

数値シミュレーション技法基礎セミナー  
第3回 熱伝導と行列解法

担当：中久喜伴益 (理学研究科・地球惑星システム学専攻)

2007年9月7日

## 目次

1. 偏微分方程式の数値解法とシミュレーション
  - 1.1 偏微分方程式の型
  - 1.2 物質の輸送を表す微分方程式
  - 1.3 偏微分方程式の数値解法と離散化の方法
  - 1.4 定常解を解く問題：境界値問題
2. 離散化方程式の求め方～熱伝導 (拡散) 方程式を例に
  - 2.1 拡散方程式
  - 2.2 有限差分法
  - 2.3 有限体積法
3. 熱伝導 (拡散) 方程式の差分方程式
  - 3.1 一次元熱伝導方程式
  - 3.2 二次元熱伝導方程式
  - 3.3 時間積分の方法
4. プレートの冷却のシミュレーション
  - 4.1 プレートの温度構造
  - 4.2 変数変換と数値積分による解法
  - 4.3 差分法による解法
  - 4.4 差分法における境界条件の与え方
  - 4.5 プレート冷却シミュレーションのアルゴリズム
5. 波動の数値解析
  - 5.1 高階の時間微分の扱い
  - 5.2 波動の数値シミュレーションのアルゴリズム
6. 連立一次方程式の解法
  - 6.1 直接法と反復法
  - 6.2 ガウスの消去法
  - 6.3 対称行列の場合
  - 6.4 特殊な係数行列を持つ連立一次方程式の解法
  - 6.5 三重対角行列のガウスの消去法
  - 6.6 帯対称行列に対する修正コレスキー法
  - 6.7 反復法による連立一次方程式の解法
  - 6.8 緩和法によるポワソン方程式の解法
7. 参考文献など
  - 7.1 参考文献
  - 7.2 参考資料

# 1. 偏微分方程式の数値解法とシミュレーション

## 1.1 偏微分方程式の型

楕円型：ポワソン方程式など

$$\nabla^2 \phi = \rho$$

境界値問題。離散化して得られる連立一次方程式を解く。

放物型：熱伝導方程式 (拡散方程式)、移流 (対流)・拡散方程式 (非圧縮粘性流れ)

$$\frac{D\phi}{Dt} = \kappa \nabla^2 \phi$$

初期値問題。現在のタイムステップの値を使って、繰り返し直後の値を求める。陽解法では連立一次方程式を解くことなしに直接値を決められる。他に比べて解くのが楽。

双曲型：移流方程式 (超音速流れ)、波動方程式

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0$$

初期値問題。数値的に不安定になりやすく解法に工夫が必要。

## 1.2 物質の輸送を表す微分方程式

生成・消滅

$$\frac{d\phi}{dt} = a\phi$$

拡散

$$\frac{\partial \phi}{\partial t} = \kappa \nabla^2 \phi$$

移流・拡散

$$\frac{D\phi}{Dt} = \kappa \nabla^2 \phi, \quad \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \kappa \nabla^2 \phi$$

## 1.3 偏微分方程式の数値解法と離散化の方法

空間の偏微分を離散化する方法には主に以下3つの方法がある。

- ・有限差分法 (FDM)

物体を点の集まりと考え、テイラー展開に基づいて、微分を差分に置き換え、差分方程式を作る。

- ・有限体積法 (FVM)

物体を小さな箱 (有限体積) に分け、体積での保存則を満たすように積分で表した保存則の式を離散化して差分方程式に置き換える。

- ・有限要素法 (FEM)

物体を小さな要素に分け、要素内部で求める関数を2次式などの次数の低い多項式で近似して表す。多項式の係数を微分方程式に適合するように最小二乗法などで決める。

他にも、粒子法やスペクトル法がある。時間方向の離散化にはどの方法で差分法を用いる。

#### 1.4 定常解を解く問題：境界値問題

定常状態の解を求める場合には時間微分をゼロと置いた式、

$$0 = \kappa \frac{\partial^2 T}{\partial x^2}$$

を離散化する。この場合は、解が境界条件だけで決まる。離散化した式は連立一次方程式となり、すべての点に対する値が同時に求められる。

## 2. 離散化方程式の求め方～熱伝導 (拡散) 方程式を例に

### 2.1 拡散方程式

一次元の熱伝導 (拡散) 方程式を考える。

$$\frac{\partial \phi}{\partial t} = \kappa \frac{\partial^2 \phi}{\partial x^2} + H$$

ここで、空間、時間の刻み幅を一定とし、それぞれ、 $\Delta x, \Delta t$  で表す。

### 2.2 有限差分法

テイラー展開を用いて、それぞれの偏微分係数を差分商で表して、それを組み合わせる。

$$\phi(x + \Delta x, t) = \phi(x, t) + \frac{\partial \phi}{\partial x} \Delta x + \frac{1}{2} \frac{\partial^2 \phi}{\partial x^2} (\Delta x)^2 + \dots \quad (1)$$

$$\phi(x - \Delta x, t) = \phi(x, t) - \frac{\partial \phi}{\partial x} \Delta x + \frac{1}{2} \frac{\partial^2 \phi}{\partial x^2} (\Delta x)^2 - \dots \quad (2)$$

#### 1 階微分

(1) -  $\phi(x, t)$

$$\frac{\partial \phi}{\partial x} = \frac{\phi(x + \Delta x, t) - \phi(x, t)}{\Delta x} + O(\Delta x)$$

(2) -  $\phi(x, t)$

$$\frac{\partial \phi}{\partial x} = \frac{\phi(x, t) - \phi(x - \Delta x, t)}{\Delta x} + O(\Delta x)$$

(1) - (2)

$$\frac{\partial \phi}{\partial x} = \frac{\phi(x + \Delta x, t) - \phi(x - \Delta x, t)}{2\Delta x} + O((\Delta x)^2)$$

#### 2 階微分 (1) + (2)

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{\phi(x + \Delta x, t) + \phi(x - \Delta x, t) - 2\phi(x, t)}{(\Delta x)^2} + O((\Delta x)^2)$$

時間微分を前進差分にとると、差分方程式は

$$\frac{\phi(x, t + \Delta t) - \phi(x, t)}{\Delta t} = \kappa \frac{\phi(x + \Delta x, t) + \phi(x - \Delta x, t) - 2\phi(x, t)}{(\Delta x)^2} + H(x) \quad (3)$$

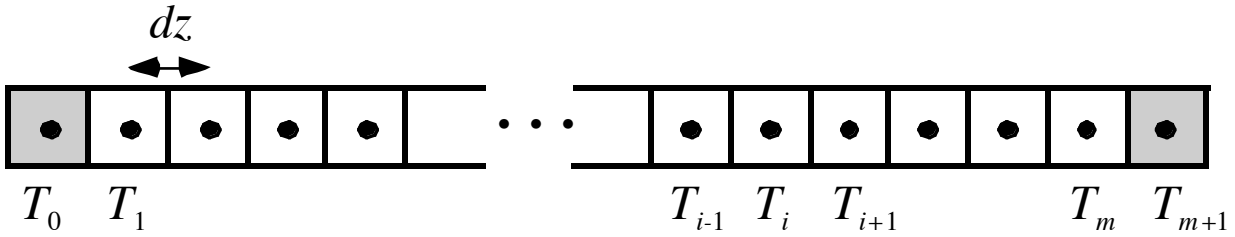
と書ける。この方法では、連立1次方程式を解くことなしに、 $\phi(x, t + \Delta t) =$  の形に式を変形できる。そのため、(3) 式は陽解法と呼ばれる。時間差分を後退差分にとった場合は、陽的に求めることが出来ない。このため、陰解法と呼ばれる。

### 2.3 有限体積法

保存則の式

$$\frac{\partial}{\partial t} \int_V \phi dv + \int_S (-\kappa \nabla \phi) \cdot \mathbf{n} dS = \int_V H dv \quad (4)$$

から出発する。いま、直角座標系で領域を有限体積に分ける。



このとき、内部では  $\phi$  は一定とする。

$$\frac{\phi(x, t + \Delta t) - \phi(x, t)}{\Delta t} \Delta x (\Delta S) = \left\{ \kappa_{x+\Delta x/2} \frac{\phi(x + \Delta x, t) - \phi(x, t)}{\Delta x} (\Delta S) - \kappa_{x-\Delta x/2} \frac{\phi(x, t) - \phi(x - \Delta x, t)}{\Delta x} (\Delta S) \right\} + H \Delta x (\Delta S)$$

ここで、 $(\Delta S)$  は x-方向の有限体積の境界面積である。1次元の時は  $\Delta S=1$ 、2次元の時は  $\Delta S=\Delta y$ 、3次元の時は  $\Delta S=\Delta y \Delta z$  である。k が一定の時には、右辺第2項が

$$\kappa \frac{\phi(x + \Delta x, t) + \phi(x - \Delta x, t) - 2\phi(x, t)}{\Delta x} (\Delta S)$$

となる。両辺を  $(\Delta S)\Delta x$  で割ると差分法から求められた式 (3) と一致する。差分方程式では慣用的にタイムステップを  $n$ 、空間座標の番号を  $i$  で表す。すなわち

$$\phi(x, t + \Delta t) \rightarrow \phi_i^{n+1}$$

$$\phi(x, y + \Delta y / 2, t + \Delta t) \rightarrow \phi_{i, j+1/2}^{n+1}$$

のように書く。(4) 式は保存則を表しているので、有限体積法により求めた離散方程式も保存則を満たしている。また、(4) 式は座標系によらず成り立つので、複雑な形状の物体の問題を計算するのに向いている。

### 3. 熱伝導 (拡散) 方程式の差分方程式

#### 3.1 一次元熱伝導方程式

一次元の熱伝導方程式は、

$$\rho C_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right)$$

熱拡散率を使うと、

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right)$$

熱伝導率が一定の場合には

$$\frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial x^2}$$

となる。

ここで、有限体積法の離散式を元に、差分方程式を作る。差分を取るためにメッシュ上で変数を定義する。下付添え字の  $i$  はメッシュの番号を表す。

拡散係数  $a$  が空間変化する場合、右辺の差分式を  $F_i$  で表すと

$$F_i = \frac{1}{\Delta x} \left[ \kappa_{i+\frac{1}{2}} \left( \frac{T_{i+1} - T_i}{\Delta x} \right) - \kappa_{i-\frac{1}{2}} \left( \frac{T_i - T_{i-1}}{\Delta x} \right) \right]$$

である。  $i+1/2$  はメッシュの境界で定義されることを示す。つまり熱拡散率はメッシュの境界で定義する。熱伝導率が一定の場合には、

$$F_i = \kappa \frac{T_{i+1} + T_{i-1} - 2T_i}{\Delta x^2}$$

となる。

#### 3.2 二次元熱伝導方程式

二次元の拡散方程式

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial z} \left( \kappa \frac{\partial T}{\partial z} \right)$$

を考える。拡散係数が一定の場合は、

$$\frac{\partial T}{\partial t} = \kappa \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial z^2} \right)$$

である。右辺の差分は、メッシュの番号を  $i, j$  で表すと、

$$F_{i,j} = \frac{1}{\Delta x} \left[ \kappa_{i+\frac{1}{2},j} \left( \frac{T_{i+1,j} - T_{i,j}}{\Delta x} \right) - \kappa_{i-\frac{1}{2},j} \left( \frac{T_{i,j} - T_{i-1,j}}{\Delta x} \right) \right] + \frac{1}{\Delta z} \left[ \kappa_{i,j+\frac{1}{2}} \left( \frac{T_{i,j+1} - T_{i,j}}{\Delta z} \right) - \kappa_{i,j-\frac{1}{2}} \left( \frac{T_{i,j} - T_{i,j-1}}{\Delta z} \right) \right]$$

となる。拡散係数が一定の場合は、

$$F_{i,j} = \kappa \frac{T_{i+1,j} + T_{i-1,j} - 2T_{i,j}}{\Delta x^2} + \kappa \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta z^2}$$

である。ただしメッシュは  $x$  と  $y$  のそれぞれの方向に一定の大きさ ( $\Delta x$ ,  $\Delta z$ ) であるとした。  $\Delta x = \Delta z$  の場合は、

$$F_{i,j} = \kappa \frac{T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j}}{\Delta x^2}$$

と表される。ある点の周りの値の合計から、ある点の値の 4 倍を引いたものである。

### 3.3 時間積分の方法

$F_{i,j}$  に  $t$  での値を用いると  $T_{i,j}^{n+1} = T_{i,j}^n + \Delta t F_{i,j}$  を陽的に計算できる。これがオイラー (Euler) 法 (一次陽解法) である。

$$T_{i,j}^{n+1} = T_{i,j}^n + \Delta t F_{i,j}^n$$

つまり、

$$T_{i,j}^{n+1} = T_{i,j}^n + \kappa \frac{T_{i+1,j}^n + T_{i-1,j}^n + T_{i,j+1}^n + T_{i,j-1}^n - 4T_{i,j}^n}{\Delta x^2} \Delta t$$

である。この式が数値的に安定となる条件は次元の場合、

$$\Delta t \leq \frac{\Delta x^2}{2\kappa}$$

となり、二次元の場合には

$$\Delta t \leq \frac{1/2\kappa}{1/\Delta x^2 + 1/\Delta z^2}$$

である。

完全陽解法は  $F_{i,j}$  に  $t + \Delta t$  の値を用いた方法である。すなわち、

$$T_{i,j}^{n+1} = T_{i,j}^n + \Delta t F_{i,j}^{n+1}$$

と書ける。  $F_{i,j}^{n+1}$  には  $n+1$  タイムステップでの値が含まれるため、陽に計算することが出来ない。そのため、1 タイムステップごとに連立一次方程式を解かなくてはならない。この式は無条件安定性を持つ。オイラー法と完全陰解法の打ち切り誤差は一次 ( $O(\Delta t)$ ) である。

$n$  ステップ目と  $n+1$  ステップ目の値の平均を取る方法が

$$T_{i,j}^{n+1} = T_{i,j}^n + \frac{\Delta t}{2} (F_{i,j}^n + F_{i,j}^{n+1})$$

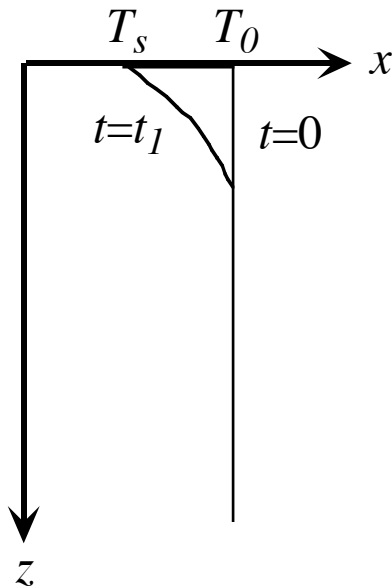
クラנק・ニコルソン (Crank-Nicholson) 法である。この方法は陰解法の一で、打ち切り誤差が二次 ( $O(\Delta t^2)$ ) の方法ができる。すなわち、この方法は無条件安定だが、時間依存解を安定に求めるためには  $\Delta t$  をある程度小さく取ることが必要である。というのは、安定性の条件とは解の振動が徐々に減衰していくという条件を意味しているのであって、解が振動しないということ必ずしも意味しないからである。



## 4. プレートの冷却のシミュレーション

### 4.1 プレートの温度構造

プレートは海嶺で生まれ、移動する。海嶺で生まれたときには熱いマントルがそのまま上昇してきたものと見なせる。海嶺から離れると、上から冷やされ表面から冷却されて行く。水平方向の温度変化を小さいと見なして、1次元のモデルを考え、プレートの温度変化を計算してみよう。



### 4.2 変数変換と数値積分による解法

1次元の熱伝導方程式 (2) から出発する。このとき内部加熱は無視する。

$$\theta = \frac{T - T_0}{T_s - T_0} \quad (10)$$

とおく。変数を

$$\eta = \frac{z}{2\sqrt{kt}} \quad (11)$$

とすると、

$$\eta \frac{d\theta}{d\eta} = \frac{1}{2} \frac{d^2\theta}{d\eta^2} \quad (12)$$

となる。これを解くと、

$$\theta = 1 - \text{erf}(\eta) = \text{erfc}(\eta) \quad (13)$$

が得られる。ただし、 $\text{erf}(x)$  は誤差関数

$$\text{erf}(\eta) = \frac{2}{\sqrt{\pi}} \int_0^\eta \exp[-\eta'^2] d\eta' \quad (14)$$

である。従って温度は

$$T = T_0 + (T_s - T_0) \text{erfc}\left(\frac{z}{2\sqrt{kt}}\right) \quad (15)$$

と表される。ただし、誤差関数は解析的に積分できないので、数値積分を利用する。数値積分の方法は授業のホームページを参照のこと。

地表の熱流量は (14) を  $z$  で微分することより求められる。

$$q(t) = -k \frac{\partial T}{\partial z} = \frac{k(T_s - T_0)}{\sqrt{\pi \kappa t}} \quad (16)$$

### 4.3 差分法による解法

この問題は差分法でも解くことが出来る。熱伝導方程式を差分法で離散化する。

左辺は、

$$\left( \frac{\partial T}{\partial t} \right)_i = \frac{T_i^{n+1} - T_i^n}{\Delta t} \quad (7)$$

である。ここで、上添え字  $n-1$  は新しい時間ステップでの温度を表す。  $T$  は古い時間ステップでの値である。下添え字  $i$  はメッシュ上での位置を表す。右辺は、

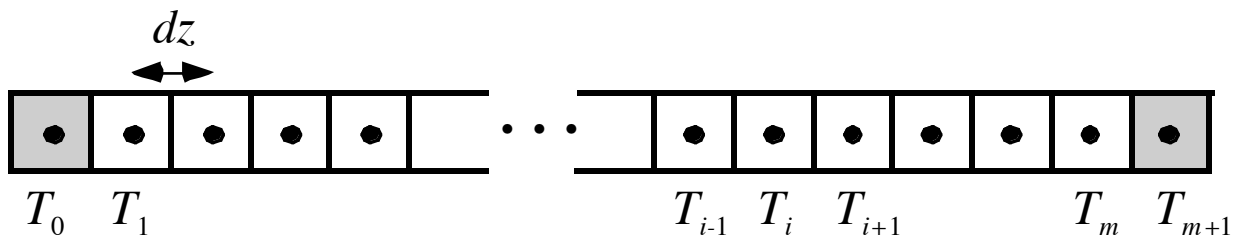
$$\left( \frac{\partial^2 T}{\partial x^2} \right)_i = \kappa \frac{T_{i+1}^n + T_{i-1}^n - 2T_i^n}{\Delta x^2} \quad (8)$$

と離散化される。ここで、  $T$  の時間ステップは古い方を取る（陽解法）。ここで、安定に計算できる条件は、

$$\Delta t \leq \frac{\Delta x^2}{2\kappa} \quad (9)$$

である。

こうすると、新しい時間ステップの温度場は古い時間ステップの値から陽的に計算できる。ただし、初期状態における温度場を与えなければならない。また、すべての時間ステップにおいて、一番端のメッシュには境界条件を与える。



### 4.4 差分法における境界条件の与え方

地表面では温度の値が与えられる。値そのものを与える境界条件をディリクレ型境界条件と言う。

$$T_0^n = T_s$$

下面が温度一定の境界条件の時には、

$$T_{m+1}^n = T_M$$

である。下面の境界条件を熱流量とする場合は次のように考える。例えば、断熱温度境界条件、つまり熱流量が0の時には、

$$q = -k \frac{\partial T}{\partial z} = 0$$

である。この様に、微分係数を与える境界条件をノイマン型境界条件と呼ぶ。差分式では、

$$k \frac{T_m - T_{m-1}}{\Delta z} = 0$$

となるので、結局、

$$T_m = T_{m-1}$$

となる。毎ステップ毎に境界の値を隣と同じにする。熱流量が与えられる場合は、

$$k \frac{T_m - T_{m-1}}{\Delta z} = q$$

であるから、

$$T_m = T_{m-1} + \frac{q \Delta z}{k}$$

となる。これらの境界条件は陰解法の時には行列の解法に組み込んで同時に解く必要がある。

#### 4.5 プレート冷却シミュレーションのアルゴリズム

プログラミングをする際のアルゴリズムは下記のようなになる。

1. 初期条件はタイムステップ0のみ

For i=1 to m

$$T_i^0 = T_M$$

2. 時間進行：オイラー法

For i=1 to m

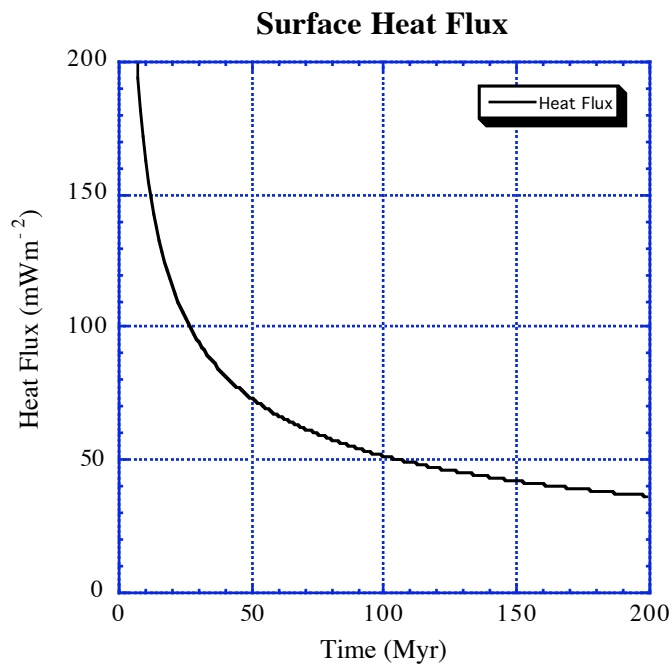
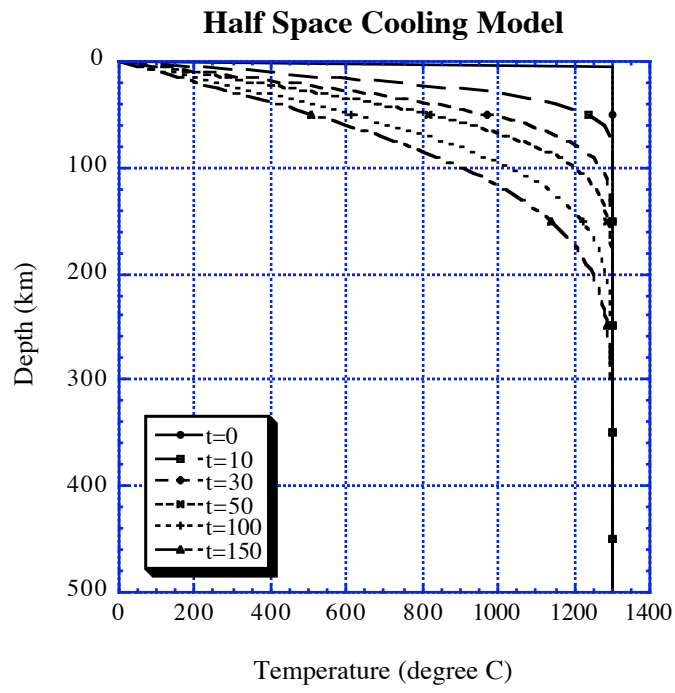
$$T_i^{n+1} = T_i^n + \Delta t \kappa \frac{T_{i+1}^n + T_{i-1}^n - 2T_{i,j}^n}{\Delta x^2}$$

3. 境界条件の設定

$$T_0^n = T_s$$

$$T_m = T_{m-1}$$

4. 新旧変数の入れ替え
5. 必要に応じて温度の値をファイルに書き出す
6. 2へ戻る



## 5. 波動の数値解析

### 5.1 高階の時間微分の扱い

時間微分が2階以上の場合は、1階微分の複数の方程式に分ける。波動方程式

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2} \quad (1)$$

は変位の速度  $u$  を使って、

$$\frac{\partial v}{\partial t} = c^2 \frac{\partial^2 u}{\partial x^2} \quad (2)$$

$$\frac{\partial v}{\partial t} = c^2 \frac{\partial^2 u}{\partial x^2} \quad (3)$$

とする。オイラー法を式(2)(3)に適用すると、

$$\frac{v_i^{n+1} - v_i^n}{\Delta t} = c^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = v_i^n$$

となるが、時間差分が1次誤差を持つ方法では正確に波を計算することが出来ない。正確に計算するためには2次以上の方法が必要である。2次の1階差分は中心差分であるので、2つの式(2)(3)の時間を  $\Delta t/2$  ずらして右辺が2つの時間の真ん中になるように計算すればよい。つまり、

$$\frac{v_i^{n+1/2} - v_i^{n-1/2}}{\Delta t} = c^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \quad (4)$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = v_i^{n+1/2} \quad (5)$$

のように計算すればよい。この方法では時間差分が中心差分となっていて2次の誤差を持つ。交互にずれた時間でとびとびに値を計算するので、この方法は蛙跳び法 (leap-frog method) と呼ばれる。(4)に(5)を代入すると、

$$\frac{u_i^{n+1} + u_i^{n-1} - 2u_i^n}{\Delta t^2} = c^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \quad (6)$$

が得られる。これは(1)に2階の時間微分を2次差分で置き換えた物と一致する。(6)を書き換えると、

$$u_i^{n+1} = 2u_i^n - u_i^{n-1} + \Delta t^2 c^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \quad (7)$$

となる。 $t=0$  と  $t=-\Delta t$  2つ時間での初期値が必要である。

## 5.2 波動の数値シミュレーションのアルゴリズム

(7) 式を用いると波動のシミュレーションを行うことが可能である。必要な条件は、

- ・ 2つの時間  $t=0$  と  $t=\Delta t$  での初期条件

For  $i = 0, m+1$

$$u_m^{-1} = 0$$

$$u_m^0 = 0$$

- ・ 境界条件

境界条件は全ての時間ステップに対して必要である。

ディリクレ型境界条件－固定端または変位震動源

この場合は温度一定の境界条件の時と同様に扱う。すなわち、

$$u_0^n = 0$$

$$u_m^n = U_0 \cos \omega t$$

ディリクレ型境界条件－自由端 (stress free) あるいは応力境界

自由端の場合には熱流量 0 の境界条件と同じように扱えばよい。つまり、

$$\frac{\partial u}{\partial x} = 0$$

より、

$$u_m^n = u_{m+1}^n$$

である。応力が与えられる場合にも同様である。

速度境界条件

境界に対して、式 (5) を計算し変位を求めればよい。元々中点蛙跳び法なので、速度の時間ステップの位置がずれているに注意することが必要であろう。

また、2次の方法では、波の後方に数値的な振動を生じる。これを防ぐにはより高次の方法や高次の人工拡散の導入が必要である。

## 6. 連立一次方程式の解法

### 6.1 直接法と反復法

連立一次方程式の解法には同じ操作を繰り返し、収束したらそこで計算をやめる反復法と元数によって決まっている計算手順の回数で厳密に解が得られる直接法の2通りの方法がある。

### 6.2 ガウスの消去法

微分方程式を離散化した場合には、連立一次方程式が現れることが多く、これを解く必要が生じる。連立一次方程式を解くためのもっとも一般的な方法はガウスの消去法であるが、単純な操作の繰り返しであるので、コンピュータ向きの方法であり、数値計算でもよく用いられ、とくに密行列（要素がゼロであるところが少ない行列）に向く方法である。この方法は直接法の1つである。

#### 前進消去

$k = 1$  から  $n-1$  まで以下の処理を行う

第  $k$  行の  $a_{k+1}$  から  $a_n$  までを  $a_{kk}$  で割る（消去済みのところはすでに0）  
 ... (1)

$b_k$  を  $a_{kk}$  で割る

$i = k+1$  から  $n$  まで以下の処理を行う

第  $k$  行の（上記の新） $a_{ik}$  倍を第  $i$  行の  $a_{i,k+1}$  から  $a_{in}$  から引く  
 （上記の新） $b_k$  の  $a_{ik}$  倍を  $b_i$  から引く

この前進消去により行列  $A$  は左下半分のすべての要素が0となっている。

#### 後退代入

$$x_n = b_n / a_{nn}$$

$k = n-1$  から 1 まで以下の処理を行う。

$$x_k = b_k - \sum_{i=k+1}^n a_{ki} x_i$$

前進消去のときにピボットが0だと割ることができないし、0に近い値となると数値計算の精度が非常に悪くなるので (1) の処理を行う前にの中から  $a_{kk}, a_{k+1,k}, \dots, a_{nk}$  の中から絶対値最大のものを選び、その行を第  $k$  行に持ってくるという操作を行う。これを部分ピボット選択という。

また、前進消去のときの左辺の行列への操作と、右辺ベクトルへの操作は独立しているので、別々に行うことも可能である。ふつうのガウスの消去法のプログラムはそのように書かれているものが多い。つまり、

左辺  $A$  に対して、

$k=1$  から  $n-1$  まで以下の処理を行う

第  $k$  行の  $a_{k,k+1}$  から  $a_{k,n}$  までを  $a_{k,k}$  で割る (消去済みのところはすでに 0)

$i=k+1$  から  $n$  まで以下の処理を行う

第  $k$  行の (上記の新)  $a_{i,k}$  倍を第  $i$  行の  $a_{i,k+1}$  から  $a_{i,n}$  までから引く

右辺  $b$  に対して、

$k=1$  から  $n-1$  まで以下の処理を行う

$b_k$  を  $a_{k,k}$  で割る

$i=k+1$  から  $n$  まで以下の処理を行う

(上記の新)  $b_k$  の  $a_{i,k}$  倍を  $b_i$  から引く

とする。これは、行列  $A$  を下三角行列  $L$  と上三角行列  $U$  に分解し、すなわち

$$Ax = b \quad \rightarrow \quad LUx = b$$

としたあと、前進消去 (代入)

$$Ux = L^{-1}b$$

と後退代入

$$x = U^{-1}L^{-1}b$$

を行っていることと同じである。

このようなガウスの消去法のプログラムの例を別紙に示す。

### 6.3 対称行列の場合

この場合には行列の対称性を利用して計算回数を半分に減らすことが可能である。このような方法の代表的なものに修正コレスキー分解法がある。修正コレスキー分解法では行列  $A$  を

$$A = LDL^T$$

のように分解する。この分解を行った後に、右辺ベクトルの前進・後退代入

$$x = (LT)^{-1}D^{-1}L^{-1}b$$

により解を求める。

### 6.4 特殊な係数行列を持つ連立一次方程式の解法

拡散方程式 (定常解や陰解法) やポワソン (Poisson) 方程式を差分化したときに得られる特別な形の連立一次方程式を解く方法として次のような方法がある。

#### 一次元の拡散方程式の場合

行列  $A$  が対角要素とその両隣だけがゼロでない値を持つ三重対角行列 (tridiagonal matrix) になる。要素数が少なく、計算量、メモリー容量ともに少ないのでガウスの消去法が用いられる。

#### 二次元の拡散方程式の場合

行列  $A$  は帯行列になる。反復法が多くの場合用いられる。大きなメモリー容量が必要だ



が、直接法も十分可能である。境界条件だけが変わる場合のような、同じ行列に対して  
 右辺だけを変えて繰り返し計算を行うときには有利になる。

### 三次元の拡散方程式の場合

行列Aは帯行列になる。計算量とメモリー容量の関係から反復法が用いられる。

## 6.5 三重対角行列のガウスの消去法

1次元熱伝導の差分方程式を陰解法で表すと、

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \kappa \frac{T_{i+1}^{n+1} + T_{i-1}^{n+1} - 2T_i^{n+1}}{\Delta x^2}$$

となる。この式は、

$$\frac{\kappa \Delta t}{\Delta x^2} T_{i-1}^{n+1} + \left(1 - 2 \frac{\kappa \Delta t}{\Delta x^2}\right) T_i^{n+1} + \frac{\kappa \Delta t}{\Delta x^2} T_{i+1}^{n+1} = T_i^n$$

と書くことができ、係数を  $a_{i,j}$  とすると、連立一次方程式

$$0T_{i-1}^{n+1} + \dots + 0T_{i-2}^{n+1} + a_{i,i-1}T_{i-1}^{n+1} + a_{i,i}T_i^{n+1} + a_{i,i+1}T_{i+1}^{n+1} + 0T_{i+2}^{n+1} + \dots + 0T_m^{n+1} = T_i^n$$

となっていることが分かる。この係数行列は対角要素とその両隣だけがゼロでない値を持つ三重対角行列となっている。

三重対角行列の構造を利用すると効率の良いガウスの消去法が得られる

### 前進消去

左辺Aに対して、

$k=1$  から  $n-1$  まで以下の処理を行う

$a_{k,k+1}$  を  $a_{k,k}$  で割る

$a_{k,k+1}$  の (上記の新)  $a_{k+1,k-1}$  倍を  $a_{k+1,k+1}$  から引く

右辺bに対して、

$k=1$  から  $n-1$  まで以下の処理を行う

$b_k$  を  $a_{k,k}$  で割る

(上記の新)  $b_k$  の  $a_{i,k}$  倍を  $b_i$  から引く

### 後退代入

$$x_n = b_n / a_{nn}$$

$k=n-1$  から 1 まで以下の処理を行う。

$$x_k = b_k - a_{k,k+1}x_{k+1}$$

行列Aの要素はほとんど0なので

$$a_{k,k-1} \rightarrow a(k,1)$$

$$a_{k,k} \rightarrow a(k,2)$$

$$a_{k,k+1} \rightarrow a(k,3)$$

のように配列にいれると記憶容量を節約できる。対称行列の場合は

$$a_{k,k-1} \rightarrow a(k,1)$$

$$a_{k,k} \rightarrow a(k,2)$$

だけでよい。このようなやりかたで行列を扱う三重対角行列用のガウスの消去法のプログラム例を Web に示す。

境界条件は次のように扱う。今、格子点数を  $m+2$  個、 $i=0\sim m+1$  とする。 $i=0$  と  $i=m+1$  は境界上の値である。ディリクレ境界条件の時には、

$$a(1,1)*T(0)+a(1,2)*T(1)+a(1,3)*T(2)=b(1)$$

より、

$$a(1,2)*T(1)+a(1,3)*T(2)=b(1)-a(1,1)*T(0)$$

となる。ノイマン型の場合は、境界での微分値を

$$G = \frac{T(m+1)-T(m)}{\Delta x}$$

とする。この式から、境界の値は、

$$T(m+1)=T(m)+G\Delta x$$

と表される。これを差分方程式に代入すると、

$$a(m,1)*T(m-1)+\{a(m,2)+a(m,3)\}*T(m)=b(1)-a(m,3)*G\Delta x$$

が得られる。具体的には

$$\frac{\kappa\Delta t}{\Delta x^2}T_{m-1}^{n+1}+1T_i^{n+1}=T_i^n-\frac{\kappa\Delta t}{\Delta x}G$$

である。

## 6.6 帯対称行列に対する修正コレスキー法

帯行列は対角要素に近いある幅の範囲だけが 0 でない要素を持ち、それ以外は 0 である行列である。つまり 0 でない要素は斜めの帯状に分布する。この構造を利用すると、0 でない要素がある範囲 (帯幅という) だけを配列に記憶し、そこだけ計算するという、うまい方法が取れる。多くの方程式は近接相互作用を表すので、これを離散化して得られる連立一次方程式は帯行列となる。拡散方程式のように偶数階の空間微分だけを持つ場合には対称行列となる。

今、下図のように  $T_{i,j}$  の周りでの差分を考える。



と表される。Web ページで紹介するサブルーチンでは、行列の下三角部分を、

$k = 1$  から  $n$  まで

$$a_{sk} \rightarrow a(1, k)$$

$$a_{wk} \rightarrow a(m, k)$$

$$a_{pk} \rightarrow a(m+1, k)$$

のように配列に入れる。それ以外の  $a(i, j)$  には 0 を入れておく。ただし、下の境界と接する部分、つまり、 $i=1 \sim m$  の下三角行列が左にはみ出す部分 ( $a_{w1}$  等) は 0 を代入する。またこのとき、境界条件から生じる項は上三角行列の時と同様右辺に移項する。

### 6.7 反復法による連立一次方程式の解法

2次元、3次元の拡散方程式を差分化した連立方程式は元数も多くなり（例えば  $100 \times 100 \times 100$  のメッシュは 100 万円）、直接法だと記憶容量も計算時間も膨大になるので、反復法が有利となる。反復法にはガウス-ザイデル法や SOR 法などの差分方程式の逐次近似による線形反復法（緩和法）と、共役勾配法などの残差ベクトルの直交性を利用する方法がある。

### 6.8 緩和法によるポワソン方程式の解法

ここでは話を簡単にするためにポワソン方程式

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial z^2} = \rho$$

の差分方程式

$$\frac{\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{i,j}}{\Delta x^2} = \rho_{i,j}$$

を解くことを考える。簡単化のために  $\Delta x = \Delta y$  とする。この式を変形すると線形反復式

$$\phi_{i,j} = (\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} + \Delta x^2 \rho_{i,j}) / 4$$

が得られる。この様な方法を線形反復法あるいは緩和法 (relaxation method) と呼ばれる。 $(i, j) = (1, 1)$  から計算を始めて、 $(i, j) = (m, n)$  まで計算をして反復 1 回が終わりである。このとき、 $\phi_{i,j-1}$  と  $\phi_{i-1,j}$  は  $\phi_{i,j}$  を計算するときにはすでに計算済みの点であるので、 $\phi_{i-1,j}$  と  $\phi_{i,j-1}$  に更新されたばかりの値を使うと収束が速くなるし、 $\phi_{i,j}$  を記憶する配列が一つですむ。このやり方をガウス-ザイデル法 (Gauss-Seidel method) という。つまり、

$$\phi_{i,j}^{n+1} = (\phi_{i+1,j}^n + \phi_{i-1,j}^{n+1} + \phi_{i,j+1}^n + \phi_{i,j+1}^{n+1} + \Delta x^2 \rho_{i,j}) / 4$$

のように反復する。また、更新された値を使わない方法もあり、ヤコビ法 (Jacobi method) と呼ばれている。プログラム中では変数をスワイプするときに上書きするとガウス-ザイデル法となり、上書きしないように書くとヤコビ法である。ヤコビ法では並列計算に向いているのに対し、ガウス-ザイデル法は依存性があるので、そのままでは並列計算に向いていない。並列計算には Red-Black 法などが必要になる。

上の式は変形すると

$$\phi_{i,j}^{n+1} = \phi_{i,j}^n + (\phi_{i+1,j}^n + \phi_{i-1,j}^{n+1} + \phi_{i,j+1}^n + \phi_{i,j+1}^{n+1} - 4\phi_{i,j}^n + \Delta x^2 \rho_{i,j}) / 4 \quad \dots (*)$$

である。もちろん行列の形を使って書くこともでき、そのときは、

$$x_i^{n+1} = \left[ b_i - \sum_{j \neq i} (L_{ij} x_j^{n+1} + U_{ij} x_j^n) \right] / a_{ii}$$

である。ただし、 $L, U$  はそれぞれ係数行列  $A$  から対角要素を除いた行列の下三角・上三角行列である。ここでは、 $\phi, \rho$  の代わりに  $x, b$  を用いた。この反復式は行列  $A$  が定値対称行列で対角要素の絶対値が他の要素の絶対値に比べて大きい (対角優位である) ときのみ収束が保証されている。ポワソン方程式を差分化した式はこの条件を満たしている。線形反復法 (緩和法) では、実際に計算するときは元の (\*) 式のように行列の形を意識しなくてよく、簡潔である。

SOR 法 (Successive Over Relaxation Method) はガウス-ザイデル法の反復を加速した改良版である。(\*) 式の代わりに、加速係数  $\omega$  を導入して、

$$\phi_{i,j}^{n+1} = \phi_{i,j}^n + \omega (\phi_{i+1,j}^n + \phi_{i-1,j}^{n+1} + \phi_{i,j+1}^n + \phi_{i,j+1}^{n+1} - 4\phi_{i,j}^n + \Delta x^2 \rho_{i,j})$$

という式で反復を行う。ただし、

$$(1.0 \leq) \omega < 2.0$$

でなければならない ( $\omega < 1.0$  のときは不足緩和といわれる)。  $\omega$  はメッシュの数により最適値が決まる。SOR 法のプログラム例を別紙に示す。このプログラムではポワソン方程式を解くように設定されているが、係数のところを工夫すれば拡散方程式の陰解法にも利用できる。もちろん、拡散係数が変化する場合でもよい。

## 7. 参考文献など

### 7.1 参考文献

本セミナーでは下記の教科書を参考にした。

戸川隼人 「数値解析とシミュレーション\*」、共立全書、1976年。

矢部 孝・観山正見・椛島成治 「パソコンによるシミュレーション物理」 朝倉書店、1992年。

新井親夫 「Fortran90 入門ー基礎から再起手続きまでー」、森北出版、1998年。

### 7.2 参考資料

参考資料やプログラムが下記のウェブページに置いてある。

<http://www.geol.sci.hiroshima-u.ac.jp/~nakakuki/>

プログラム等 Lecture 1→地球内部物理学演習

Fortran の使い方 Lecture 1→地球内部物理学演習→Fortran90 利用法 (詳細版)