非线性方程几种数值解法的Matlab程序

吴专保[1,2]

(1.湖南师范大学 数学与计算机科学学院,长沙 410081;2.岳阳职业技术学院 公共课部, 湖南 岳阳 414000)

摘 要:为研究非线性方程数值解，给出了二分法、简单迭代法和牛顿迭代法的Matlab程序，并进行了近似计算。结果表明，牛顿迭代法收敛最快。

关键词:非线性方程;Matlab程序;二分法;迭代法
中图分类号:O241.6 文献标识码:A 文章编号:1671 - 2153(2007)02 - 0020 - 03

0 引 言

在科学研究和科学计算中常常碰到非线性方程求解问题。非线性方程的解一般不能解析求出。所以数值解法显得非常重要,而数值解法在实际中的实现则更为重要。本文将介绍几种数值解法在Matlab中的实现程序。

1 二分法

设$f(x)$在$[a,b]$连续,假定$f(a)<0$,$f(b)>0$,取中点$x_0=\frac{a+b}{2}$,检查$f(x_0)$符号。若$f(x_0)=0$,则x_0就是一个根;若$f(x_0)>0$,记a为a_1,x_0为b_1,则得有根区间$[a_1,b_1]$;若$f(x_0)<0$,记x_0为a_1,b为b_1,则得有根区间$[a_1,b_1]$。后两种情况都得到有根区间$[a_1,b_1]$,它的长度为原区间的一半。对$[a_1,b_1]$,令$x_1=\frac{a_1+b_1}{2}$,再用同样的方法,可得新的有根区间$[a_2,b_2]$,它的长度为$[a_1,b_1]$的一半,如此反复进行下去,其中每一个区间是前一区间的一半。有$\lim\limits_{x\to\infty} x_n=\lim\limits_{x\to\infty}\frac{a_n+b_n}{2}=x^*$,这就是方程的根。而$x_n=\frac{a_n+b_n}{2}$即为方程的近似根,且有估计误差

$$|x_n-x^*|\le\frac{b-a}{2^{n+1}} \tag{1}$$

下面用二分法求

$$f(x)=x^3+4x^2-10 \tag{2}$$

在区间[1,2]上的根.因为二分法只能求单根,首先可以搜索函数(2.2)在区间[1,2]的根的情况。在Matlab命令窗中输入[2]:

```
x=1：0.01：2;
y=x^3+4*x^2-10;
plot(x,y)
```

图1为$f(x)$在区间[1,2]的图像。由图1可以看出,式(2)在区间[1,2]间有惟一的一个大于1.35而小于1.4的单根。

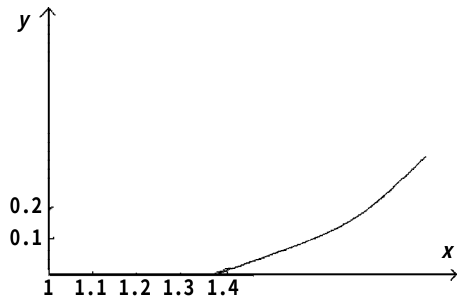

图1 $f(x)$在区间[1,2]的图像

●Matlab实现

先建立erfen.m

```
function y=erfen(fun,a,b,esp)
```

收稿日期:2006-12-25
基金项目:国家自然科学基金资助项目(10271046)
作者简介:吴专保(1973-)，男，湖南岳阳人，讲师，在读硕士研究生,研究方向为计算数学。

© 1994-2007 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

```
if nargin<4 esp=1e-4;end
if feval (fun,a)*feval (fun,b)<0
    n=1;
    c=(a+b)/2;
    while c>esp
       if feval (fun,a)*feval(fun,c)<0
             b=c;c=(a+b)/2;
       elseif feval (fun,c)*feval (fun,b)<0
             a=c;c=(a+b)/2;
 else y=c;esp=10000;
          end
          n=n+1;
      end
 format long
      y=c;
 elseif feval (fun,a)==0
      y=a;
   elseif feval (fun,b)==0
        y=b;
   else disp ('may not be a root');
   end
   n
```

再建立 fc.m 函数文件

```
function y=fc(x);
y= x^3+4*x^2-10;
```

最后在Matlab命令窗中输入:

```
erfen('fc',1,2,0.0001)
 n=
   50
 ans=
   1.36523001341410
```

即迭代50次计算结果为1.36523001341410,精度已经非常高。如果希望精度达到10^{-4},只需要迭代13次结果是1.365173390。二分法优点是计算简单,收敛性有保证;缺点是收敛不快,不能求复根及重根。

2 简单迭代法

将函数$f(x)=0$非线性方程变为同解方程:$x=\phi(x)$,其中函数$\phi(x)$为连续函数。然后构造迭代格式

$$x_{k+1}=\phi(x_k) \tag{3}$$

并选取初值x_0。根据初值x_0及迭代格式(3)可产生迭代序列$\{x_k\}$[3]。

下面用简单迭代法计算式(2)。由式(2)可以得到下列等价方程

$x=\phi_1(x)=x-f(x)=x-x^3-4x^2+10$,　　$x=\phi_2(x)=\left(\frac{10}{x}-4x\right)^{\frac{1}{2}}$

$x=\phi_3(x)=\frac{1}{2}\left(10-x^3\right)^{\frac{1}{2}}$,　　$x=\phi_4(x)=\left(\frac{10}{x+4}\right)^{\frac{1}{2}}$

但第一个方程是发散的,第二个方程不适定,只有后面两个方程具有收敛性。用第4个方程计算如下。

● MATLAB 实现

先建立iterate.m

```
function y=iterate (x)
x1=g(x);
    n=1;
    while (abs(x1-x)>=1.0e-6)&(n<=1000)
      x=x1;
        x1=g(x);n=n+1;
end
format long
x1
n
```

再建立g.m函数文件

```
function y=g(x);
y=(10/(x+4))^(1/2) ;
```

最后在Matlab命令窗中输入:

若设初值为1.5,则

```
iterate(1.5)
x1=
1.36522559416052
n =
  5
```

即迭代5次达到精度10^{-4},计算结果为1.36522559416052。

3 牛顿迭代法

对于方程$f(x)=0$,若已知根x^*的一个近似值x_k,可将$f(x)$在x_k处展成一阶泰勒公式并取其线性部分近似,即用线性方程

$$f(x_k)+f'(x_k)(x-x_k)=0 \tag{4}$$

近似。若$f'(x_k)\neq 0$,方程(4)的根记作x_{k+1},则得

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)} \quad (k=0,1,\cdots) \tag{5}$$

这就是牛顿法的迭代程序[4,5]。下面用牛顿迭代法计算式(2)。

● Matlab 实现

先建立 newton.m

```
function y=newton (x0)
x1=x0-fs (x0)/df (x0);
n=1;
```


© 1994-2007 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

```
while (abs (x1-x0)>=1.0e-6)&(n<=100000000)
     x0=x1;
    x1=x0-fs (x0)/df (x0);n=n+1;
end
x1
n
```

建立 fs.m 函数文件

```
function y=fs (x)
y=x^3+4x^2-10;
```

建立 df.m 函数文件

```
function y=df (x)
y=3*x^2+8*x;
```

最后在 Matlab 命令窗中输入:

若设初值为 1.5，则

```
Newton (1.5)
n =
    3
ans=
    1.36526201487463
```

即迭代3次达到精度10^{-4},计算结果为1.36526201487463。将上述 3 种方法的每步结果列表，如表 1 所示。从表1可以看出,牛顿迭代法收敛最快．简单迭代法次之，二分法最慢。

4 结束语

计算数学的第 3 个问题是在计算机中实现，目前这个问题显得非常迫切。本文编写了 3 种非线性方程数值解法的程序。当然,如果想看到每一次的结果,主程序需改动一下,或在命令窗口依次输入迭代格式。并且每次计算的时间还受计算机的配置所影响。

表1 $f(x)$在区间[1,2]上的数值解

迭代次数	二分法结果	简单迭代法结果	牛顿迭代法结果
0	1.5	1.5	1.5
1	1.25	1.348399725	1.37333333333333
2	1.375	1.367376372	1.36526201487463
3	1.3125	1.364957015	
4	1.34375	1.365264748	
5	1.359375	1.36522559416052	
6	1.3671875		
7	1.36328125		
8	1.365234375		
9	1.364257813		
10	1.364746094		
11	1.364990235		
12	1.365112305		
13	1.365173390		

参考文献:

[1]曾金平,李郴良.数值计算方法[M].湖南:湖南大学出版社,2004.

[2]王沫然.Matlab与科学计算[M].北京:电子工业出版社,2003.

[3]李庆扬,关冶,白峰衫.数值计算原理[M].北京:清华大学出版社,2000.

[4]RICHARD L B, DOUGLAS FAIRES J.Numerical Analysis[M].北京:高等教育出版社,2001.

[5]JOHN D S,JERRY B K. Elementary Numerical Computing with Mathematica[M].New York: McGrew-Hill,Inc,1993.

Procedures of several methods for the numerical solution of the nonlinear equation in Matlab

WU Zhuan-bao[1,2]

(1. Department of Mathematic, Mathematics and Computer Science College Hunan Normal University , Changsha 410081,China;2. Department of Basic Courses, Yueyang Higher Vocational and Technical College,Yueyang 414000,China)

Abstract: To study numerical solution of the nonlinear equation, the article presents the procedures of the bisection method, the simple iterate method and the Newton iterate method in Matlab,and has approximately calculated. The Newton iterate method is the fastest.

Key words: nonlinear equation; procedures in Matlab; bisection method; iterate method

© 1994-2007 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net