



数值分析(2)

Numerical Analysis

计算机系 软件所 喻文健



本课件更适合于用电脑观看
(手机用于雨课堂答题)

现场的同学务必佩戴口罩，
不要看雨课堂视频

建议上课时将**纸**、**笔**放在旁边!

第二章 非线性方程求根

- 引言
- 二分法
- 不动点迭代法
- 牛顿法
- 牛顿法的改进
- 有关的实用技术

非线性方程基本理论

■ 何谓非线性方程?

$$f(x) = 0 \quad (\text{默认实数域上})$$

■ 一般地，解的存在性和个数很难确定

(1) $e^x + 1 = 0$. 此方程无实数解.

(2) $e^{-x} - x = 0$. 此方程有一个解.

(3) $\cos x = 0$. 此方程有无穷多个解.

(4) $x^3 - 6x^2 + 5x = 0$. 此方程有三个解.

■ 特例是n次多项式方程

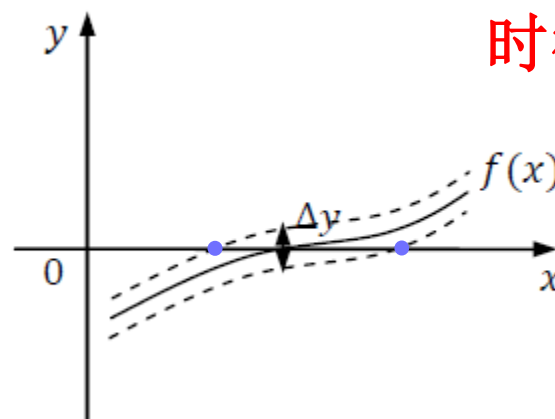
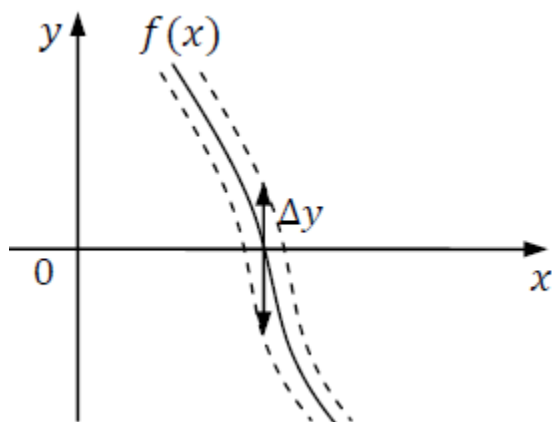
■ 实际问题中，一般求区间 $[a, b]$ 上的实数解

非线性方程基本理论

$$f^{(m)}(x^*) \neq 0, m > 1$$

- **m重根** x^* : $f(x^*) = f'(x^*) = \dots = f^{(m-1)}(x^*) = 0$
- 问题的敏感性: 输入数据扰动对解的误差的影响
- 一种易于分析的情况: 设 $f(x) = y$, 求 $y = 0$ 对应的 x
绝对条件数 $\left| \frac{\Delta x}{\Delta y} \right| \approx \left| \frac{1}{f'(x^*)} \right|$
← 解 ← 输入
- $|f'(x^*)|$ 越小, 问题越敏感

当 x^* 是重根
时很敏感



二分法

二分法

(连续函数)

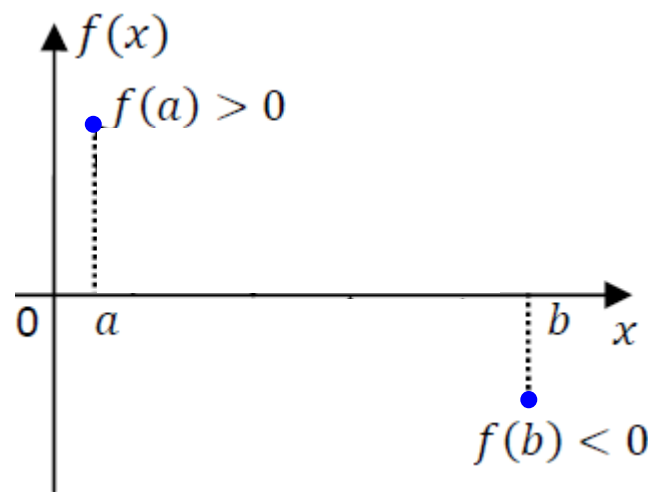
- **定理2.1**: 若 $f(x) \in C[a, b]$, 且 $f(a)f(b) < 0$, 则区间 (a, b) 内至少有一实根

- **有根区间**, 可以逐渐让它缩小

- **二分法**: 逐次将有根区间一分为二, 得到区间序列 $\{(a_k, b_k)\}$, 近似解 $x_k = (a_k + b_k)/2$. 并且

$$|x_k - x^*| < (b_k - a_k)/2 = (b_0 - a_0)/2^{k+1}, k = 0, 1, \dots$$

- 可估计要达到某个误差限所需的二分次数, 也是计算函数值的次数



二分法

算法2.1: 二分法

输入: a, b , 函数 $f(x)$; 输出: x .

While $(b - a) > \varepsilon$ **do**

$x := a + (b - a)/2$;

If $\text{sign}(f(x)) = \text{sign}(f(a))$ **then**

$a := x$;

Else

$b := x$;

End

End

$x := a + (b - a)/2$.

或者 $\varepsilon \cdot |b|$

$\text{sign}()$ 表示取符号的函数。这里忽略了 $f(x)$ 或 $f(a) = 0$ 的情况（若成立, 直接退出）

- 算法稳定性: 运算简单, 误差逐渐缩小, 比较稳定

二分法

算法2.1中误差
阈值 ε 不能太小了

- 浮点运算下, 二分法的结果准确度的极限
- **例:**求解方程 $f(x) = x^2 - 2 = 0$, 初始区间为 $[1, 2]$.
- 程序运行, 迭代**52**次后, 区间不再缩小
- 打印出的最后几个区间端点值

a = 3ff6a09e667f3bc8

(‘>>format hex’按16进制显示)

a = 3ff6a09e667f3bcc

b = 3ff6a09e667f3bce

b = 3ff6a09e667f3bcd

相邻的两个浮点数

$\rightarrow = \lfloor \log_2 |x^*| \rfloor$

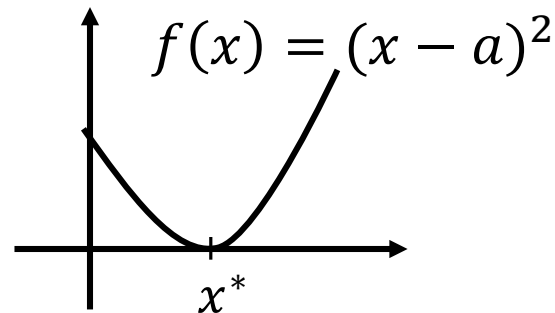
定理2.2

- 最小的有根区间长度为: $2^E \cdot 2^{\varepsilon_{\text{mach}}}$, E 为准确解 x^* 的浮点数**指数** 双精度下, 解的误差限最小为 2^{E-52}

$$|e_r| \leq 2^{\varepsilon_{\text{mach}}}$$

二分法的总结

- 求单变量方程 $f(x) = 0$ 的实根的可靠算法，它逐次缩小解的范围，总能收敛。
- 在实际浮点数系统上，解的误差有个下限。
- 缺点：收敛速度较慢、无法求偶数重的重根；初始有根区间有时不方便确定、可能从多个根中随机得一个
- 常将二分法与其他方法结合起来使用





不动点迭代法

不动点迭代

$$f(x) = 0 \iff x = g(x)$$

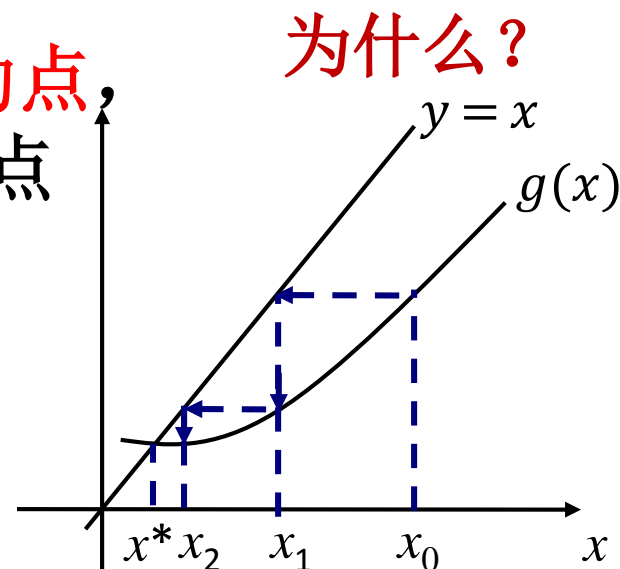
推出迭代法: $\begin{cases} x_{k+1} = g(x_k), & (k = 0, 1, \dots) \\ \text{给定 } x_0 \end{cases}$

■ 好处: 若 $\{x_k\}$ 序列收敛到 x^* , 则 x^* 是原方程的解

■ 满足 $x = g(x)$ 的 x 称为 $g(x)$ 的**不动点**,
几何意义是 $y = g(x)$ 与 $y = x$ 的交点

■ 上述求解 $f(x) = 0$ 的方法称为
不动点迭代法

计算过程的几何含义 \rightarrow



不动点迭代

$$f(x) = 0 \longleftrightarrow x = g(x) \longrightarrow x_{k+1} = g(x_k)$$

■ **例2.4:** 求 $x^4 - x - 2 = 0$ 在 $x_0 = 1.5$ 附近的根

■ 一种变形(A): $x = \sqrt[4]{x + 2} \longrightarrow \begin{cases} x_{k+1} = \sqrt[4]{x_k + 2} \\ x_0 = 1.5 \end{cases}$

$$x_1 = 1.3678, x_2 = 1.3547, \dots, x_4 = 1.3532, x_5 = 1.3532$$

■ 还有其他变形吗? (5位有效数字不变)

■ (B): $x = x^4 - 2 \longrightarrow \begin{cases} x_{k+1} = x_k^4 - 2 \\ x_0 = 1.5 \end{cases}$ (计算量较小)

$$x_1 = 1.5^4 - 2 = 3.1, x_2 = x_1^4 - 2 \approx 85, \dots, \text{越来越大, 发散了!}$$

关键问题: 如何判断收敛性?

全局收敛的充分条件

■ **定理2.3** $g(x) \in C[a, b]$, 若

(1) 对 $\forall x \in [a, b]$, 有 $a \leq g(x) \leq b$

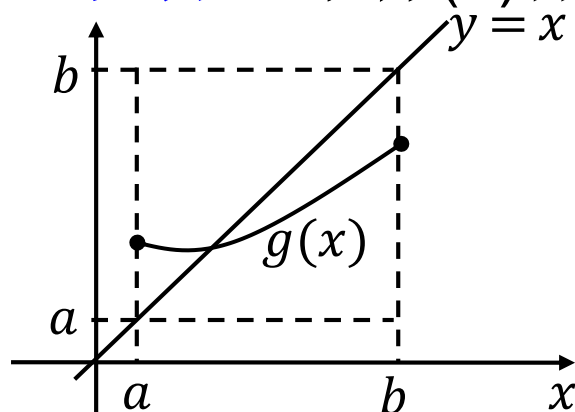
(2) $\exists L \in (0, 1)$, $\forall x_1, x_2 \in [a, b]$ 有 $|g(x_1) - g(x_2)| \leq L|x_1 - x_2|$

可改为 $< |x_1 - x_2|$

则 $g(x)$ 在 $[a, b]$ 上存在不动点, 且唯一

■ 详细证明请自己看书 pp. 37~38

■ **说明:** 条件(1)保证了存在不动点, 可用图形解释



$g(x)$ 曲线在虚线正方形内, 水平方向横跨 $[a, b]$, 则必与对角线相交, 这是连续函数的性质

唯一性的证明用反证法

全局收敛的充分条件 $f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$

■ **定理2.4 (充分条件)** 若 $g(x)$ 满足定理2.3中的两个条件

(1) 对 $\forall x \in [a, b]$, 有 $a \leq g(x) \leq b$

(2) $\exists L \in (0, 1)$, $\forall x_1, x_2 \in [a, b]$ 有 $|g(x_1) - g(x_2)| \leq L|x_1 - x_2|$

则对 $\forall x_0 \in [a, b]$, 不动点迭代法 $x_{k+1} = g(x_k)$ **都收敛**到

不动点 x^* , 且 $|x_k - x^*| \leq \frac{L^k}{1-L} |x_1 - x_0|$

■ **证明:** 考察误差序列 (条件(1)是算法执行的前提)

$$|x_k - x^*| = |g(x_{k-1}) - g(x^*)| \leq L|x_{k-1} - x^*| \leq \cdots \leq L^k |x_0 - x^*|$$

$$\text{则 } \lim_{k \rightarrow \infty} L^k |x_0 - x^*| = 0 \quad \longrightarrow \quad \lim_{k \rightarrow \infty} x_k = x^*$$

后面的不等式实际意义不大

全局收敛的充分条件 $f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$

■ **定理2.4 (充分条件)** 若 $g(x)$ 满足两个条件

(1) 对 $\forall x \in [a, b]$, 有 $a \leq g(x) \leq b$

(2) $\exists L \in (0, 1)$, $\forall x_1, x_2 \in [a, b]$ 有 $|g(x_1) - g(x_2)| \leq L|x_1 - x_2|$

则对 $\forall x_0 \in [a, b]$, 不动点迭代法 $x_{k+1} = g(x_k)$ 都收敛, ...

■ **说明:** 可用于判断不动点迭代法的**全局收敛**, 但条件(2)不够直观, 常替换为 **(2)' $\forall x \in [a, b], |g'(x)| < 1$**

这就是**定理2.5** 为什么可以换? 或者 $|g'(x)| \leq L < 1$ 便于使用

$$|g(x_1) - g(x_2)| = |g'(\xi)(x_1 - x_2)|$$

$$\because |g'(x)| \leq L \longrightarrow |g(x_1) - g(x_2)| \leq L|x_1 - x_2|$$

$L = |g'(\bar{x})|$ 为最大值 (闭区间上的函数一定有)

全局收敛的充分条件 $f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$

■ 定理2.5 若 $g(x)$ 满足两个条件

- (1) 对 $\forall x \in [a, b]$, 有 $a \leq g(x) \leq b$ 则对 $\forall x_0 \in [a, b]$, 不动点迭代法收敛 ...
(2)' $\forall x \in [a, b], |g'(x)| < 1$

■ 例2.5: 用定理2.5分析上例中方法(A), (B)的收敛性, 在区间 $[1, 2]$ 上考察

(A) $g(x) = \sqrt[4]{x+2}$, $g(x)$ 值域 $[\sqrt[4]{3}, \sqrt[4]{4}] \subset [1, 2]$

又 $g'(x) = \frac{1/4}{\sqrt[4]{(x+2)^3}}$, 当 $x \in [1, 2]$ 时, 其值 < 1 , 满足条件(2)'!

(B) $\forall x \in [1, 2], g(x) = x^4 - 2, \in [-2, 14]$ 不满足条件(1), 无法说明是否收敛

■ 说明: 1. 全局, 充分

思考: 从图形角度, 举不满足条件

2. 全局收敛要求较高, 也较难判断 (1)和(2)但收敛的例子

局部收敛

$$f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$$

- **定义2.2** 若 $g(x)$ 有不动点 x^* , $\exists x^*$ 的邻域 $D: [x^* - \delta, x^* + \delta]$, 使 $\forall x_0 \in D$, 迭代法 $x_{k+1} = g(x_k)$ 都收敛到 x^* , 则称该方法局部收敛 (强调存在某个邻域)

- **定理2.6** 设 x^* 是 $g(x)$ 的不动点, 若 $|g'(x^*)| < 1$, 且 $g'(x)$ 在 x^* 的某邻域上连续, 则迭代法 $x_{k+1} = g(x_k)$ 局部收敛

- **证明:** 即证明 $\exists x^*$ 的邻域 D , 在它上迭代法全局收敛.

(利用定理2.5) 首先, $|g'(x^*)| < 1$ 且局部连续, 则存在 x^* 的某个邻域 D 使 $\forall x \in D, |g'(x)| \leq L < 1$, 这个 L 是介于 $g'(x^*)$ 与 1 之间的数. 因此满足条件(2)'.

又 $\forall x \in D, |g(x) - x^*| \leq L|x - x^*| < |x - x^*|, \therefore g(x) \in D$ 满足条件(1). 根据定理2.5, 收敛!

局部收敛

$$f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$$

■ **定理2.6** 设 x^* 是 $g(x)$ 的不动点, 若 $|g'(x^*)| < 1$, 且 $g'(x)$ 在 x^* 的某个邻域上连续, 则迭代法 $x_{k+1} = g(x_k)$ 局部收敛

■ **注意: 1.**只需看 $g'(x^*)$, 因此局部收敛易判断

2. $|g'(x^*)| < 1$ 是充分条件, 但也有一定的必要性.

因为若 $|g'(x^*)| > 1$, 则在 x^* 附近局部 $|g'(x)| > 1$.

$$|x_{k+1} - x^*| = |g(x_k) - x^*| = |g'(\xi)(x_k - x^*)| \overset{\text{可能}}{>} |x_k - x^*|$$

误差有放大的趋势

$|g'(x^*)| = 1$ 的例子: $x_{k+1} = b - x_k$ 不收敛

$x_{k+1} = \frac{x_k^3}{3} - x_k$ 收敛 (在0附近的区间上)

稳定性与收敛阶

$$f(x) = 0 \longrightarrow x_{k+1} = g(x_k)$$

- 与二分法类似, 由于每步迭代计算后都通过判停准则来估计解的准确度, 因此只要解收敛, 误差即越来越小, 计算过程是稳定的.

(合理设置判停准则,
稍后讲牛顿法时介绍)

只需关心**收敛性**

- 收敛阶 (评估收敛快慢)

- **例:** 误差 $|x_k - x^*|$ 按如下**3**种情形缩小, 哪种收敛更快?

情形**1**: $10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, \dots$

情形**2**: $10^{-1}, 10^{-3}, 10^{-5}, 10^{-7}, \dots$

情形**3**: $10^{-1}, 10^{-2}, 10^{-4}, 10^{-8}, \dots$

} 线性

平方(超线性)

稳定性与收敛阶

$$f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$$

- **定义2.3:** 迭代解序列 $\{x_0, x_1, \dots, x_k, \dots\}$ 收敛. 若误差

$$e(x_k) = x_k - x^* \text{ 满足 } \lim_{k \rightarrow \infty} \frac{|e(x_{k+1})|}{|e(x_k)|^p} = c, \quad (c \neq 0)$$

则称 p 阶收敛 (收敛阶为 p)

(不考虑舍入误差)

- **注意:** 对一个收敛的迭代法(过程), 上述数值 p 是唯一的
- **例:** 二分法的收敛阶? 大体上是线性(1阶)收敛, $c = 0.5$
- **定理2.7** 迭代法 $x_{k+1} = g(x_k)$, 若 $g^{(p)}(x)$ 在不动点 x^* 附近连续, 整数 $p \geq 2$, 则该方法在 x^* 的邻域上 p 阶收敛 \longleftrightarrow
 $g'(x^*) = g''(x^*) = \dots = g^{(p-1)}(x^*) = 0$, 且 $g^{(p)}(x^*) \neq 0$.
推论: $g'(x^*) = 0 \Leftrightarrow$ 至少2阶局部收敛
先证明充分性: 易知局部收敛, 然后看 $e(x_{k+1}) = ?$

稳定性与收敛阶

$$f(x) = 0 \Rightarrow x_{k+1} = g(x_k)$$

- **定理2.7** $x_{k+1} = g(x_k)$, 若 $g^{(p)}(x)$ 在不动点 x^* 附近连续, 整数 $p \geq 2$, 则该迭代法在 x^* 邻域上 p 阶收敛 \longleftrightarrow

$$g'(x^*) = g''(x^*) = \dots = g^{(p-1)}(x^*) = 0, \text{ 且 } g^{(p)}(x^*) \neq 0.$$

- **证明: 充分性.** $e(x_{k+1}) = g(x_k) - g(x^*)$

在 x^* 处做 Taylor 展开

$$e(x_{k+1}) = \frac{g^{(p)}(\xi_k)}{p!} (x_k - x^*)^p + \cancel{\frac{g'(x^*)}{1!} (x_k - x^*)} + \cancel{\frac{g''(x^*)}{2!} (x_k - x^*)^2} + \dots + \cancel{\frac{g^{(p-1)}(x^*)}{(p-1)!} (x_k - x^*)^{p-1}} + \frac{g^{(p)}(\xi_k)}{p!} (x_k - x^*)^p$$

$$\lim_{k \rightarrow \infty} \frac{|e(x_{k+1})|}{|e(x_k)|^p} = \lim_{k \rightarrow \infty} \frac{|g^{(p)}(\xi_k)|}{p!} = \frac{|g^{(p)}(x^*)|}{p!} \neq 0$$

矛盾!
方法 q 阶收敛.

必要性 用反证法证明: 设 x^* 处直到 φ 的 q 阶导数才 $\neq 0$, ($q \neq p$)

牛顿法

牛顿法

解 $f(x) = 0$



- 也叫**Newton-Raphson**方法

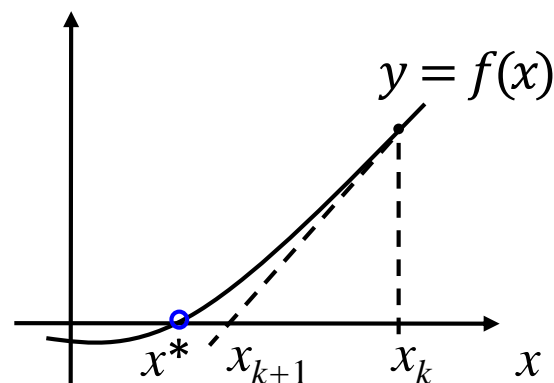
- 优点: 1.减少不动点迭代法构造的盲目性

- 2.较好的收敛性 (收敛阶)

- 原理: 用切线近似曲线 $f(x)$

切线方程 $P(x) = f(x_k) + (x - x_k)f'(x_k) = 0$

$$\rightarrow x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (\text{假设 } f'(x_k) \neq 0)$$



也是不动点迭代法

- 考察局部收敛性

$$g(x) = x - \frac{f(x)}{f'(x)} \quad \begin{aligned} &\rightarrow g'(x) = 1 - \frac{[f'(x)]^2 - f(x)f''(x)}{[f'(x)]^2} = \frac{f(x)f''(x)}{[f'(x)]^2} \\ &\text{设 } f'(x^*) \neq 0, \text{ 即单根} \quad \rightarrow g'(x^*) = 0 \end{aligned}$$

牛顿法

$$\text{解 } f(x) = 0, \quad x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

- 迭代函数 $g(x) = x - \frac{f(x)}{f'(x)}$, $g'(x^*) = 0 \Rightarrow$ 至少**2阶收敛**

又 $g''(x^*) = \frac{f''(x^*)}{f'(x^*)}$, 一般 $\neq 0$. (请自行推导) \therefore 为**2阶收敛**

- **定理2.8** 设 x^* 是方程 $f(x) = 0$ 的单根, 且 $f(x)$ 在 x^* 附近有连续的二阶导数, 则牛顿法局部二阶收敛.

\searrow $g(x)$ 的一阶导连续

- **例2.7** 求方程 $f(x) = x^4 - x - 2 = 0$ 在 1.5 附近的根
- 解: 牛顿法公式 $x_{k+1} =$ $k = 0, 1, 2 \dots$

将 $x_0 = 1.5$ 代入, 课本**2.4.1**小节的表2-2列出了计算结果
收敛速度比二分法、不动点迭代法都快. (4步, 5位精度)

牛顿法

$$\text{解 } f(x) = 0, \quad x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

- **例2.8** 求方程 $f(x) = x^2 - c = 0, (c > 0)$ 的正根
- 解: 牛顿法公式 $x_{k+1} = x_k - \frac{x_k^2 - c}{2x_k} = \frac{1}{2} \left(x_k + \frac{c}{x_k} \right), k = 0, 1, 2 \dots$

由于是单根, 此迭代法局部**2阶收敛**

事实上, 上述公式在 $(0, +\infty)$ 上全局收敛, 是计算 \sqrt{c} 稳定、有效的方法(计算机中使用).

- **说明**: 它不满足定理**2.4, 2.5**的全局收敛条件, 需从别的角度证明其全局收敛性 (请大家自己看书)
- 方程 $f(x) = 0$ 有重根的情况
 - 结论是只有线性阶的收敛速度 (课本2.4.2小节)
 - 并不比二分法(若可行)更好

迭代法的判停准则

■ 算法2.2 不动点迭代法

$k := 0;$

While $|f(x_k)| > \varepsilon_1$ 或 $|x_k - x_{k-1}| > \varepsilon_2$ **do**

$x_{k+1} := g(x_k);$

$k := k + 1;$

End

判停准则决定了解的准确度、及迭代步数(计算量)

相比二分法, 较难设置

■ $|f(x_k)| \leq \varepsilon_1$, 称为残差判据

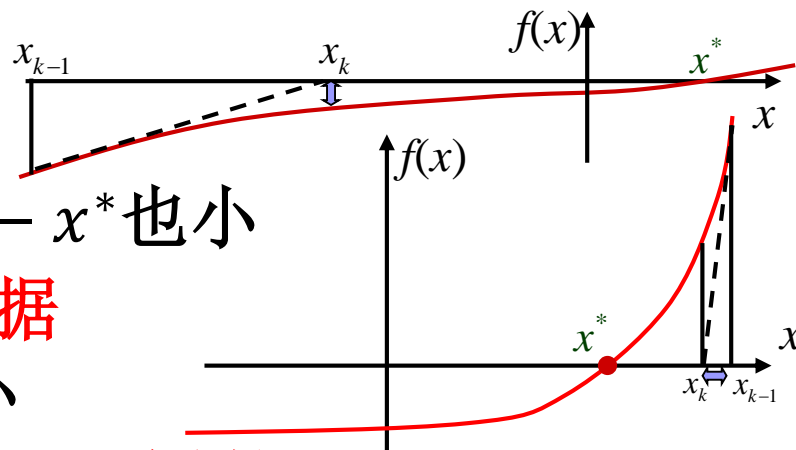
缺点: $f(x_k)$ 小并不意味着 $x_k - x^*$ 也小

■ $|x_k - x_{k-1}| \leq \varepsilon_2$, 称为误差判据

缺点: 并不意味着 $x_k - x^*$ 很小

■ $|x_k - x_{k-1}| \leq \varepsilon_3 |x_k|$, 称为相对误差判据

特点: 阈值为相对量, 适合 $|x^*|$ 很小的情况



牛顿法的问题

- 局部收敛, 依赖于初始解的设定 (见"Java演示")
- 对 $f(x)$ 的连续性要求高
 - 由于连续性不好, 对任意初值都不收敛的例子

$$f(x) = \text{sign}(x - a)\sqrt{|x - a|} = 0$$

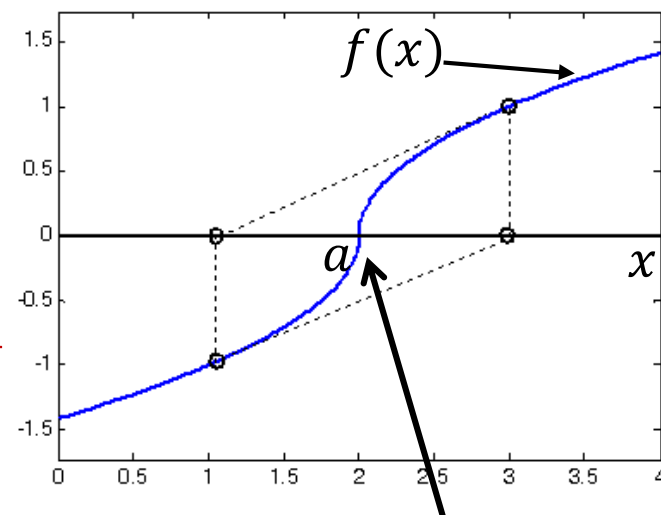
$$x_{k+1} = x_k - 2(x_k - a)$$

$$\Rightarrow x_{k+1} - a = -(x_k - a)$$

$$g'(x^*) = -1$$

迭代解围绕 $x = a$ 点来回跳动

- 需要计算导数 $f'(x)$



一阶导数不连续!



割线法与抛物线法

割线法与抛物线法

■ 割线法 (secant method)

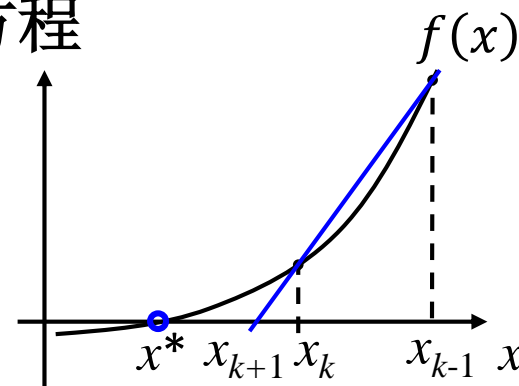
□ 为避免导数计算, 用割线 $P(x)$ 近似 $f(x)$

□ 过 x_k, x_{k-1} 对应函数曲线上点的割线方程

$$P_1(x) = f(x_k) + \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} (x - x_k)$$

□ 求 $P_1(x) = 0$ 的解, 令其为 x_{k+1}

$$x_{k+1} = x_k - \frac{f(x_k)}{f(x_k) - f(x_{k-1})} (x_k - x_{k-1})$$



□ 是一种“广义”的不动点迭代法. 关于其收敛性, 有

■ **定理2.9** 若 $f(x)$ 在根 x^* 某邻域内二阶连续可导且 $f'(x) \neq 0$, 当 x_0, x_1 充分接近 x^* 时, 割线法按 $p \approx 1.618$ 阶收敛 (超线性)

$$\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k| \cdot |e_{k-1}|} = c$$

割线法与抛物线法

■ 割线法

- 将牛顿法中的导数计算替换成近似导数
- 属于一种拟牛顿法 (quasi-Newton method)

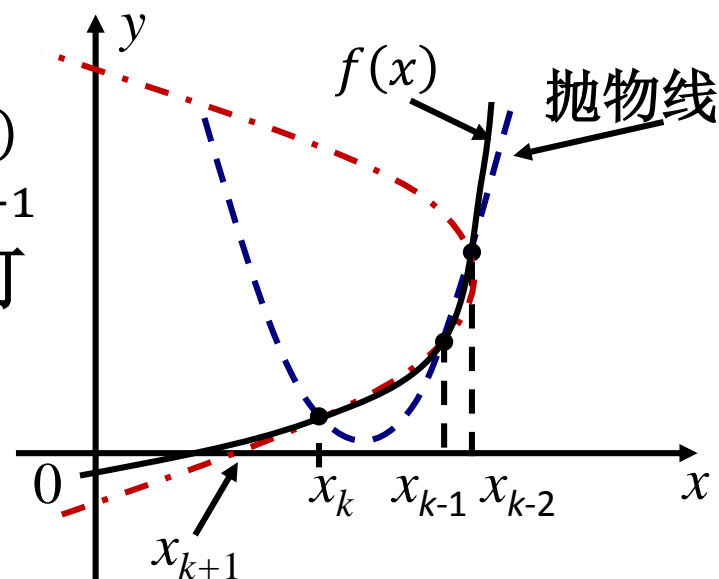
■ 抛物线法

二次多项式近似 $f(x)$
□ $x_k, x_{k-1}, x_{k-2} \xrightarrow{\quad\quad\quad} x_{k+1}$

□ y 看成 x 的抛物线函数，它可能与横轴无交点

□ x 看成是 y 的二次函数(侧向抛物线)，一定能得到 x_{k+1}

□ 这个方法叫逆二次插值法，局部收敛阶 $p \approx 1.839$





实用的求根技术

实用的方程求根技术

- 牛顿下山法
- 多项式方程求根 (2.6.2小节不要求)
- 通用求根算法**zeroin** (简略)

牛顿下山法

是一种阻尼牛顿法

■ 防止牛顿法迭代过程发散

- 一系列比例因子 $\lambda_i \in (0, 1]$, 令 $x_{k+1} = x_k - \lambda_i f(x_k)/f'(x_k)$
- 保证 $|f(x_{k+1})| < |f(x_k)|$.

算法2.5: 阻尼牛顿法

$k := 0$;

While 不满足收敛判据 **do**

$s := f(x_k)/f'(x_k)$;

$x_{k+1} := x_k - s$;

$i := 0$;

While $|f(x_{k+1})| \geq |f(x_k)|$ **do**

$x_{k+1} := x_k - \lambda_i s$; {使用因子序列 $\{\lambda_i\}$ }

$i := i + 1$;

End

$k := k + 1$;

End

$x := x_k$.

λ_i 的值从1开始递减, 也叫“下山”因子

通用求根算法zeroin

- 求解 $f(x) = 0$ 的方法比较 (无法算导数 $f'(x)$ 的情况)
 - 割线法/抛物线法: 需要局部收敛性、函数的光滑性, 但收敛速度快
 - 二分法: 全局收敛、只需函数连续, 但收敛较慢
 - 将两者结合, 得稳定快速的Brent方法(1973)
- Zeroin算法主要步骤
 - 输入有根区间 $[a, b]$
 - 用 b 表示最优解, 与 a 构成有根区间, 而 c 为次优解
 - 重复下面的步骤, 直到 $|f(b)|$ 足够小或 $|a - b|$ 足够小
 - 调整 a, b, c 的值; 执行逆二次插值法/割线法/二分法

通用求根算法zeroin

■ 迭代步中如何调整 a, b, c ?

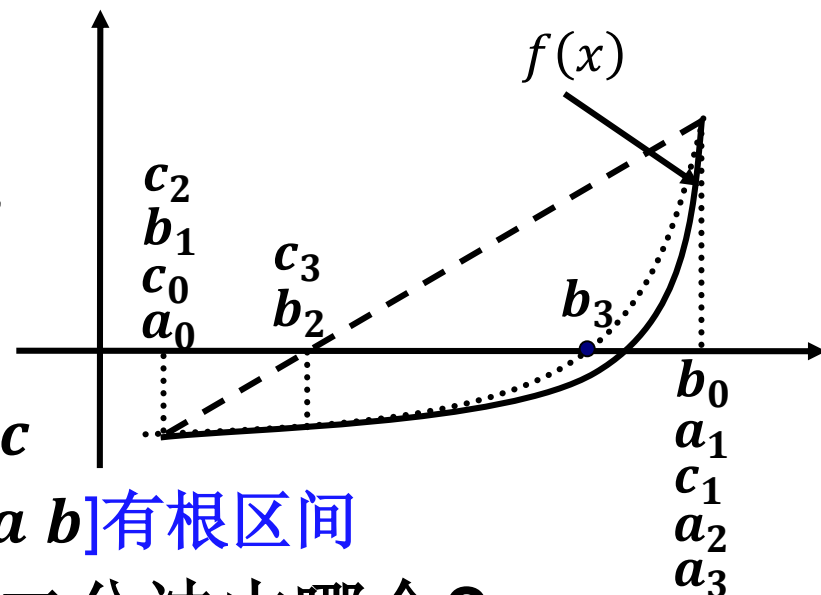
纠正
上一步
解的偏
离

- 若 $f(a)$ 的正负号与 $f(b)$ 相同, 将 c 的值(前一个 b)赋给 a
- 若 $|f(a)| < |f(b)|$, 则对调 a, b 的值, 然后将 a 的值赋给 c

□ **保证:** b 是最优解, c 次优解, $[a, b]$ 有根区间

■ 执行逆二次插值法/割线法/二分法中哪个?

- 若 $c \neq a$, 利用 a, b, c 以及它们的函数值作逆二次插值法的一步迭代, 否则执行一步割线法
- 若逆二次插值法/割线法得到的近似解足够满意(相邻解之差的缩小程度、在有根区间内), 用它更新 b , 否则执行一步二分法更新 b 的值; 改变之前 b 的值赋给 c



通用求根算法zeroin

■ 算法的特点/优点

- 本身不要求函数 $f(x)$ 具有光滑性
- 不需要计算导数 $f'(x)$
- 初始解为有根区间，不需要接近准确解
- 按“逆二次插值, 割线法, 二分法”的优先顺序生成下一步解，保证较快的收敛速度

■ 二分法的改进, 求单根/奇数重的重根 (Matlab中fzero)

详细的Matlab算法程序见课本2.6.3小节

应用实例

■ Matlab中有关命令

- **fzero**: 单变量非线性方程求根
- **fsolve**: 非线性方程组的求解
- **roots**: 求多项式方程的所有根 (与矩阵特征值有关!)
- 定义函数 $f(x)$, 并作为输入参数的方法: 匿名函数"@", 复杂的函数需要用.m文件定义

■ 实例

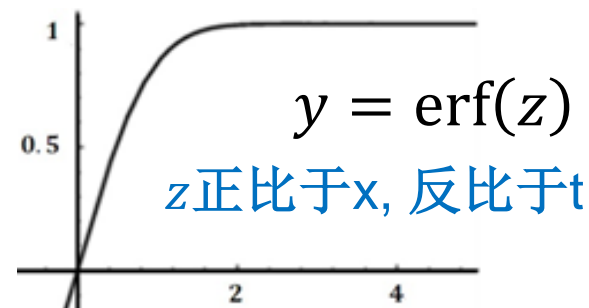
- 解多项式方程: $x^4 - x - 2 = 0$
- 牛顿法不收敛的问题: $f(x) = \text{sign}(x - a)\sqrt{|x - a|} = 0$
- “应用实例”: 城市水管干线埋入地下的深度

应用实例

■ 城市水管干线埋入地下的深度

- 在保证水管不冻结的前提下，尽量将其埋得浅一点
- 土壤温度 $T(x, t)$, x 离地面的距离, t 寒流持续时间

$$\begin{array}{l} x \nearrow, t \searrow \quad \frac{T(x, t) - T_s}{T_i - T_s} \nearrow \text{最大值} 1 \\ x \searrow, t \nearrow \quad \frac{T(x, t) - T_s}{T_i - T_s} \searrow \text{最小值} 0 \end{array}$$



- T_s 是寒流来后的(地上)气温(-10°)
- T_i 是寒流到来前的土壤的常温(20°)
- 可以设 $\frac{T(x, t) - T_s}{T_i - T_s} = \text{erf}\left(\frac{x}{2\sqrt{\alpha t}}\right)$

见 *Tpipe.m*

- 寒流最长持续时间为 t_m , 求温度 $T(x, t_m) = 0$ 对应的 x