Modified Bisection Algorithm for multiple roots of nonlinear equation with the R software

Kouider Mohammed Ridha

Department of Mathematics, Applied Mathematics Laboratory , University of Mohamed Khider, Biskra, Algeria,
mohakouider@gmail.com

Abstract

The most basic problem in numerical analysis (methods) is the root finding problem. In this paper we are interesting in represent numerical method which is the Modified Bisection Algorithm(MBA) referred to by Tanakan, (2013, [9]) for finding the multi-roots of a function. Hence in this study we programming the MBA for multi-roots with the R software.

Keywords: Bisection method , Kouider method, Modified Bisection Algorithm , The R software version i386 3.3.3.

1. Introduction

A modified bisection algorithms is much more efficient than the bisection method. Furthermore, it is faster than the Newton method, and don not count the derivative of a function at the reference point, which is not always easy. In the practice, the initial solution is really important for the Newton method. But some initial solutions can make the method Newton diverges. Hence, by the intermediate value theorem a modified bisection algorithms, it works when f is a continuous function on $[\alpha;\beta]$ where $(\alpha,\beta)\in\mathbb{R}^2$ with the initial condition $f(\alpha)f(\beta)<0$ existing with each iteration.

A modified bisection algorithms can reduced the number of iterations which less than the iteration number of the bisection method and nearby to the iteration number of Newton method, for numerical results see Tanakan ([13], section[9]) .

2.Algorithm for a new Modified Bisection Method for multi-roots

In this section we provide an algorithm that relies on the method of a new MBM which finding the roots of function. In this work took the error are less than the tolerance, which is linked to condition $\left|\theta_{n+1}-\theta_n\right|$ less than the tolerance because we computed numerically on the space B.

We can state an alternative modified bisection algorithms for multi-roots as the following ,

- Step1: Divide the interval into several intervals by $h=(b-a)/t$ where $t\in\mathbb{N}$ is the number of intervals for example $t=10$.
- Step2: For $i=1,\ldots,10$, and tolerance 1×10^{-7} ; and $\alpha_1=\alpha_i$, $\beta_1=\beta_{i+1}$ where $f(\alpha_i)f(\beta_{i+1})<0$.
- Step3: For $n\geq1$, compute $\theta_n^*=(\alpha_n+\beta_n)/2$.
- Step4: Compute for a sub-interval $\left[\alpha_n^*;\beta_n^*\right]$ by

$$\left[\alpha_n^*;\beta_n^*\right]=\begin{cases}\left[\alpha_n;\theta_n^*\right] & if \quad f(\alpha_n)f(\theta_n^*)<0\\ \left[\theta_n^*;\beta_n\right] & if \quad f(\theta_n^*)f(\beta_n)<0\end{cases}$$

- Step 5: Compute,

$$\theta_{n+1}=\beta_n^*-f\left(\beta_n^*\right)\frac{\beta_n^*-\alpha_n^*}{f\left(\beta_n^*\right)-f\left(\alpha_n^*\right)} \quad \text{or}$$

$$\theta_{n+1}=\alpha_n^*-f\left(\alpha_n^*\right)\frac{\beta_n^*-\alpha_n^*}{f\left(\beta_n^*\right)-f\left(\alpha_n^*\right)}$$

- Step 6: If $\left|\theta_{n+1}-\theta_n^*\right|<1\times10^{-7}$, then stop program (i.e $\theta_{n+1}=\theta_n^*$) the zero is θ_{n+1} . ELSE

$$\left[\alpha_{n+1}^*;\beta_{n+1}^*\right]=\begin{cases}\left[\alpha_n^*;\theta_{n+1}\right] & if \quad f(\alpha_n^*)f(\theta_{n+1})<0\\ \left[\theta_{n+1};\beta_n^*\right] & if \quad f(\theta_{n+1})f(\beta_n^*)<0\end{cases}$$

and set $n=n+1$, GOTO Step3.

To alert, this the Modified Bisection Algorithms is based on two method: Bisection method and Kouider method (MB). Hence ,we note that the MBA is the Kouider method see[6] with value sacoring precedents found by Bisection method θ_n^* for an account θ_{n+1} by Kouider method

2.1 A new MBM for multi-roots With The R Software

This algorithm is with the R software version i386 3.3.3. The R is an integrated suite of software facilities for data manipulation, calculation and graphical display. R is very much a vehicle for newly developing methods of interactive data analysis. It has developed rapidly, and has been extended by a large collection of packages, many people use R as a statistics system. There are about 25 packages supplied with R (called \standard" and \recommended" packages) and many more are available through the CRAN family of Internet sites (via http://CRAN.R-project.org) and elsewhere.

Next we present our algorithm for multi-roots with the R software as the following ,

```
KouiderRoot<-function(f,a,b,num=10,eps=1e-07)
{h=abs(b-a)/num
i=0
j=0
a1=b1=d=0
```

```
while(i<=num)
{  a1=a+i*h
b1=a1+h
if(f(a1)==0){print(a1)
       print(f(a1))}
if (f(b1)==0){print(b1)
      print(f(b1))}
else if (f(a1)*f(b1)<0) {
repeat{
    c=(a1+b1)/2
    if (f(a1)*f(c)<0){
      m= (a1-c)/(f(a1)-f(c))
      d=c-f(c)*m
      if(f(a1)*f(d)>0){
      if(f(c)*f(d)<0)
      a1<-c
      b1<-d }
      else b1<-d
      }
     else
     { m= (b1-c)/(f(b1)-f(c))
     d=c-f(c)*m
      if(f(b1)*f(d)>0){
      if(f(c)*f(d)<0)
      a1<-c
      b1<-d }
      else a1<-d
      }
   if(abs(c-d)<eps)
         break
}
print(j+1)
j=j+1
print(d)
print(f(d))
}
i=i+1
}
if(j==0)
print("finding root is fail")
else print("finding root is successful")
}
```

```
##########################
```

Example1: we go to finding the roots of $f(x)=x^2-2$ on the interval $[-2;2]$

```
f <- function(x){x^2-2*x+1}
KouiderRoot(f,-3,3)
###########################
> f <- function(x){x^2-2*x+1}
> KouiderRoot(f,-3,3)
[1] "finding root is fail"
##############################
```

Finding the root is fail because the initial condition is not satisfied i.e $f(-3)f(3)>0$

Example2: we go to finding the roots of $f(x)=x^2-2x$ on the interval $[0;2]$

```
f <- function(x){x^2-2*x}
KouiderRoot(f,0,2)
##########################
> f <- function(x){x^2-2*x}
> KouiderRoot(f,0,2)
[1] 0
[1] 0
[1] 2
[1] 0
[1] 2
[1] 0
[1] "finding root is fail"
############################
```

Finding the root is fail because the initial condition is not satisfied i.e $f(0)f(2)=0$ or we have the bounds interval is the roots.

Example3: we go to finding the roots of $f(x)=x^2-2$ on the interval $[-2;2]$

```
 f <- function(x){x^2-2}
KouiderRoot(f,-2,2)
##############################
>  f <- function(x){x^2-2}
> KouiderRoot(f,-2,2)
[1] 1
[1] -1.414214
[1] -4.440892e-16
[1] 2
[1] 1.414214
[1] -4.440892e-16
[1] "finding root is successful"
```

```
######################################
```

Example4: we go to finding the roots of $f(x)=x^3-2x+1$ on the interval $[-2;2]$

```
f <- function(x){x^3-2*x+1}
KouiderRoot(f,-2,2)
#####################################
> f <- function(x){x^3-2*x+1}
> KouiderRoot(f,-2,2)
[1] 1
[1] -1.618034
[1] 0
[1] 2
[1] 0.618034
[1] 1.110223e-16
[1] 3
[1] 1
[1] -4.440892e-16
[1] "finding root is successful"
#####################################
```

Example5:
we go to finding the roots of $f(x)=\exp(-x)+\cos(x)$ on the interval $[1;8]$

```
f <- function(x){exp(-x) +cos(x)}
KouiderRoot(f,1,8)
###############################
> f <- function(x){exp(-x) + cos(x)}
> KouiderRoot(f,1,8)
[1] 1
[1] 1.74614
[1] -1.110223e-16
[1] 2
[1] 4.703324
[1] -3.469447e-16
[1] 3
[1] 7.85437
[1] 9.703609e-18
[1] "finding root is successful"
################################
```

4. Conclusion

In this study, we represented the numerical method MBA of looking for the multi-roots of a zero equation with the R software. In order to apply the MBA for multi-roots finding, correctly on the interval $[\alpha;\beta]$ we have to present the initial condition, is the intermediate value theorem for all $x \in [\alpha;\beta]$,. More precisely, we know that the function is continuous, and it requires previous knowledge of two initial guesses α and β such that $f(\alpha)$ and $f(\beta)$ have opposite signs with out $f(\alpha)\neq 0$ and $f(\beta)\neq 0$ see the example 1 and the example 2.

Reference

[1] Autar, K.K and Egwu, E (2008), http://www.numericalmethods.eng.usf.edu Retrieved on 20th February, 2014.

[2] Ehiwario, J.C., Aghamie, S.O, “Comparative Study of Bisection, Newton-Raphson and Secant Methods of Root- Finding Problems”, IOSR Journal of Engineering Vol. 04, ||V1|| PP 01-07 , 2014

[3] Emmanuel Paradis, “R for Beginners”, Institut des Sciences de l’Evolution ´ Universit´e Montpellier II F-34095 Montpellier c´edex 05 France, 2005 https://cran.r-project.org/doc/contrib/Paradis-rdebuts_en.pdf.

[4] Jay, Villanueva, “Approximation of equations with a hand-held calculator”, Florida Memorial University Miami, FL, jvillanu@fmuniv.edu. http://archives.math.utk.edu/ICTCM/VOL23/C012/paper

[5] Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liquet, “The R Software: Fundamentals of Programming and Statistical Analysis”,. Springer. http://web.maths.unsw.edu.au/~lafaye/textes/bookR2013

[6] Kouider M.R, “A New Numerical Method for multi-roots finding with the R software”, 2019 https://papers.ssrn.com › sol3 › papers.cfm?abstract_id=3440013

[7] Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liquet, “The R Software: Fundamentals of Programming and Statistical Analysis”,. Springer. http://web.maths.unsw.edu.au/~lafaye/textes/bookR2013

[8] Srivastava,R.B and Srivastava, S, “Comparison of Numerical Rate of Convergence of Bisection, Newton and Secant Methods”, Journal of Chemical, Biological and Physical Sciences. Vol 2(1) pp 472- 479 , 2011.

[9] Tanakan ,S,“A New Algorithm of Modified Bisection Method for Nonlinear Equation”, Applied Mathematical Sciences 7.123, 2013, 6107-6114.

[10] Y. Cohen, J. Cohen ,“Statistics and Data with R: An Applied Approach Through Examples”,. Wiley, 2008