牛顿迭代法在计算机中的实现

周晓庆*

摘 要 详细讨论了如何通过 C 语言编程进行牛顿迭代法的求解,并给出了相关的程序代码。

关键词 牛顿迭代法 C 语言 程序

中图分类号 TP311 **文献标识码** A **文章编号**:1002-2422(2006)05-0034-02

The Implementation of Newton Iteration Method in the Computer

Zhou Xiaoqing

Abstract The paper discusses in detail how to solve the nonlinear equations by making the C language program,and gives the relevant procedure code.

Keyword Newton Iteration Method C Language Program

牛顿迭代法实际上是用近似方程代替原方程去求根,它把非线性方程线性化,以线性方程的解逐步逼近非线性方程的解,本文讨论了怎样用 C 语言编制程序,并通过计算机来求解该类方程的解,下面就对这一程序进行详细讨论。

1 牛顿迭代法简介

解非线性方程 $f(x)=0$ 的牛顿法是把非线性方程线性化的一种近似方法。把 $f(x)$ 在 x0 点附近展开成泰勒级数:

$$f(x)=f(x_0)+(x-x_0)f'(x_0)+(x-x_0)^2\frac{f''(x_0)}{2!}+\cdots$$

取其线性部分,作为非线性方程 $f(x)=0$ 的近似方程,则有:

$$f(x_0)+f'(x_0)(x-x_0)=0$$

设 $f'(x_0)\neq 0$ 则其解为:

$$x_1=x_0-f(x_0)/f'(x_0)$$

再把 $f(x)$ 在 x_1 附近展开成泰勒级数,也取其线性部分作 $f(x)=0$ 的近似方程。若 $f(x_1)\neq 0$,则得:

$$x_2=x_1-f(x_1)/f'(x_1)$$

这样,得到牛顿法的一个迭代序列:

$$x_{n+1}=x_n-f(x_n)/f'(x_n)$$

牛顿迭代法的收敛速度:如果 $f(x)$ 在零点附近存在连续的二阶微商,ξ 是 $f(x)$ 的一个重零点,且初始值 x_0 充分接近于 ξ,那么牛顿迭代是收敛的,其收敛速度是二阶的,即平方收敛速度。

2 程序功能说明

本程序采用牛顿法,求实系数高次代数方程

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

在初始值 x_0 附近的一个根。

(1)程序中调用 qiugf 和 qiudf 函数,其中 qiugf 是待求根的实数代数方程的函数,qiudf 是方程一阶导数的函数。

(2)参数说明

a 一维实数组,输入参数,按升幂存放方程系数。

n 整变量,输入参数,方程阶数。

x0 实变量,输入参数,初始迭代值。

nn 整变量,输入参数,允许的最大迭代次数。

eps1 实变量,输入参数,控制根的精度。

3 程 序 代 码

```
# include <stdio.h>
# include <math.h>
double qiugf(int a[ ],int n,double x);
double qiudf(int a[ ],int n,double x);
main()
{int b,n,nn,i,a[100],x[100];
double x0,eps1,y;
printf("输入方程的阶 n:");
scanf("%d",&n);
printf("输入初始迭代值 x0:");
scanf("%f",&x0);
printf("输入控制根精度的值 eps1:");
scanf("%f",&eps1);
printf("输入最大迭代次数值 nn:");
scanf("%d",&nn);
printf("请依次输入方程的系数:");
for(i=0;i<n;i++)
{
scanf("%d",&a[i]);
```

* 周晓庆 西华师范大学计算中心讲师(四川南充 637002),研究方向:管理信息系统,计算机网络 收稿日期:2006-03-11

```
}
x[ 1] =x0 ;
b=1 ;
i=1 ;
while( abs( b) >eps1*x[ i] )
{
i=i+1 ;
x[ i] =x[ i- 1] - qiug( a,n,x[ i- 1] ) /qiud( a,n,x[ i- 1] ) ;
b=x[ i] - x[ i- 1] ;
if( i>nn) error( "nn is full") ;
return ;
}
y=x[ i] ;
printf( "该方程的解为%f :",y) ;
printf( "该次迭代运算的最终迭代次数为 &d :",i) ; } }
double qiug( int a[ ] ,int n,double x)
{
int i ;
double y=0.0 ;
for( i=1 ;i<n+1 ; i++)
{
y=y+a[ i] *x( n+1- i) ;
return y ;}
{
double qiud( int a[ ] ,int n,double x)
{
int i ;
double y=0.0 ;
for(i=1; i<n; i++)
{
y=y+a[ i] *( n+1- i) *x[ n- i] ;
return y ;
}
}
```

4 实 例 分 析

用牛顿法求下面方程的根 :(x) $=x^3+2x^2+10x-20$

运行结果 :

输入方程的阶 n :3

输入初始迭代值 x0 :1

输入最大迭代次数值 nn :1000

输入控制根精度的值 eps1 :1e- 8

请依次输入方程的系数 :

1 2 10 - 20

该方程的解为 :1.368808107821373

该次迭代运算的最终迭代次数为 :6

5 结 束 语

牛顿迭代法虽然简单,但是也有局限性。在牛顿迭代法中,选取适当迭代初始值是求解的前题,当迭代的初始值 x_0 在某根的附近时迭代才能收敛到这个根，有时会发生从一个根附近跳向另一个根附近的情况,尤其在导数 $f'(x_0)$ 数值很小时。如图 1 所示。

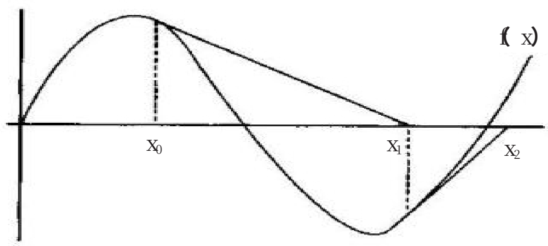

图 1 牛顿迭代法中根的跳跃情况

因此,我们也可以用对分法、逐次迭代法、双点割线法、单点割线法等进行迭代运算，这些方法也都可以通过编程来进行计算,不过,它们都各有自己的优缺点,我们这里就不一一讨论了。

参 考 文 献

[1] 张世禄. 计算方法[M] . 成都 :电子科技大学出版社,1999.

[2] 张培强. MATLAB 语言- - - 演算纸式的科学工程语言[M] . 北京 :中国科技大学出版社,2003.

[3] 张纪元,沈守范. 计算机构学[M] . 北京 :国防工业出版社,2004.