数値計算

京都大学大学院情報学研究科
大木 健太郎
連絡先： ohki@i.kyoto-u.ac.jp

第7回
- 非線形方程式の数値解法
- 代数方程式の数値解法（Strum の方法）

本日の内容

- 非線形方程式の数値解法
  - Newton法
  - 反復法と収束
  - 反復法の誤差解析
  - 加速法
- 代数方程式の数値解法
  - Sturm の方法

最適化問題と代数方程式のゼロ点

- 理学・工学や実社会において，数理最適化はよく用いられる.
  - （例）適当な制約条件の下で，下の評価関数を最小化

$$
\begin{aligned}
S(x) &= \frac{1}{2}(x - x_*)^\top A(x - x_*) \\
J(x) &= \exp(\|x - x_*\|_2^2) - x^\top A x
\end{aligned}
$$

⇒ 最適性の必要条件： $\frac{\partial}{\partial x}S(x) = 0,\ \frac{\partial}{\partial x}J(x) = 0$

極値を精度よく求めることは，極めて重要

非線形方程式の数値解法

- 先ほどの極値を求める問題は,
  $f(x) = 0$ を満たす $x \in \mathbb{R}^n$ を求める問題

- 代表的な方法として, Newton法
  - 線形近似して解を逐次更新していく

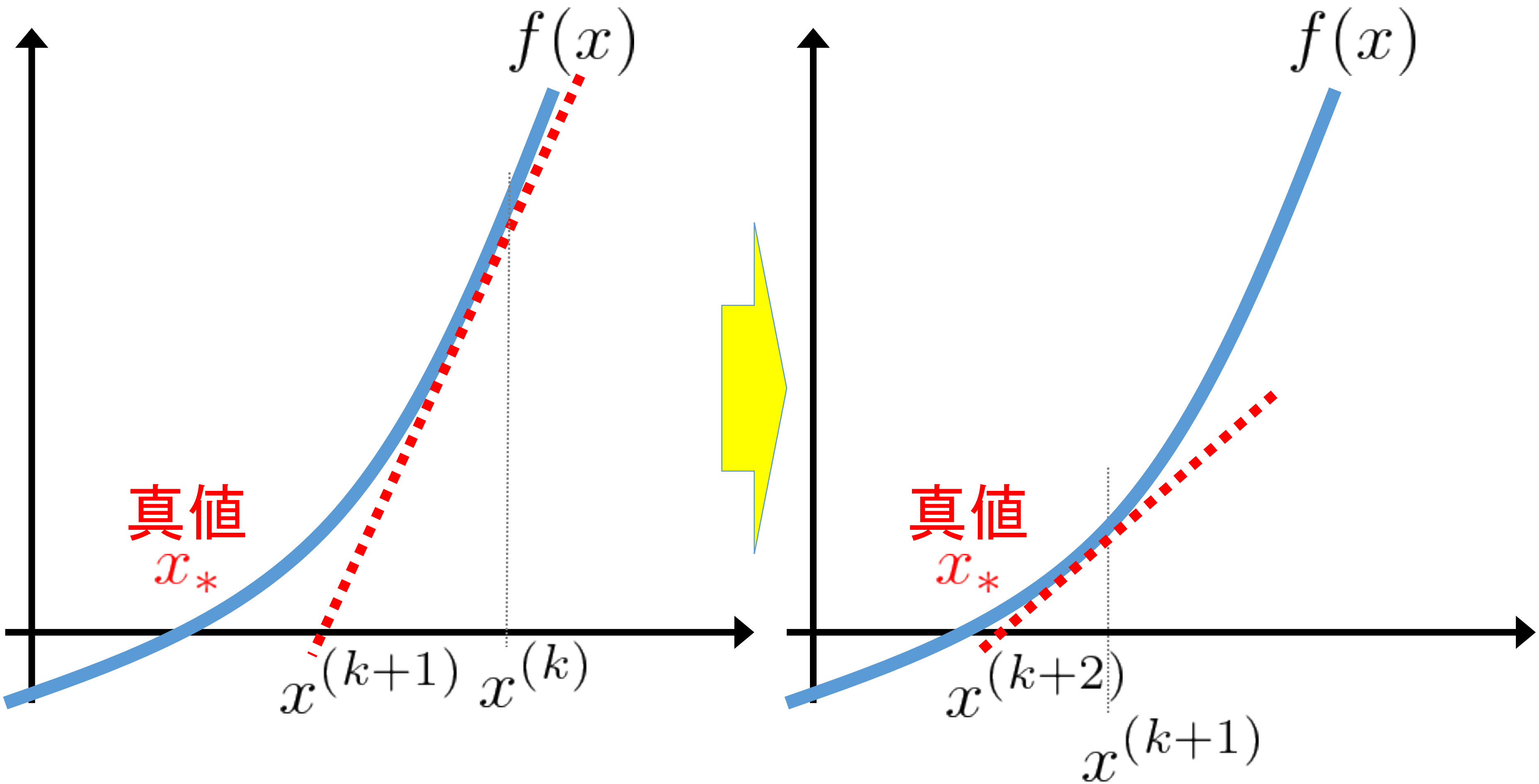
（例）1次元の場合

$$f(x) \simeq f(x^{(k)}) + f'(x^{(k)})(x^{(k+1)} - x^{(k)}) = 0$$

$$\Rightarrow \quad x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}$$

Newton法の幾何学的解釈

非線形の問題 ⇒ 線形の問題にして徐々に近づける



Newton法で扱える問題の例

- 平方根を求める問題: $f(x) = x^2 - a$
- n乗根を求める問題: $f(x) = x^n - a$
- 指数を求める問題: $f(x) = \log x - a$

n 変数方程式のNewton法

- n 変数だと n次元:

$$f(x) := \frac{\partial}{\partial x} S(x) \in \mathbb{R}^n \qquad f : \mathbb{R}^n \to \mathbb{R}^n$$

Taylor 展開

$$f(x) \simeq f(x^{(k)}) + J(x^{(k)})(x - x^{(k)}) = 0$$

$$J(x) := \left(\frac{\partial f_i(x)}{\partial x_j} \right)_{ij} = \frac{\partial^2}{\partial x \partial x^\top} S(x) \qquad \det J(x) \neq 0, \quad \forall x \in \mathbb{R}^n \setminus \{x_*\}$$

⇒ $$x^{(k+1)} = x^{(k)} - J^{-1}(x^{(k)}) f(x^{(k)})$$

Newton法の収束速度

- 前段の誤差の二乗に比例して小さくなる

$$x^{(k+1)} - x_* = J^{-1}(x^{(k)})O(\|x^{(k)} - x_*\|^2)$$

- J の正則性が成り立てば，二次収束が保証されるので，古くからよく使われている

反復法としてのNewton法

$$\varphi(x) = \frac{1}{f'(x)}$$ として，Newton 法は次の形で表せる

$$x^{(k+1)} = \phi(x^{(k)}), \quad \phi(x) = x - \underline{\varphi(x) f(x)}$$

なぜこの形か？

⇒ $x = \phi(x)$ ならば，$x = x - \varphi(x)\underbrace{f(x)}_{=0}$

解が収束すれば，方程式の根である

※ $\varphi(x) \neq 0$ は保証する必要がある

他の反復法による数値解法

$$x^{(k+1)} = \phi(x^{(k)}), \quad \phi(x) = x - \varphi(x) f(x)$$

- Newton 法 $\varphi(x) = \dfrac{1}{f'(x)}$

  各ステップで線形近似

- 線形逆補間法 $\varphi(x) = \dfrac{x - x_0}{f(x) - f(x_0)}$

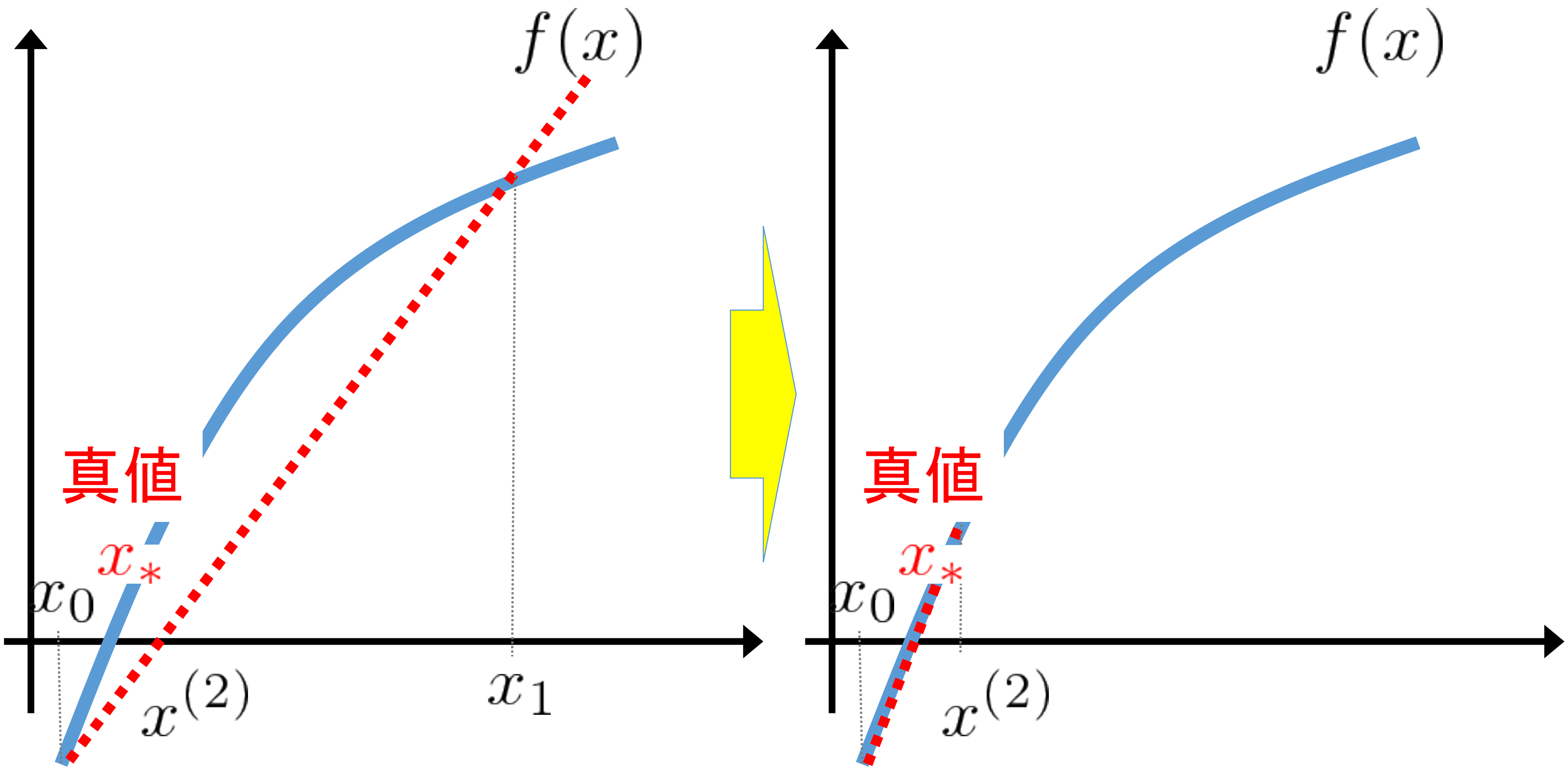
  初期値のみでなく第一ステップの値も用いる

- von Mises 法 $\varphi(x) = \dfrac{1}{f'(x_0)}$

  初期値の傾きをずっと使う

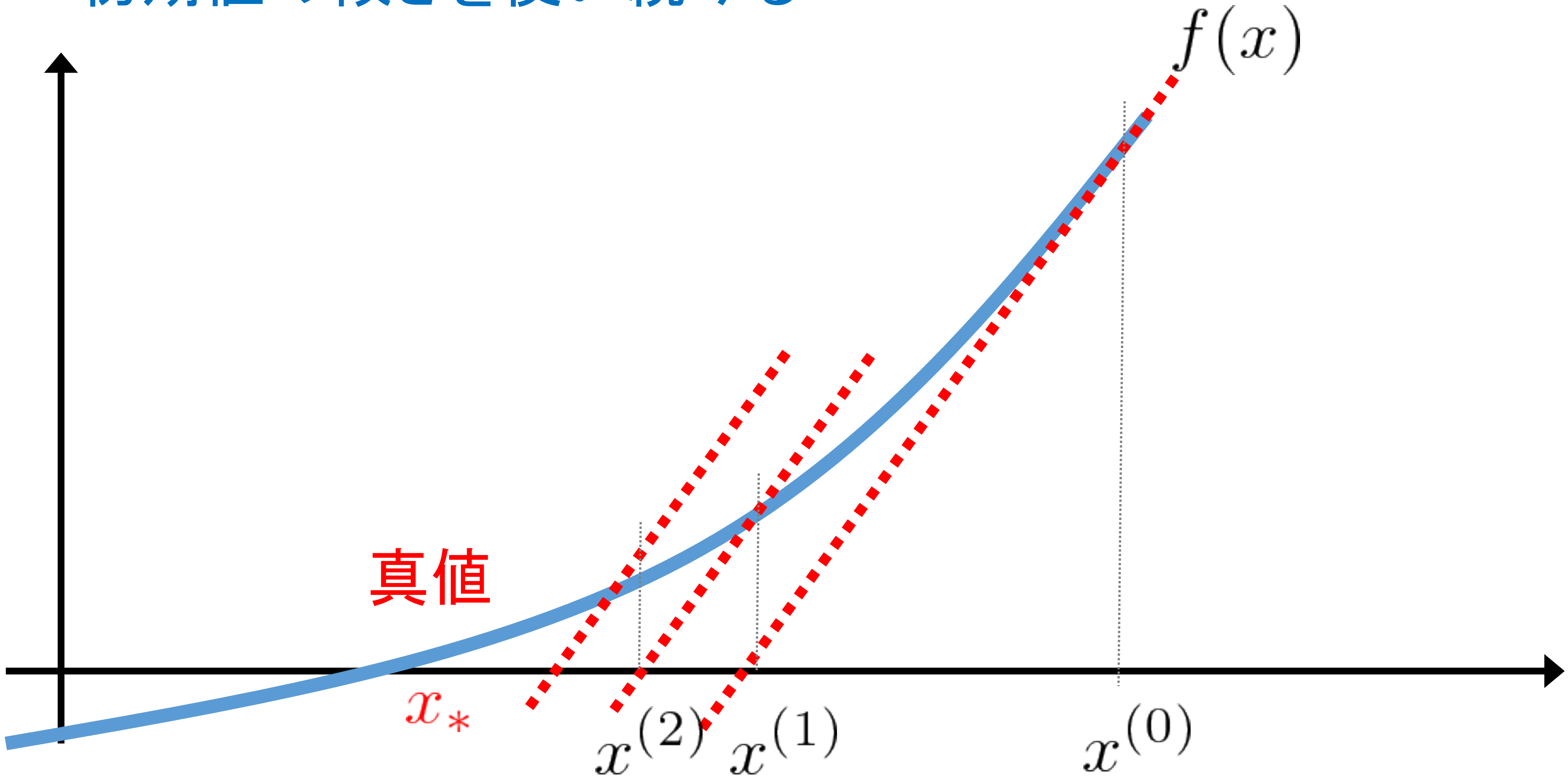
線形逆補間法の幾何学的解釈

$f(x_0)f(x_1) < 0$ となるペア $x_0, x_1 \in \mathbb{R}$ を見つけておく



von Mises 法の幾何学的解釈

初期値の傾きを使い続ける



本日の内容

- 非線形方程式の数値解法
  - Newton法
  - 反復法と収束
  - 反復法の誤差解析
  - 加速法
- 代数方程式の数値解法
  - Sturm の方法

反復法の収束

- 反復法によって生成される数列が,
  求めたい解に近づかなければ意味がない

$$x^{(k+1)} = \phi(x^{(k)})$$

- 収束の議論は**縮小写像の原理**

縮小写像の原理(復習)

仮定

$$\forall x, y \in \mathbb{R}^n, \ \|\phi(x) - \phi(y)\| \leq q\|x - y\|$$

かつ, q は $0 < q < 1$

⇒ $$x^{(k+1)} = \phi(x^{(k)})$$ で更新される解は

$$x = \phi(x)$$ を満たす唯一解に収束

$\mathbb{R}^n$ 全体で考えると, 多くの写像が縮小写像にならない

定義域を制限した縮小写像

- 非線形方程式の場合,
  写像が縮小写像か否かは定義域にも依存する

$$x^{(k+1)} = \phi(x^{(k)}) \qquad \phi(x) = x^2$$

$$x, y \in K = [0, 0.3], \ |\phi(x) - \phi(y)| \le q|x - y|$$

⇒ ϕ は $K = [0, 0.3]$ 上で縮小写像

縮小写像の原理(定理2.1)

仮定

- $K \subset \mathbb{R}^n$ は閉領域かつ $\phi : K \to K$
- $\forall x, y \in K \subset \mathbb{R}^n,\ \|\phi(x) - \phi(y)\| \leq q\|x - y\|$

  かつ，q は $0 < q < 1$

$$x^{(k+1)} = \phi(x^{(k)}), \quad x^{(0)} \in K$$

で更新される解は

$x = \phi(x)$ を満たす唯一解に収束

本日の内容

- 非線形方程式の数値解法
  - Newton法
  - 反復法と収束
  - 反復法の誤差解析
  - 加速法
- 代数方程式の数値解法
  - Sturm の方法

反復法の誤差解析

丸め誤差なし $x^{(k+1)} = \phi(x^{(k)})$

丸め誤差あり $y^{(k+1)} = \phi(y^{(k)}) + \epsilon^{(k)}$

丸め誤差

仮定 $\|\epsilon^{(k)}\| \leq \epsilon, \ k = 1, 2, \cdots$

$$\|y^{(N)} - x_*\| \leq \frac{1}{1-q}\left\{q^N \|\underline{\phi(x^{(0)})} - x^{(0)}\| + \epsilon\right\}$$

教科書の評価

これは分からない・・・

反復法の誤差解析：修正版

$$y^{(1)} - x^{(0)} = \phi(x^{(0)}) - x^{(0)} + \epsilon^{(0)}$$

$$\Rightarrow y^{(1)} - x^{(0)} - \epsilon^{(0)} = \phi(x^{(0)}) - x^{(0)}$$

$$\|\phi(x^{(0)}) - x^{(0)}\| \leq \|y^{(1)} - x^{(0)}\| + \epsilon$$

$$\|y^{(N)} - x_*\| \leq \frac{1}{1-q}\left\{q^N \left(\|y^{(1)} - x^{(0)}\| + \epsilon\right) + \epsilon\right\}$$

初期値，事前情報（丸め誤差の大きさ），第一ステップの値のみで評価可能

初期値の選択

- 反復法の収束は，適切な初期値の選択が重要

$$T_1 = \{y \mid \|y - x^{(0)}\| \le \rho\}$$

仮定 $\quad \|\phi(x^{(0)}) - x^{(0)}\| + \epsilon \le (1-q)\rho$

丸め誤差の大きさ

$q^N \le 1$ より，

$$\Rightarrow \quad \|y^{(N)} - x_*\| \le \frac{1}{1-q}\left\{\|\phi(x^{(0)}) - x^{(0)}\| + \epsilon\right\} \le \rho$$

初期値の選択:修正版

$$T_1 = \{y \mid \|y - x^{(0)}\| \leq \rho\}$$

$\boxed{\text{仮定}}$ $\|y^{(1)} - x^{(0)}\| + 2\epsilon \leq (1-q)\rho$

$$\Rightarrow \quad \|y^{(N)} - x_*\| \leq \frac{1}{1-q}\left\{\|y^{(1)} - x^{(0)}\| + 2\epsilon\right\} \leq \rho$$

よって, $y^{(N)} \in T_1,\ N = 1, 2, \cdots$

何らかの先見的情報を用いて
初期値を真値の十分近くにとる工夫が大事

本日の内容

- 非線形方程式の数値解法
  - Newton法
  - 反復法と収束
  - 反復法の誤差解析
  - 加速法
- 代数方程式の数値解法
  - Sturm の方法

収束の速度

反復法 $x^{(k+1)} = \phi(x^{(k)}),\ x^{(0)} \in \underline{K} \subset \mathbb{R}^n$

有界閉集合

仮定 ϕ は $(K, \|\bullet\|)$ 上で縮小写像

➡ 収束先を x^* とする

定義：収束の速度

- $\displaystyle \lim_{k\to\infty} \frac{\|x^{(k+1)} - x^*\|}{\|x^{(k)} - x^*\|} = \mu$ ➡ 線形収束，1次収束
- $\displaystyle \lim_{k\to\infty} \frac{\|x^{(k+1)} - x^*\|}{\|x^{(k)} - x^*\|} = 0$ ➡ 超1次収束
- $\displaystyle \lim_{k\to\infty} \frac{\|x^{(k+1)} - x^*\|}{\|x^{(k)} - x^*\|^p} > 0, \quad p > 1$ ➡ p 次収束

Newton 法は真値の近傍で2次収束

Aitken の加速法

少ないステップで収束した方が嬉しい

⇒ **加速法**：より速い収束列を作る方法

簡単のために $n=1$ とする

$\{x^{(k)}\}$ ：収束列（収束先：x^*，線形収束）

$$y^{(k)} = x^{(k)} - \frac{(x^{(k+2)} - x^{(k+1)})^2}{x^{(k+2)} - 2x^{(k+1)} + x^{(k)}}$$

で生成する列 $\{y^{(k)}\}$ は元の収束列よりも速くに x^* 収束する：

$$\lim_{k\to\infty} \frac{y^{(k)} - x^*}{x^{(k+2)} - x^*} = 0$$

※ 証明は [山本哲朗：数値解析入門，p. 74，サイエンス社（2003）] 参照

関数の微分を必要としないアルゴリズム

関数 f が微分できない場合，Newton 法は使えない

⇒ Steffensen 反復法： $x = \phi(x)$ の根を求めるアルゴリズム

$$\phi(x) = x + f(x)$$ として

$$z^{(k+1)} = \phi(\phi(z^{(k)})) - \frac{(\phi(\phi(z^{(k)})) - \phi(z^{(k)}))^2}{\phi(\phi(z^{(k)})) - 2\phi(z^{(k)}) + z^{(k)}}$$

- ✓ Aitken 加速と似ているが，Aitken 加速が発散しても初期値の選び方によって収束
- ✓ 収束速度はNewton法と同程度

本日の内容

- 非線形方程式の数値解法
  - Newton 法
  - 反復法と収束
  - 反復法の誤差解析
  - 加速法
- 代数方程式の数値解法
  - Sturm の方法

代数方程式

- 代数方程式
  - 多項式：（例）　$$f(x) = \sum_{j=0}^{m} a_j x^j$$
  - 代数方程式：　$$f(x) = 0$$
- 根の存在が保証され，根の数も保証される
- 非線形関数の近似して議論できる

代数方程式の根に関する数値計算法

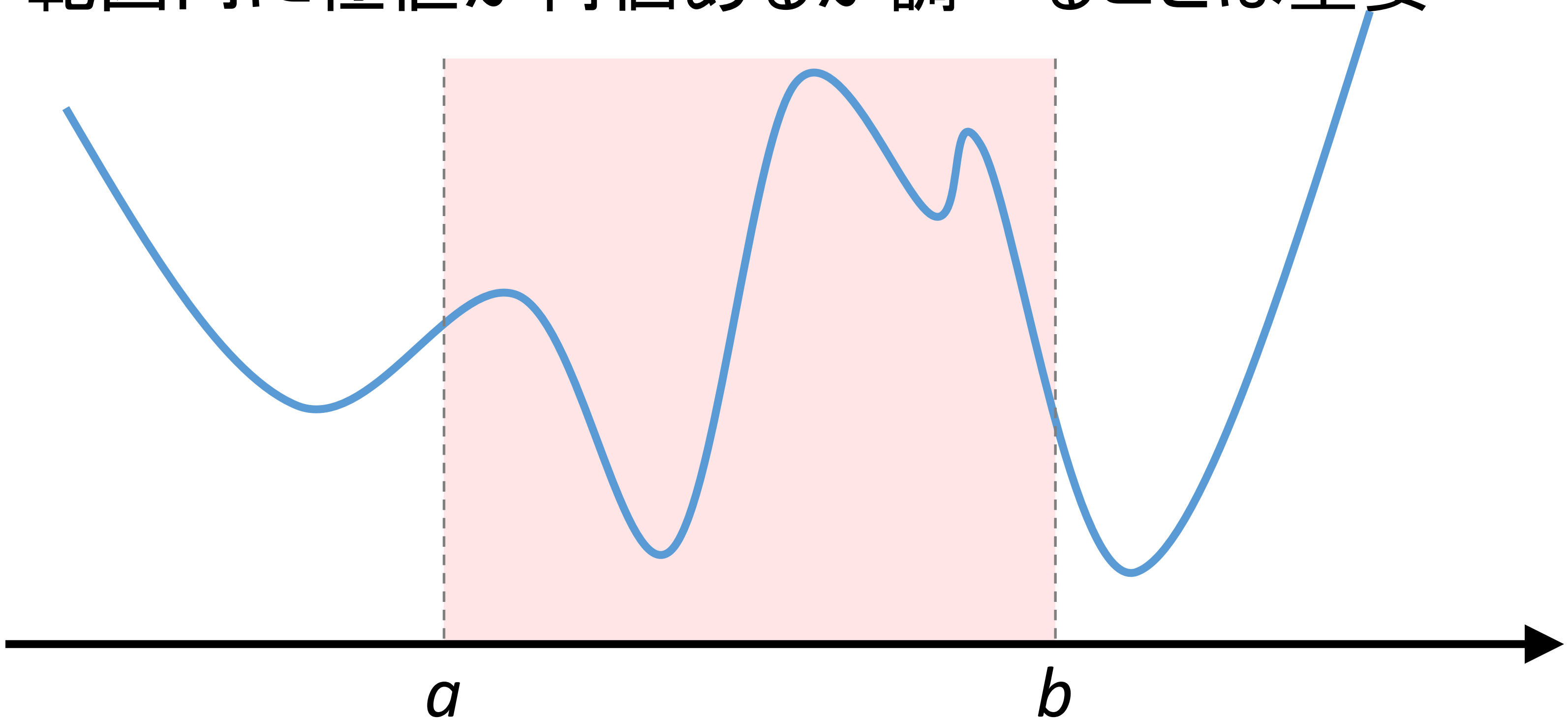
- ある指定した範囲に根がいくつ存在するか
  $\Rightarrow$ Strum の方法

- 複素数の根を求める
  $\Rightarrow$ Bairstow-Hitchcock 法

- 全ての根を求める
  $\Rightarrow$ DKA 法 (Durand-Kerner-Aberth法)

根の数を調べること

- 最適化の問題では，パラメータを動かせる範囲が制限されることが多い
- 範囲内に極値が何個あるか調べることは重要



Sturm列

$[a, b], \quad a, b \in \mathbb{R}, a < b$

定義（Sturm 列）

実係数多項式の列 $f(x),\ f_1(x),\ \cdots,\ f_l(x)$ が区間[*a*, *b*]において**Sturm 列**であるとは，次の4つを満たすことをいう.

(1) $x \in [a, b],\ f_k(x) = 0 \Rightarrow f_l(x) \neq 0,\ k, l = 0, 1, \cdots, l-1,\ |k - l| = 1$

(2) $f_k(x_0) = 0 \Rightarrow f_{k-1}(x_0) f_{k+1}(x_0) < 0,\ k = 1, \cdots, l-1$

(3) $(f_l(x) > 0,\ \forall x \in [a, b])$ or $(f_l(x) < 0,\ \forall x \in [a, b])$

(4) $f(x_0) = 0 \Rightarrow f(x_0) f_1(x_0) > 0$

Sturm列の例

次はSturm 列の定義を満たす

$$f(x) = x^2 + a_1 x + a_0, \quad a_0, a_1 \in \mathbb{R} \setminus \{0\}, a_1 \neq 2a_0$$

$$f_1(x) = 2x + a_1$$

$$f_2(x) = a_0 - \frac{1}{4} a_1^2$$

※ f は重複する根を持たないことに注意

Sturmの定理

仮定

実係数多項式の列 $f(x),\ f_1(x),\ \cdots,\ f_l(x)$ が区間[*a*, *b*]においてSturm 列であり, $f(a)f(b) \neq 0$.

$N(x)$: $x \in [a, b]$ において $[f(x), f_1(x), \cdots f_l(x)]$ を左から見たときの符号反転の数

定理2.2 (Sturmの定理)

$f(x)$ の区間 [*a*, *b*] に存在する零点の個数は,

$$n_0 = N(a) - N(b)$$

符号反転回数の数え方の注意

- 次の場合は，一回反転したものと数える

$f_{k-1}(x)$	$f_k(x)$	$f_{k+1}(x)$
$+$	0	$-$

Sturm列の生成法

1. 多項式 f は区間 $[a, b]$ で重根を持たないものとする
   （Sturm 列の条件(1)を満たすため）
2. まず微分し，1つ目の多項式を作る
3. f を f_1 で割り算し，剰余のマイナスを f_2 とする.
4. 以降，f_k を f_{k-1} で割り算し，剰余のマイナスを f_{k+1} とする.
5. $f_l(x)$ = 非ゼロの定数になったら終了

$$
\begin{aligned}
f(x) &= x^2 + a_1 x + a_0, \quad a_0, a_1 \in \mathbb{R} \setminus \{0\},\ a_1 \neq 2a_0 \\
f_1(x) &= f'(x) = 2x + a_1 \\
f_2(x) &= \underbrace{\left(\frac{1}{2}x + \frac{1}{4}a_1\right)}_{g_1(x)} f_1(x) - f(x) = -a_0 + \frac{1}{4}a_1^2
\end{aligned}
$$

次数を下げるようにする

重根のある場合のSturm列

$$f(x) = (x+1)^2(x-5) = x^3 - 3x^2 - 9x - 5$$

のSturm 列を計算（重根の場合）

Scilabコード例

```
x = poly(0,'x');
f = x^3 – 3 x^2 – 9 x -5;
f1 = 3 x^2 – 6 x – 9;
[f2, g1] = pdiv(f, f1);
f2=-f2;
[f3, g2] = pdiv(f1, f2);
```

$$f_2(x) = \frac{1}{8}(x+1) \neq \text{定数}$$

$$g_2(x) = -\frac{9}{8}x + \frac{3}{8}$$

$$f_3(x) = 0$$

重根のある場合のSturm列

$$\tilde{f}(x) = \frac{f(x)}{f_2(x)} = 8(x+1)(x-5) = 8x^2 - 32x - 40$$

のSturm 列を計算（重根の場合）

Scilabコード例

```
x = poly(0,'x');
f = 8x^2 – 32 x – 40;
f1 = 16 x – 32;
[f2, g1] = pdiv(f, f1);
f2=-f2;
```

Sturm列

$$\tilde{f}(x) = 8x^2 - 32x - 40$$

$$\tilde{f}_1(x) = 16x - 32$$

$$\tilde{f}_2(x) = 72$$

Sturmの方法

- Sturm の定理を使って,
- 区間 $[a, b]$ を与えて，根の数を調べる
- 区間 $[a, (a+b)/2]$, $[(a+b)/2, b]$ の根の数を調べる
- どんどん2分割していき，根が一つしかない区間を十分小さくして，その区間内の適当な値を根の近似値として採用する

Sturmの方法：具体例

- $f(x) = x^2 - 1x - 2 = 0$ を求める（区間[-10,10]）
- Sturm 列

$$f(x) = x^2 - 1x - 2$$
$$f_1(x) = 2x - 1$$
$$f_2(x) = 2 + 0.25 = 2.25$$

- 区間[-10, 10] 内の根の数

$$\left[f(10),\quad f_1(10),\quad f_2(10)\right] = \left[88,\quad 19\quad 2.25\right] \Rightarrow N(10) = 0$$
$$\left[f(-10),\quad f_1(-10),\quad f_2(-10)\right] = \left[108,\quad -21\quad 2.25\right] \Rightarrow N(-10) = 2$$

$$\Rightarrow \quad n_{[-10,10]} = N(-10) - N(10) = 2$$

Sturmの方法：具体例

- 区間を2分割して，根の数を計算
  - 区間 [-10,0] の根の数
  - 区間 [0,10] の根の数
- 再び区間を2分割して，根の数の計算を繰り返す

$$n_{[-10,0]} = 1$$

$$n_{[0,10]} = 1$$

$$\left[f(10),\quad f_1(10),\quad f_2(10)\right] = \left[88,\quad 19 \quad 2.25\right] \;\Rightarrow\; N(10) = 0$$

$$\left[f(0),\quad f_1(0),\quad f_2(0)\right] = \left[-2,\quad -1 \quad 2.25\right] \;\Rightarrow\; N(0) = 1$$

$$\left[f(-10),\quad f_1(-10),\quad f_2(-10)\right] = \left[108,\quad -21 \quad 2.25\right] \;\Rightarrow\; N(-10) = 2$$

次回の講義予定

- 代数方程式の数値解法