International Multidisciplinary Research Journal 2012, 2(1):01-04
ISSN: 2231-6302
Available Online: http://irjs.info/

INTERNATIONAL MULTIDISCIPLINARY
RESEARCH JOURNAL

Numerical accuracy of bisection method

R. B. Srivastava* and Narendra Deo Dixit

Department of Mathematics, M. L. K. P. G. College, Balrampur, U. P., India

Abstract
A computer program in C++ language has been developed to calculate square roots of numbers from 1 to 25 in interval [0, 6] using bisection method. Accuracy of bisection method has been found out in each calculation. Lowest accuracy has been observed in the calculation of square root of 1 in the interval [0, 6] and percentage error is equal to 0.000381469700. Highest accuracy has been observed in the evaluation of square root of 13 in the interval [0, 6] and percentage error is equal to 0.000000905160. Average percentage error of bisection method in the calculation of square roots of natural numbers from 1 to 25 has been found to be 0.000041549568 which indicates that the accuracy of bisection method can be increased by reducing tolerance value.

Keywords: Bisection method, numerical accuracy, percentage error, Intermediate-Value theorem, algorithm.

INTRODUCTION

Numerical analysis involves the study of methods of computing numerical data. Many problems across mathematics can be reduced to linear algebra, this too is studied numerically. Numerical solutions to differential equations require the determination not of a few numbers but of an entire function; in particular, convergence must be judged by some global criterion.[1-5]

The bisection method is the simplest and most robust algorithm for finding the root of a one-dimensional continuous function on a closed interval. The basic idea is a follows-

Suppose that $f(x)$ is a continuous function defined over an interval $[a; b]$ and $f(a)$ and $f(b)$ have opposite signs. Then by the intermediate value theorem, there exists at least one $r \varepsilon [a; b]$ such that $f(r) = 0$. The method is iterative and each iteration starts by breaking the current interval bracketing the root into two subintervals of equal length. One of the two subintervals must have endpoints of different signs. This subinterval becomes the new interval and the next iteration begins. Thus we can define smaller and smaller intervals such that each interval contains r by looking at subintervals of the current interval and choosing the one in which $f(x)$ changes signs. This process continues until the width of the interval containing a root shrinks below some predetermined error tolerance.[6-8] This method's major drawback is that it is the slowest among all the root finding methods. However, the method always converges to a solution and would be good to use as a starter for one of the other methods.

MATERIAL AND METHOD

The bisection method is developed with the support of the *Intermediate-Value Theorem*. Let us consider a continuous function $f(x)$ over the interval $[a, b]$. If
$f(a)f(b) < 0$

Then the values of $f(a)$ and $f(b)$ must be nonzero and each of different sign. That is, if $f(a)$ is negative, $f(b)$ must be positive.[9-13] Likewise, if $f(a)$ is positive, $f(b)$ must be negative. In light of the Intermediate-Value theorem, the intermediate-value that we seek is zero, which certainly lies between the positive and negative numbers represented by $f(a)$ and $f(b)$. The conclusion of the Intermediate-Value theorem is that there *is* a point, ξ , between a and b where $f(\xi)=0$. That is, the solution to $f(x)=0$ lies somewhere between a and b.

The bisection method checks the midpoint, c, of the interval $[a, b]$. If $f(c) = 0$, the task of finding a root is complete. If $f(c) \neq 0$, then for the same reason, there must be a root in either $[a, c]$ if $f(a)f(c) < 0$ or on $[c, b]$ if $f(c)f(b) < 0$.

This procedure defines the bisection method. At each step the interval in which there is guaranteed to be a root of the equation is halved (bisected), and the method terminates as soon as the width of the interval containing the root is less than some error tolerance $\delta > 0$. Since the bisection method keeps a bounded interval where there is at least one root at each step, it falls in the category of bracketing methods.[14-18]

An algorithmic definition of the bisection method is as follows [7]
Inputs: the function $f(x)$, the initial interval [a, b] and the stopping tolerance δ

```
if ( (f(a) * f(b) ≥ 0) then
begin
c=(a + b) / 2
while ( f(c) < δ )
Begin
c=(a + b)/2
if ( sign f(a)) = (sign f(c)) then a=c else b=c
end
return c;
end
else
write("Root does not exist")
```

Received: Sept 13, 2011; Revised: Dec 17, 2011; Accepted: Dec 27, 2011.

*Corresponding Author

R. B. Srivastava
Department of Mathematics
M. L. K. P. G. College, Balrampur, U. P., India

Tel: +91-9415036245; Fax:
Email: rambux@gmail.com

Computer program for bisection method developed by us is given below-

```
#include<conio.h>
#include<stdio.h>
#include<math.h>
//Bisection method
void main(void)
{
   FILE *fpt;
   int n;
   float a[1000],b[1000],c[1000],delta,rl,ru,d,aa;
   double f(float x);
   //avr is the variable whose square root is to be calculated
   double avr=25.0;
   clrscr();
   //Filename to store result
   fpt=fopen("nddb25.txt", "w");
   rl=0; ru=6.0; n=1; a[1]=rl; b[1]=ru; aa=fabs(rl-ru);
   //Value of function f(x)
   fprintf(fpt,"f(x)=x^2-25\n");
   fprintf(fpt,"rl= %6.2f\n",rl);
   fprintf(fpt,"ru= %6.2f\n",ru);
   //to check existence of root between the interval
   d=f(rl)*f(ru);
   delta=0.00001;
   fprintf(fpt,"    n      a[n]          b[n]         c[n]   f(c[n])\n");
   printf("    n          a[n]          b[n]         c[n]   f(c[n])\n");
   if (d<0)
   {
    while(aa > delta)
    {
      if (a[n]==b[n]) break;
      c[n]=(a[n]+b[n])/2;
      if (((f(c[n])>0) && (f(a[n])>0)) || ((f(c[n])<0) && (f(a[n])<0)))
      {
         a[n+1]=c[n];
         b[n+1]=b[n];
      }
      else
      {
         b[n+1]=c[n];
         a[n+1]=a[n];
      }
      aa=fabs(f(c[n]));
      fprintf(fpt,"%3d %15.12f %15.12f %15.12f %18.12f\n", n,a[n], b[n],
      c[n], f(c[n]));
      printf("%3d %15.12f %15.12f %15.12f %18.12f\n", n,a[n], b[n],
      c[n],f(c[n]));
      if (aa > delta)    n=n+1;
    }
      printf("Root= %20.15f\n",c[n]);
      printf("Value of function=%20.15f\n", f(c[n]));
      printf("No. of iterations=%3d\n",n);
      printf("Actual value of root=%15.12f\n",sqrt(avr));
      fprintf(fpt,"Actual value of root=%15.12f\n",sqrt(avr));
      printf("\n");
      getch();
     }
   else
     {
       printf("There is no root in the given interval\n");
       getch();
     }
   fclose(fpt);
 }
//Function definition
double f(float x)
{
   double r;
   r=x*x-25;
   return(r);
}
```

With the help of above computer program, square roots of the natural numbers from 1 to 25 in the interval [0, 6] have been calculated. For this, the following functions have been taken

$f(x) = x^2 - n = 0$ where n = 1, 2, 3, ……. , 25

Numerical accuracy of bisection method has been has been measured by percentage error and defined as follows–

Percentage error = error in the value of square root * 100/actual value of square root

Numerical accuracy of bisection method is inversely proportional to percentage error.

RESULT AND DISCUSSION

Square roots of natural numbers from 1 to 25 have been calculated using bisection method in the interval [0, 6] with stopping tolerance 0.00001.

Calculation of square root of 1 by bisection method

Bisection method has been applied to calculate the roots of equation

$f(x) = x^2 – 1 = 0$

In the interval [0, 6] using computer program developed by us. Initial value, last value, middle point of interval and value of function at middle point of the interval in each iteration is included in Table-1. Estimated value of square root of 1 by bisection method after each iteration is shown in Graph-1.

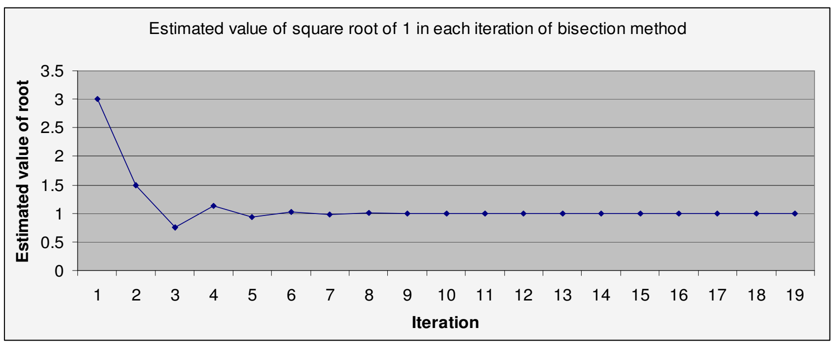
Table1. Initial value, last value, middle point of interval and value of function at middle point of the interval in the calculation of square root of 1 by bisection method

Number of iterations	Initial value (a)	Last value (b)	Middle point (c)	Value of function *f(x)* at x=c
1	0.000000000000	6.000000000000	3.000000000000	8.000000000000
2	0.000000000000	3.000000000000	1.500000000000	1.250000000000
3	0.000000000000	1.500000000000	0.750000000000	-0.437500000000
4	0.750000000000	1.500000000000	1.125000000000	0.265625000000
5	0.750000000000	1.125000000000	0.937500000000	-0.121093750000
6	0.937500000000	1.125000000000	1.031250000000	0.063476562500

7	0.937500000000	1.031250000000	0.984375000000	-0.031005859375
8	0.984375000000	1.031250000000	1.007812500000	0.015686035156
9	0.984375000000	1.007812500000	0.996093750000	-0.007797241211
10	0.996093750000	1.007812500000	1.001953125000	0.003910064697
11	0.996093750000	1.001953125000	0.999023437500	-0.001952171326
12	0.999023437500	1.001953125000	1.000488281250	0.000976800919
13	0.999023437500	1.000488281250	0.999755859375	-0.000488221645
14	0.999755859375	1.000488281250	1.000122070312	0.000244155526
15	0.999755859375	1.000122070312	0.999938964844	-0.000122066587
16	0.999938964844	1.000122070312	1.000030517578	0.000061036088
17	0.999938964844	1.000030517578	0.999984741211	-0.000030517345
18	0.999984741211	1.000030517578	1.000007629395	0.000015258847
19	0.999984741211	1.000007629395	0.999996185303	-0.000007629380

Actual value of square root of 1	1.000000000000
Calculated value of square root of 1 by bisection method	0.999996185303
Difference between actual and calculated values of square root of 1	0.000003814697
Percentage error in the value of square root of 1 calculated by bisection method	0.000381469700

Graph 1. Estimated value of square root of 1 by bisection method after each iteration



Similarly, the square roots of natural numbers from 2 to 25 have been calculated with the help of computer program to draw the appropriate conclusion.

CONCLUSION

Lowest percentage error has been obtained in the calculation of square root of 13 in the interval [0, 6] and is equal to 0.000000905160. It means if roots lies in the middle of the interval then the error in the calculated value of root of equation by bisection method is least.

Highest percentage error has been obtained in the calculation of square root of 1 in the interval [0, 6] and is equal to 0.000381469700. It means if roots lies in the beginning of the interval then the error in the calculated value of root of equation by bisection method is greatest. Exact value of root, value of root calculated by bisection method and percentage error in the calculation of root by bisection method is shown in Table-2. Average percentage error of bisection method in the calculation of square roots of natural numbers from 1 to 25 has been found to be 0.000041549568 which indicates that the accuracy of bisection method can be increased by reducing tolerance value.

Table 2. Numerical accuracy of bisection method in the calculation of roots of functions $f(x) = x^2 - n$; n=1, 2, …., 25

S. No.	Function	Exact Value of root	Value of root obtained by bisection method	Percentage error in bisection method
1	$f(x)=x^2-1$	1.000000000000	0.999996185303	0.000381469700
2	$f(x)=x^2-2$	1.414213562373	1.414215087891	0.000107870412
3	$f(x)=x^2-3$	1.732050807569	1.732051849365	0.000060148120
4	$f(x)=x^2-4$	2.000000000000	1.999998092651	0.000095367450
5	$f(x)=x^2-5$	2.236067977500	2.236066818237	0.000051843817
6	$f(x)=x^2-6$	2.449489742783	2.449490547180	0.000032839370
7	$f(x)=x^2-7$	2.645751311065	2.645750999451	0.000011777902
8	$f(x)=x^2-8$	2.828427124746	2.828427314758	0.000006717939
9	$f(x)=x^2-9$	3.000000000000	2.999998569489	0.000047683700
10	$f(x)=x^2-10$	3.162277660168	3.162277221680	0.000013866208
11	$f(x)=x^2-11$	3.316624790355	3.316623687744	0.000033244973
12	$f(x)=x^2-12$	3.464101615138	3.464100837708	0.000022442471
13	$f(x)=x^2-13$	3.605551275464	3.605551242828	0.000000905160
14	$f(x)=x^2-14$	3.741657386774	3.741657257080	0.000003466218
15	$f(x)=x^2-15$	3.872983346207	3.872983932495	0.000015137891
16	$f(x)=x^2-16$	4.000000000000	3.999999046326	0.000023841850
17	$f(x)=x^2-17$	4.123105625618	4.123106002808	0.000009148201

S. No.	Function	Exact Value of root	Value of root obtained by bisection method	Percentage error in bisection method
18	$f(x)=x^2-18$	4.242640687119	4.242639541626	0.000026999529
19	$f(x)=x^2-19$	4.358898943541	4.358900070190	0.000025847101
20	$f(x)=x^2-20$	4.472135955000	4.472136497498	0.000012130624
21	$f(x)=x^2-21$	4.582575694956	4.582574844360	0.000018561526
22	$f(x)=x^2-22$	4.690415759823	4.690415382385	0.000008047005
23	$f(x)=x^2-23$	4.795831523313	4.795831203461	0.000006669375
24	$f(x)=x^2-24$	4.898979485566	4.898979663849	0.000003639187
25	$f(x)=x^2-25$	5.000000000000	5.000000953674	0.000019073480
Average percentage error in bisection method				0.000041549568

REFERENCES

[1] Speck, G. P., 1977. *New Zealand Math. Mag.*, 14(1):34-36.

[2] Garey M. R., Johnson D. S., Stockmeyer L., 1976. *Theoretical Computer Science*, 1(3):237-267.

[3] Nievergelt, Yves,1994. *SIAM Rev.,* 36(2): 258-264.

[4] Bui T., 1984. *IEEE FOCS*, 181-191.

[5] Jerrum M., 1993. Sorkin G. B., *IEEE FOCS*, 94-103.

[6] Baushev A.N., Morozova E.Y,2007. *Lectures notes in engineering and computer science*, 2:801-803.

[7] Torczon, V, 1991. *SIAM J. Optim.*, 1:123-145.

[8] Torczon V.,1997. *SIAM J. Optim*, 7(1):1-25.

[9] Kazuo Murota,1982. *SIAM Journal on Numerical Analysis*, 19(4): 793-799.

[10] Reddien G. W.,1978. *SIAM Journal on Numerical Analysis*, 15(5): 993-996.

[11] Hart, V. G.; Howard, L. N.,1978. *Austral. Math. Soc. Gaz*., 5 (3): 73-89.

[12] Speck, G. P.,1977. *New Zealand Math. Mag*., 14 (1): 34-36.

[13] Franklin F.,1981. *American Journal of Mathematics*, 4(1/4): 275-276.

[14] Fiduccia C. M., Mattheyses R. M.,1982. *Proceedings of the 19th Design Automation Conference*, 175-181.

[15] Rall L. B.,1974. *SIAM Journal on Numerical Analysis*, 11(1): 34-36.

[16] Jorge J. More,1971. *SIAM Journal on Numerical Analysis*, 8(2): 325-336.

[17] McKinnon, K.I.M.,1998. *SIAM J. Optim*, 9:148-158.

[18] Kernighan B., Lin S.,1970. *The Bell System Technical Journal*, 49(2):291-307.