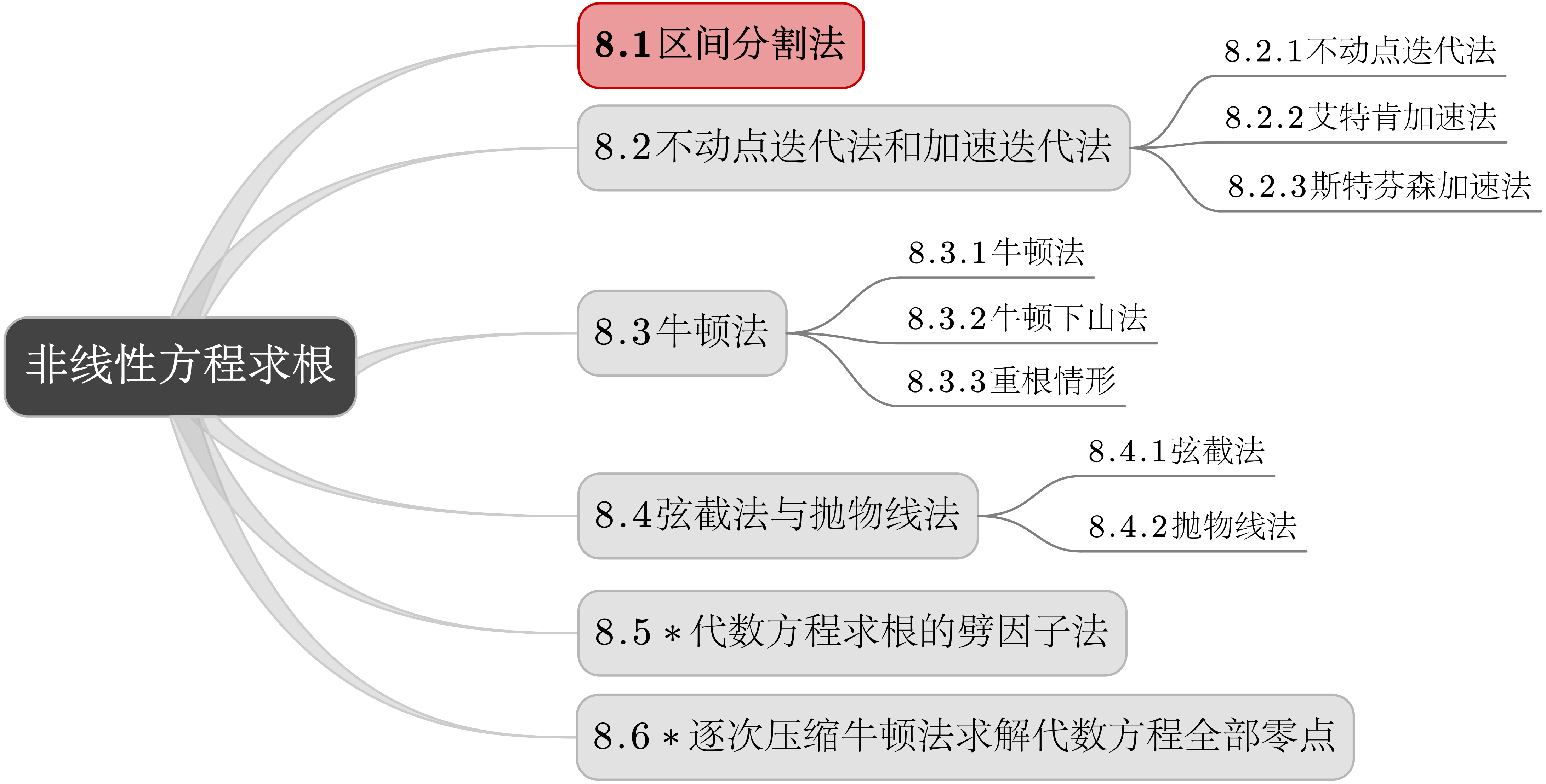
Python 数值分析 算法实践

Python Numerical Analysis Algorithm Practice

第8章 非线性方程求根

讲授：X X X

日期：2024年4月19日


非线性方程求根
8.1区间分割法
8.2不动点迭代法和加速迭代法
8.2.1不动点迭代法
8.2.2艾特肯加速法
8.2.3斯特芬森加速法
8.3牛顿法
8.3.1牛顿法
8.3.2牛顿下山法
8.3.3重根情形
8.4弦截法与抛物线法
8.4.1弦截法
8.4.2抛物线法
8.5 * 代数方程求根的劈因子法
8.6 * 逐次压缩牛顿法求解代数方程全部零点

8.1 区间分割法

二分法的求解步骤:

(1)计算$f(x)$在有根区间$[a,b]$端点处的值$f(a)$和$f(b)$.

(2)计算$f(x)$在区间中点$m=\dfrac{a+b}{2}$的值$f(m)$，并作判断. 若$f(m)=0$(通常算法设计时判断$|f(m)|\leqslant\varepsilon$)，则$m$是根，计算过程结束. 否则，检验$f(a)f(m)<0$，则$m$以代替$b$，否则以$m$代替$a$.

反复执行步骤(2),直到区间的长度小于精度误差ε，此时区间中点即为方程的近似根. 二分法算法简单，且总是收敛的，但缺点是收敛速度较慢,通常不单独使用二分法求根.

8.1 区间分割法

试值法(Method of False Position)又称为试位法(Regula Falsi Method)，试值法是二分法的改进，其思想为：假设$f(a)\cdot f(b)<0$，则经过点$(a,f(a))$和$(b,f(b))$的割线L与x轴的交点$(c,0)$，可得到一个更好的近似值.

割线L的斜率k可有两种方式表示：

$$k=\frac{f(b)-f(a)}{b-a} \text{ 或 } k=\frac{0-f(b)}{c-b}.$$

由于割线L的斜率相等，则可求得交点的横坐标c：

$$\frac{f(b)-f(a)}{b-a}=\frac{0-f(b)}{c-b} \Longrightarrow c=b-\frac{f(b)(b-a)}{f(b)-f(a)}.$$

8.1 区间分割法

可分以下三种情况迭代计算:

(1) $f(a)\cdot f(c)<0$, 则在 $[a,c]$ 内有一个零点;

(2) $f(b)\cdot f(c)<0$, 则在 $[c,b]$ 内有一个零点;

(3) $f(c)=0$, 则 c 是零点.

二分法的收敛判别准则不适宜试值法, 尽管区间长度越来越小, 但它可能不趋近于0. 故修改判别准则如下:

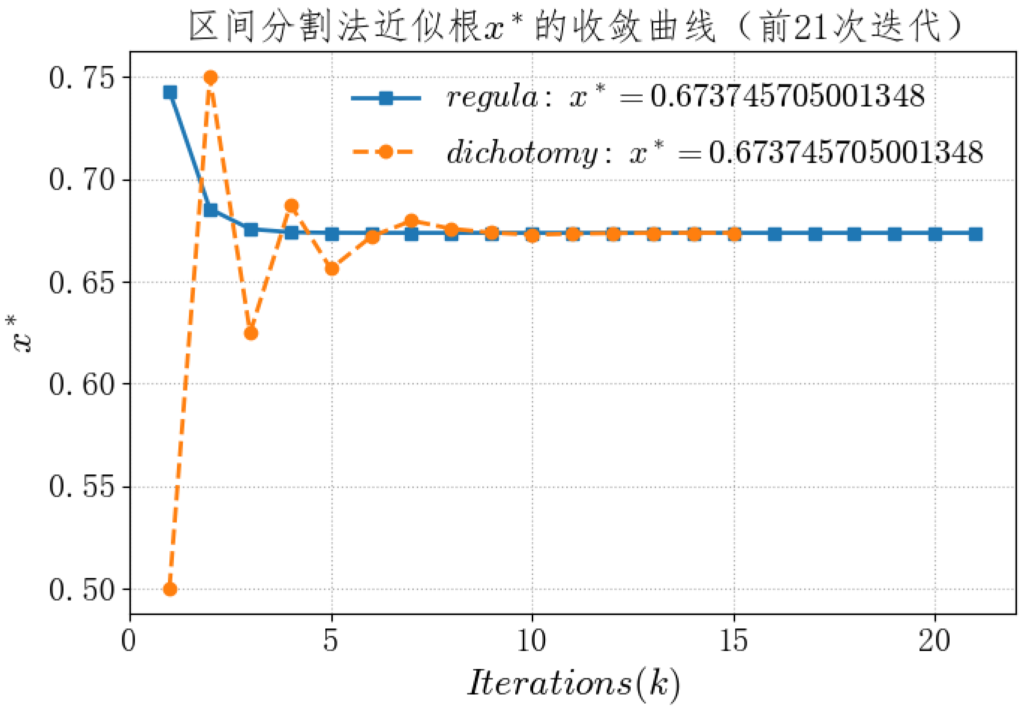
(1)横坐标的封闭性判别条件: $|x_n-x_{n-1}|\leqslant\delta$ 或 $\dfrac{2|x_n-x_{n-1}|}{|x_n|+|x_{n-1}|}\leqslant\delta$;

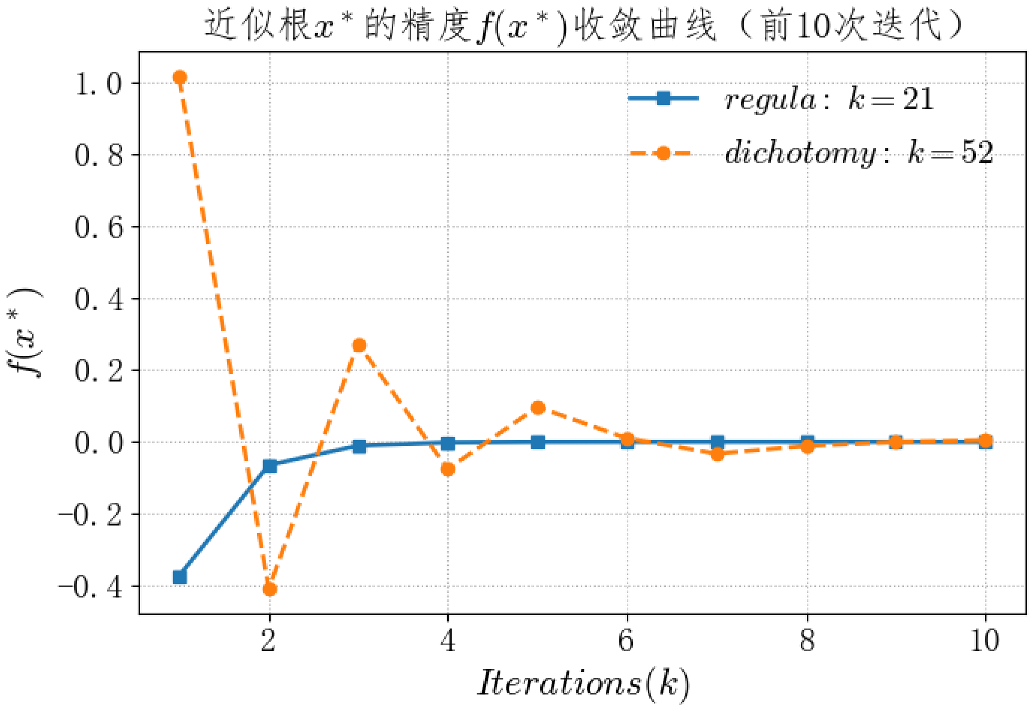
(2)纵坐标的封闭性判别条件: $|f(x_n)|\leqslant\varepsilon$.

8.1 区间分割法

例1:采用试值法和二分法求解$e^{-3x}\sin(4x+2)+4e^{-0.5x}\cos 2x-0.5=0$在区间$[0,1]$和$[3,4]$内的近似根，精度要求$\varepsilon=10^{-16}$，近似根保留15位有效数字.

(1) 在区间$[0,1]$

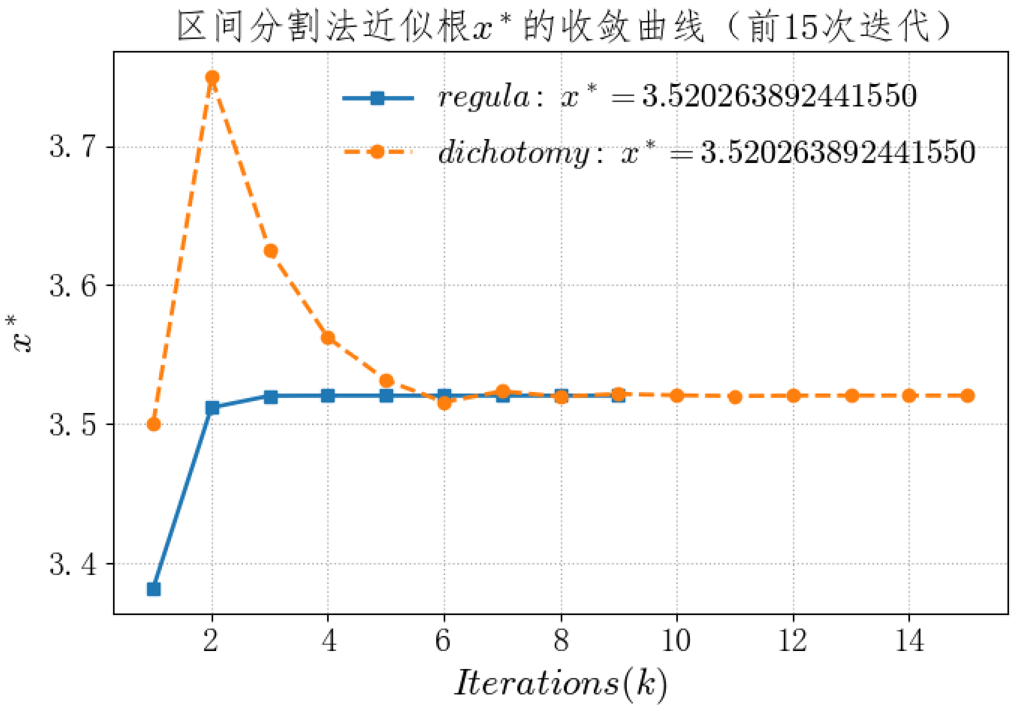


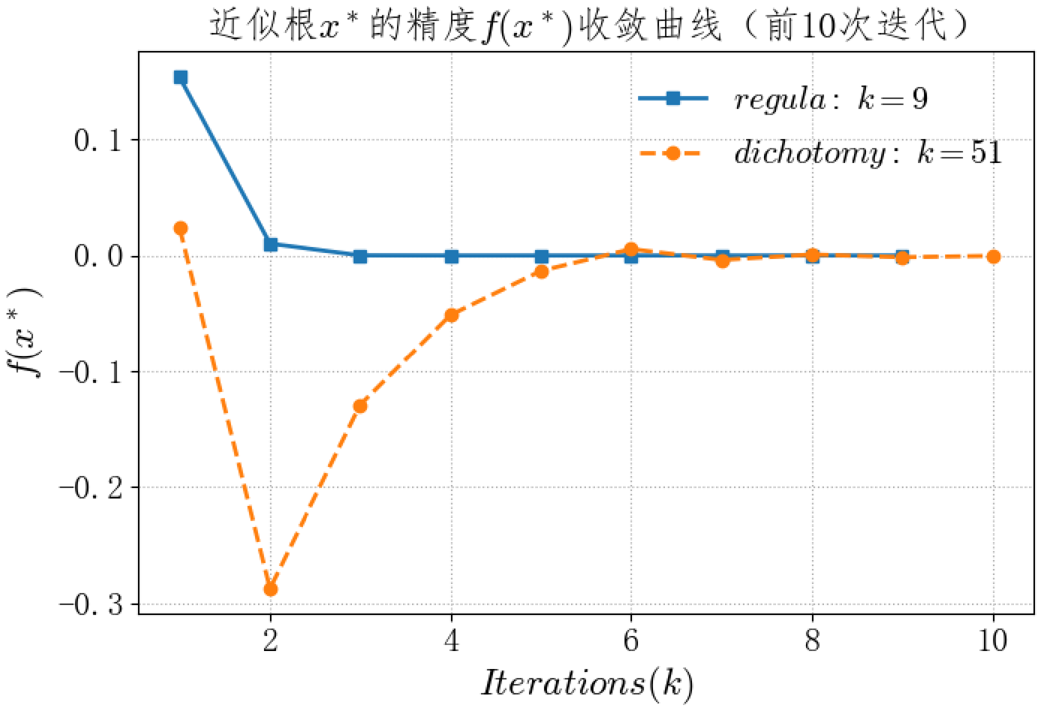


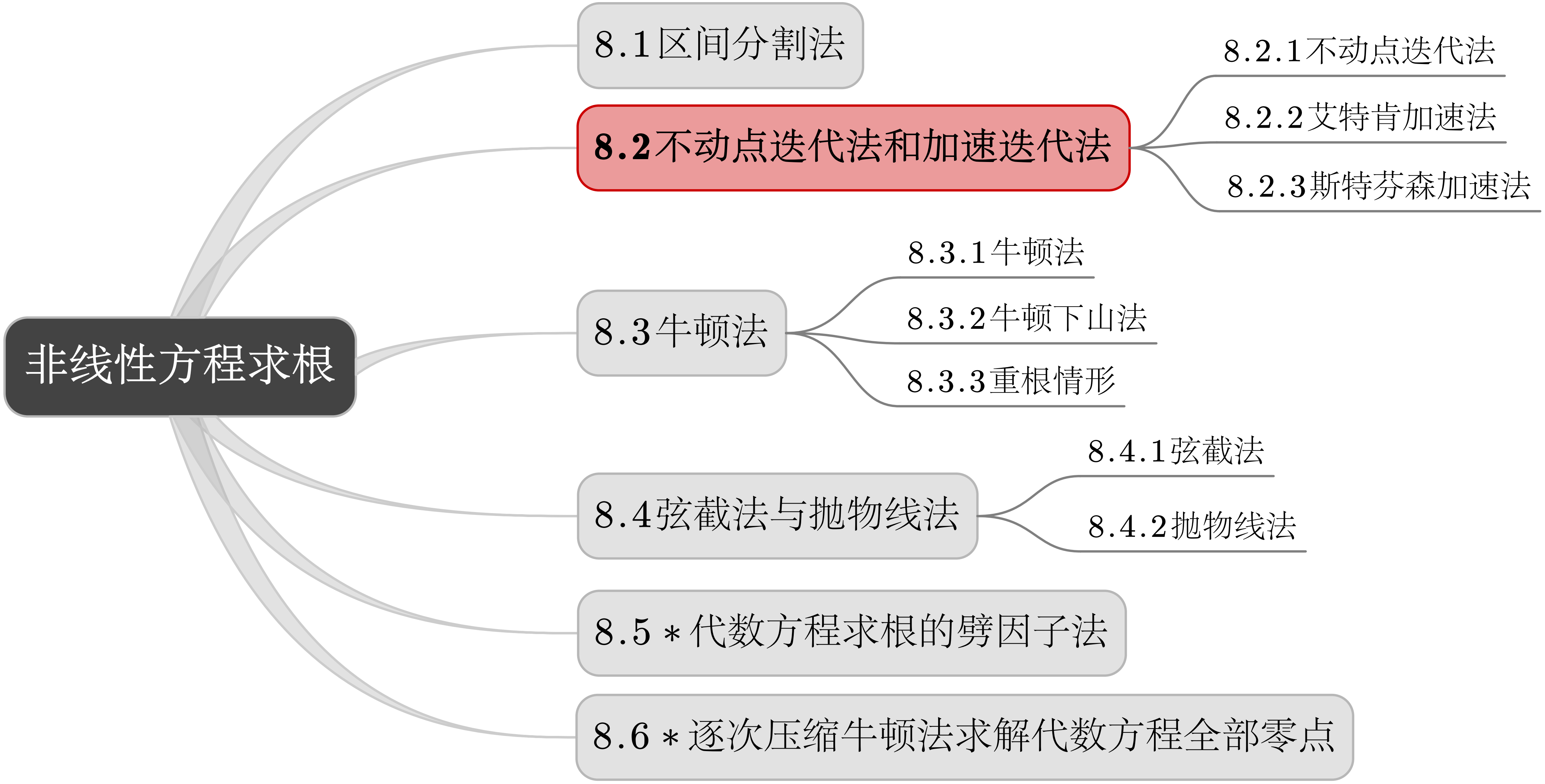
8.1 区间分割法

例1:采用试值法和二分法求解$e^{-3x}\sin(4x+2)+4e^{-0.5x}\cos 2x-0.5=0$在区间$[0,1]$和$[3,4]$内的近似根，精度要求$\varepsilon=10^{-16}$，近似根保留15位有效数字.

(1) 在区间$[3,4]$






非线性方程求根
8.1区间分割法
8.2不动点迭代法和加速迭代法
8.2.1不动点迭代法
8.2.2艾特肯加速法
8.2.3斯特芬森加速法
8.3牛顿法
8.3.1牛顿法
8.3.2牛顿下山法
8.3.3重根情形
8.4弦截法与抛物线法
8.4.1弦截法
8.4.2抛物线法
8.5 ∗ 代数方程求根的劈因子法
8.6 ∗ 逐次压缩牛顿法求解代数方程全部零点

8.2 不动点迭代法和加速迭代法

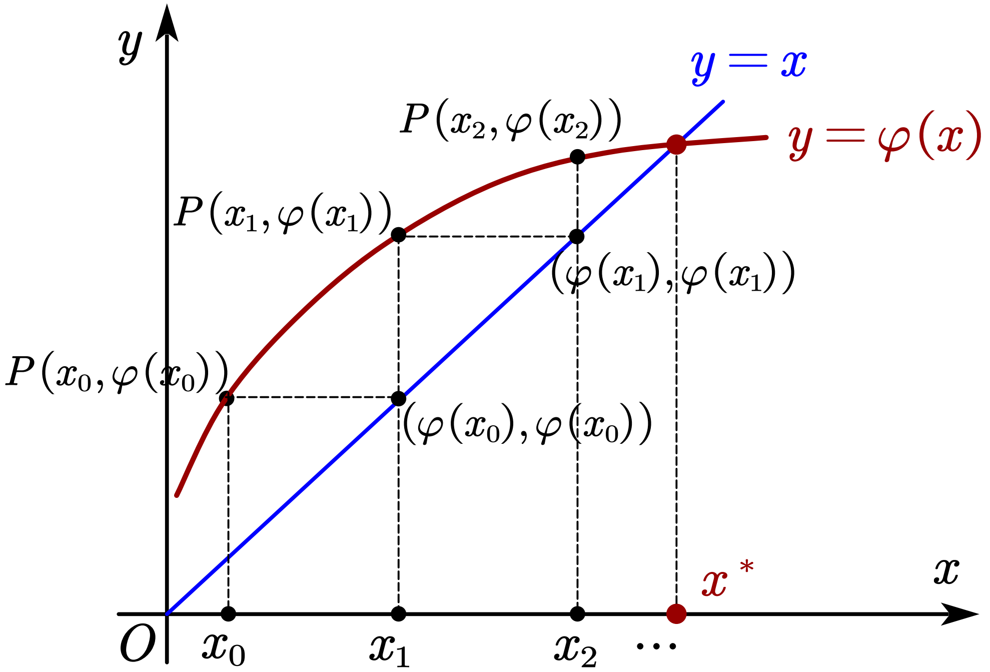
不动点迭代法又称为皮卡(Picard)迭代法、逐次逼近法，不动点迭代法是求方程在某区间内单根的近似值的重要方法.

用不动点迭代法求方程$f(x)=0$的单根x^*的主要步骤为:

(1)把$f(x)=0$变形为$x=\varphi(x)$，称$\varphi(x)$为迭代函数.

(2)以$x_{k+1}=\varphi(x_k)$，$k=0,1,\cdots$为迭代公式，以x^*附近的某一个值x_0为迭代初值，反复迭代，得到迭代序列：$x_1,x_2,\cdots$.

(3)若此序列收敛，当k充分大时，则必收敛于精确根x^*.



8.2 不动点迭代法和加速迭代法

设用迭代公式$x_{k+1}=\varphi(x_k)$求方程$x=\varphi(x)$在区间$[a,b]$内的一个根x^*，根据微分中值定理有

$$x^*-x_{k+1}=\varphi(x^*)-\varphi(x_k)=\varphi'(\xi)(x^*-x_k),\quad \xi\in(x^*,x_k)$$

如果存在常数$L\,(0\leq L<1)$，使得对于$\forall x\in[a,b]$一致成立$|\varphi'(x)|\leq L$，则$|x^*-x_{k+1}|\leq L|x^*-x_k|$，对迭代误差有$e_k=|x^*-x_k|$，同样有$e_k\leq L^k e_0$，由于$0\leq L<1$，因而迭代收敛.

实际应用中，通常只在不动点x^*的邻近考察其收敛性，即局部收敛性. 设$\varphi(x)$在方程$x=\varphi(x)$的根x^*的邻近有连续的一阶导数，且成立$|\varphi'(x^*)|<1$，则迭代过程$x_{k+1}=\varphi(x_k)$在x^*邻近具有局部收敛性.

8.2.2 埃特金加速法

埃特金(Aitken)加速法用来加快不动点迭代法的收敛速度. 先用不动点迭代法算出序列$\{x_k\}$, 再对此序列作修正得到$\{\bar{x}_k\}$. 具体方法:用埃特金加速法对不动点迭代法$x=\varphi(x)$迭代过程加速得到的迭代序列记为$\{x_k\}_{k=0}^{\infty}$,则计算出x_k、x_{k+1}、x_{k+2}后, 对x_{k+1}作以下修正:

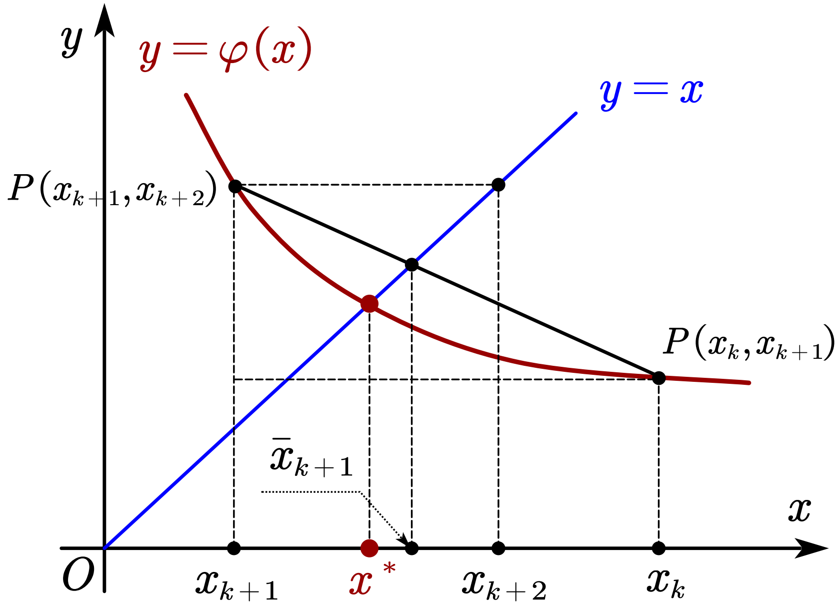
$$\bar{x}_{k+1}=x_k-\frac{(x_{k+1}-x_k)^2}{x_{k+2}-2x_{k+1}+x_k},\ k=0,1,\cdots.$$

然后用$\bar{x}_{k+1}$来逼近方程的根.

$$y=x_{k+1}+\frac{x_{k+2}-x_{k+1}}{x_{k+1}-x_k}(x-x_k)=x\xrightarrow{\text{记}x=\bar{x}_{k+1}}$$

$$\bar{x}_{k+1}=\frac{x_{k+2}x_k-x_{k+1}x_{k+1}}{x_{k+2}-2x_{k+1}+x_k}$$

$$=x_k-\frac{(x_{k+1}-x_k)^2}{x_{k+2}-2x_{k+1}+x_k}.$$



8.2.3 斯特芬森加速法

埃特金方法不管原序列$\{x_k\}$是怎样产生的，对$\{x_k\}$进行加速计算，得到序列$\{\bar{x}_k\}$. 如果把埃特金加速技巧与不动点迭代法结合,则可得到如下的迭代法:

$$y_k=\varphi(x_k),\ z_k=\varphi(y_k),\ x_{k+1}=x_k-\frac{(y_k-x_k)^2}{z_k-2y_k+x_k},\ k=0,1,\cdots.$$

称为斯特芬森(Steffensen)迭代法，斯特芬森迭代法是二阶收敛的. 某些情况下，即使不动点迭代法和埃特金加速法不收敛，但斯特芬森法仍可能收敛.

8.2 不动点迭代法和加速迭代法

例2:用不动点迭代法、埃特金加速法和斯特芬森加速法求解非线性方程$x^4+2x^2-x-3=0$的根，精度要求$\varepsilon=10^{-16}$，迭代初值$x_0=1.0$，最大迭代次数200.

对方程进行变形,构造迭代公式如下:

(1)若变形为$x=\sqrt{\sqrt{x+4}-1}$，则得迭代公式$x_{k+1}=\sqrt{\sqrt{x_k+4}-1}$，$k=0,1,\cdots$;

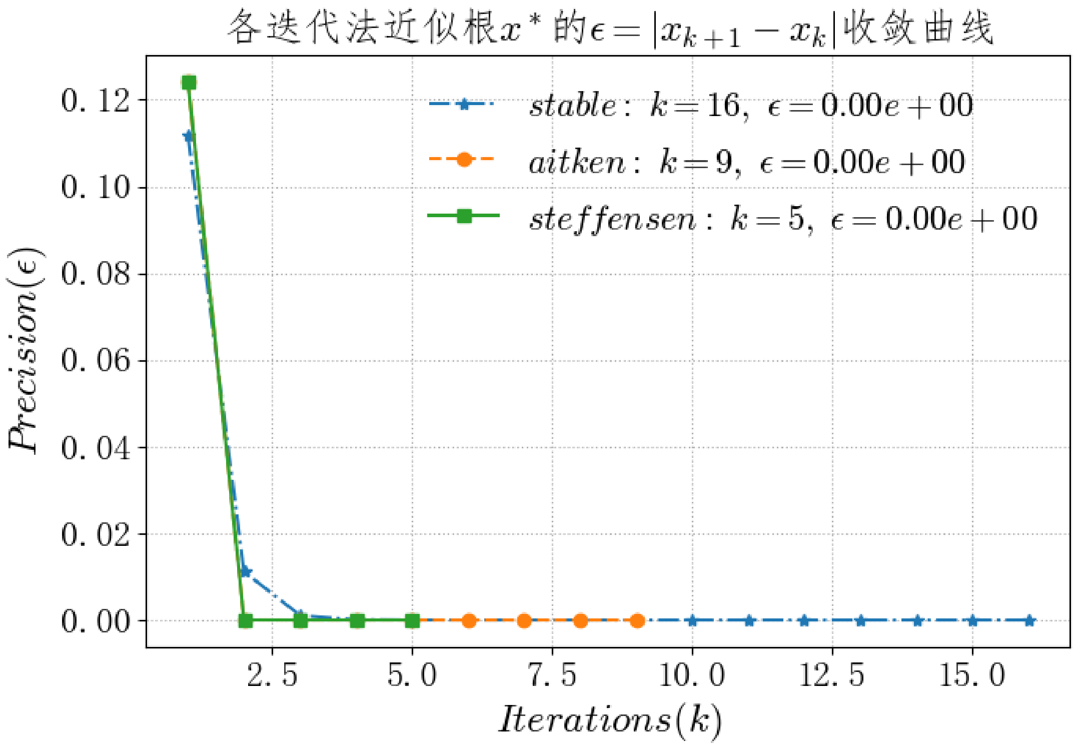
(2)若变形为$x=\sqrt[4]{3+x-2x^2}$，则得迭代公式$x_{k+1}=\sqrt[4]{3+x_k-2x_k^2}$，$k=0,1,\cdots$;

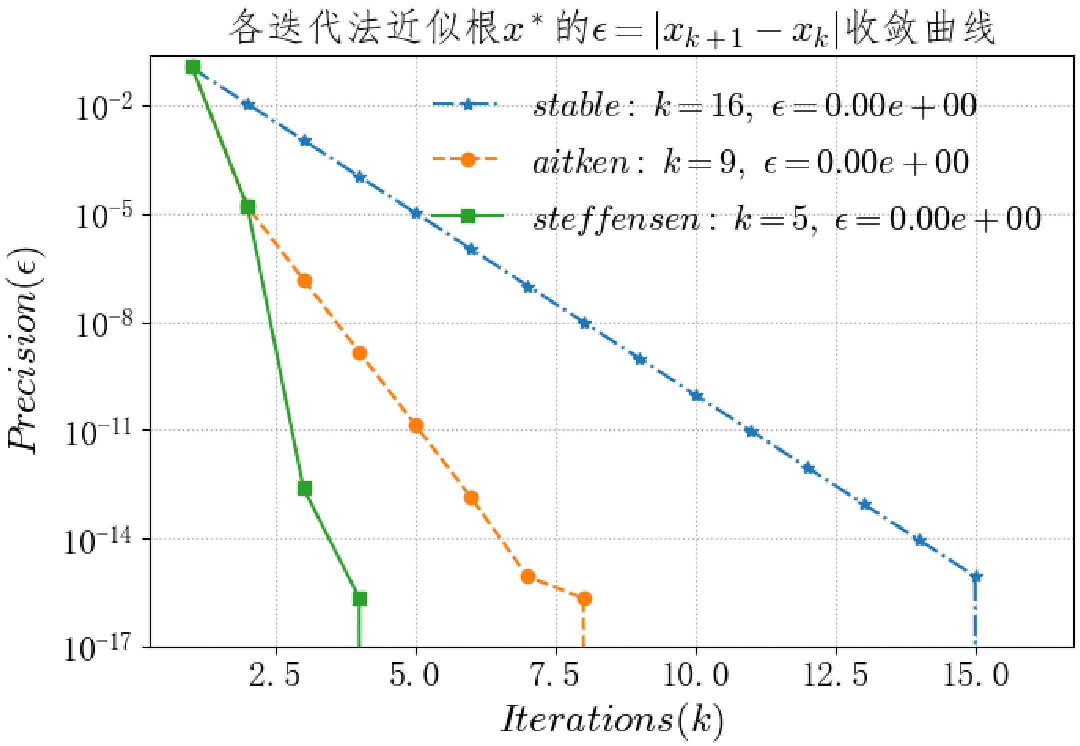
(3)若变形为$x=x^4+2x^2-3$，则得迭代公式$x_{k+1}=x_k^4+2x_k^2-3$，$k=0,1,\cdots$.

8.2 不动点迭代法和加速迭代法

(1)若变形为$x=\sqrt{\sqrt{x+4}-1}$, 则得迭代公式$x_{k+1}=\sqrt{\sqrt{x_k+4}-1}$, $k=0,1,\cdots$;

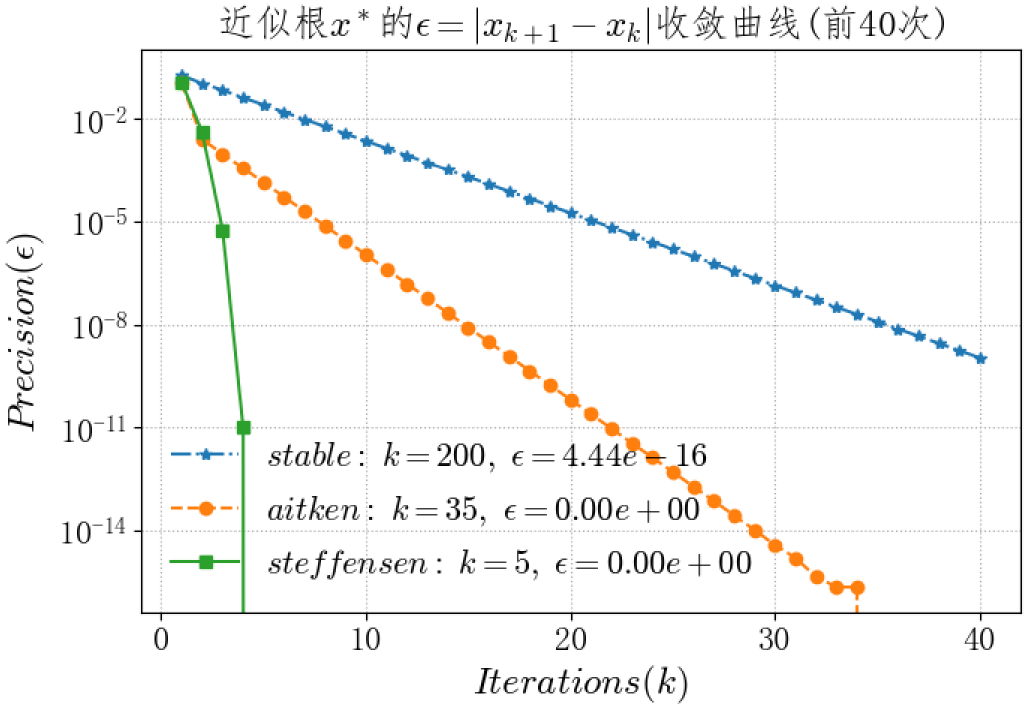
三种方法最终在满足精度要求下的解均为$x^*=1.124123029704315$.

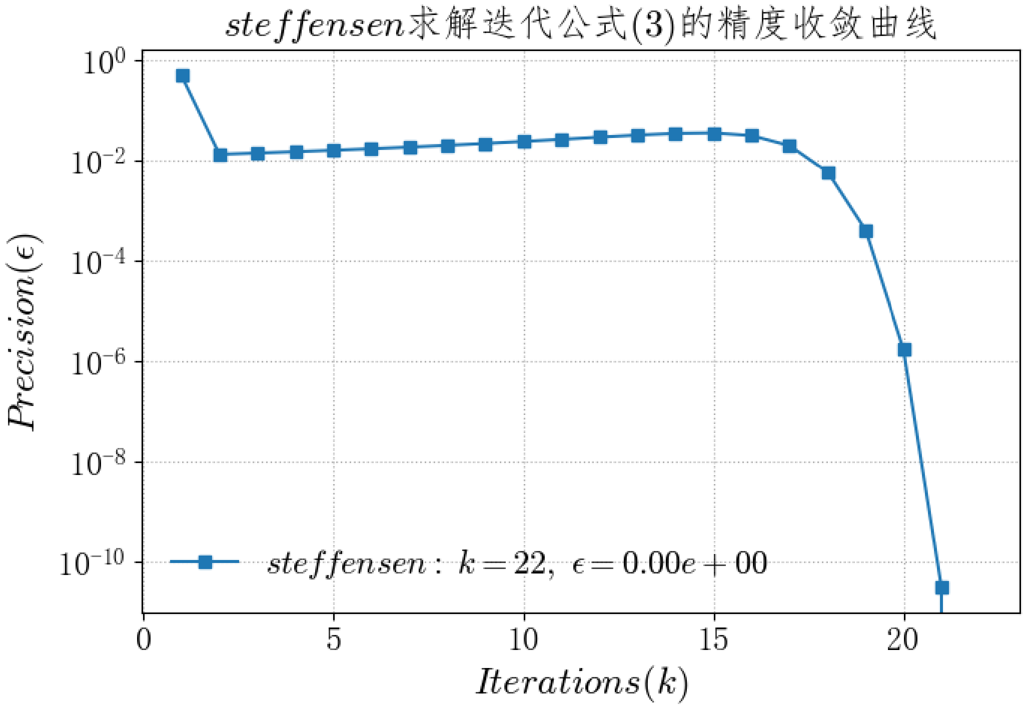


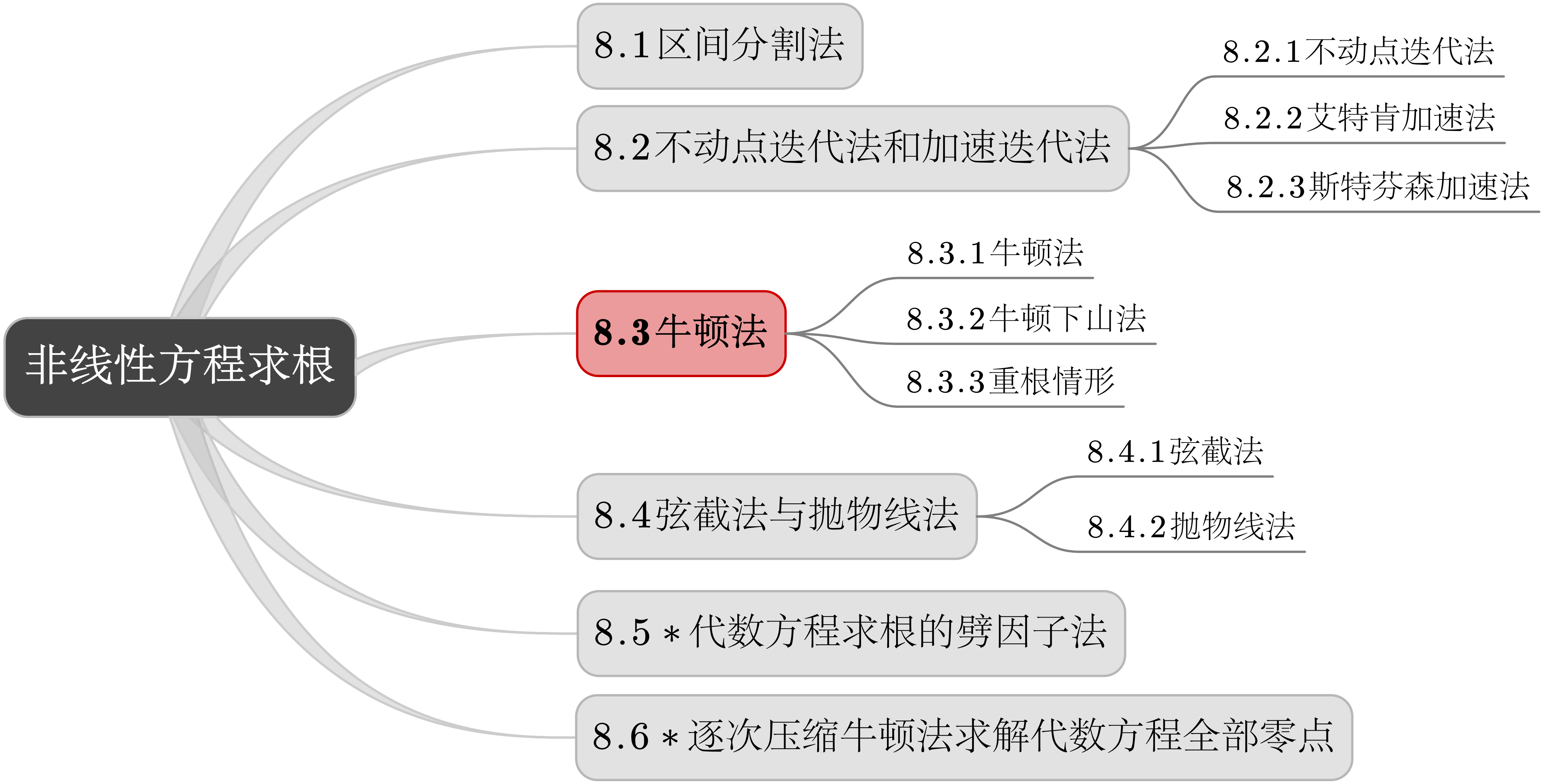


8.2 不动点迭代法和加速迭代法

对于(2)情形下的迭代公式，不动点迭代法达到最大迭代次数200，精度为$\varepsilon = 4.44\times10^{-16}$，收敛速度较慢，而埃特金加速法迭代35次，斯特芬森加速法迭代5次. 对于(3)情形下的迭代公式，不动点迭代法和埃特金加速法均不收敛，迭代过程近似根的值非常大而出现溢出现象，斯特芬森加速法迭代22次收敛，近似根仍为$x^* = 1.124123029704315$.






非线性方程求根
8.1区间分割法
8.2不动点迭代法和加速迭代法
8.2.1不动点迭代法
8.2.2艾特肯加速法
8.2.3斯特芬森加速法
8.3牛顿法
8.3.1牛顿法
8.3.2牛顿下山法
8.3.3重根情形
8.4弦截法与抛物线法
8.4.1弦截法
8.4.2抛物线法
8.5 ∗代数方程求根的劈因子法
8.6 ∗逐次压缩牛顿法求解代数方程全部零点

8.3 牛顿迭代法

牛顿迭代法实质上是一种线性化方法，其基本思想是将非线性方程$f(x)=0$逐步归结为某种线性方程来求解. 牛顿迭代法又称切线法.

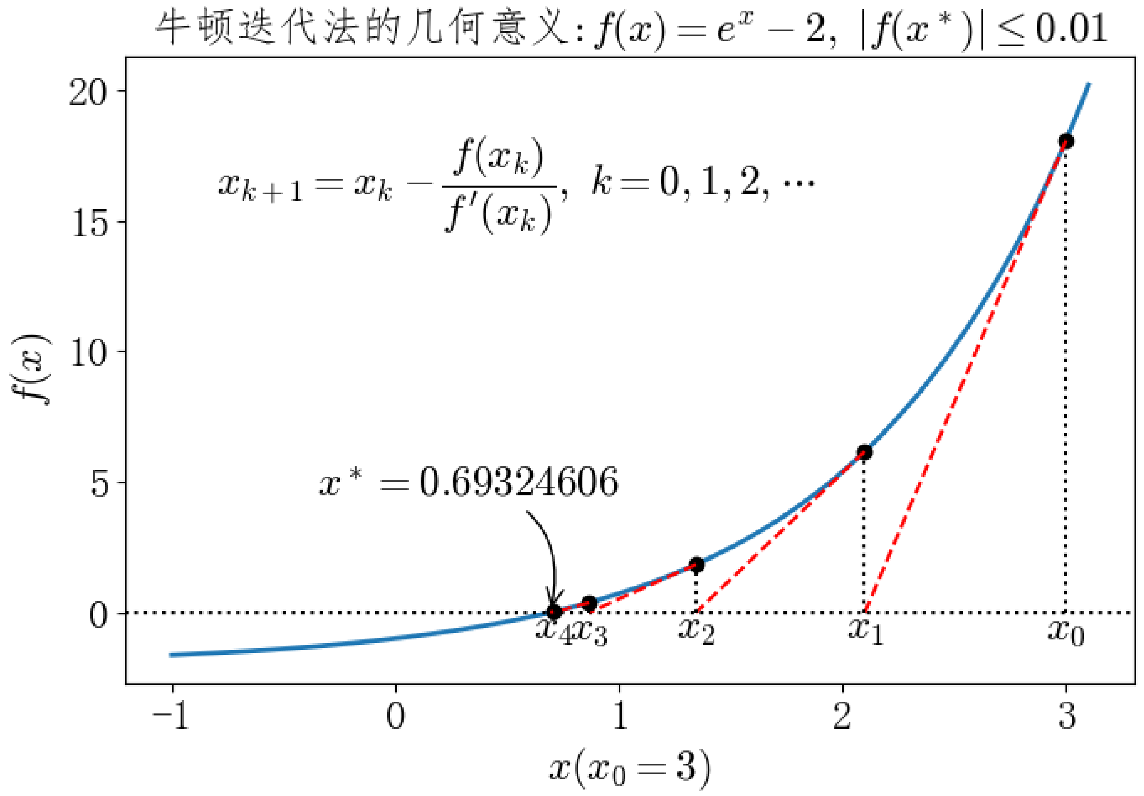
用牛顿迭代法求的单根的主要步骤:

(1)构造牛顿法的迭代公式

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)},\ k=0,1,\cdots.$$

(2)以x^*附近的某一个值x_0为迭代初值，代入迭代公式，反复迭代，得到序列$x_1,x_2,\cdots$.

(3)若序列收敛，当迭代次数k充分大时，则必收敛于精确根x^*.



8.3 牛顿迭代法

哈利法(Halley)可用于加速牛顿法收敛，哈利迭代公式:

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)}\left(1-\frac{f(x_k)f''(x_k)}{2(f'(x_k))^2}\right)^{-1},\ k=0,1,\cdots.$$

哈利法在单根情况下可达到三阶收敛.

简化迭代公式为$x_{k+1}=x_k-\lambda f(x_k)$，$k=0,1,\cdots$. 为保证收敛，系数$\lambda$只需满足$0<\lambda<2/f'(x_k)$即可. 如果$\lambda$取常数$\dfrac{1}{f'(x_0)}$，则称简化牛顿法，也称平行弦法. 简化牛顿法线性收敛.

8.3 牛顿迭代法

牛顿法的收敛性依赖初值x_0的选取，如果x_0偏离所求根x^*较远，则牛顿法可能发散. 牛顿下山法，引入下山因子$\lambda > 0$，保证函数值稳定下降$|f(x_{k+1})| < |f(x_k)|$的同时，加速收敛速度. 牛顿下山法公式：

$$x_{k+1} = x_k - \lambda \frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, \cdots.$$

下山因子的取法：从$\lambda = 1$开始，逐次减半，即$\lambda = 1, \frac{1}{2}, \frac{1}{2^2}, \cdots$，直到满足下降条件$|f(x_{k+1})| < |f(x_k)|$.

8.3 牛顿迭代法

将求重根问题化为求单根问题进行牛顿法求解. 对$f(x)=(x-x^*)^m g(x)$, 令函数

$$\mu(x)=\frac{f(x)}{f'(x)}=\frac{(x-x^*)g(x)}{mg(x)+(x-x^*)g'(x)}.$$

则化为求$\mu(x)=0$的单根x^*的问题, 对它用牛顿法是二阶收敛的. 其迭代函数为

$$\varphi(x)=x-\frac{\mu(x)}{\mu'(x)}=x-\frac{f(x)f'(x)}{[f'(x)]^2-f(x)f''(x)}.$$

从而构造迭代方法

$$x_{k+1}=x_k-\frac{f(x_k)f'(x_k)}{[f'(x_k)]^2-f(x_k)f''(x_k)},\ k=0,1,\cdots.$$

8.3 牛顿迭代法

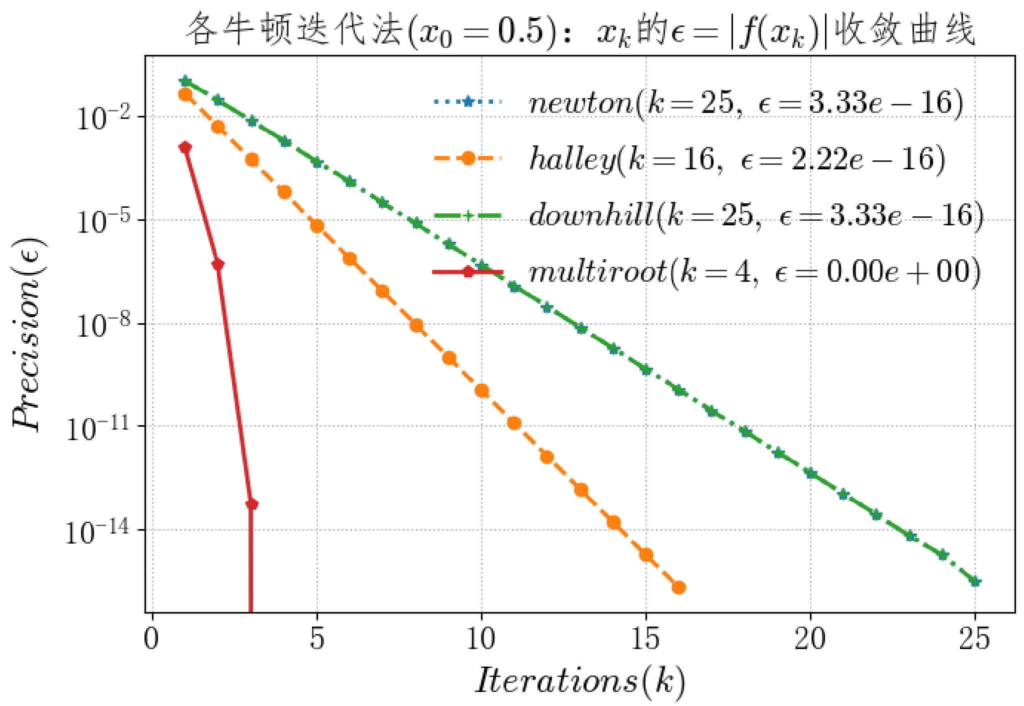
例3:分别采用不同的牛顿法求解方程$(x-1)\left(\sin(x-1)+3x\right)-x^3+1=0$在$x_0=0.5$和$x_0=2.5$附近的近似根，精度要求$\varepsilon=10^{-15}$，并比较各方法的效率.

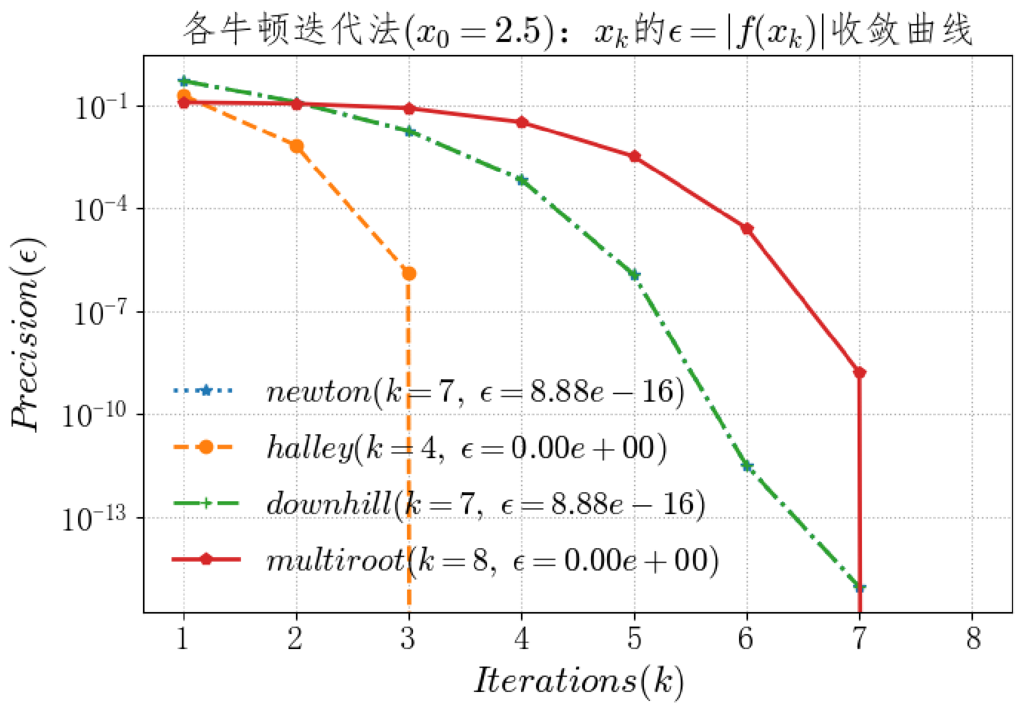
表 8-1　不同牛顿法在初值 0.5 时近似根的迭代结果

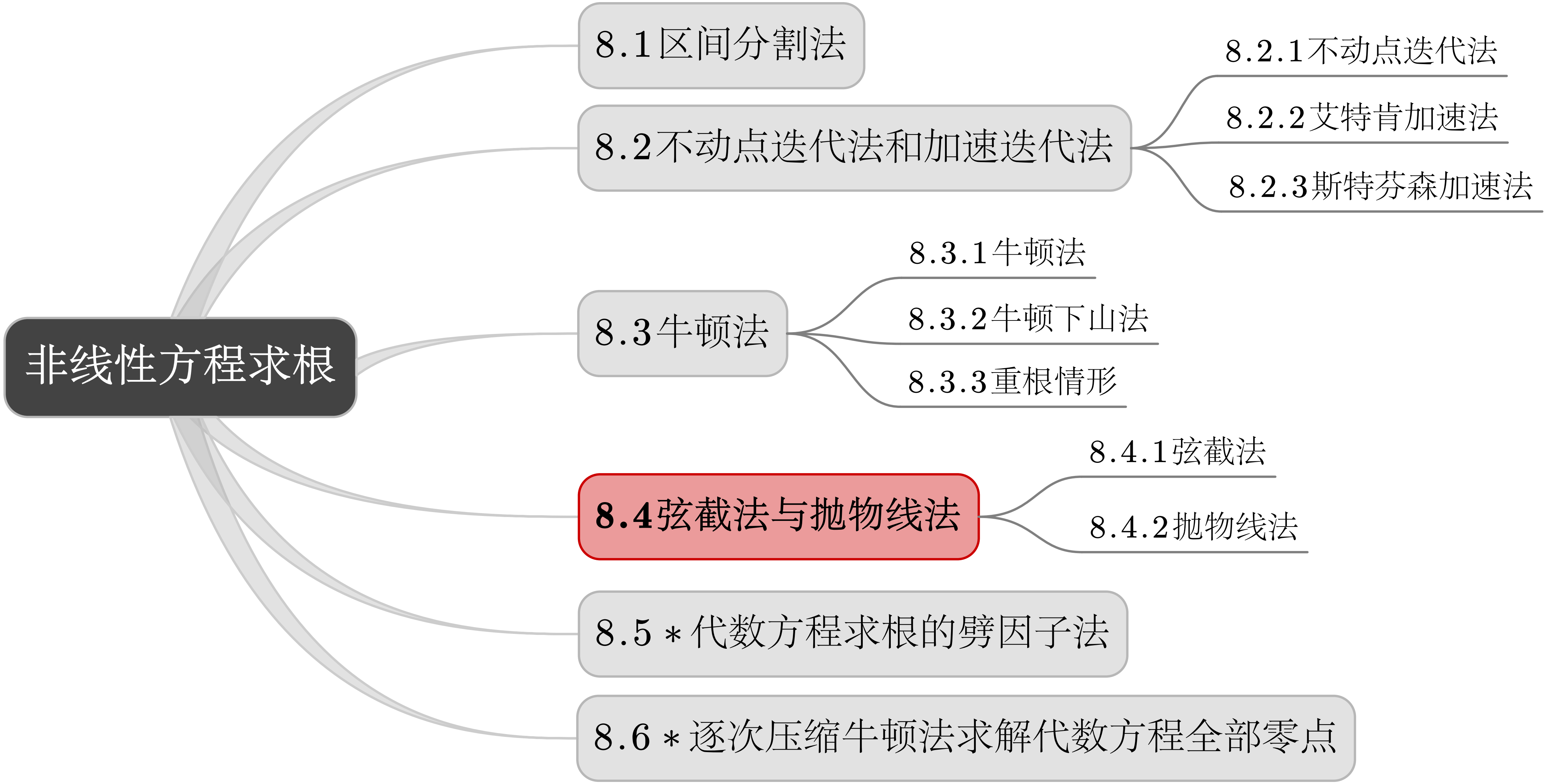
迭代方法	迭代次数	近似根	近似根的精度
牛顿法	25	0.99999997996933098765	3.330669073875470e−16
哈利法	16	0.99999998577509119357	2.220446049250313e−16
牛顿下山法	25	0.99999997996933098765	3.330669073875470e−16
重根情形	4	0.99999999904738023915	0.000000000000000e+00

8.3 牛顿迭代法

例3：分别采用不同的牛顿法求解方程$(x-1)(\sin(x-1)+3x)-x^3+1=0$在$x_0=0.5$和$x_0=2.5$附近的近似根，精度要求$\varepsilon=10^{-15}$，并比较各方法的效率.






非线性方程求根
8.1区间分割法
8.2不动点迭代法和加速迭代法
8.2.1不动点迭代法
8.2.2艾特肯加速法
8.2.3斯特芬森加速法
8.3牛顿法
8.3.1牛顿法
8.3.2牛顿下山法
8.3.3重根情形
8.4弦截法与抛物线法
8.4.1弦截法
8.4.2抛物线法
8.5 * 代数方程求根的劈因子法
8.6 * 逐次压缩牛顿法求解代数方程全部零点

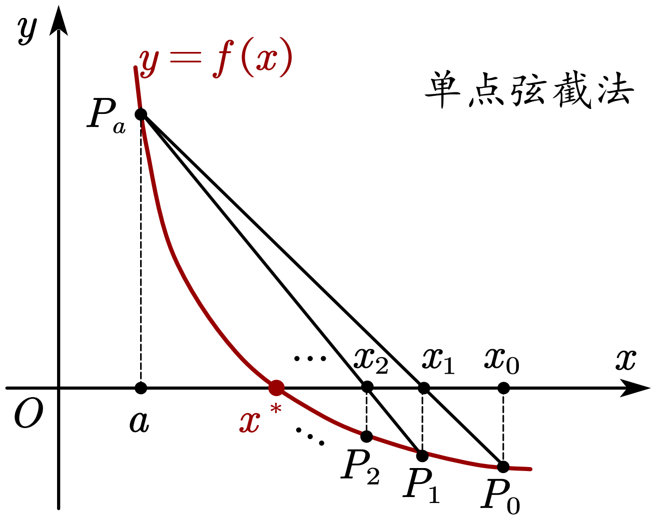
8.4 弦截法与抛物线法

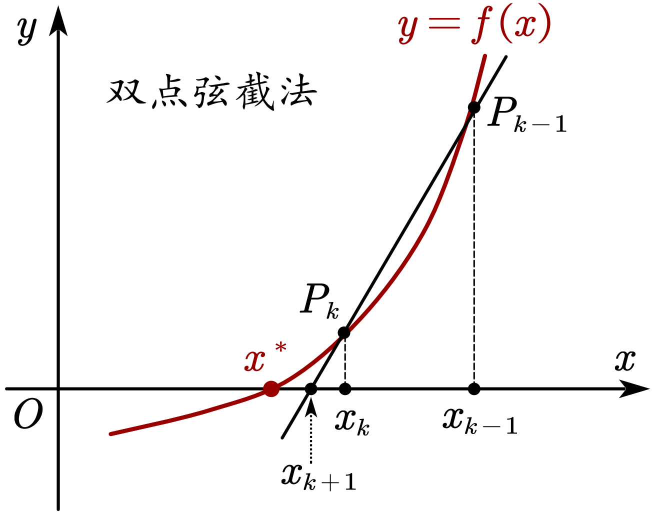
Python 数值分析 算法实践

为避免计算函数的导数 $f'(x)$，使用差商 $\dfrac{f(x_k)-f(x_{k-1})}{x_k - x_{k-1}}$ 代替导数，便得到迭代公式：

$$x_{k+1} = x_k - \frac{f(x_k)}{f(x_k)-f(x_{k-1})}(x_k - x_{k-1}),\ k = 1, 2, \cdots.$$

称作双点弦截法，收敛阶约为1.618. 双点弦截法启动需要两个迭代初值和. 几何意义如右图所示，以弦 $\overline{P_{k-1}P_k}$ 为斜率，通过点 $(x_k, f(x_k))$ 构成的直线与 x 轴的交点即为 x_{k+1}.



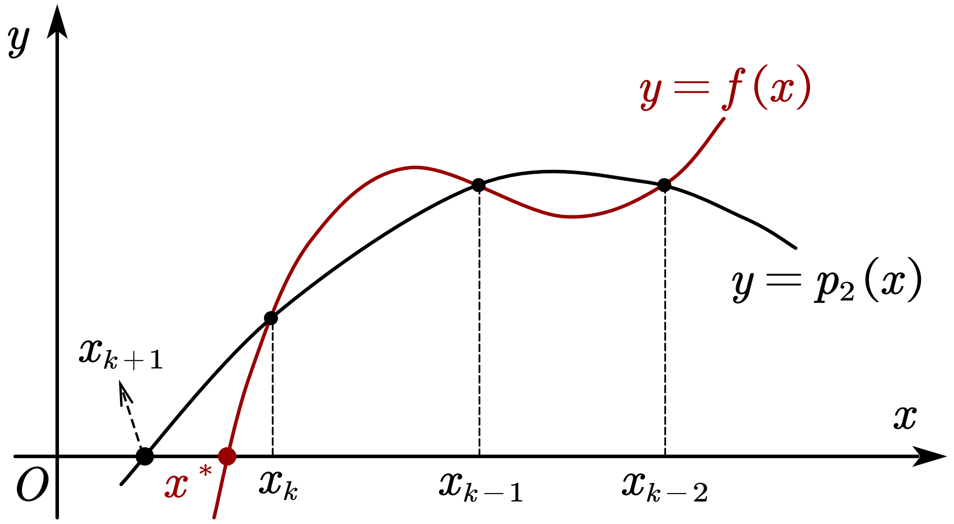


8.4 弦截法与抛物线法

设已知方程$f(x)=0$的三个近似根x_k,x_{k-1},x_{k-2}，以这三点为节点构造二次插值多项式$p_2(x)$，并适当选取$p_2(x)$的一个零点x_{k+1}作为新的近似根，该迭代过程称为抛物线法，亦称为密勒(Müller)法. 公式

$$x_{k+1}=x_k-\frac{2f(x_k)}{\omega_k+\operatorname{sgn}(\omega_k)\sqrt{\omega_k^{\,2}-4f(x_k)f[x_k,x_{k-1},x_{k-2}]}},\quad k=2,3,\cdots.$$

其中$\omega_k=f[x_k,x_{k-1}]+f[x_k,x_{k-1},x_{k-2}](x_k-x_{k-1})$. 几何意义如图所示.



8.4 弦截法与抛物线法

例5:用双点弦截法和抛物线法求解方程$xe^x-1=0$在区间$[0,1]$的近似根，精度要求$\varepsilon=10^{-16}$.

表 8-2　弦截法和抛物线法近似根及精度迭代过程

k	弦截法近似根及精度		抛物线法近似根及精度	
1	0.567285874779347	3.940384008427333e−04	0.567143288110075	−6.354608839131970e−09
2	0.567142994791206	−8.168598065738664e−07	0.567143290409784	0.000000000000000e+00
3	0.567143290375262	−9.539125045421315e−11		
4	0.567143290409784	0.000000000000000e+00		