ちょっと難しいのですが） u の点 $(x,0)$ における t についての展開 $u(x,\tau)=u(x,0)+\tau\frac{\partial u}{\partial t}(x,0)+\frac{\tau^2}{2}\frac{\partial^2 u}{\partial t^2}(x,0)+\cdots$ において、関係式 $\frac{\partial^2 u}{\partial t^2}(x,0)=\frac{\partial^2 u}{\partial x^2}(x,0)$ が成立するとして、この $\frac{\partial^2 u}{\partial x^2}(x,0)$ を二階中心差分商で近似したものです。

数列 $\{U_{i,j}\}$ に関する方程式 (5),(6),(7),(8) は二つの添字 i, j を含んでいますが、 (5) を漸化式として、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出来ます。熱伝導方程式の場合は、二番目の添字のところには、 $j, j+1$ しか現れませんでしたが、 (5) では $j-1$, $j, j+1$ と 3 つのものが現れています。 $j+1$ での値を求めるために、一段前の j での値のみならず、もう一段前の $j-1$ での値が必要になっているわけです。このことは、もとの方程式が時刻 t に関して二階であることに対応しています。そのため計算を出発させるためには、 $j=0$ での値だけでなく、 $j=1$ での値も必要になりますが、それは時刻に関する一階の微分を指定している初期条件 (3) に由来する (7) で与えられています。

熱方程式の場合と同様に、 h, τ と性質の異なる刻み幅が 2 つありますが、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？今回も結論を天下りに述べるだけで我慢することにしますが、「λ を $0 < \lambda \leq 1$ と選んで固定したまま、 h,τ を小さくする」で OK です。

3　例題プログラムによる実験

前回の熱方程式は（時間が経つと温度が場所に依らず一様になるという）現象としては少々地味なものでしたが、今回の波動方程式は工夫次第で色々な現象が観察できてなかなか面白いものであると思います。

今回例題プログラムとして用意した reidai8-x ($x = 1, 2, 3, 4$) は、 (5),(6),(7),(8D-8N) をプログラミングしたものです（reidai8-4 は Fortran のプログラム reidai8-4.f ではなく、実行形式のまま配ります。）。プログラムの先頭付近に

```
logical Neu
parameter (Neu = .false.)
```

のように論理型の定数 Neu の値を設定しているところがありますが、これが .false. の場合、Dirichlet 境界条件 (8D) を指定したことになり、反対に Neu が .true. の場合は Neumann 境界条件 (8N) を指定したことになります（初めてだという人も多いでしょうが、容易に想像できるように .true. は「真」、.false. は「偽」を意味する Fortran の用語です。）。最初の状態では Neu は .false. ですから Dirichlet 条件の場合のシミュレーションをすることになります。

プログラム reidai8-1.f では、初期値は

$$f(x) = \sin x + \sin 3x + \sin 5x, \quad g(x) = 0 \quad (0 \leq x \leq 2\pi)$$

というものです。実行するには、いつものように f77x reidai8-1.f としてコンパイルしてから、

```
waltz11% reidai8-1
  空間の分割数 (<=  1000), λ = 時間刻み幅 / 空間刻み幅を入力してください。
100 1                          ← 100 分割、λ =1 を指定
```

のようにします。するとウィンドウが現われて弦の振動のシミュレーションが始まりますが、おそらくは見覚えのある[4]アニメーションが見られることでしょう。

次の reidai8-2.f は、 x の範囲（$-\pi \leq x \leq \pi$）と初期値以外全く reidai8-1.f と同じものです。初期値は

$$f(x) = \begin{cases} 1 + \cos 4x & |x| \leq \pi/4 \\ 0 & \pi/2 \leq x \leq \pi \end{cases}$$

[4]第 3 回目のアニメーションの例と同じものです。計算方法はまったく異なっていますが。

$$g(x) = 0$$

という一つの山があるものです。このプログラムの実行の仕方は上のものと同じです。こちらは是非とも Neumann 境界条件の場合も試してみて、波が境界で反射される仕方を比べてみてください。

上の二つの例では、いずれも $g = 0$ としてあります。これは時刻 $t = 0$ で弦をそっと放した（＝速度が 0）場合で、波の山が左右に二つに割れて、それぞれ同じ速さで反対向きに進んでいくことになりますが、 g を適当に設定すると、「山」の動きをコントロール出来ます。詳しい解説は省きますが、 `reidai8-3.f`, `reidai8-4` を試してみてください。

注意 8-1　前回の熱方程式では、山の高さは時間の経過につれて低くなる一方（逆に谷は浅くなる一方）でしたので、最初に山、谷を機械的に調べればグラフを描くべき範囲が分かったのですが、今回はそう簡単には行きません。山と山が衝突すると高くなりますし、一方で谷と谷が衝突すると低くなるからです。

注意 8-2 今回の現象は周期現象であることが分かっていて、その周期も簡単に計算できるので、どれくらいの間シミュレーションを続ければ十分か分かることになります（例題プログラムではこのことを全く考慮していませんが...）。

注意 8-3　`reidai8-2` などで、波と波がぶつかると、「はね返る」ようにも感じられますが、それは間違いで、むしろ「通り抜ける」と見る方が正しいことに注意して下さい。これは異なる高さの波が衝突する `reidai8-4` の結果を見れば納得できるでしょう。

問題 8-1　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 $0 < \lambda \le 1$ を満足しない場合を試す。 (3) Neumann 条件と Dirichlet 条件で波の反射の仕方が異なることを確認する。

問題 8-2　波の速さが 1 であることを確かめなさい。余裕があれば c を含んだ方程式のシミュレーションを行って、波の速さが c になることを確かめなさい。

問題 8-3　（唯一 Fortran プログラムを公開していない） reidai8-4 の Fortran プログラムはどんなものか推測しなさい（＝同じ動作をするプログラムを自分で作りなさい）。（ヒント: `reidai8-3.f` を調べてみましょう。）