

Topic : Bisection Method - Roots of Equations

Simulation : Pitfall - Slow Convergence

Language : Mathematica 4.1

Authors : Nathan Collier, Autar Kaw

Date : 11 July 2002

Abstract : This simulation illustrates the slow convergence of the bisection method of finding the root of an equation

$f[x] = 0$.

■ **INPUTS: Enter the Following**

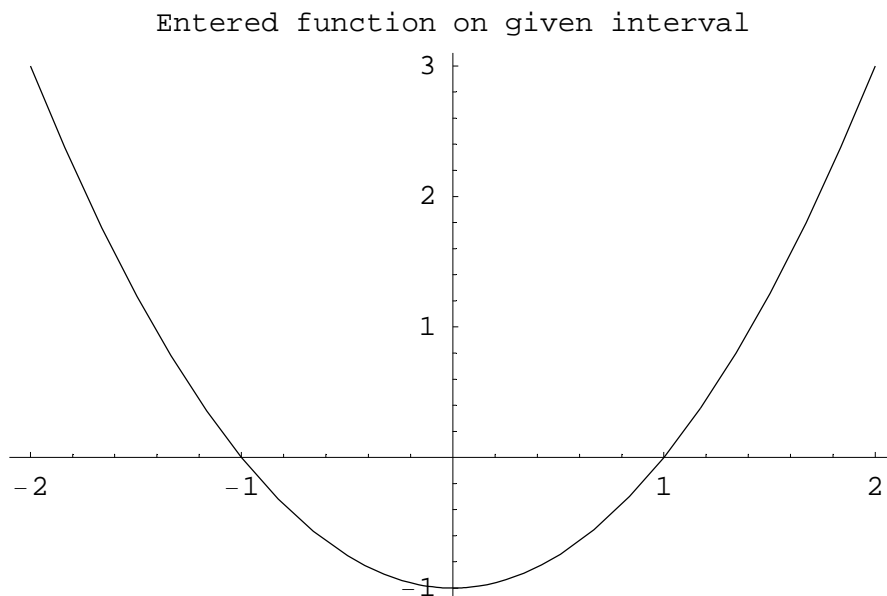
Function in $f[x] = 0$

```
In[209]:= f[x_] := x^2 - 1
```

Range of 'x' you want to see the function

```
In[210]:= x_b = -2;  
x_e = 2;
```

```
In[212]:= curve = Plot[f[x], {x, x_b, x_e}, PlotLabel ->  
"Entered function on given interval", TextStyle -> {FontSize -> 11}];
```



Lower initial guess

```
In[213]:= x_1 = -1.25;
```

Upper initial guess

```
In[214]:= x_u = -0.5;
```

Maximum number of iterations

```
In[215]:= n_maximum = 30;
```

Counting from the left, enter the number of the root desired

```
In[216]:= numroot = 1;
```

■ SOLUTION

```
In[217]:= maxi = f[x_b];
mini = f[x_b];
step = (x_e - x_b) / 10;
Do[ If[f[i] > maxi, maxi = f[i]];
  If[f[i] < mini, mini = f[i]], {i, x_b, x_e, step}];
tot = maxi - mini;
mini = mini - 0.1 * tot;
maxi = maxi + 0.1 * tot;
```

Check first if the lower and upper guesses bracket the root of the equation

```
In[224]:= f[x_l]
```

```
Out[224]= 0.5625
```

```
In[225]:= f[x_u]
```

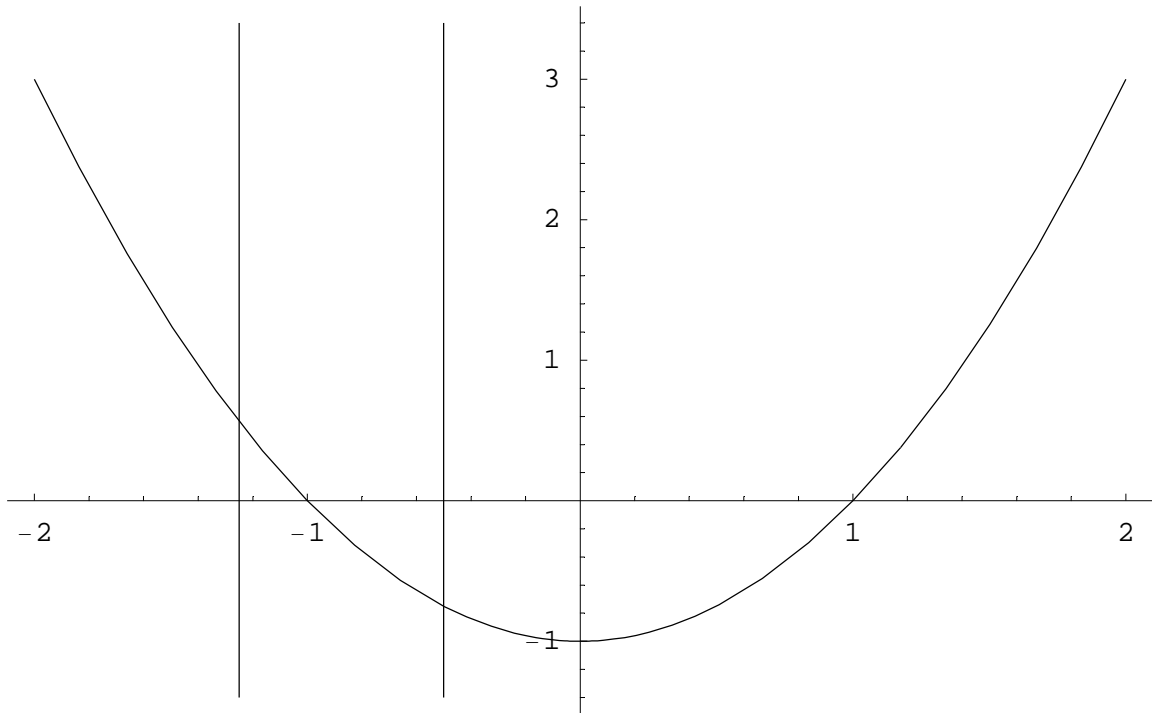
```
Out[225]= -0.75
```

```
In[226]:= % * %%
```

```
Out[226]= -0.421875
```

```
In[227]:= Show[Graphics[Line[{{xu, maxi}, {xu, mini}}]], curve,
Graphics[Line[{{xl, maxi}, {xl, mini}}]], Axes → True, PlotLabel →
"Entered function on given interval with upper and lower guesses",
TextStyle → {FontSize → 11}];
```

Entered function on given interval with upper and lower guesses



■ True Solution

This is the solution found by Mathematica

```
In[228]:= xactual = Root[f[x], numroot]
```

```
Out[228]= -1
```

■ Value of root as a function of iterations

Here the bisection method algorithm is applied to generate the values of the roots, thre error, absolute relative true error, approximate error, absolute relative approximate error, and the number of significant digits at least correct in the estimated root as a function of number of iterations.

```
In[229]:= Array[xr, nmaximum];
```

```
In[230]:= For[i = 1; xu = xu; xl = xl, i <= nmaximum, i++,
xr[i] = (xu + xl) / 2; If[f[xr[i]] * f[xu] ≤ 0, xl = xr[i], xu = xr[i]]]
```

■ Absolute true error

```
In[231]:= Array[Et, n_maximum];
In[232]:= For[i = 1, i <= n_maximum, i++, Et[i] = Abs[xactual - xr[i]]]
```

■ Absolute relative true error

```
In[233]:= Array[et, n_maximum];
In[234]:= For[i = 1, i <= n_maximum, i++, et[i] = Abs[Et[i] / xactual * 100]]
```

■ Absolute approximate error

```
In[235]:= Array[Ea, n_maximum];
In[236]:= For[i = 1, i <= n_maximum, i++, If[i <= 1, Ea[i] = 0, Ea[i] = Abs[xr[i] - xr[i - 1]]]]
```

■ Absolute relative approximate error

```
In[237]:= Array[ea, n_maximum];
In[238]:= For[i = 1, i <= n_maximum, i++, If[i <= 1, ea[i] = 0, ea[i] = Abs[Ea[i] / xr[i] * 100]]]
```

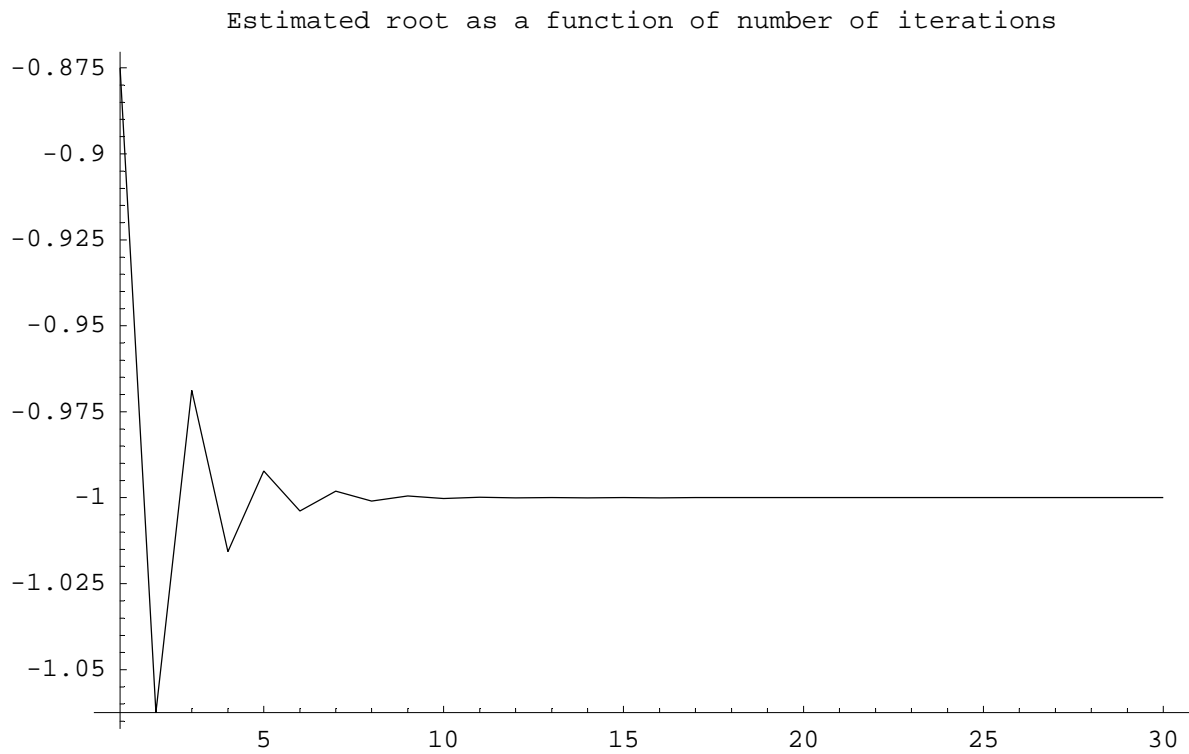
■ Significant digits at least correct

```
In[239]:= Array[sigdig, n_maximum];
In[240]:= For[i = 1, i <= n_maximum, i++, If[(ea[i] ≥ 5) || (i <= 1),
    sigdig[i] = 0, sigdig[i] = Floor[(2 - Log[10, Abs[ea[i] / 0.5]])]]]
```

■ Graphs

```
In[241]:= xrplot = Table[xr[i], {i, 1, n_maximum}];
```

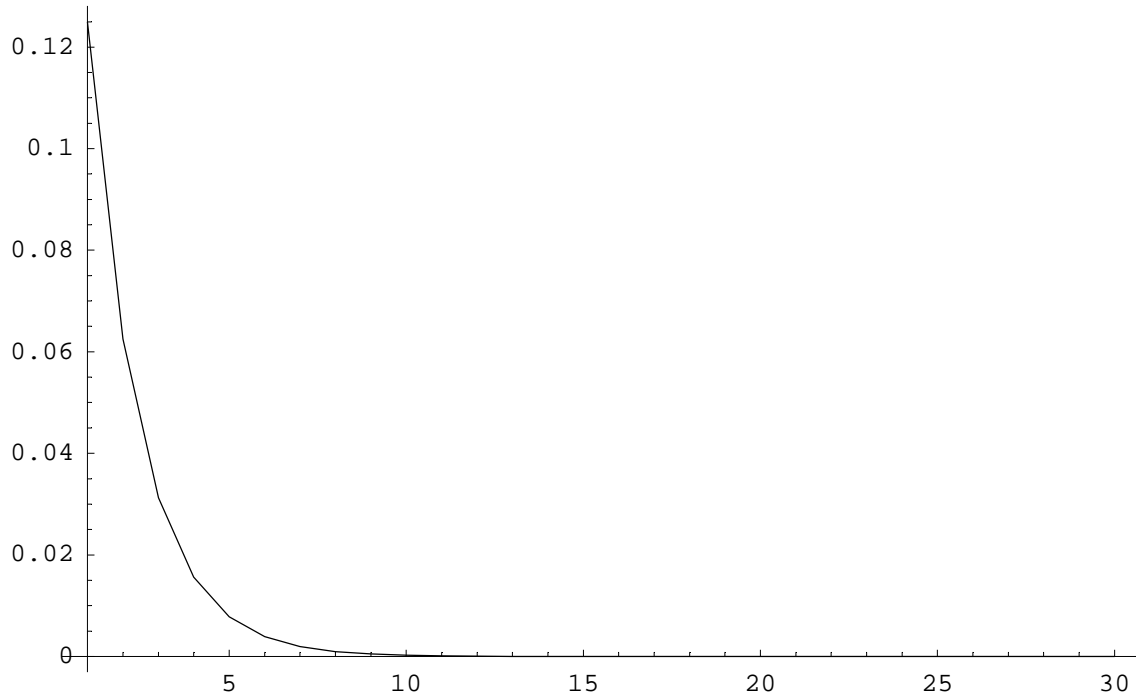
```
In[242]:= ListPlot[xrplot, PlotJoined -> True,  
  PlotRange -> All, AxesOrigin -> {1, Min[xrplot]},  
  PlotLabel -> "Estimated root as a function of number of iterations"];
```



```
In[243]:= Etplot = Table[Et[i], {i, 1, nmaximum}];
```

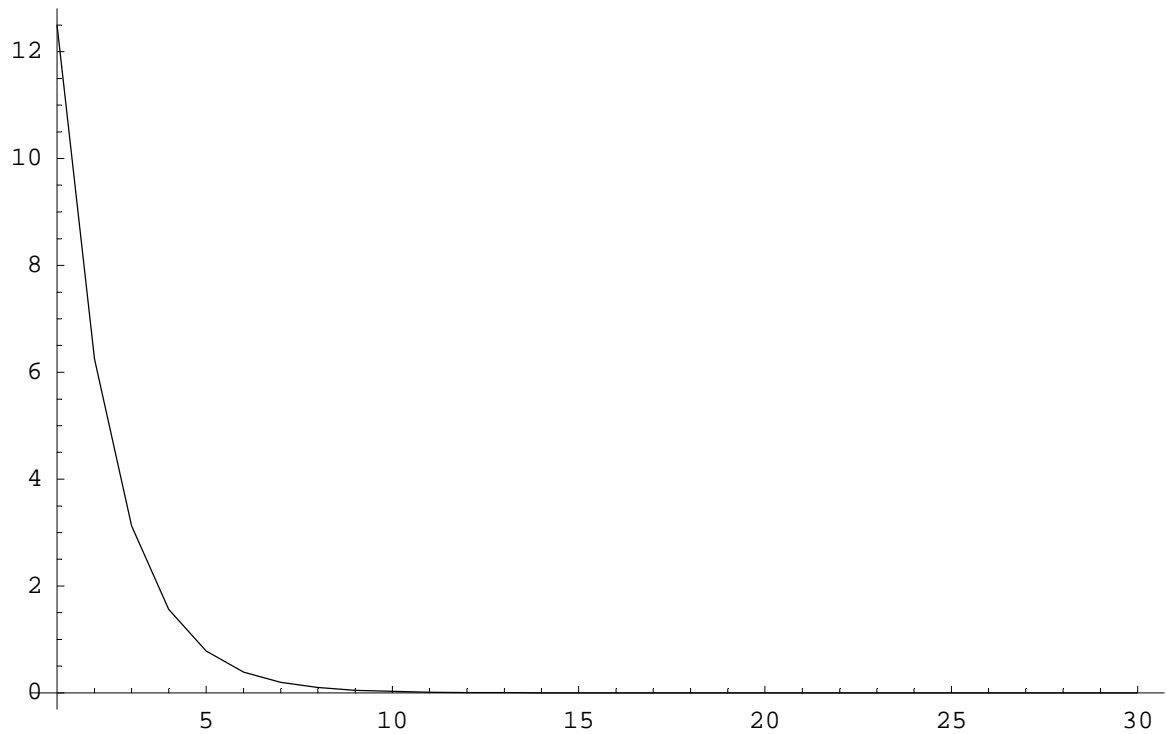
```
In[244]:= ListPlot[Etplot, PlotJoined -> True,  
  PlotRange -> All, AxesOrigin -> {1, Min[Etplot]},  
  PlotLabel -> "Absolute true error as a function of number of iterations"];
```

Absolute true error as a function of number of iterations



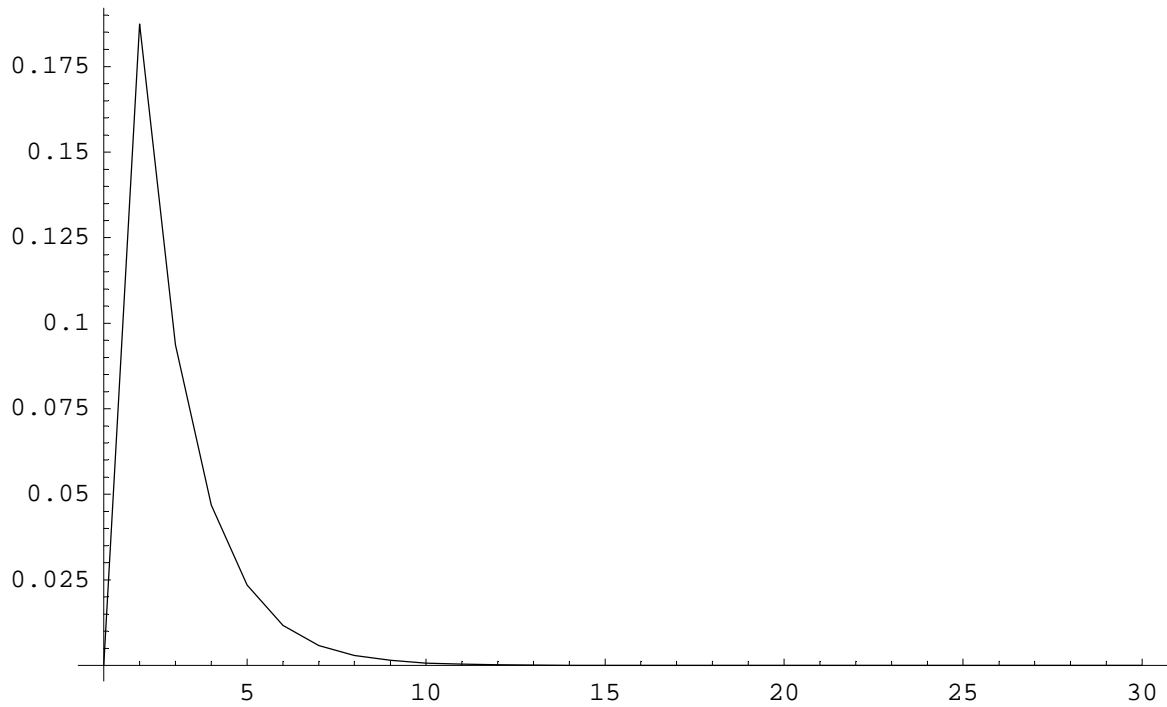
```
In[245]:= etplot = Table[et[i], {i, 1, nmaximum}];
```

```
In[246]:= ListPlot[etplot, PlotJoined -> True,  
  PlotRange -> All, AxesOrigin -> {1, Min[etplot]}, PlotLabel ->  
  "Absolute relative true error as a function of number of iterations";  
  Absolute relative true error as a function of number of iterations
```



```
In[247]:= Eaplot = Table[Ea[i], {i, 1, nmaximum}];
```

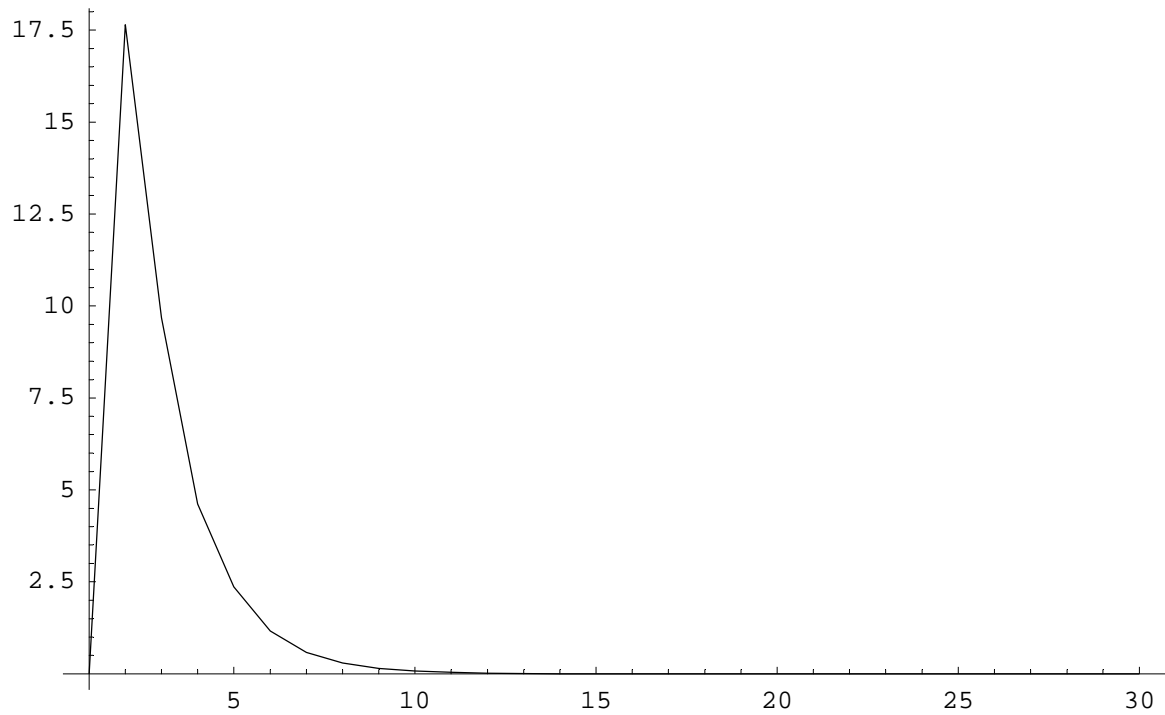
```
In[248]:= ListPlot[Eaplot, PlotJoined -> True,  
  PlotRange -> All, AxesOrigin -> {1, Min[Eaplot]}, PlotLabel ->  
  "Absolute approximate error as a function of number of iterations";  
  Absolute approximate error as a function of number of iterations
```



```
In[249]:= eaplot = Table[ea[i], {i, 1, nmaximum}];
```



```
In[250]:= ListPlot[eaplot, PlotJoined -> True,  
  PlotRange -> All, AxesOrigin -> {1, Min[eaplot]},  
  PlotLabel -> "Absolute relative approximate error  
  as a function of number of iterations";  
  Absolute relative approximate error as a function of number of iterations
```



```
In[251]:= sigdigplot = Table[sigdig[i], {i, 1, nmaximum}];
```

```
In[252]:= << Graphics`Graphics`
```

```
In[253]:= BarChart[sigdigplot];
```

