

情報数学C

Mathematics for Informatics C

第9回 偏微分方程式の数値解法
(放物型方程式, 楕円型方程式, 双曲型方程式)

情報メディア創成学類
藤澤誠

今回の講義内容

- **今日の問題**
- 境界値問題
- 有限差分法
- 1階偏微分方程式：移流方程式
- 2階偏微分方程式

今回の講義で解く問題

$$\frac{\partial f(x, y)}{\partial x} + \frac{\partial f(x, y)}{\partial y} = g(x, y)$$

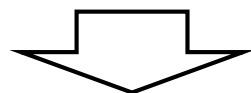
今回の講義で解く問題

偏微分方程式 (PDE : Partial Differential Equation)

2つ以上の独立変数を含む未知関数 $f(\boldsymbol{x})$ とその偏微分で構成される方程式

$$g\left(\boldsymbol{x}, f(\boldsymbol{x}), \frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}}, \frac{\partial^2 f(\boldsymbol{x})}{\partial \boldsymbol{x}^2}, \dots, \frac{\partial^n f(\boldsymbol{x})}{\partial \boldsymbol{x}^n}\right) = 0$$

ここで $n \geq 1$ で $\boldsymbol{x} = (x_1, x_2, \dots, x_m)$ が独立変数



求めたいのは、「未知関数 $f(\boldsymbol{x})$ 」

⇒ 数値解析としては「 \boldsymbol{x} が与えられたときの $f(\boldsymbol{x})$ の値を求める」ということ ($f(\boldsymbol{x})$ がどういう関数なのかは分かっていない)

今回の講義で扱う問題

偏微分方程式は**どんなところで使われる?**(前回と同じ)

自然現象, 工学モデルの多くはその支配方程式が常微分方程式/偏微分方程式*の形になる

物理

ニュートンの運動方程式($m \frac{\partial^2 x}{\partial t^2} = F$)が微分方程式であり、多くの力学の支配方程式はこれから導かれているため微分方程式になる

工学

熱力学の熱拡散方程式や流体力学のナビエ・ストークス方程式, 電磁気学のマクスウェル方程式, 量子力学のシュレディンガー方程式などほとんどの工学分野は微分方程式で表された支配方程式を基礎としている

これらの微分方程式のほとんどが**非線形微分方程式で数学的に解くことが難しい**(境界条件が簡単なら解けたりするけど実際の問題には適用しづらい)

*常微分方程式と偏微分方程式の違いは独立変数の数(要は $\frac{df}{dx}$ か $\frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}$ の違い)

今回の講義で解く問題

偏微分方程式を解いた結果の $f(x)$ は関数？

数値計算では具体的な数値が求まる. 関数 $f(x)$ の形(多項式とか)が決まっていないのにどうやって何を求めるのか？

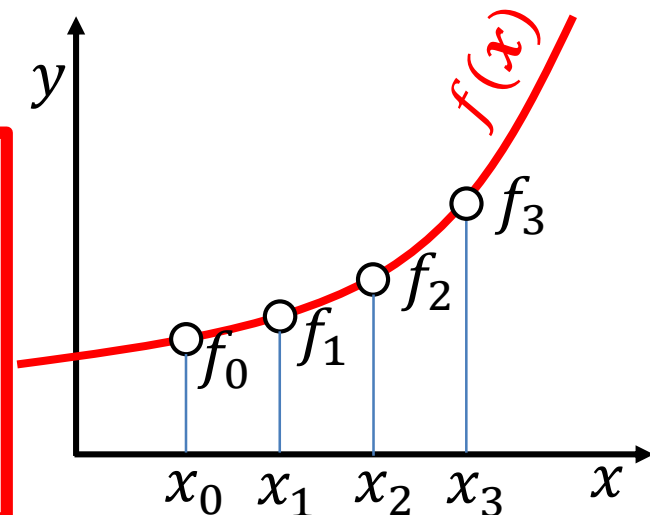
離散点 x_1, x_2, \dots, x_n における値 f_1, f_2, \dots, f_n を

- 初期値 (x_0, f_0) から順番に求めていく

⇒ 初期値問題

- 境界条件 $[a, b]$ ($f(a) = \alpha$, $f(b) = \beta$)に基づいて求める

⇒ 境界値問題



今回の講義で解く問題

今回の講義で扱う「偏微分方程式」についての前提条件

- 微分方程式の右辺項 $g(x, f(x), \dots)$ が**数式として与えられ**, 初期値 (x_0, f_0) が値として, 境界条件が条件式として**得られた**とき, **離散点 x_i における f_i の値**を求める($i = 1, 2, \dots$)
- $f(x)$ は**2つ以上の変数 x で表される**とする(つまり, df/dx ではなく $\partial f/\partial x$ と表される). 今回扱うのは $f(x)$ の境界条件が与えられたときの**境界値問題**
- $g(x)$ は**線形でも非線形でもどのような形でも良い**.

今回の講義内容

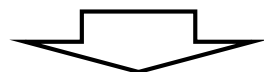
- 今日の問題
- **境界値問題**
- 差分法
- 1階偏微分方程式：移流方程式
- 2階偏微分方程式

境界値問題

境界値問題とは？

前回(常微分方程式)までは初期値問題:

x の範囲 $[a, b]$ に対して, $x_0 = a$ における値 $y_0 = f(x_0)$
を用いて, $y_i = f(x_i)$ ($i = 1, 2, \dots, n$)を順番に求めていく



最終的に y_n がどのような値になるのかは分からない

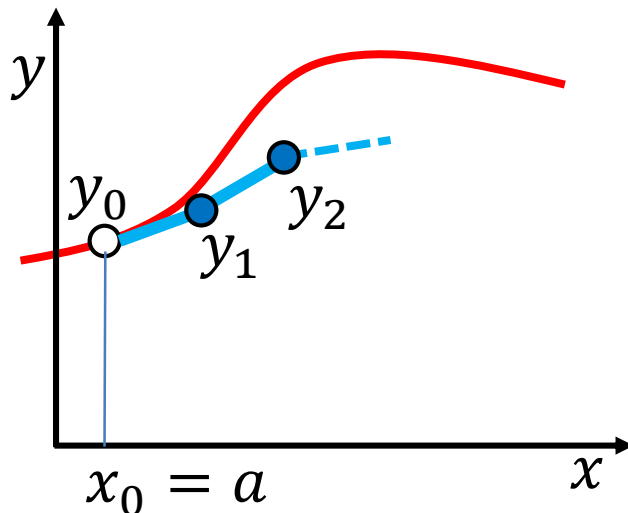
- 変数が1つだけの常微分方程式の場合, **時間 t** を変数にしてモデル化することが多く, ほとんどの場合, 初期値問題でOK
- 偏微分方程式の場合, 時間以外の別の軸(多くは**空間**)が含まれ, 空間軸の場合は**両端点での値/条件が決まっている問題**が多い

境界値問題

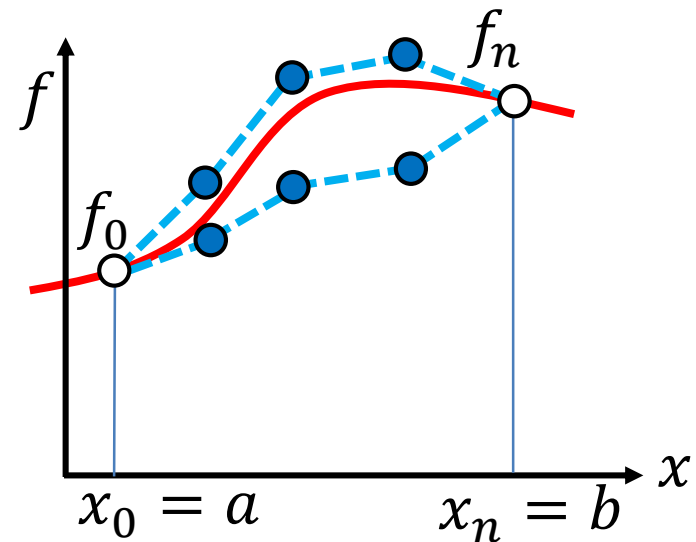
境界値問題とは？

2点境界値問題(1次元):

x の範囲 $[a, b]$ に対して, $x_0 = a$ における値 f_0 と $x_n = b$ における値 f_n を用いてその間の値 f_i ($i = 1, 2, \dots, n-1$)を求める



初期値問題

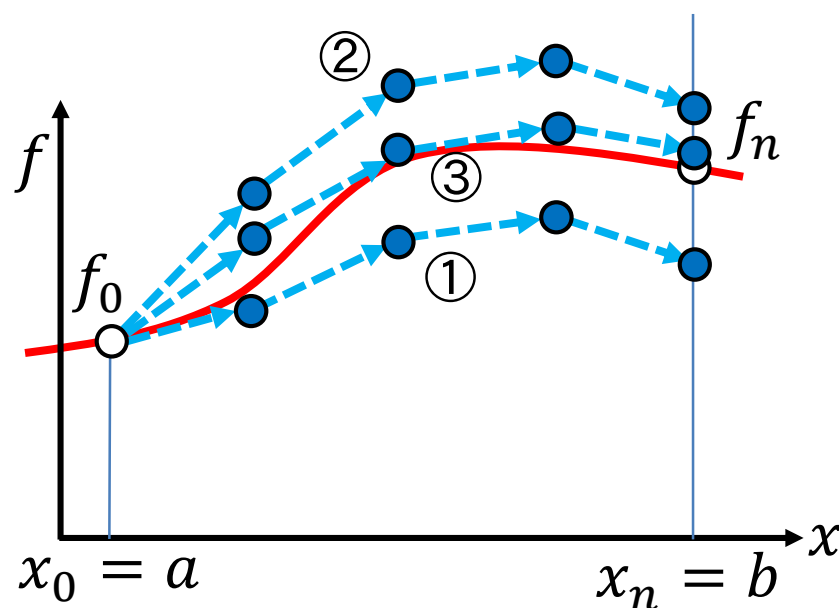


境界値問題

境界値問題

境界値問題を解くための方法1: 狙い撃ち法

初期値問題との違いは始点だけでなく終点も指定されること
⇒ 初期値問題と同じ方法で f_n まで求めて、指定値/条件に外れていたら修正する



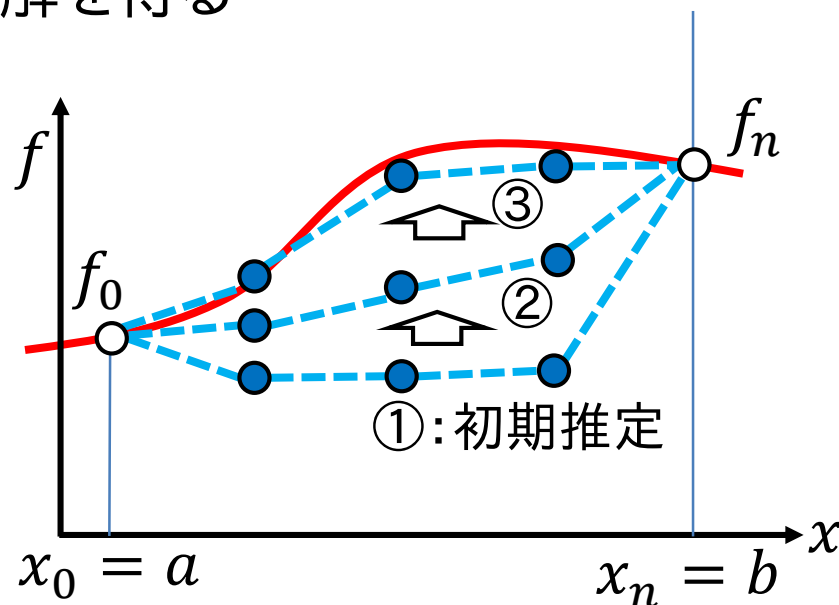
- ①: 1回目の試行
- ②: 2回目の試行
- ③: 3回目の試行
- ⋮

x_n でのずれを式で表し($B(x, f)$),
それを0にする
⇒ $B(x, f) = 0$ で求根問題となる
るのでニュートン法などが使える

境界値問題

境界値問題を解くための方法2:緩和法

初期推定値を $i = 0 \sim n$ のすべての点で与え、微分方程式及び境界値/条件に合うようにすべての点の値を修正する。この修正を何度も反復(緩和)することで最終的に近似解を得る



- ①: 全体の初期推定値を決める
- ②: 全体を修正(反復1回目)
- ③: 全体を修正(反復2回目)
- ⋮

複数の変数値を一度に解く
⇒線形システム $Ax = b$ の形になるのでCG法などのソルバーが使える

境界値問題

境界条件の種類

$f(x)$ に関する微分方程式の $x = a, b$ における境界条件を考える

ディリクレ(Dirichlet)条件

境界においてとるべき**値を指定**する

$$f(a) = f_a, \quad f(b) = f_b$$

⇒ 周囲環境の温度や密度, 壁面における速度など

ノイマン(Neumann)条件

境界においてとるべき条件を**勾配値として指定**する

$$\frac{\partial f}{\partial x}(a) = f_a, \quad \frac{\partial f}{\partial x}(b) = f_b$$

⇒ 流入/流出条件や大きな空間の一部を切り取った場合など

境界値問題

境界値問題について

- 今回の講義では基本的に**緩和法**を用いる
(実際の計算では境界条件を**多数の地点で同時に満たす**必要があり, その場合でも緩和法は**安定して解ける**)
- 偏微分方程式では基本的に,
 - **時間微分**については**初期値問題**として,
 - **空間微分**については**境界値問題**として
解く(2つの組み合わせになることがほとんど)

今回の講義内容

- 今日の問題
- 境界値問題
- **有限差分法**
- 1階偏微分方程式：移流方程式
- 2階偏微分方程式

有限差分法

以下の境界条件付きの2階微分方程式を(数値計算で)解くことを考えてみる

$$\frac{d^2 f(x)}{dx^2} = g(x) \quad \text{境界条件: } f(a) = \alpha, f(b) = \beta$$

$x \in [a, b]$ を n 分割し, 各点 x_i ($i = 0, 1, \dots, n$)での $f(x_i) = f_i$ を求めるとすると緩和法の手順は:

1. f_i ($i = 0, 1, \dots, n$)の初期値を設定
2. 微分方程式に合うように f_i を修正
3. 解が収束したら終了, 収束してなければ2に戻る

となるがここで**問題**となるのが手順2の「**微分方程式に合うように**」のところ \Rightarrow コンピュータで計算できるように
微分方程式を四則演算だけで表さなければならない

有限差分法

前回の講義で説明した**前進オイラー法**と微分の定義の関係を思いだそう！

前進オイラー法の式: $f_{i+1} = f_i + hg(x_i, f_i)$

⇩ 微分の定義で $h \rightarrow 0$ の極限をとらないものに相当

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x_i + h) - f(x_i)}{h} \approx \frac{f(x_i + h) - f(x_i)}{h}$$

この考え方を使えば、微分を四則演算だけに置き換えられる！

刻み幅を $h = (b - a)/n$ とすると微分は以下の式で近似できる

$$\frac{df}{dx} \approx \frac{f_{i+1} - f_i}{h}$$

前進差分近似

有限差分法

微分を差分に置き換えて近似する方法を**有限差分法**
(FDM: Finite Differential Method)もしくは単に**差分法**と呼ぶ

前進差分近似では計算点を前に進めたときの値 f_{i+1} を用いたが、
後ろ(x_{i-1})での値 f_{i-1} を用いても同様に近似できる

$$\frac{df}{dx} \approx \frac{f_i - f_{i-1}}{h}$$

後退差分近似

前進/後退差分近似の**誤差**は前進オイラーの1区間の場合
の誤差と同じく $O(h^2)$ (n 区間では $O(h)$ なので1次精度)

⇒ **精度**を上げるには？

⇒ **2次以上の微分**に対応するには？

有限差分法

前進/後退差分の式はテイラー展開からも求められる.

$$\underline{f(x_i + h) = f(x_i) + hf'(x_i) + \frac{h^2}{2}f''(x_i) + O(h^3)}$$

1次項までを変形すると前進差分近似になる

$$\underline{f(x_i - h) = f(x_i) - hf'(x_i) + \frac{h^2}{2}f''(x_i) + O(h^3)}$$

1次項までを変形すると後退差分近似になる

2次項まで考えれば精度が上げられそうだけど f'' が邪魔なので、両式の差をとって f'' の項を消してみる

$$f(x_i + h) - f(x_i - h) = 2hf'(x_i) + O(h^3)$$

$\Rightarrow O(h^3)$ 部分を無視して変形すると...

有限差分法

f_{i+1} と f_{i-1} を用いた差分式

$$\frac{df}{dx} \approx \frac{f_{i+1} - f_{i-1}}{2h}$$

中心差分近似

中心差分近似の誤差は $O(h^3)$ (n 区間全体で2次精度)

⇒ 他にも高精度な差分法があるがこれは後で説明

⇒ まずは設定した境界値問題を解くために**2次微分を近似**する方法を考える

有限差分法


3次項までテイラー展開:

$$f(x_i + h) = f(x_i) + hf'(x_i) + \frac{h^2}{2}f''(x_i) + \frac{h^3}{6}f'''(x_i) + O(h^4)$$

$$f(x_i - h) = f(x_i) - hf'(x_i) + \frac{h^2}{2}f''(x_i) - \frac{h^3}{6}f'''(x_i) + O(h^4)$$

今度は2つの式を足してみる

$$f(x_i + h) + f(x_i - h) = 2f(x_i) + h^2f''(x_i) + O(h^4)$$

 $f(x_i)$ を f_i として, f'' について解く

$$f''(x_i) \approx \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} : 2次微分の差分法による近似
(n 区間で3次精度)$$

有限差分法

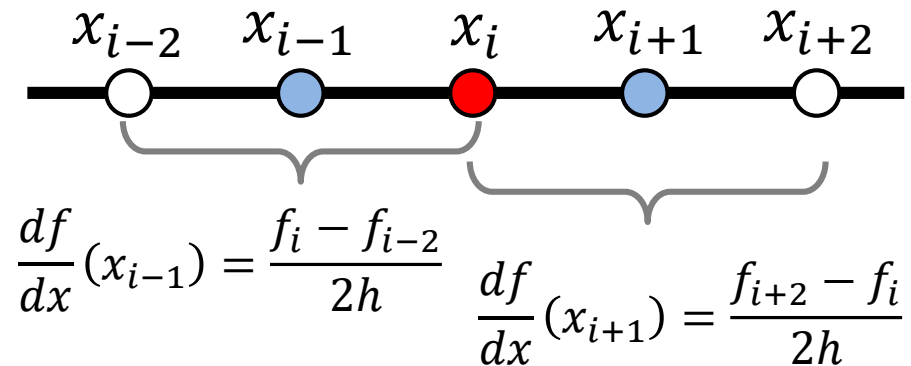
2次微分の近似式の意味を考えてみよう

差分法は1次元においては刻み幅 h で格子分割した点での値を用いて微分を近似する方法

⇒ 2次微分を1次微分を更に微分したものと考えれば、「微分値の差分」として中心差分近似できる

$$\frac{d^2 f}{dx^2} = \frac{d}{dx} \left(\frac{df}{dx} \right) \approx \frac{\frac{df}{dx}(x_{i+1}) - \frac{df}{dx}(x_{i-1})}{2h}$$

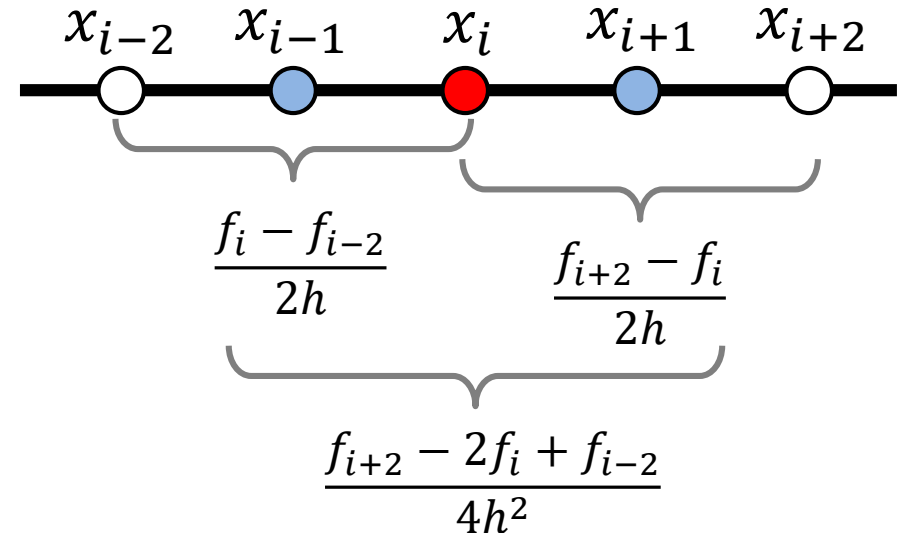
$\frac{df}{dx}(x_{i+1})$ と $\frac{df}{dx}(x_{i-1})$ は x_{i+1}, x_{i-1} での微分なのでそれぞれ右図のように中心差分近似できる



有限差分法

それぞれ代入して整理すると:

$$\frac{d^2 f}{dx^2} \approx \frac{f_{i+2} - 2f_i + f_{i-2}}{4h^2}$$



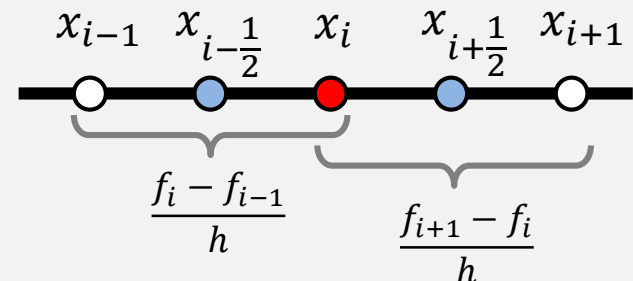
刻み幅 $2h \rightarrow h$ とすると:

$$\frac{d^2 f}{dx^2} \approx \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}$$

同じ式が導出できた!

($2h \rightarrow h$ とすると $f_{i+2} = f(x_i + 2h)$ が $f_{i+1} = f(x_i + h)$ になることに注意)

$2h \rightarrow h$ にすると右図のようになると考えればよい (最初から $x_{i-\frac{1}{2}}$ と $x_{i+\frac{1}{2}}$ での微分を中心差分で近似しても導出できる)



有限差分法

境界条件付き2階微分方程式の問題に戻ろう

$$\frac{d^2 f(x)}{dx^2} = g(x) \quad \text{境界条件: } f(a) = \alpha, f(b) = \beta$$

中心差分近似を用いることで左辺を差分式に近似できる

$$\frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} = g(x_i)$$



$$f_{i+1} - 2f_i + f_{i-1} = h^2 g(x_i)$$

この式が x_1 から x_{n-1} までの $n - 1$ 個の計算点で定義される
(x_0 と x_n での $f(x)$ は境界条件で決まっている)

有限差分法

$g_i = g(x_i)$ と書くと方程式は:

$$\begin{cases} f_0 - 2f_1 + f_2 = h^2 g_1 \\ f_1 - 2f_2 + f_3 = h^2 g_2 \\ \vdots \\ f_{n-2} - 2f_{n-1} + f_n = h^2 g_n \end{cases} \xrightarrow{\text{境界条件 } f_0 = \alpha \text{ より}} -2f_1 + f_2 = h^2 g_1 - \alpha$$

$$\xrightarrow{f_n = \beta \text{ より}} f_{n-2} - 2f_{n-1} = h^2 g_{n-1} - \beta$$

両辺に -1 を掛けて*,行列とベクトルでまとめて表す

$$\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-2} \\ f_{n-1} \end{pmatrix} = \begin{pmatrix} \alpha - h^2 g_1 \\ -h^2 g_2 \\ \vdots \\ -h^2 g_{n-2} \\ \beta - h^2 g_{n-1} \end{pmatrix}$$

$Ax = b$ の線形システムの形になっている!

⇒ ガウスザイデル法や共役勾配法などで解けば f_i が得られる

有限差分法

境界条件付き2階微分方程式を数値計算で解く.

$$\frac{d^2 f(x)}{dx^2} = g(x) \quad \text{境界条件: } f(a) = \alpha, f(b) = \beta$$

$x \in [a, b]$ を n 分割し, 各点 x_i ($i = 0, 1, \dots, n$)での $f(x_i) = f_i$ を求めるとすると緩和法の手順は:

1. f_i ($i = 0, 1, \dots, n$)の初期値を設定
2. 差分式を線形システムソルバで解くことで
 $f_1 \sim f_{n-1}$ を求める

反復処理や収束判定は2の線形システムソルバ内で行っている

有限差分法

緩和法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

```
double h = (x1-x0)/n; // 空間刻み幅
// 線形システムの係数行列と右辺項ベクトルの計算
vector<double> b(n-1, 0.0), x(n-1, 0.0);
vector< vector<double> > A(n-1, b);
for(int i = 1; i < n-2; ++i){
    double xi = x0+h*(i+1);
    A[i][i-1] = -1; A[i][i] = 2; A[i][i+1] = -1;
    b[i] = -h*h*g(xi);
}
A[0][0] = 2; A[0][1] = -1; b[0] = -h*h*g(x0+h)+alpha;
A[n-2][n-3] = -1; A[n-2][n-2] = 2; b[n-2] = -h*h*g(x1-h)+beta;
for(int i = 0; i < n-1; ++i) x[i] = f[i+1];

// CG法で線形システムを解く
int max_iter = 100;
double eps = 1e-6;
cg_solver(A, b, x, n-1, max_iter, eps);

// 結果を配列fに戻す
for(int i = 0; i < n-1; ++i) f[i+1] = x[i];
f[0] = alpha; f[n] = beta; // 境界条件
```

1次元配列bで初期化することで
 $n-1 \times n-1$ の2次元配列になる
(同時に0で初期化されている)

$i = 2 \sim n-2$ での係数行列,
右辺項の計算

$i = 1$ と $i = n-1$ での係数行列,
右辺項の計算(配列が0スタートなの
で要素位置が1つずれることに注意)

f_i の大きさが $n+1$ に対して, $Ax = b$
での x は $f_1 \sim f_{n-1}$ の $n-1$ なので別
の配列を使って解いた後に f_i に戻す
ようにしている

有限差分法

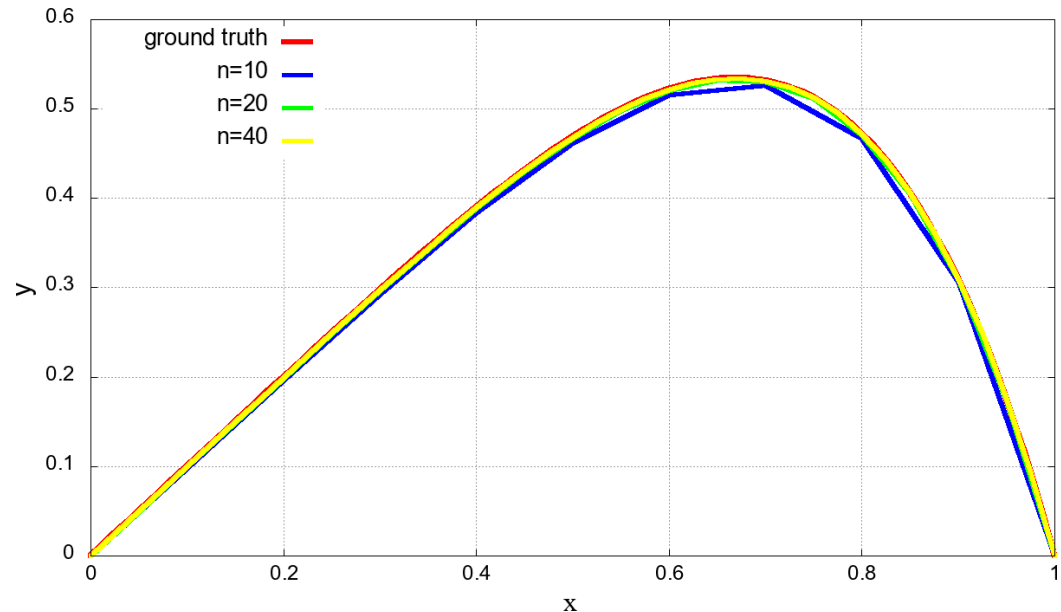
緩和法の実行結果例

$$\frac{d^2 f(x)}{dx^2} = -20x^3 \quad \text{境界条件: } f(0) = 0, f(1) = 0$$

\Rightarrow 解析解 $y = x - x^5$

範囲 $[0, 1]$, 初期値 $f_i^{(0)} = 0$, 刻み幅 $h = 0.1$

```
f(0) = 0, error = 0
f(0.1) = 0.09834, error = 0.00165
f(0.2) = 0.19648, error = 0.0032
f(0.3) = 0.29302, error = 0.00455
f(0.4) = 0.38416, error = 0.0056
f(0.5) = 0.4625, error = 0.00625
f(0.6) = 0.51584, error = 0.0064
f(0.7) = 0.52598, error = 0.00595
f(0.8) = 0.46752, error = 0.0048
f(0.9) = 0.30666, error = 0.00285
f(1) = 0, error = 0
avg. error = 0.00458333
```



$n = 10$ で平均誤差約0.0046, $n = 20$ で約0.0011, $n = 40$ で約0.00027であり,
分割数が増える(刻み幅が小さく)なれば精度は上がっていく

今回の講義内容

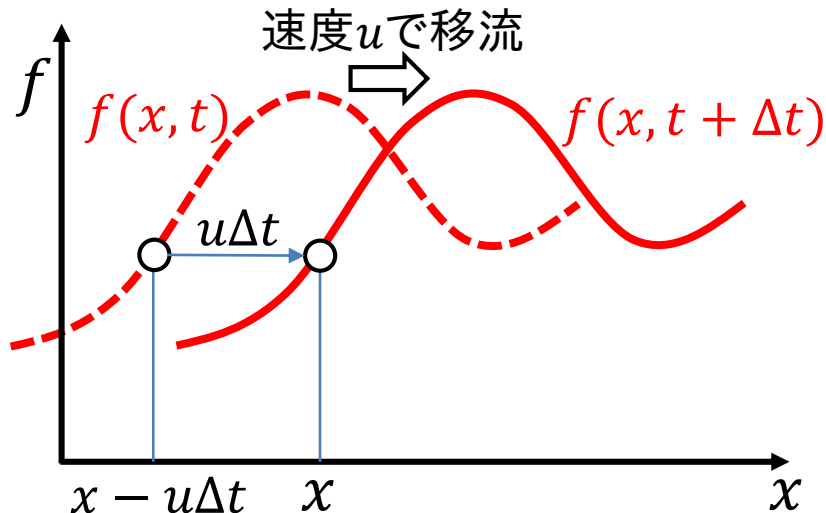
- 今日の問題
- 境界値問題
- 有限差分法
- **1階偏微分方程式：移流方程式**
- 2階偏微分方程式

1階偏微分方程式

境界値問題も解けるようになったので, 具体的な偏微分方程式を差分法で解いてみよう!

移流方程式(1階偏微分方程式の例)

位置 x と時間 t に関する関数 $f(x, t)$ が一定速度 u で流される場合(移流)を考えてみる



ある点 x での値は Δt 前に $x - u\Delta t$ にあった値が流れてきたと考えると:

$$f(x, t + \Delta t) = f(x - u\Delta t, t)$$

1階偏微分方程式

移流方程式(続き)

前ページ青枠の式の左辺を t で, 右辺を x についてテイラー展開:

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

2次項以降を無視して整理すると:

$$\boxed{\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = 0} \quad : \text{移流方程式}$$

移流方程式は何らかの速度場が与えられたときに関数(分布)がどう変化するかを計算するのに使われる

例) 煙や水の移流シミュレーション, 画像中の領域追跡など(レベルセット法が有名, 移流方程式自体は流体シミュレーションでは必須)

移流方程式

これまでに学んだ方法で移流方程式を解いてみよう

移流方程式の左辺第二項を右辺に移項して, $F(x, t)$ と置き換えると

$$\frac{\partial f}{\partial t} = -u \frac{\partial f}{\partial x} = F(x, t) \quad (u \text{は定数値とする})$$

前回の講義で見慣れた形であり, 現在の f から未来の f の計算したいとすると, **初期値問題**として解ける. **時間軸方向の刻み幅**を Δt として**前進オイラー法**で近似すると:

$$f^{(k+1)} = f^{(k)} + \Delta t F(x, t)$$

$F(x, t)$ に含まれる**空間微分**を**境界値問題**として刻み幅 h で**中心差分近似**すれば:

$$f_i^{(k+1)} = f_i^{(k)} - \frac{u\Delta t}{h} \frac{f_{i+1}^{(k)} - f_{i-1}^{(k)}}{2}$$

FTCS法

$$f_i^{(k+1)} = f_i^{(k)} - \frac{u\Delta t}{h} \frac{f_{i+1}^{(k)} - f_{i-1}^{(k)}}{2}$$

この更新式は、時間微分を前進オイラー法で、
空間微分を中心差分で近似するので

FTCS法 と呼ばれる(Forward-Time Centered-Space)

FTCS法で重要なのは $\nu = \frac{u\Delta t}{h}$ でこれは**クーラン数**と呼ばれる。

⇒ クーラン数は**計算の安定性**に関係する！

安定性に関して説明するためにも、
まずはコードと実行例を見てみよう

FTCS法

移流方程式のFTCS法による解法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

```
double h = (xn-x0)/n; // 空間刻み幅
double nu = u*dt/h;
for(int i = 1; i <= n-1; ++i){
    // ∂f/∂xの差分による近似
    double gx = (f0[i+1]-f0[i-1])/2.0;
    // fの値を更新
    f1[i] += -nu*gx;
}
f1[0] = alpha;
f1[n] = beta;
```

クーラン数 $\nu = \frac{u\Delta t}{h}$ の計算

中心差分 $\frac{f_{i+1}^{(k)} - f_{i-1}^{(k)}}{2}$ の計算

境界条件(ディリクレ条件)の設定

境界部分を考慮して反復の範囲が $1 \sim n-1$ になっていることに注意(f は $0 \sim n$ で定義されている)
この処理を $t = 0$ から Δt ずつ何回も反復することで時間とともに関数が移流する挙動がシミュレーションできる。

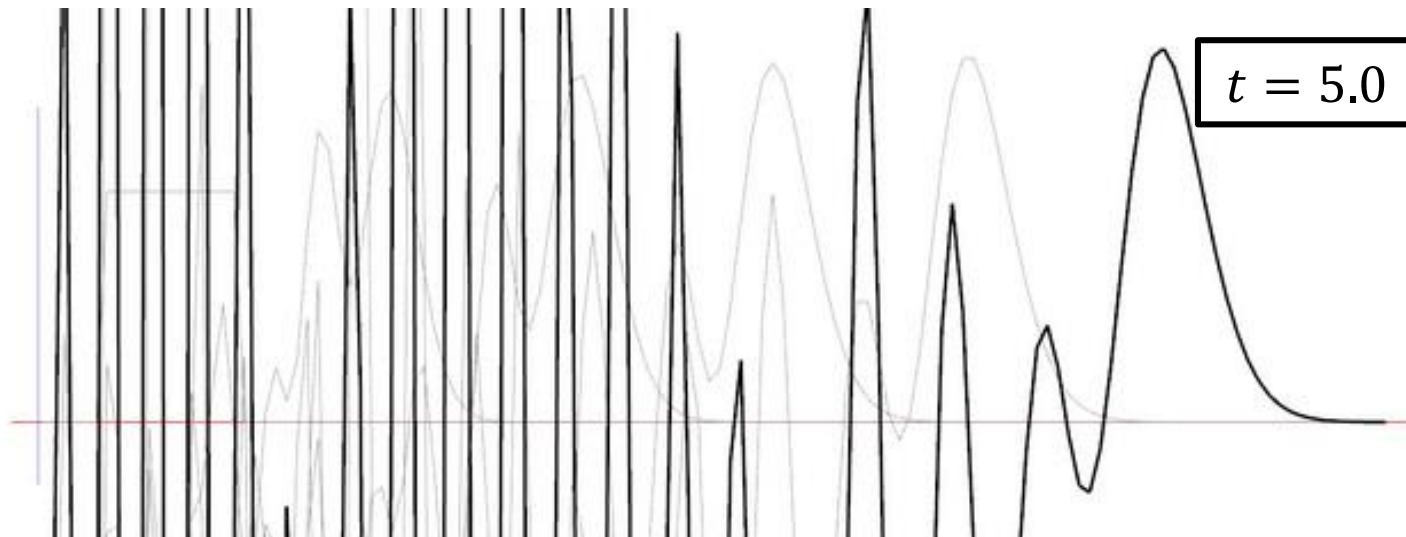
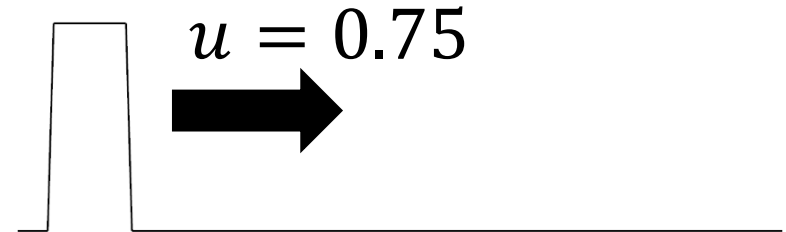
Githubのコード例では前進オイラーだけでなく、ホインやRK4とも組み合わせられるようにしているので注意(関数ポインタを多用). 境界条件の設定方法も違う(こちらに関数ポインタを利用)

FTCS法

FTCS法による結果: 矩形波の一定速度での移流

$f(x, t)$ の初期値として, 右のような矩形波(範囲 $x_a < x < x_b$ で1, それ以外で0となる関数)を設定

範囲 $[0, 5]$ を128分割にして,
 $u = 0.75, \Delta t = 0.1h/|u|$ で移流



⇒ 非常に不安定. **なぜ不安定なのか**を解析してみよう

FTCS法

von Neumannの安定性解析でFTCSが不安定になることを確かめてみる

安定性を確かめるために f が**波の形**だったとして, $f(x, t) = \cos(wx)$ (w は波数)を複素数に拡張した以下の式を実際に差分式に代入して解析してみる($\theta = w\Delta x$ とおく)

$$f_j^{(k)} = e^{ij\theta}$$

ここでの i は虚数単位を表し, これと区別するために空間方向の刻み幅を j にしています.

次のステップで波の形は変わらずに増幅(or減衰)している
とすると増幅係数 g を導入して:

$$f_j^{(k+1)} = g e^{ij\theta}$$

これらをFTCSの更新式に入れたときに $|g| < 1$ なら収束するので安定, $|g| > 1$ なら振幅が増えていって不安定(発散)になることが分かる.

FTCS法

von Neumannの安定性解析

実際にFTCS法での $|g|$ を計算してみよう.

$$f_j^{(k+1)} = f_j^{(k)} - \frac{\nu}{2} (f_{j+1}^{(k)} - f_{j-1}^{(k)}) \quad \left(\text{ここで } \nu = \frac{u\Delta t}{h} \right)$$

↓ $f_j^{(k+1)} = g e^{ij\theta}, f_j^{(k)} = e^{ij\theta}$ を代入

$$g e^{ij\theta} = e^{ij\theta} - \frac{\nu}{2} (e^{i(j+1)\theta} - e^{i(j-1)\theta})$$

↓ 両辺を $e^{ij\theta}$ で割る($e^{i(j+1)\theta} = e^{ij\theta} e^{i\theta}$ であることに注意)

$$g = 1 - \frac{\nu}{2} (e^{i\theta} - e^{-i\theta}) = 1 - i\nu \sin \theta$$

オイラーの公式より $i \sin \theta = \frac{1}{2}(e^{i\theta} - e^{-i\theta})$ なので

複素数の絶対値なので $|g| = \sqrt{1 + \nu^2 \sin^2 \theta} > 1$.

⇒ FTCS法は**無条件不安定**である*

移流方程式

移流方程式の**安定**した数値計算のためには？

- RK4や予測子修正子法, 陰的解法などの方法を時間微分に使う(前回の講義で説明済み)
- **空間微分**にもっと**安定**するような**or高精度な差分法**を使う
⇒ これからこちらについて説明していく
 - 風上差分
 - Lax-Wendroff
 - セミラグランジュ法
 - その他(QUICK,QUICKEST,河村・桑原法,ENO,WENO)
⇒ その他に挙げた方法は名前と特徴だけ紹介

風上差分法

風上差分法

移流方程式は速度 u による移流を表す

⇒ 情報が**風上から風下に流れている**と考えるなら、
差分にも**風上の値**を使うべきでは？

$$\begin{cases} f_i^{(k+1)} = f_i^{(k)} - v(f_i^{(k)} - f_{i-1}^{(k)}) & (u > 0 \text{ の場合}) \\ f_i^{(k+1)} = f_i^{(k)} - v(f_{i+1}^{(k)} - f_i^{(k)}) & (u < 0 \text{ の場合}) \end{cases}$$

風上差分法の安定性

$$g = 1 - v(1 - e^{-i\theta}) \Rightarrow |g| = \sqrt{1 - 2v(1 - v)(1 - \cos \theta)}$$

$(1 - \cos \theta) \geq 0$ ということを考えると $0 \leq v \leq 1$ なら $|g| \leq 1$ となる
ので風上差分は**安定**(精度は**1次精度**)

風上差分法

移流方程式の風上差分法による解法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

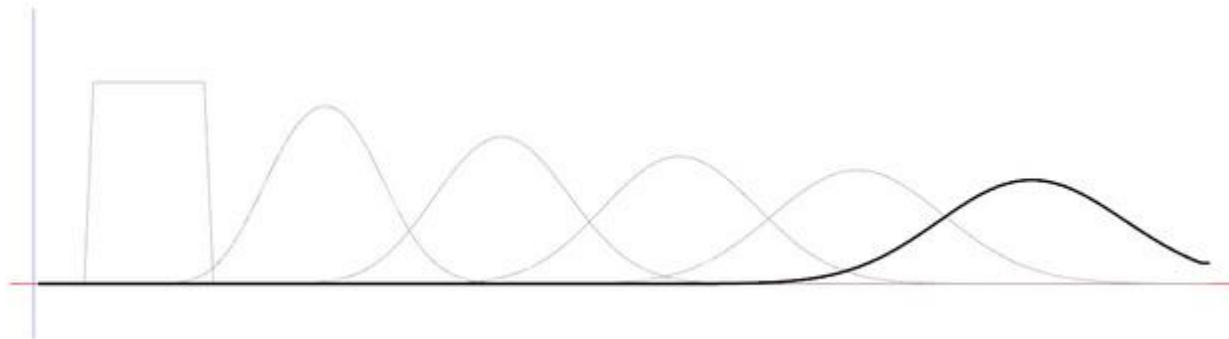
```
double h = (xn-x0)/n; // 空間刻み幅
double nu = u*dt/h; // クーラン数
for(int i = 1; i <= n-1; ++i){
    //  $\partial f / \partial x$ の差分による近似
    double gx;
    if(u > 0){
        gx = f0[i]-f0[i-1];
    }
    else{
        gx = f0[i+1]-f0[i];
    }
    // fの値を更新
    f1[i] += -nu*gx;
}
f1[0] = alpha;
f1[n] = beta;
```

速度 u の符号によって処理を分ける(多次元なら各軸方向毎に処理)

境界条件(ディリクレ条件)

風上差分法

風上差分法の結果: 矩形波の一定速度での移流
FTCS法の時と同じく矩形の移流(条件は全く同じ)



時間方向の刻み幅は同じだけど**非常に安定**している(矩形形状が崩れているので**精度が良いわけではない**ことに注意).

⇒**安定条件は?** ($\Delta t = 0.1h/|u|$ と設定)

$0 \leq v \leq 1$ (つまり $0 \leq \frac{u\Delta t}{h} \leq 1$) となるためには, $\Delta t \leq \frac{h}{|u|}$ となるように

時間方向の刻み幅を設定すればよい

⇒ この条件を**CFL条件**(Courant-Friedrich-Lewy条件)という

Lax-Wendroff法

Lax-Wendroff法 (ラックス・ウェンドロフ法)

$f_i^{(k+1)} = f(x_i, t + \Delta t)$ をテイラー展開すると:

$$f_i^{(k+1)} = f(x_i, t) + \Delta t \frac{\partial f}{\partial t}(x_i, t) + \frac{\Delta t^2}{2} \frac{\partial^2 f}{\partial t^2}(x_i, t) + O(\Delta t^3)$$

移流方程式を導出するときに**テイラー展開の2次項まで**考えれば

$\frac{\partial f}{\partial t} = -u \frac{\partial f}{\partial x}$ であるとともに $\frac{\partial^2 f}{\partial t^2} = u^2 \frac{\partial^2 f}{\partial x^2}$ なので,これを代入:

$$f_i^{(k+1)} = f(x_i, t) - u \Delta t \frac{\partial f}{\partial x}(x_i, t) + \frac{u^2 \Delta t^2}{2} \frac{\partial^2 f}{\partial x^2}(x_i, t) + O(\Delta t^3)$$

$\partial f / \partial x$ と $\partial^2 f / \partial x^2$ を**中心差分**で近似する:

$$f_i^{(k+1)} = f_i^{(k)} - u \Delta t \frac{f_{i+1}^{(k)} - f_{i-1}^{(k)}}{2h} + \frac{u^2 \Delta t^2}{2} \frac{f_{i+1}^{(k)} - 2f_i^{(k)} + f_{i-1}^{(k)}}{h^2} + O(\Delta t^3)$$

Lax-Wendroff法

Lax-Wendroff法

3次項以降を無視して整理($v = u\Delta t/h$)

$$f_i^{(k+1)} = f_i^{(k)} - \frac{v}{2} (f_{i+1}^{(k)} - f_{i-1}^{(k)}) + \frac{v^2}{2} (f_{i+1}^{(k)} - 2f_i^{(k)} + f_{i-1}^{(k)})$$

Lax-Wendroff法の誤差は前ページより $O(\Delta t^3)$, 計算精度としては n 回計算で**2次精度**

Lax-Wendroff法の安定性

$$g = 1 - \frac{v}{2} (e^{i\theta} - e^{-i\theta}) + \frac{v^2}{2} (e^{i\theta} - 2 + e^{-i\theta})$$
$$\Rightarrow |g| = \sqrt{1 - 2v^2(1 - v^2)(1 - \cos \theta)}$$

これも風上差分と同じく $0 \leq v \leq 1$ なら $|g| \leq 1$ となる.

\Rightarrow **CFL条件を満たす**ならLax-Wendroff法は**安定**

Lax-Wendroff法

移流方程式のLax-Wendroff法による解法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

```
double h = (xn-x0)/n; // 空間刻み幅
double nu = u*dt/h; // クーラン数
for(int i = 1; i <= n-1; ++i){
    // ∂f/∂xの差分による近似
    double gx = (f0[i+1]-f0[i-1])/2.0-nu*(f0[i+1]-2*f0[i]+f0[i-1])/2.0;
    // fの値を更新
    f1[i] += -nu*gx;
}
f1[0] = alpha;
f1[n] = beta;
```

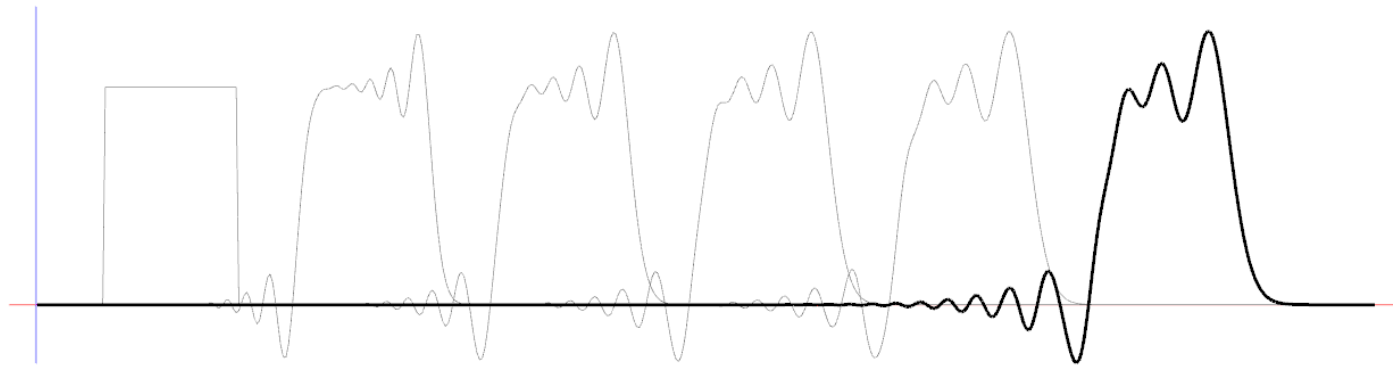
他のコードと同じような処理にするために

$$f_i^{(k+1)} = f_i^{(k)} - \nu \left(\frac{1}{2} (f_{i+1}^{(k)} - f_{i-1}^{(k)}) + \frac{\nu}{2} (f_{i+1}^{(k)} - 2f_i^{(k)} + f_{i-1}^{(k)}) \right)$$

という風に変形している

Lax-Wendroff法

Lax-Wendroff法の結果: 矩形波の一定速度での移流
FTCS法の時と同じく矩形の移流(条件はここまでと全く同じ)



振動はしているものの、**振幅は大きくはならず**, 元の形状の維持(一定速度移流なので元の形を保ったまま移動すべき)という点では**風上差分より優れている**(1次精度と2次精度の違い)

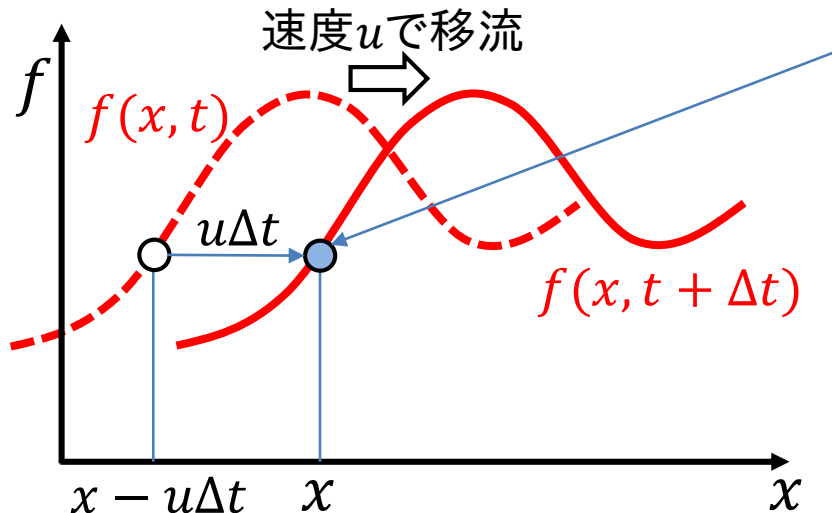
セミ・ラグランジュ法

セミ・ラグランジュ法(semi-Lagrangian method)

移流方程式の元になった式を思い出してみよう

$$f(x, t + \Delta t) = f(x - u\Delta t, t)$$

微分方程式を差分化しなくてもこれをそのまま計算できるのでは？ \Rightarrow セミ・ラグランジュ法*



この点における値は前のステップで $x - u\Delta$ の値が流れてきたと考えればよい

$$f_i^{(k+1)} = f(x_i - u\Delta t, t)$$

\Rightarrow 流れてきた位置を
バックトレースで求めている

セミ・ラグランジュ法

$f(x_i - u\Delta t, t)$ を**どうやって計算**するのか？

関数値 f が格納されているのは離散点 x_i ($i = 0, 1, \dots, n$)でのみ.

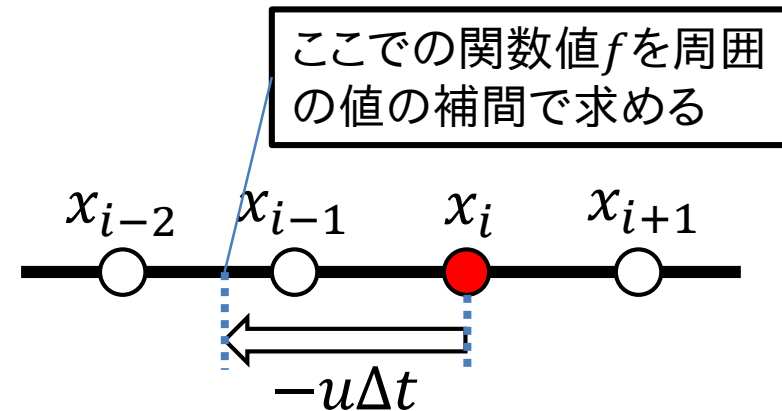
$u\Delta t = h$ ならよいけど必ずしもそうとは限らない

⇒ 何らかの補間を用いて求める必要がある！

- **線形補間**

風上差分と同じ(1次精度).

ただし, **CFL条件を満たさなくても安定して計算可能**



- **CIP法**(Constrained Interpolation Profile scheme)

スプライン補間を使って**3次多項式で補間**. 傾き(導関数)を格納する必要があるが**急激に変化する関数にも対応可能**

セミ・ラグランジュ法

移流方程式のセミ・ラグランジュ法による解法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

```
double h = (xn-x0)/n; // 空間刻み幅
for(int i = 1; i <= n-1; ++i){
    // 現在の位置からバックトレースした位置を求める
    double x = x0+i*h;
    x -= u*dt; // バックトレース

    // xが含まれるグリッド番号の計算
    int ib = (int)((x-x0)/h);
    if(ib < 0) ib = 0;
    if(ib >= n) ib = n-1;
    // グリッド位置からの距離を $[0,1]$ で正規化
    double dx = (x-ib*h)/h;

    // fの値を線形補間で求める
    f1[i] = (1-dx)*f0[ib]+dx*f0[ib+1];
}
```

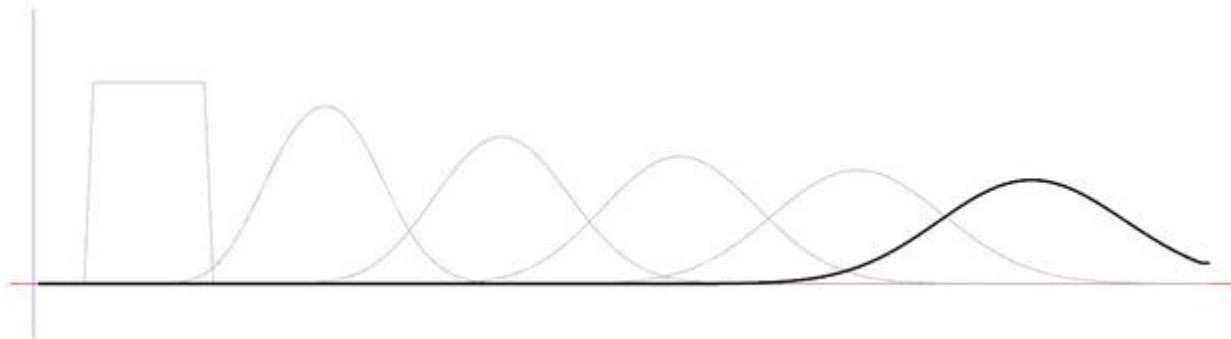
バックトレースした位置 x が含まれる区間番号(左側)は、範囲内の最小値を原点とした座標値 $x - x_0$ を刻み幅 h で割って小数点以下切り捨てで求められる(番号が $0 \sim n$ になるようなクランプ処理も忘れずに)

線形補間用に含まれる区間の左端からの距離を刻み幅で割ったものを計算しておく

セミ・ラグランジュ法

セミ・ラグランジュ法(線形補間)の結果

FTCS法の時と同じく矩形の移流(条件はここまでと全く同じ)



結果としては風上差分法とほぼ同じ(線形補間が1次精度なので当然だが). ただし, こちらの方法はCFL条件に関係なく安定している(Δt を大きくしても安定)
⇒ 高次の補間を用いれば精度は上げられる*

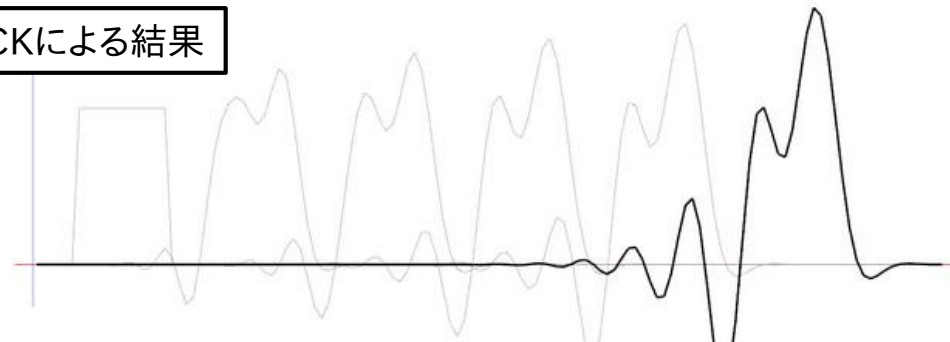
*高次補間を使った場合の例は[こちらのリンク](#)参照

1階偏微分方程式

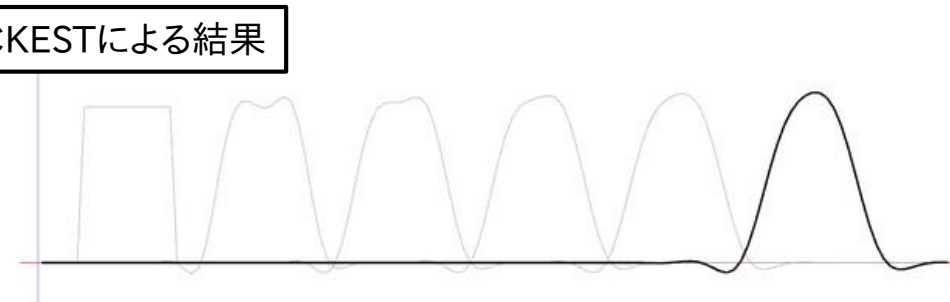
その他の高精度差分法*

- **QUICK** : f_{i-1}, f_i, f_{i+1} の3点+風上側のもう一つの $f(f_{i-2}$ or $f_{i+2})$ を使った差分法(2次精度)
- **QUICKEST** : Lax-Wendroff法+風上側のもう一つの $f(f_{i-2}$ or $f_{i+2})$ を使った差分法(3次精度)

QUICKによる結果



QUICKESTによる結果



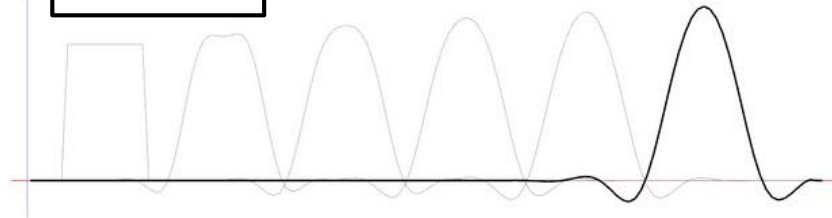
*これらの方法についての詳細は[こちらのリンク](#)を参照

1階偏微分方程式

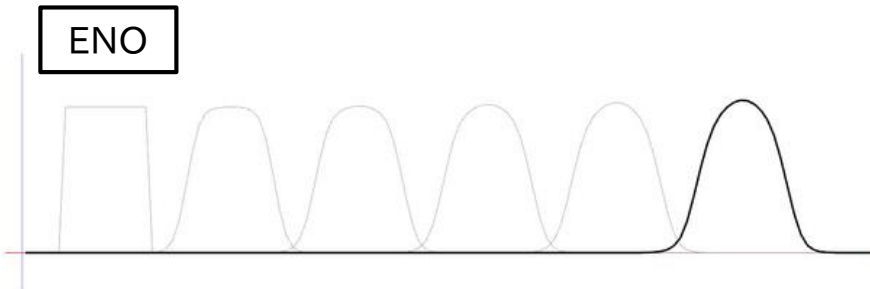
その他の高精度差分法

- 河村・桑原スキーム：4次精度中心差分+4次微分項の3次精度の風上差分
- ENO,WENO：風上差分を改良して3次多項式で近似した3次精度差分法.WENOはENOで重み付き平均をとることで5次精度にしたもの

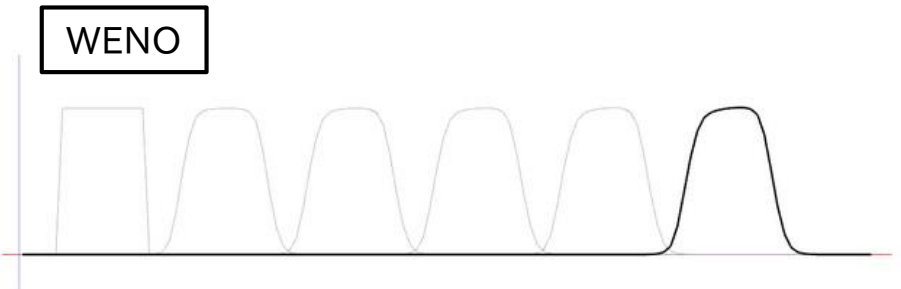
河村・桑原



ENO



WENO



1階偏微分方程式

移流方程式の近似手法について*

- **保存系の差分法**：物体が流れることを考えるとある領域内に入ってきた量(流束)と出てきた量(流束)は等しくなる(非圧縮なら). このような保存を考えた差分法もある(保存系風上差分, 保存系 Lax-Wendroffなど). 保存系は特に速度が場所によって変化する場合に重要.
- **Godunovの定理**：「移流方程式で全領域で同じ差分法をつかうと2次精度以上で単調性を維持できない(数値振動する)」というもの. 単調性を維持するために**流束制限関数**というものを導入して安定性を高める方法がある.
- **TVDスキーム**：計算点毎に**値の変化量の総和(TV:Total Variation)**を定義したとき, TVが時間とともに増大しないという条件(**TVD条件**)を満たした方法

*今回の講義では時間の関係で説明しないが, 今後困ったことがあったらこれらのキーワードで調べると良さそう

1階偏微分方程式

2次元以上での移流方程式はどうなる？

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} = 0 \quad : \text{2次元移流方程式}$$

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} + w \frac{\partial f}{\partial z} = 0 \quad : \text{3次元移流方程式}$$

u, v, w はそれぞれ x, y, z 方向速度

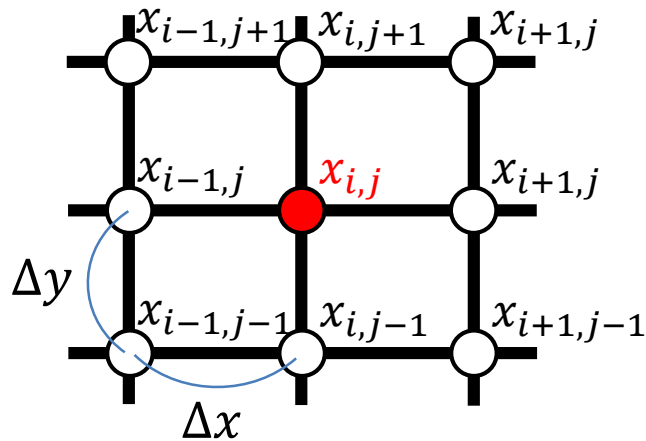
⇒ 全部まとめてベクトル表記すると：

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

$$\mathbf{u} = (u, v, w)^T, \nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)^T \Rightarrow \nabla \text{は勾配 (gradient) と呼び, } f \text{ がスカラー値の場合 } \nabla f \text{ はベクトルになることに注意}$$

1階偏微分方程式

2次元の場合で計算空間を格子状に分割してみる



x 方向刻み幅を Δx , y 方向刻み幅を Δy とすると中心差分は:

$$\frac{\partial f}{\partial x} \approx \frac{f_{i+1,j} - f_{i-1,j}}{2\Delta x} \quad \frac{\partial f}{\partial y} \approx \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta y}$$

時間微分に前進オイラー, 空間微分に中心差分を用いた(要はFTCS)
2次元移流方程式は:

$$f_{i,j}^{(k+1)} = f_{i,j}^{(k)} + \frac{u\Delta t}{\Delta x} \frac{f_{i+1,j} - f_{i-1,j}}{2} + \frac{v\Delta t}{\Delta y} \frac{f_{i,j+1} - f_{i,j-1}}{2}$$

⇒ 3次元でも同じ. 基本的には1次元をそのまま拡張でOK. ただし, セミラグランジュ法の場合は補間を多次元で行わなければならないことに注意

今回の講義内容

- 今日の問題
- 境界値問題
- 有限差分法
- 1階偏微分方程式：移流方程式
- **2階偏微分方程式**

2階偏微分方程式の分類分け

1階偏微分方程式をここまで説明してきたが、実際の物理現象の多くは**2階偏微分方程式でモデル化**される

⇒ 2階偏微分方程式についても考えてみよう!

どんなモデル化がされているのかを説明するために、まずは**2階偏微分方程式を分類分け**する

2階偏微分方程式を定数係数 A, B, C を使って以下の形で定義する

$$A \frac{\partial^2 f}{\partial x^2} + 2B \frac{\partial^2 f}{\partial x \partial y} + C \frac{\partial^2 f}{\partial y^2} = g \left(x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

xy 平面内のある曲線上でこの方程式が解けるとする

⇒ この曲線を特性曲線と呼ぶ

2階偏微分方程式の分類分け

特性曲線に沿った方向の微分(特性曲線の傾き) dy/dx について以下が成り立つ

$$A dy^2 + 2B dx dy + C dx^2 = 0$$

傾き dy/dx について解くと:

$$\frac{dy}{dx} = \frac{-B \pm \sqrt{B^2 - AC}}{A}$$

判別式 $B^2 - AC$ の符号によって特性曲線が実数or複素数or一つのみとなり, **特性曲線がどうなるかで2階微分方程式を分類分けできる.**

$B^2 - AC > 0$: 特性曲線は2つの実定数で表される \Rightarrow **双曲型**

$B^2 - AC = 0$: 特性曲線は1つの実定数で表される \Rightarrow **放物型**

$B^2 - AC < 0$: 特性曲線は2つの複素定数で表される \Rightarrow **楕円型**

2階偏微分方程式の分類分け

各分類で代表的な2階偏微分方程式

双曲型($B^2 - AC > 0$) :

$$\frac{\partial^2 f}{\partial t^2} - c^2 \frac{\partial^2 f}{\partial x^2} = 0 \quad : \text{波動方程式 (波の運動を表す)}$$

放物型($B^2 - AC = 0$) :

$$\frac{\partial f}{\partial t} = a \frac{\partial^2 f}{\partial x^2} \quad : \text{拡散方程式 (熱伝導など物体/物理量が広がっていく(拡散する)様子を表す)}$$

楕円型($B^2 - AC < 0$) :

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = g(x, y) \quad : \text{ポアソン方程式 (静電ポテンシャル, 重力ポテンシャル場などを求めるのに使われる)}$$

$\Rightarrow g(x, y) = 0$ のときはラプラス方程式となる

拡散方程式(FTCS法)

拡散方程式を数値計算で解いてみよう*

$$\frac{\partial f}{\partial t} = a \frac{\partial^2 f}{\partial x^2}$$

移流方程式の時と同じく, まずは前進オイラー法+中心差分で近似してみる(FTCS法)

$$\frac{f_i^{(k+1)} - f_i^{(k)}}{\Delta t} = a \frac{f_{i+1}^{(k)} - 2f_i^{(k)} + f_{i-1}^{(k)}}{\Delta x^2}$$

安定性は? $g = 1 - 2\mu(1 - \cos \theta)$

$\mu = a\Delta t/\Delta x^2$ が0以上1/2以下なら安定. $a > 0$ なら安定条件は:

$$\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{a} \quad \Rightarrow \quad \text{空間方向の分割幅}\Delta x\text{に対して}\Delta t\text{をかなり小さくする必要はあるが無条件不安定ではない.}$$

拡散方程式(FTCS法)

拡散方程式のFTCSによる解法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

```
double dx = (xn-x0)/n; // 空間刻み幅
double mu = a*dt/(dx*dx);
for(int i = 1; i <= n-1; ++i){
    f1[i] = f0[i]+mu*(f0[i+1]-2*f0[i]+f0[i-1]);
}
f1[0] = alpha;
f1[n] = beta;
```

移流の時と同じく, 境界部分を考慮して反復の範囲が $1 \sim n-1$ になっていることに注意(f は $0 \sim n$ で定義されている). また, 2次元化に備えて $h \rightarrow \Delta x$ としている.

拡散方程式(FTCS法)

拡散方程式をFTCS法で解いた結果

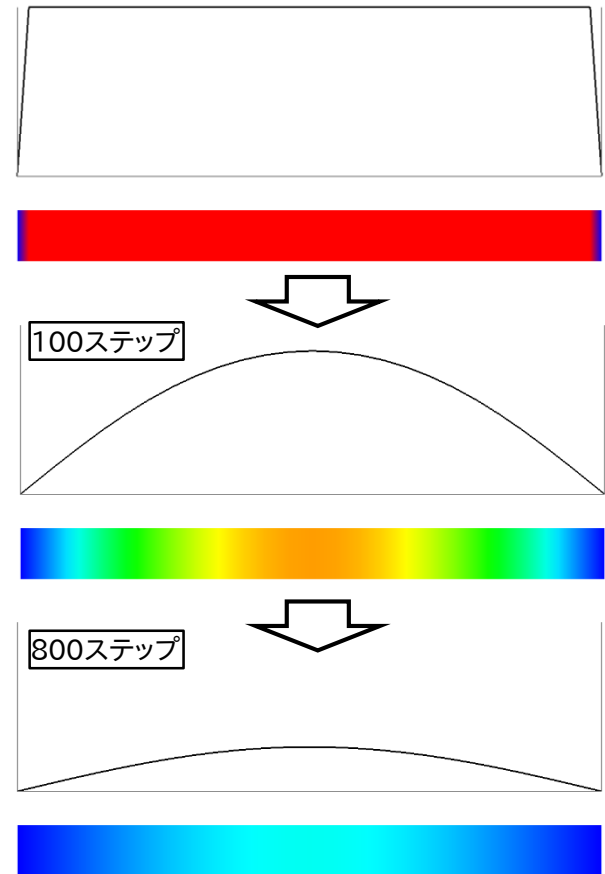
初期値 $f_i = 1$, 境界条件 $f_0 = f_n = 0$
計算範囲 $[0,1]$, 分割数 $n = 50 (\Delta x = 0.01)$
拡散係数 $a = 0.02$



安定条件 $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{a}$ から $\Delta t = 0.01$ と設定

f を温度場とすると, 両端が低温状態のときの物体内部の熱の変化(熱伝導)が拡散方程式で再現できていることが分かる

⇒ 安定条件を満たさなかったら?



$\Delta t = 0.01$

拡散方程式(FTCS法)

拡散方程式をFTCS法で解いた結果

初期値 $f_i = 1$, 境界条件 $f_0 = f_n = 0$
計算範囲 $[0,1]$, 分割数 $n = 50 (h = 0.01)$
拡散係数 $a = 0.02$

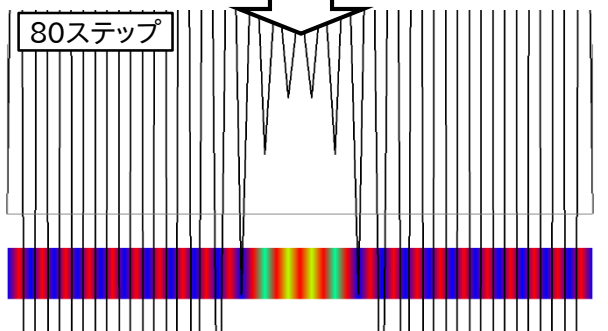


安定条件 $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{a}$ から少しだけ外れるように $\Delta t = 0.011$ と設定

条件を満たしていないと振動が発生し、解が発散してしまっている

⇒ 時間方向の刻み幅(時間ステップ幅)の制限が強い

⇒ 制限をなくすには？



$\Delta t = 0.011$

拡散方程式(クランク・ニコルソン法)

拡散方程式を陰的解法で解いてみよう

前回の講義で説明した陰的解法(後退オイラー法など)を適用してみよう.

$$f_{i+1} = f_i + \Delta t \phi(\mathbf{x}_{i+1}, f_{i+1}, \mathbf{x}_i, f_i)$$

クランク・ニコルソン法

$$\frac{f_i^{(k+1)} - f_i^{(k)}}{\Delta t} = a \left(\lambda \frac{f_{i+1}^{(k+1)} - 2f_i^{(k+1)} + f_{i-1}^{(k+1)}}{\Delta x^2} + (1 - \lambda) \frac{f_{i+1}^{(k)} - 2f_i^{(k)} + f_{i-1}^{(k)}}{\Delta x^2} \right)$$



展開して $f^{(k+1)}$ の項だけを右辺に集めれば

$$A \mathbf{f}^{(k+1)} = \mathbf{b}(\mathbf{f}^{(k)})$$

線形システム*の形になるのでこれを共役勾配法などで解けばよい

*係数行列や右辺項ベクトルは[境界値問題での緩和法](#)と同じような形になる

拡散方程式

クランク・ニコルソン法のコード例

(微分方程式を範囲 $[x_0, x_n]$ で n 分割して解く)

```
double dx = (xn-x0)/n; // 空間刻み幅
double eta = a*dt/(dx*dx);

// 線形システムの係数行列と右辺項ベクトルの計算
vector<double> b(n-1, 0.0), x(n-1, 0.0);
vector< vector<double> > A(n-1, b);
for(int i = 0; i < n-1; ++i){
    int j = i+1; // 行列Aのi行目はf[i+1]の式となる
    double xi = a+dx*j;
    if(i != 0) A[i][i-1] = -cn_lambda*eta;
    A[i][i] = 1+2*cn_lambda*eta;
    if(i != n-2) A[i][i+1] = -cn_lambda*eta;
    b[i] = f0[j]+(1-cn_lambda)*eta*(f0[j+1]-2*f0[j]+f0[j-1]);
}
b[0] += cn_lambda*eta*f0[0];
b[n-2] += cn_lambda*eta*f0[n];
for(int i = 0; i < n-1; ++i) x[i] = f0[i+1];

// CG法で線形システムを解く
int max_iter = 100; double eps = 1e-6;
cg_solver(A, b, x, n-1, max_iter, eps);

// 結果を配列fに戻す
for(int i = 0; i < n-1; ++i) f1[i+1] = x[i];
```

1次元配列bで初期化することで
 $n-1 \times n-1$ の2次元配列になる
(同時に0で初期化されている)

f_i ($i = 1 \sim n-1$)についての係数
行列の行, 右辺項の計算
(配列が0スタートなので要素位置が1
つずれることに注意)

$i = 1$ と $i = n-1$ での係数行列,
右辺項に境界値 $f_0 = \alpha, f_n = \beta$ の
項が含まれる

f_i の大きさが $n+1$ に対して, $Ax = b$ で
の x は $f_1 \sim f_{n-1}$ の $n-1$ なので別の配
列を使って解いた後に f_i に戻すように
している

拡散方程式

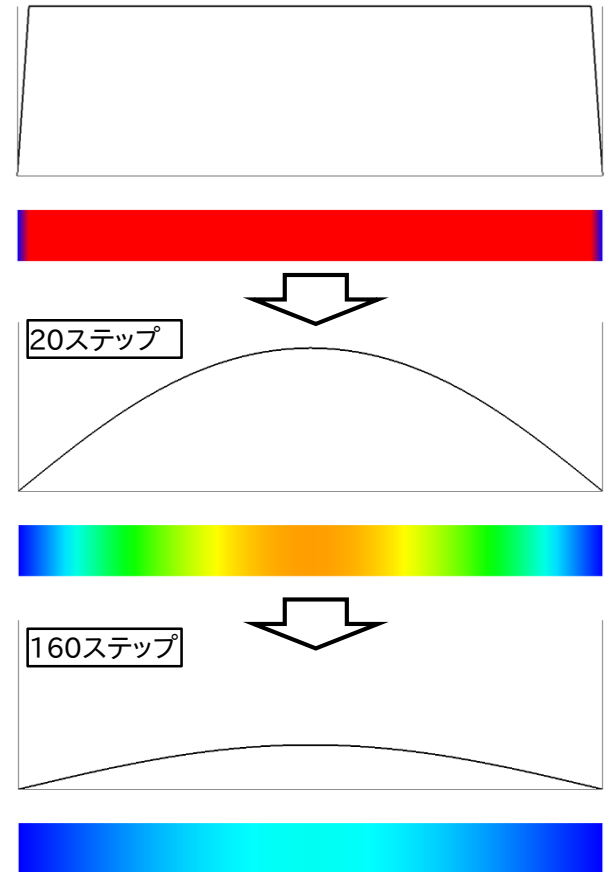
クランク・ニコルソン法で拡散方程式を解いた結果

初期値 $f_i = 1$, 境界条件 $f_0 = f_n = 0$
計算範囲 $[0,1]$, 分割数 $n = 50$ ($h = 0.01$)
拡散係数 $a = 0.02$



FTCS法の安定条件 $\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{a}$ から
大きく外れる $\Delta t = 0.05$ を設定

条件を満たす最大値の5倍でも**安定して計算**できている. 計算時間は掛かるがその分**少ないステップ数で同じ結果**が得られている



$\Delta t = 0.05$

拡散方程式

2次元の拡散方程式はどう離散化する？

$$\frac{\partial f}{\partial t} = a \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \right) \quad \Rightarrow \quad \frac{\partial f}{\partial t} = a \nabla^2 f$$

基本的には1次元の時と同じ。2次元の移流方程式の時と同様に格子状に計算点を設定し、前進オイラー法+中心差分(FTCS法)で近似した場合は：

$$\frac{f_{i,j}^{(k+1)} - f_{i,j}^{(k)}}{\Delta t} = a \left(\frac{f_{i+1,j}^{(k)} - 2f_{i,j}^{(k)} + f_{i-1,j}^{(k)}}{\Delta x^2} + \frac{f_{i,j+1}^{(k)} - 2f_{i,j}^{(k)} + f_{i,j-1}^{(k)}}{\Delta y^2} \right)$$

⇓ $\Delta x = \Delta y = h$ の場合

$$f_{i,j}^{(k+1)} = f_{i,j}^{(k)} + \frac{a\Delta t}{h^2} \left(f_{i+1,j}^{(k)} + f_{i-1,j}^{(k)} + f_{i,j+1}^{(k)} + f_{i,j-1}^{(k)} - 4f_{i,j}^{(k)} \right)$$

波動方程式

楕円型のポアソン方程式は差分法の例で解き、
放物型の拡散方程式も差分法で近似解が求められた。

⇒ **双曲型の波動方程式***は？

$$\frac{\partial^2 f}{\partial t^2} - c^2 \frac{\partial^2 f}{\partial x^2} = 0 \quad \Rightarrow \quad \frac{\partial^2 f}{\partial t^2} = c^2 \nabla^2 f$$

これまでと同じくそれぞれの微分を差分法で近似すれば良い。
ただし、時間微分が2次微分になっているので、 $f^{(k+1)}$ をもとめるために、 $f^{(k)}$ だけでなく $f^{(k-1)}$ も使うことになる。

中心差分+中心差分(CTCS法)で近似した場合は：

$$\frac{f_i^{(k+1)} - 2f_i^{(k)} + f_i^{(k-1)}}{\Delta t^2} = c^2 \left(\frac{f_{i+1}^{(k)} - 2f_i^{(k)} + f_{i-1}^{(k)}}{\Delta x^2} \right)$$

*1次元の波動方程式は $\left(\frac{\partial^2}{\partial t^2} - c^2 \frac{\partial^2}{\partial x^2} \right) f = 0$ とすると因数分解できて、
結局移流方程式と同じになるとも考えられる

差分法以外の解法

有限差分法(FDM:Finite Differential Method)以外で偏微分方程式を解く方法*

- **有限体積法**(FVM:Finite Volume Method) : 分割した領域(区間)毎に積分を含む**保存方程式**を適用する方法. 非構造格子向き.
- **有限要素法**(FEM:Finite Element Method) : 物理量を値ではなく**形状関数の線形結合**で表すことで高精度に微分方程式を解くことができる. こちらも格子形状の自由度が高く**色々な分野で使われている**とても有名な手法.
- **境界要素法**(BEM:Boundary Element Method) : 空間を領域(要素)で区切るのではなく, **領域を囲む境界のみ**を離散化, 解析する方法.
- **粒子法**(Particle Method) : 自由に動き回れる**粒子を計算点**として, **周囲の粒子の重み付き平均**で物理量を定義, 微分方程式を離散化して解く方法.

講義内容のまとめ

- 今日の問題：偏微分方程式
- 境界値問題
 - 境界条件, 緩和法
- 有限差分法
 - 前進/後退差分近似, 中心差分近似
- 1階偏微分方程式：移流方程式
 - FTCS法, 風上差分, Lax-Wendroff法, セミ・ラグランジュ法
- 2階偏微分方程式
 - 拡散方程式, クランク・ニコルソン法

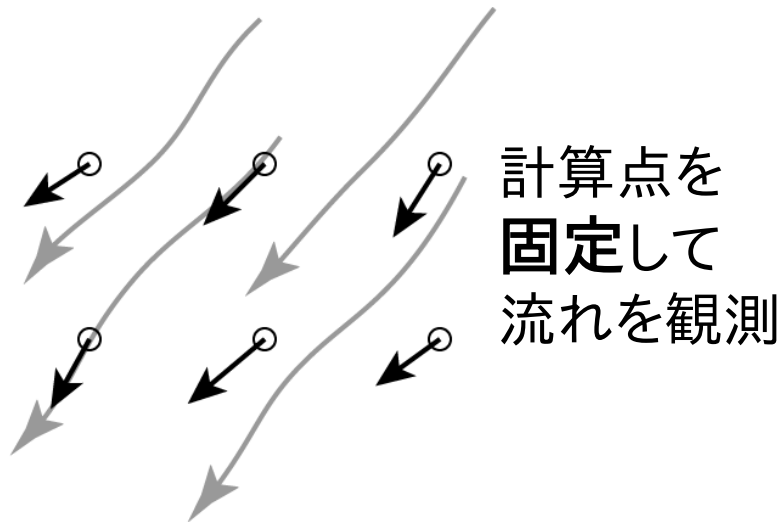
Appendix

(以降のページは補足資料です)

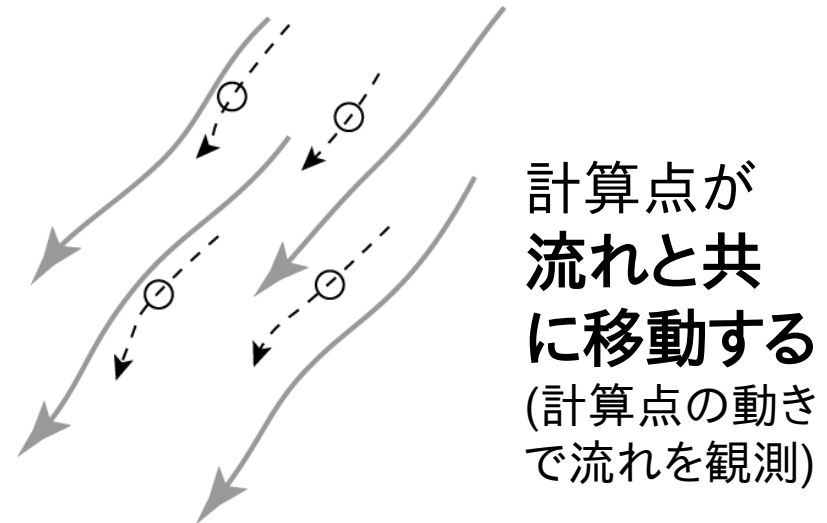
オイラー/ラグランジュ的手法

移流方程式のような流れに関する方程式の離散化方法は計算点の考え方で大きく2種類に分けられる

オイラー的アプローチ



ラグランジュ的アプローチ



格子分割する方法は基本的にオイラー的な方法なんだけど、移流元をバックトレースするのは計算点を移動するラグランジュ的。そのためセミ・ラグランジュ法という

特性曲線

移流方程式 $\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = 0$ において, この式が常に満たされているような関数 f を具体的に求めてみると,

$$f(x, t) = x - ut$$

としたときに

$$\frac{\partial(x - ut)}{\partial t} + u \frac{\partial(x - ut)}{\partial x} = -u + u = 0$$

となって微分方程式を常に満たす.
一方で定数を微分しても0になるのだから,
微分方程式を満たすということは $f(x, t)$ の
値が x, t によらず一定値になっているといえる.

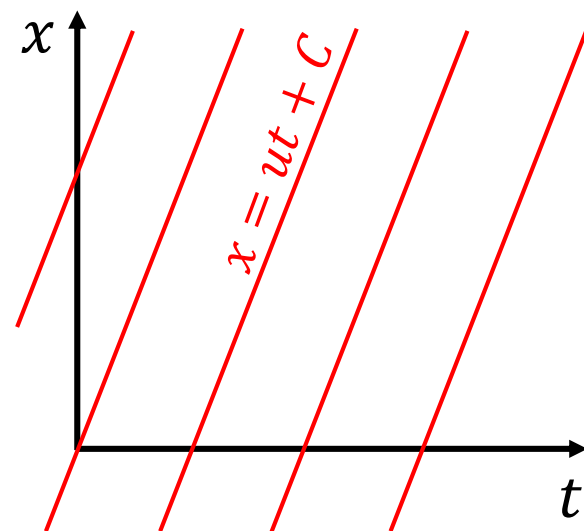


$$f(x, t) = x - ut = C \quad \Rightarrow \quad \underline{x = ut + C}$$

C : 定数

微分方程式を

常に満たす線 \Rightarrow 特性曲線



拡散方程式

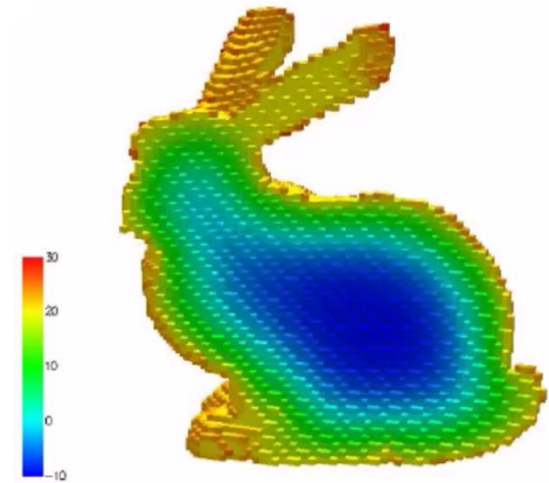
拡散方程式の具体的な例:

熱伝導方程式

物体内部での熱の伝わりをモデル化したもの(熱力学,伝熱工学)

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

T : 温度
 κ : 熱伝導係数



磁気拡散方程式

抵抗によって磁場が拡散・減衰する減少をモデル化したもの(磁気流体力学:MHD)

$$\frac{\partial \mathbf{B}}{\partial t} = \eta \nabla^2 \mathbf{B}$$

\mathbf{B} : 磁場
 η : 磁気拡散率