第五章 非线性方程的数值解法

肖 明

微电子科学与技术学院

邮箱：xiaom37@mail.sysu.edu.cn

回顾

第一章 数值计算的误差

- 绝对误差和相对误差
- 有效数字

第二章 插值法

- 拉格朗日插值和牛顿插值
- 分段低次插值

第三章 最小二乘法

- 最小二乘法
- 加权最小二乘法

第四章 数值积分

- 牛顿-柯特斯公式
- 龙贝格算法和高斯型求积公式

课程安排

第五章 非线性方程的数值解法

- 二分法
- 迭代法
- 牛顿-雷扶生方法

第六章 方程组的数值解法

- 高斯消元法
- 追赶法
- 平方根法、迭代法

第七章 常微分方程的数值解法

- 欧拉公式、龙格-库塔公式
- 阿达姆斯方法
- 算法的稳定性及收敛性

第五章---非线性方程的数值解法（引言）

微电子领域中的非线性现象

- 非线性元器件（电容、电感、晶体管、光电器件等）
- 载流子输运现象（能带理论、异质结）
- 微电子制造过程中的非线性（磁控溅射、刻蚀、扩散、原子沉积等）
- 微电子器件应用（模拟、算法等）

在现代计算机（尤其是超级计算机）快速发展的情况下，针对非线性系统在给定初值和演化规则之后在一定时间以内的变化情况，通过**方法合适**、**精度足够**的模拟手段，求得**近似解**是通常的解决方案。

1 **方程求根与二分法**

2 迭代法及其收敛性

3 牛顿法-雷扶生方法

4 牛顿下山法与弦截法

本章主要讨论单变量非线性方程

$$f(x)=0$$

的求根问题

本章重点研究对象

多项式方程

$$f(x)=a_0x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n \quad (a_0\neq 0),$$

其中系数$a_i(i=0,1,\cdots,n)$为实数.

的求根问题

非线性方程的根

- 若方程$f(x^*)=0$，则x^*称为函数$f(x)$的零点
- 若方程 $f(x)=(x-x^*)^m g(x)$,

  其中m为正整数，且$g(x^*)\neq 0$.

  则：(1) 当$m=1$时，则称x^*为单根，

  (2) 当$m>1$称x^*为m重根，或x^*为$f(x)$的m重零点.
- 若x^*是$f(x)$的m重零点,且$g(x)$充分光滑，则

$$f(x^*)=f'(x^*)=\cdots=f^{(m-1)}(x^*)=0,\ f^{(m)}(x^*)\neq 0.$$

n次方程在复数域有且只有n个根

根的存在性---零点定理

条件：1^0 $f(x)\in C^0[a,b]$，

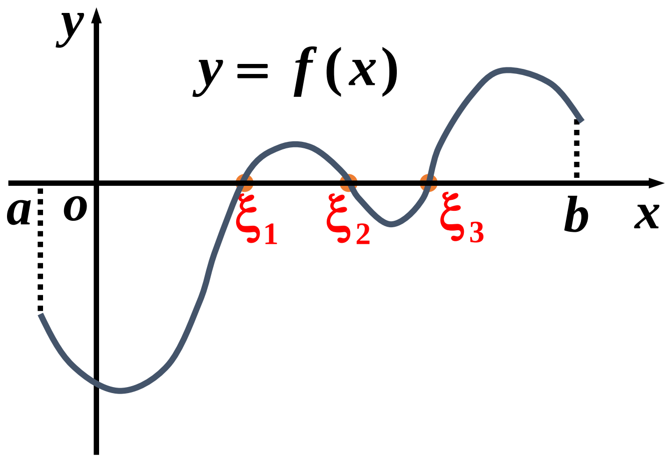
2^0 $f(a)f(b)<0$

结论：至少有一点 $x_0\in(a,b)$，使得 $f(x_0)=0$

即方程 $f(x)=0$ 在 (a,b) 内至少存在一个实根.

零点定理

的几何解释:



零点定理的应用范例

例 1 证明方程 $x^3-4x^2+1=0$ 在区间 $(0,1)$ 内至少有一个根.

证明 显然 $f(x)=x^3-4x^2+1\in C[0,1]$,

又 $f(0)=1>0,\quad f(1)=-2<0$

故据零点定理, 至少存在一点 $\xi\in(0,1)$,使 $f(\xi)=0$,

即 $\xi^3-4\xi^2+1=0$

原命题得证.

零点定理的应用范例

例 2 **证明方程** $x = e^{x-3} + 1$ **至少有一个不超过 4 的正根。**

证明 **令** $f(x) = x - e^{x-3} - 1$

显然 $f(x)$ 在闭区间$[0,4]$上连续,

且 $f(0) = -e^{-3} - 1 < 0$

$f(4) = 4 - e^{4-3} - 1 = 3 - e > 0$

根据零点定理，在开区间 $(0, 4)$ **内至少存在一点**

$\xi \in (0, 4)$, 使 $f(\xi) = 0$, **原命题得证 .**

零点定理的推论

设 (1) $f(x)\in C^0(-\infty,+\infty)$,

(2) $\lim\limits_{x\to+\infty} f(x)=A,\ \lim\limits_{x\to-\infty} f(x)=B,\ AB<0$

则 $f(x)$在$(-\infty,+\infty)$至少有一个零点.

设 (1) $f(x)\in C^0(-\infty,+\infty)$,

(2) $\lim\limits_{x\to+\infty} f(x)=+\infty(-\infty),\ \lim\limits_{x\to-\infty} f(x)=-\infty(+\infty)$

则 $f(x)$在$(-\infty,+\infty)$至少有一个零点.

奇次方程$a_0x^{2n+1}+a_1x^{2n}+\ldots+a_{2n}x+a_{2n+1}=0$

至少有一个实根.

高次方程的根

n次方程在复数域有且只有n个根

当$n \geq 5$时,方程

$$f(x) = a_0 x^n + a_1 x^{n-1} + \cdots + a_{n-1} x + a_n \quad (a_0 \neq 0),$$

的根不能用公式表示,因此，通常对$n \geq 3$的多项式方程求根与一般连续函数方程$f(x) = 0$一样多可采用迭代法求根.

搜索法：

先求出使 $f'(x)=0$ 的点，然后将这些点放在定义域内，将定义域分成几部分，算出驻点处的函数值，即可知道方程的有根区间。

牛顿二分法

设 $f(x)$ 在 $[a,b]$ 上连续且单调，$f(a)\cdot f(b)<0$，则在 $[a,b]$ 有且仅有一个根。

根存在，但未必好求，可用对分法：

不妨设 $f(a)<0$，$f(b)>0$

(1)若 $f\left(\frac{a+b}{2}\right)=0$ 则根为 $x=\frac{a+b}{2}$，否则：若 $f\left(\frac{a+b}{2}\right)<0$，

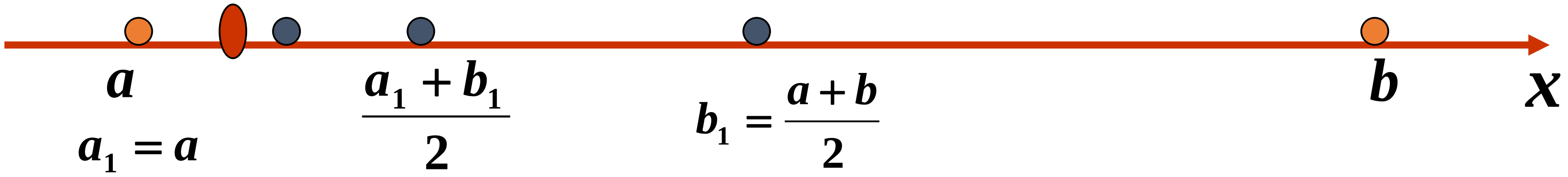
令 $a_1=\frac{a+b}{2}$，$b_1=b$ 反之 $b_1=\frac{a+b}{2}$，$a_1=a$.

(2) 对 $[a_1,b_1]$ 区间重复(1)的计算，并产生 $[a_2,b_2],\cdots$

牛顿二分法

$$b_1 - a_1 = \frac{b-a}{2}, b_2 - a_2 = \frac{b_1 - a_1}{2} = \frac{b-a}{2^2}$$

考察有根区间$[a,b]$,



如此反复二分下去，即可得出一系列有根区间

$$[a,b] \supset [a_1,b_1] \supset [a_2,b_2] \supset \ldots \supset [a_k,b_k] \supset \ldots,$$

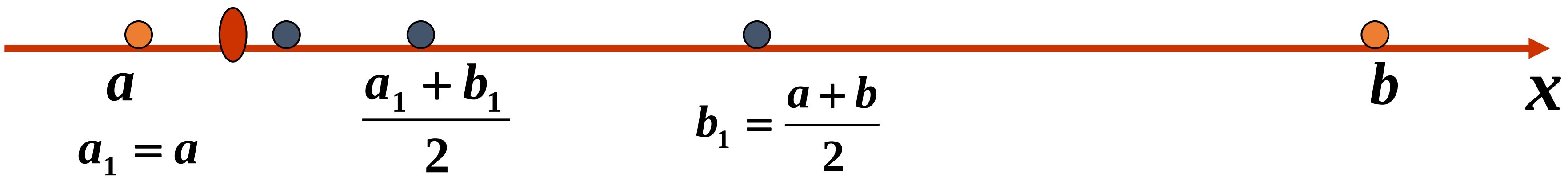
其中

$$[a_k,b_k]\text{的长度} b_k - a_k = (b-a)/2^k \xrightarrow{\text{当}k\to\infty\text{时}} 0,$$

即这些区间最终必收敛于一点x^*,该点显然就是所求的根.

目标：寻找一个非常小的区间，使函数的正负号在该区间上发生改变。

牛顿二分法



由于

$$\boxed{|x^*-x_k|\le\frac{b_k-a_k}{2}=\frac{b-a}{2^{k+1}}},$$

只要二分足够多次（即k充分大），

则有 $$|x^*-x_k|<\varepsilon$$

这里ε为预定的精度.

不难得出：

$$n>\frac{\ln(b-a)-\ln\varepsilon}{\ln 2}-1$$

对分区间次数的估计 由 $|x^*-x_n|\le\frac{b_n-a_n}{2}=\frac{b-a}{2^{n+1}}<\varepsilon$

例 2

求方程 $f(x)=x^3-x-1=0$ 在区间 $[1,\quad 1.5]$ 内的一个实根,要求准确到小数后第2位.

解（二分法） $a=1.0, b=1.5, f(a)<0, f(b)>0$

取中点 $x_0=1.25$，将区间二等分,

$\because f(x_0)<0, \therefore$ 令 $a_1=x_0=1.25, b_1=b=1.5,$

得新的有根区间 $[a_1, b_1]$

如此二分下去即可。现估计二分次数

$$\left|x^*-x_n\right|<0.005 \Rightarrow n \geq 6.95$$

所以二分7次可达到要求。

牛顿二分法-MATLAB程序实现

例 2

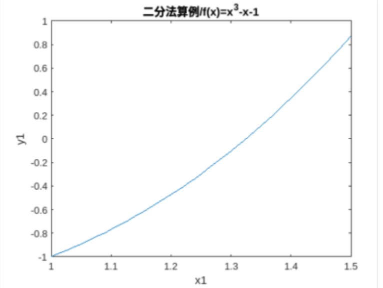
求方程 $f(x)=x^3-x-1=0$ 在区间 $[1,\ 1.5]$ 内的一个实根,要求准确到小数后第2位.

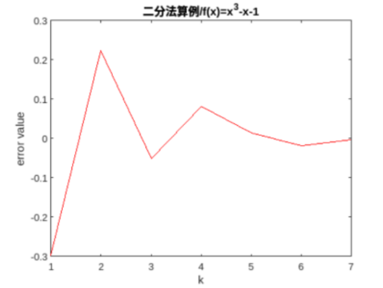
```matlab
clc;
clear all;
close all;

f= @(beta) beta.^(3)-beta-1;
a =1;%赋值a
b = 1.5;%赋值b
eps = 1e-2;%赋值eps
x1 = a:0.01:b;
y1=x1.^3-x1-1;
plot(x1,y1);
xlabel('x1');%设置横轴坐标
ylabel('y1');%设置纵轴坐标
title('二分法算例/f(x)=x^3-x-1');
T = bisect(f,a,b,eps);%调用函数
data = [T(:,6) T(:,end)]; %输出求解的beta  C error
save('data93552.txt','data','-ascii');
```

```matlab
function T = bisect(f,a,b,eps)
%输入
%f代表输入的函数 a, b代表区间范围[a,b],eps是输入的误差
%T代表输出的参数
%包括迭代次数   左区间a   a点函数值   右区间b  b点函数值   区间a和b的中点值xk    xk点函数值
k=1%设置初始值
x=(a+b)/2;%设置初始区间中点
fprintf('        k          a          f(a)         b          f(b)         xk          f(xk)\n ');%输
T=[k,a,f(a),b,f(b),x,f(x)];%对T赋值
while abs(T(k,4)-T(k,6))>eps/2   %判断和误差的大小
    k=k+1;%循环计数
    if  f(x)*f(a)==0   %判断当函数值为0的时候
        a=a;%左区间重新赋值
        b=x;%右区间重新赋值
        x=(a+b)/2;%区间中点重新赋值
        T=[T;k,a,f(a),b,f(b),x,f(x)];%对T赋值
        break
    elseif  f(x)*f(a)>0  %判断当f(x)和f(a)同号的情况
        a=x;%左区间重新赋值
        b=b;%右区间重新赋值
        x=(a+b)/2;%区间中点重新赋值
        T=[T;k,a,f(a),b,f(b),x,f(x)];%对T赋值
    elseif  f(x)*f(a)<0   %判断当f(x)和f(a)异号的情况
        a=a;%左区间重新赋值
        b=x;%右区间重新赋值
        x=(a+b)/2;%区间中点重新赋值
        T=[T;k,a,f(a),b,f(b),x,f(x)];%对T赋值
    end

end
disp(T);%输出变量T
fprintf('经过%d次迭代，函数方程根的近似解为：x=%.8f\n',k-1,T(k-1,6))%输出迭代过程
error = T(:,7);%误差
figure(2);%新建一个窗口
plot(1:k,error,'r');%画图
xlabel('k');%设置横轴坐标
ylabel('error value');%设置纵轴坐标
title('二分法算例/f(x)=x^3-x-1');
end
```





k = 1

k	a	f(a)	b	f(b)	xk	f(xk)
1.0000	1.0000	-1.0000	1.5000	0.8750	1.2500	-0.2969
2.0000	1.2500	-0.2969	1.5000	0.8750	1.3750	0.2246
3.0000	1.2500	-0.2969	1.3750	0.2246	1.3125	-0.0515
4.0000	1.3125	-0.0515	1.3750	0.2246	1.3438	0.0826
5.0000	1.3125	-0.0515	1.3438	0.0826	1.3281	0.0146
6.0000	1.3125	-0.0515	1.3281	0.0146	1.3203	-0.0187
7.0000	1.3203	-0.0187	1.3281	0.0146	1.3242	-0.0021

经过6次迭代，函数方程根的近似解为：x=1.32031250

```
>>
```

精确解为:
x =1.324718

牛顿二分法 优点：对函数要求低，计算简单;
缺点：收敛慢且对有偶数重根的情况不适合。

二分法-MATLAB程序实现

例 3 计算sqrt（2）

方法I:

/MATLAB Drive/sqrt2.m

```
clc;
clear all;
k=0;
a=1;
b=2;
M=2;
while b-a>eps
    x = (a+b)/2;
    if x^2>M
        b=x;
    else
        a = x;
    end
    k = k+1;
end
fprintf('经过 %d 次迭代，函数方程根的近似解为:\n a=%.16f\n b=%.16f',k,a,b)%输出迭代过程
```

命令行窗口

```
经过 52 次迭代，函数方程根的近似解为:
 a=1.4142135623730949
 b=1.4142135623730951
>>
```

二分法-MATLAB程序实现

例 3　计算sqrt（2）

方法2：

```
clc;
clear all;
k=0;
a=1;
b=2;
M=2;
f=@(x) x.^2-M;
k=0;
while abs(b-a)>eps*abs(b)
    x=(a+b)/2
    if sign(f(x))==sign(f(b))
        b=x;
    else
        a=x;
    end
    k=k+1;
end

fprintf('经过 %d 次迭代，函数方程根的近似解为:\n a=%.16f\n b=%.16f',k,a,b)%输出迭代过程
```

```
    1.4142


x =

    1.4142


x =

    1.4142

经过 52 次迭代，函数方程根的近似解为:
 a=1.4142135623730949
 b=1.4142135623730951
>>
```

不动点迭代法

构造不动点方程，以求得近似根。

即由方程 $f(x)=0$ 变换为其等价形式 $x=\varphi(x)$，然后建立迭代格式，

$$x_{k+1}=\varphi(x_k)$$

当给定初值 x_0 后，由迭代格式可求得数列 $\{x_k\}$。此数列可能收敛，也可能不收敛。如果 $\{x_k\}$ 收敛于 x^*，则它就是方程的根。因为：

$$x^*=\lim_{k\to\infty}x_{k+1}=\lim_{k\to\infty}\varphi(x_k)=\varphi(\lim_{k\to\infty}x_k)=\varphi(x^*)$$

故 k 充分大时，x_k 可作为方程根的近似值

按上述方法构造迭代格式来求解方程的方法称为简单迭代法或逐次迭代法。

非线性方程的求解---不动点迭代

不动点迭代法： 将方程 $f(x)=0$ 改写为: $x=\varphi(x)$.

若要求x^*满足$f(x^*)=0$,则$x^*=\varphi(x^*)$;反之亦然，

称x^*为函数的一个不动点

求$f(x)$的零点等价于求不动点

按下列公式反复迭代

$$x_{k+1}=\varphi(x_k) \qquad (k=0,1\cdots)$$

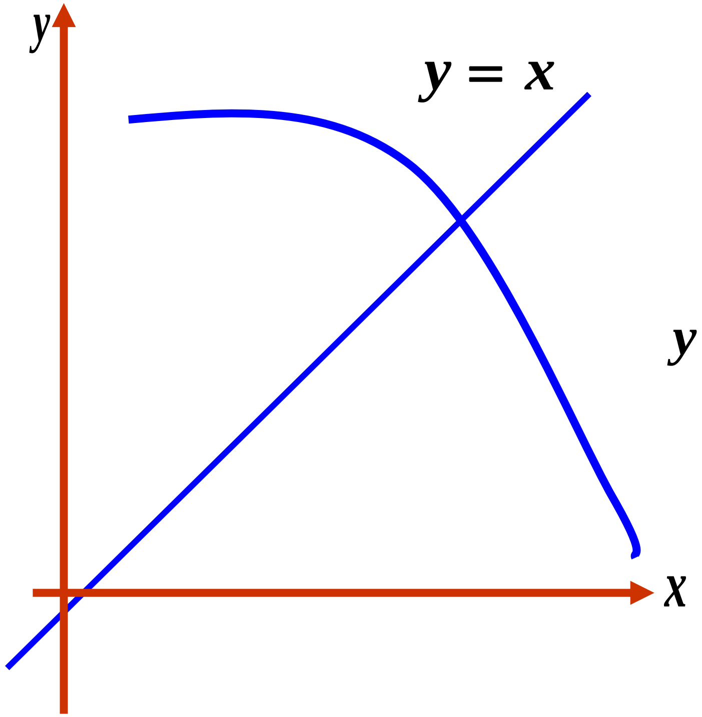
$\varphi(x)$称为迭代函数

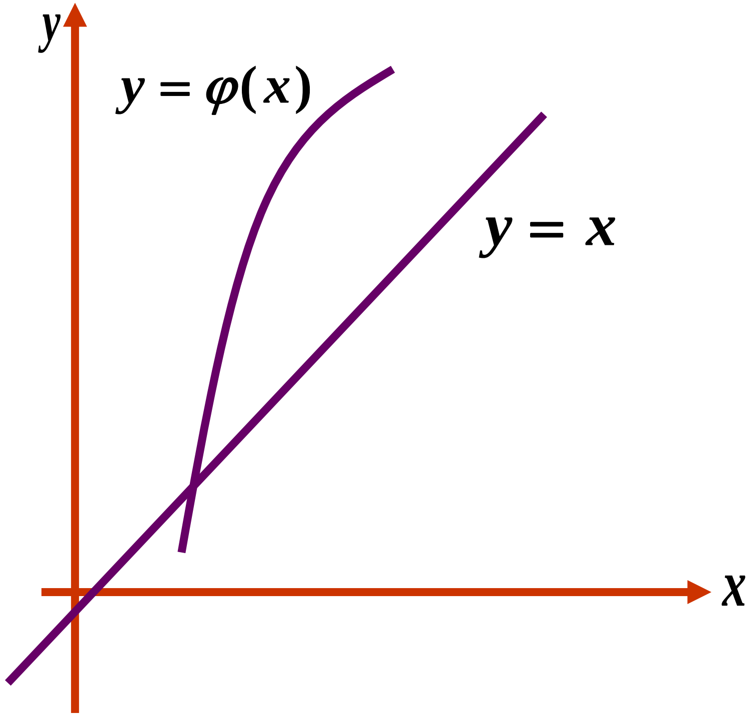
非线性方程的求解---不动点迭代

因为：$f(x) = x - \varphi(x)$

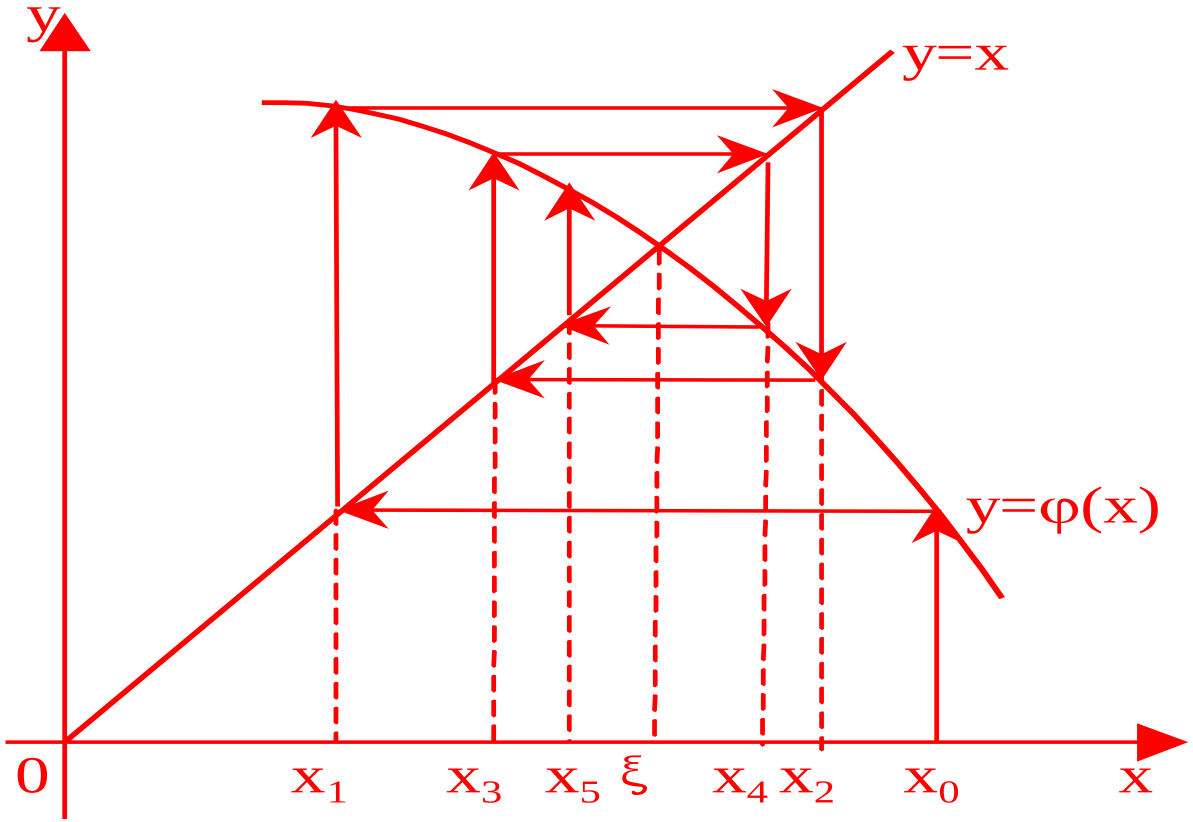
所以：　　$f(x) = 0$ 的解 $\Leftrightarrow$ $x = \varphi(x)$ 的解

又　　$x = \varphi(x)$ 的解 $\Leftrightarrow \begin{cases} y = \varphi(x) \\ y = x \end{cases}$ 的解（两条线的交点）

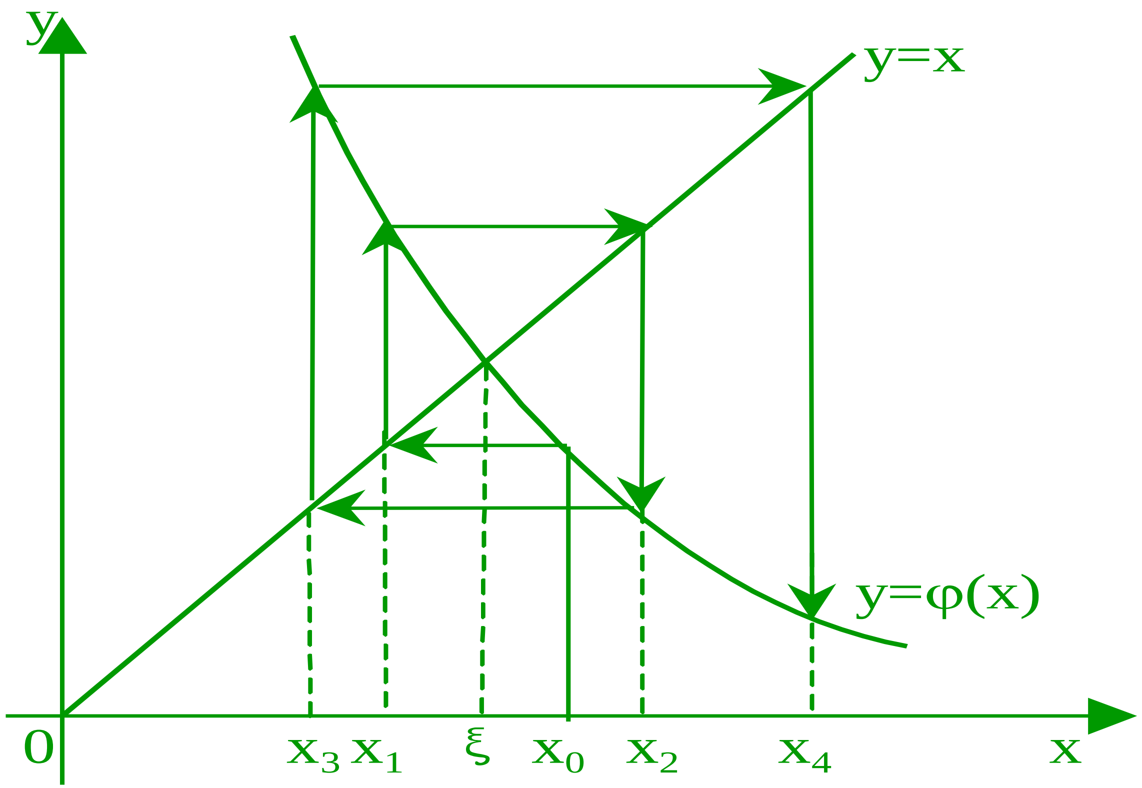




非线性方程的求解---不动点迭代法的几何意义



收敛



发散

非线性方程的求解---不动点迭代法

例1 求方程 $f(x)=x^3-x-1=0$ 在 $x_0=1.5$ 附近的根 x^*.

解析 设方程改写成下列形式

$$x=\sqrt[3]{x+1}$$

据此建立迭代公式

$$x_{k+1}=\sqrt[3]{x_k+1}\qquad (k=0,1,2,\cdots).$$

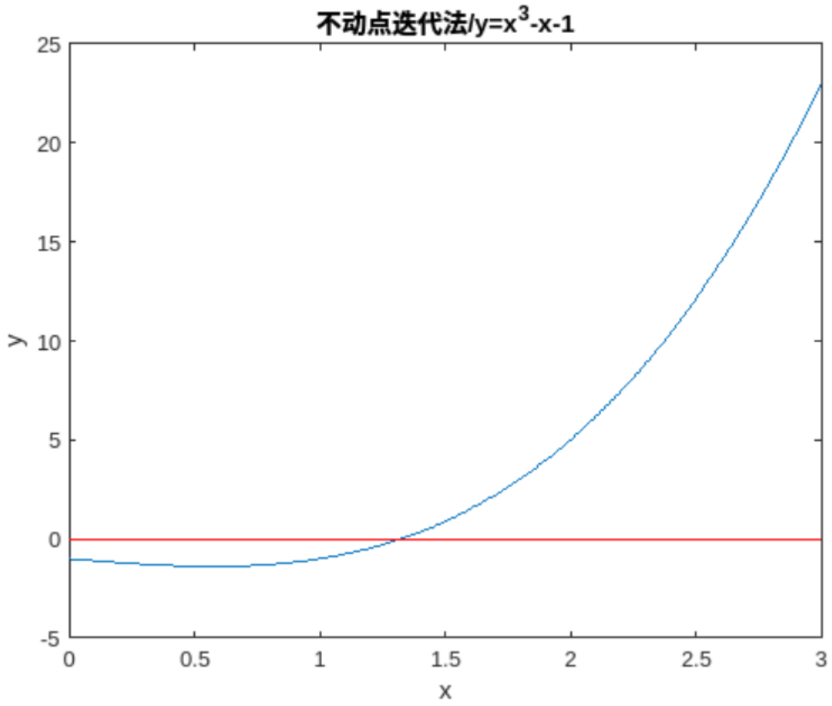
非线性方程的求解---不动点迭代法

```
clear
clc
% x=(x+1)^(1/3);
x=1.5; % 初始值
%f(x)=x.^3-x-1;
esp=1e-6; % 迭代终止条件
N=100; % 最大迭代次数
y=zeros(N, 1); % 暂存x变量的空间
for t=1:N
    x=fun(x);
    y(t)=x;
    fprintf('第 %d 次, x=%f\n', t, x);
    if t>1
        if abs(y(t)-y(t-1))<esp
            break;
        end
    end
end

% 画出函数曲线
xx=0:0.01:3;
yy=xx.^3-xx-1;
figure(1)
plot(xx, real(yy));
hold on
z=0*ones(1, length(xx));
plot(xx, z, 'r');
xlabel('x');
ylabel('y');
title('不动点迭代法/y=x^3-x-1');
%saveas(gcf,sprintf('不动点迭代法.jpg'),'bmp');
```

```
function x=fun(x)
x=(x+1).^(1./3);   %% x的迭代函数
end
```

```
第 1 次, x=1.357209
第 2 次, x=1.330861
第 3 次, x=1.325884
第 4 次, x=1.324939
第 5 次, x=1.324760
第 6 次, x=1.324726
第 7 次, x=1.324719
第 8 次, x=1.324718
第 9 次, x=1.324718
```



例 2 求方程 $f(x)=x^3+10x-20=0$

在$x_0=1.5$附近的根x^*,要求精确到六位小数。

解析 设方程分别改写成下列形式

(1) $x=x^3+11x-20$ (2) $x=\dfrac{20}{x^2+10}$

据此建立迭代公式

(1) $x_{n+1}=x_n^3+11x_n-20$ $(n=0,1,2,\cdots).$

(2) $x_{n+1}=\dfrac{20}{x_n^2+10}$ 结果有:

非线性方程的求解---迭代法的收敛性问题

(1) $x_{n+1} = x_n^3 + 11x_n - 20$

```
第 1 次, x=-0.125000
第 2 次, x=-21.376953
第 3 次, x=-10023.860932
第 4 次, x=-1007175483753.888428
第 5 次, x=-1021681283411173471400728984459149312.000000
第 6 次, x=-10664642762800224311930821814442358810284982131153690543584527505815448723
第 7 次, x=-Inf
第 8 次, x=-Inf
第 9 次, x=-Inf
第 10 次, x=-Inf
第 11 次, x=-Inf
第 12 次, x=-Inf
第 13 次, x=-Inf
第 14 次, x=-Inf
第 15 次, x=-Inf
第 16 次, x=-Inf
第 17 次, x=-Inf
```

发散

(2) $x_{n+1} = \dfrac{20}{x_n^2 + 10}$

```
第 1 次, x=1.632653
第 2 次, x=1.579086
第 3 次, x=1.600831
第 4 次, x=1.592020
第 5 次, x=1.595593
第 6 次, x=1.594144
第 7 次, x=1.594732
第 8 次, x=1.594493
第 9 次, x=1.594590
```

收敛

非线性方程的求解---迭代法的收敛性问题

压缩映象原理 1 设$\varphi(x)$在$[a,b]$满足以下条件:

1° 对任意$x \in [a,b]$有$a \le \varphi(x) \le b$.

2° 存在常数$0 < L < 1$，使对$\forall$，$x, y \in [a,b]$都有

$$|\varphi(x) - \varphi(y)| \le L|x - y|$$

则$x = \varphi(x)$在$[a,b]$上存在唯一的根x^*.

且对任意的$x_0 \in [a,b]$, $x_{n+1} = \varphi(x_n) \longrightarrow x^*$,并有

$$|x_n - x^*| \le \frac{L^n}{1-L}|x_1 - x_0|$$

解析 （一、证明存在惟一性）

由于

$$|\varphi(x)-\varphi(y)| \le L|x-y| \longrightarrow \varphi(x) \in C^0[a,b]$$

$$a \le \varphi(x) \le b \longrightarrow \begin{cases} \psi(a) = a-\varphi(a) \le 0 \\ \psi(b) = b-\varphi(b) \ge 0 \end{cases}$$

所以：$\psi(x) = x-\varphi(x) = 0$在$[a,b]$内有根

又 $\left|x_2{}^*-x_1{}^*\right| = \left|\varphi(x_2{}^*)-\varphi(x_1{}^*)\right| \le L\left|x_2{}^*-x_1{}^*\right| \longrightarrow x_2{}^* = x_1{}^*$

非线性方程的求解---迭代法的收敛性问题

（二、证明收敛与初值的无关性）

$\forall x_0 \in [a,b]$

又 $|x_n - x^*| = |\varphi(x_{n-1}) - \varphi(x^*)| \le L|x_{n-1} - x^*| \le \ldots$

$$\ldots \le L^n |x_0 - x^*| \xrightarrow{0<L<1} 0$$

$$\therefore \quad \lim_{n\to\infty} x_n = x^*$$

非线性方程的求解---迭代法的收敛性问题

三、证明：对$\forall$的$x_0 \in [a,b]$,有：$\left|x_n - x^*\right| \le \dfrac{L^n}{1-L}\left|x_1 - x_0\right|$

$$\left|x_{n+p} - x_n\right| \le \left|x_{n+p} - x_{n+p-1}\right| + \left|x_{n+p-1} - x_{n+p-2}\right| + \ldots + \left|x_{n+1} - x_n\right|$$

$$\le L^{p-1}\left|x_{n+1} - x_n\right| + L^{p-2}\left|x_{n+1} - x_n\right| + \ldots + L^2\left|x_{n+1} - x_n\right| + \left|x_{n+1} - x_n\right|$$

$$= (L^{p-1} + L^{p-2} + \cdots + 1)\left|x_{n+1} - x_n\right|$$

$$= \frac{1-L^p}{1-L}\left|x_{n+1} - x_n\right| \le \frac{1-L^p}{1-L} L^n \left|x_1 - x_0\right|$$

$$\xrightarrow{p\to\infty} \frac{L^n}{1-L}\left|x_1 - x_0\right|$$

非线性方程的求解---迭代法的收敛性问题

压缩映象原理 2

设$\varphi(x)$在$[a,b]$满足以下条件：

1° 对任意$x \in [a,b]$有$a \le \varphi(x) \le b$.

2° 存在常数$0 < L < 1$，使对$\forall$，$x, y \in [a,b]$都有

$$|\varphi'(x)| \le L$$

则$x = \varphi(x)$在$[a,b]$上存在唯一的根x^*.

且对任意的$x_0 \in [a,b]$, $x_{n+1} = \varphi(x_n) \longrightarrow x^*$,并有

$$|x_n - x^*| \le \frac{L^n}{1-L}|x_1 - x_0|$$

非线性方程的求解---迭代法的收敛性问题

只要证明: $\left|\varphi'(x)\right| \le L \Rightarrow \left|\varphi(x)-\varphi(y)\right| \le L\left|x-y\right|$

显然 $\forall x, y \in [a,b]$

$$\left|\varphi(x)-\varphi(y)\right| \xrightarrow{\text{中值定理}} \left|\varphi'(\xi)\right|\left|x-y\right| \le L\left|x-y\right|$$

例 2 求方程 $f(x)=x^3+10x-20=0$

在$x_0=1.5$附近的根x^*,要求精确到六位小数。

解析 设方程分别改写成下列形式

$$(1)\quad x=x^3+11x-20\xrightarrow{\Delta}\varphi_1(x)\qquad (2)\quad x=\frac{20}{x^2+10}\xrightarrow{\Delta}\varphi_2(x)$$

$$(1)\ |\varphi_1'(x)|=|3x^2+11|>1$$

所以迭代法发散

$$(2)\quad x=\frac{20}{x^2+10}\xrightarrow{\Delta}|\varphi_2'(x)|=\left|-\frac{40x}{(x^2+10)^2}\right|<1\qquad 1\le\varphi(x)=\frac{20}{x^2+10}\le 2$$

所以迭代法收敛

非线性方程的求解---迭代法的收敛性问题

例1 求方程 $f(x)=x^3-x-1=0$ 在 $x_0=1.5$ 附近的根 x^*.

解：$\varphi_1(x)=\sqrt[3]{x+1},\ \varphi_1'(x)=\frac{1}{3}(x+1)^{-2/3}$

在区间[1,2]中，$|\varphi_1'(x)|\le\frac{1}{3}(\frac{1}{4})^{1/3}<1,$

又因 $1\le\sqrt[3]{2}\le\varphi_1(x)\le\sqrt[3]{3}\le 2,$

所以迭代法收敛.

而当 $\varphi_2(x)=x^3-1$ 时，$\varphi_2'(x)=3x^2$ 在区间[1,2]中,

$$|\varphi_2'(x)|>1$$

所以迭代法发散.

非线性方程的求解---迭代法的收敛性问题

例 3　用不同方法求方程$x^2-3=0$的根$x^*=\sqrt{3}$.

解析

这里$f(x)=x^2-3$,可改写成各种不同的等价形式$x=\varphi(x)$,其中不动点为$x^*=\sqrt{3}$.由此构造不同的迭代法:

(1) $x_{k+1}=x_k^2+x_k-3, \varphi(x)=x^2+x-3, \varphi'(x)=2x+1,$

$$\varphi'(x^*)=\varphi'(\sqrt{3})=2\sqrt{3}+1>1.$$

非线性方程的求解---迭代法的收敛性问题

例 3 用不同方法求方程$x^2-3=0$的根$x^*=\sqrt{3}$.

(2) $x_{k+1}=\dfrac{3}{x_k}, \varphi(x)=\dfrac{3}{x},$ $\varphi'(x^*)=-1$

(3) $x_{k+1}=x_k-\dfrac{1}{4}(x_k^2-3), \varphi(x)=x-\dfrac{1}{4}(x^2-3),$

$$\varphi'(x)=1-\frac{1}{2}x,$$

$$\varphi'(x^*)=1-\frac{\sqrt{3}}{2}\approx 0.134<1.$$

(4) $x_{k+1}=\dfrac{1}{2}(x_k+\dfrac{3}{x_k}), \varphi(x)=\dfrac{1}{2}(x+\dfrac{3}{x}),$

$$\varphi'(x)=\frac{1}{2}(1-\frac{1}{x^2}), \varphi'(x^*)=\varphi'(\sqrt{3})=0.$$

取$x_0=2$，对上述4种迭代法，计算三步所得的结果如下:

用不同方法求方程$x^2-3=0$的根$x^*=\sqrt{3}$.

方法一

$$x_{k+1}=x_k^2+x_k-3, \varphi(x)=x^2+x-3, \varphi'(x)=2x+1,$$
$$\varphi'(x^*)=\varphi'(\sqrt{3})=2\sqrt{3}+1>1.$$

1.5
2.
1.5
2.
1.5

发散

方法二

$$x_{k+1}=\frac{3}{x_k}, \varphi(x)=\frac{3}{x}, \qquad \varphi'(x^*)=-1$$

3.
9.
87.
7653.
5.85761×10^7

发散

方法三

$$x_{k+1}=x_k-\frac{1}{4}(x_k^2-3), \varphi(x)=x-\frac{1}{4}(x^2-3),$$
$$\varphi'(x)=1-\frac{1}{2}x, \quad \varphi'(x^*)=1-\frac{\sqrt{3}}{2}\approx 0.134<1.$$

1.75
1.73437
1.73236
1.73209
1.73206

收敛

用不同方法求方程$x^2-3=0$的根$x^*=\sqrt{3}$.

方法四

$$x_{k+1}=\frac{1}{2}(x_k+\frac{3}{x_k}),\varphi(x)=\frac{1}{2}(x+\frac{3}{x}),$$

$$\varphi'(x)=\frac{1}{2}(1-\frac{3}{x^2}),\varphi'(x^*)=\varphi'(\sqrt{3})=0.$$

1.75
1.73437
1.73236
1.73209
1.73205

收敛

非线性方程的求解---迭代法的收敛性问题

例 3　用不动点迭代法计算sqrt（2）

迭代函数

/MATLAB Drive/sqrt2_budong.m

```
clear
clear all;
clc;
f=@(x) x.^2-2;
% x=(x+1)^(1/3);
a=1.5; % 初始值
esp=1e-6; % 迭代终止条件
N=100; % 最大迭代次数
y=zeros(N, 1); % 暂存x变量的空间
for t=1:N
    a=fun(a);
    y(t)=a;
    fprintf('第 %d 次, x=%f\n', t, a);
    if t>1
        if abs(y(t)-y(t-1))<esp
            break;
        end
    end
end

fprintf('经过 %d 次迭代, 函数方程根的近似解为:\n a=%.16f\n',t,a)%输出迭代过程

% 画出函数曲线
xx=0:0.01:3;
yy=xx.^2-2;
figure(1)
plot(xx, real(yy));
hold on
z=0*ones(1, length(xx));
plot(xx, z, 'r');
xlabel('x');
ylabel('y');
title('不动点迭代法/y=x^2-2');
%saveas(gcf,sprintf('不动点迭代法.jpg'),'bmp');

function x=fun(x)
x=(x+2./x)/2;  %% x的迭代函数
%x=x-(x.^2-2)/3;
end
```

命令行窗口

```
第 1 次, x=1.416667
第 2 次, x=1.414216
第 3 次, x=1.414214
第 4 次, x=1.414214
经过 4 次迭代, 函数方程根的近似解为:
 a=1.4142135623730949
>> a

a =

    1.4142

>>
```

非线性方程的求解---迭代法的收敛性问题

例 3 用不动点迭代法计算sqrt（2）

迭代函数2

```
/MATLAB Drive/sqrt2_budong.m
clear
clear all;
clc;
f=@(x) x.^2-2;
% x=(x+1)^(1/3);
a=1.5; % 初始值
esp=1e-6; % 迭代终止条件
N=100; % 最大迭代次数
y=zeros(N, 1); % 暂存x变量的空间
for t=1:N
    a=fun(a);
    y(t)=a;
    fprintf('第 %d 次, x=%f\n', t, a);
    if t>1
        if abs(y(t)-y(t-1))<esp
            break;
        end
    end
end

fprintf('经过 %d 次迭代，函数方程根的近似解为:\n a=%.16f\n',t,a)%输出迭代过程

% 画出函数曲线
xx=0:0.01:3;
yy=xx.^2-2;
figure(1)
plot(xx, real(yy));
hold on
z=0*ones(1, length(xx));
plot(xx, z, 'r');
xlabel('x');
ylabel('y');
title('不动点迭代法/y=x^2-2');
%saveas(gcf,sprintf('不动点迭代法.jpg'),'bmp');

function x=fun(x)
%x=(x+2./x)/2;  %% x的迭代函数
x=x-(x.^2-2)/3;
end
```

命令行窗口

```
第 1 次, x=1.416667
第 2 次, x=1.414352
第 3 次, x=1.414221
第 4 次, x=1.414214
第 5 次, x=1.414214
经过 5 次迭代，函数方程根的近似解为:
 a=1.4142135882194871
>> a

a =

    1.4142

>>
```

非线性方程的求解---总结

- 牛顿二分法是实现单实变量非线性方程零点可靠的求解方法，但存在收敛速度慢和对有偶数重根不适用等问题。
- 不动点迭代法是逐次逼近法，但是迭代函数的选择对求解结果的收敛性和求解速率有较大影响。
- 为了保证迭代过程收敛，要求迭代函数的导数满足条件 $|\varphi'(x)| < 1$

谢谢大家！

肖　明

微电子科学与技术学院

邮箱：xiaom37@mail.sysu.edu.cn

电话：17817227706

办公室：中山大学珠海校区公共实验楼2楼A214室