THE BISECTION METHOD

Most methods for solving $f(x) = 0$ are <u>iterative</u> <u>methods</u>. We begin with the simplest of such methods, one which most people use at some time.

We assume we are given a function $f(x)$; and in addition, we assume we have an interval $[a, b]$ containing the root, on which the function is continuous. We also assume we are given an error tolerance $\varepsilon > 0$, and we want an approximate root $\widetilde{\alpha}$ in $[a, b]$ for which

$$|\alpha - \widetilde{\alpha}| \leq \varepsilon$$

We further assume the function $f(x)$ changes sign on $[a, b]$, with

$$f(a)\, f(b) < 0$$

Algorithm Bisect(f, a, b, ε). Step 1: Define

$$c = \frac{1}{2}(a + b)$$

Step 2: If $b - c \leq \varepsilon$, accept c as our root, and then stop.

Step 3: If $b - c > \varepsilon$, then check compare the sign of $f(c)$ to that of $f(a)$ and $f(b)$. If

$$\text{sign}(f(b)) \cdot \text{sign}(f(c)) \leq 0$$

then replace a with c; and otherwise, replace b with c. Return to Step 1.

Denote the initial interval by $[a_1, b_1]$, and denote each successive interval by $[a_j, b_j]$. Let c_j denote the center of $[a_j, b_j]$. Then

$$\left|\alpha - c_j\right| \leq b_j - c_j = c_j - a_j = \frac{1}{2}\left(b_j - a_j\right)$$

Since each interval decreases by half from the preceding one, we have by induction,

$$|\alpha - c_n| \leq \left(\frac{1}{2}\right)^n (b_1 - a_1)$$

An example from the matlab program ‘bisect.m’ is given in an accompanying file. It is for the function.

$$f(r) = P_{in}\left[(1+r)^{N_{in}} - 1\right] - P_{out}\left[1 - (1+r)^{-N_{out}}\right]$$

Checking, we see that $f(0) = 0$. Therefore, with a graph of $y = f(r)$ on $[0, 1]$, we see that $f(x) < 0$ if we choose x very small, say $x = .001$. Also $f(1) > 0$. Thus we choose $[a, b] = [.001, 1]$. Using $\varepsilon = .000001$ yields the answer

$$\widetilde{\alpha} = .02918243$$

with an error bound of

$$|\alpha - c_n| \leq 9.53 \times 10^{-7}$$

for $n = 20$ iterates. We could also have calculated this error bound from

$$\frac{1}{2^{20}}(1 - .001) = 9.53 \times 10^{-7}$$

Suppose we are given the initial interval $[a, b] = [1.6, 4.5]$ with $\varepsilon = .00005$. How large need n be in order to have

$$|\alpha - c_n| \le \varepsilon$$

Recall that

$$|\alpha - c_n| \le \left(\frac{1}{2}\right)^n (b - a)$$

Then ensure the error bound is true by requiring and solving

$$\left(\frac{1}{2}\right)^n (b - a) \le \varepsilon$$

$$\left(\frac{1}{2}\right)^n (4.5 - 1.6) \le .00005$$

Dividing and solving for n, we have

$$n \ge \log\left(\frac{2.9}{.00005}\right) = 15.82$$

Therefore, we need to take $n = 16$ iterates.

ADVANTAGES AND DISADVANTAGES

Advantages: 1. It always converges.
2. You have a guaranteed error bound, and it decreases with each successive iteration.
3. You have a guaranteed rate of convergence. The error bound decreases by $\frac{1}{2}$ with each iteration.

Disadvantages: 1. It is relatively slow when compared with other rootfinding methods we will study, especially when the function $f(x)$ has several continuous derivatives about the root α.
2. The algorithm has no check to see whether the ε is too small for the computer arithmetic being used. [This is easily fixed by reference to the unit round of the computer arithmetic.]

We also assume the function $f(x)$ is continuous on the given interval $[a, b]$; but there is no way to confirm this.