

Computational Physics

Homework 2

Yi-Hsuan Hsu

09/18/2014

1 Problem 1

Root-finding using bisection method, regula falsi, secant method, Newton's method. Observing error and convergence property in each method.

1.1 Algorithms

1.1.1 Bisection Method

Bisection method take average point of an interval to form a new subinterval repeatedly. In general, bisection method can always find a root (given a continuous function with at least one root).

$$\epsilon_{n+1} = \text{constant} \times \epsilon_n^m \quad (1)$$

where ϵ equal to difference of subinterval. Bisection method is convergent linearly, relatively slow compare to other method.

1.1.2 Regula Falsi Method

Instead of average point as the end of subinterval, Regula Falsi using weight average in the fact that the closer $f(x_n)$ to zero, the closer x_n to the root.

$$w = [f(b_n) \cdot a_n - f(a_n) \cdot b_n] / [f(b_n) - f(a_n)] \quad (2)$$

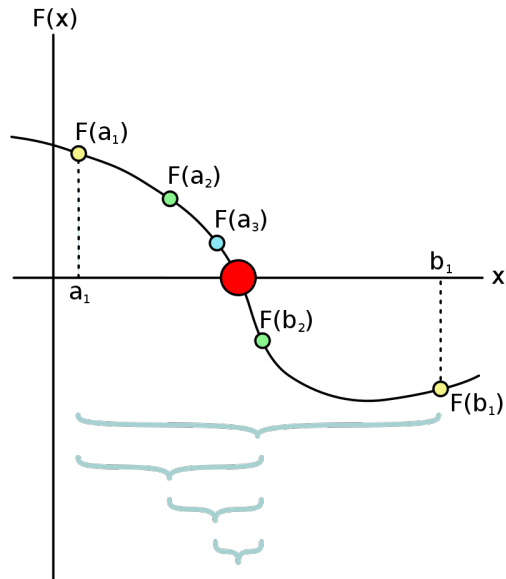


Figure 1: In each iteration, take bisects of interval into subinterval. Make sure the left hand end times right hand end always get minus sign, we will approach close enough to root. Pic From Wiki

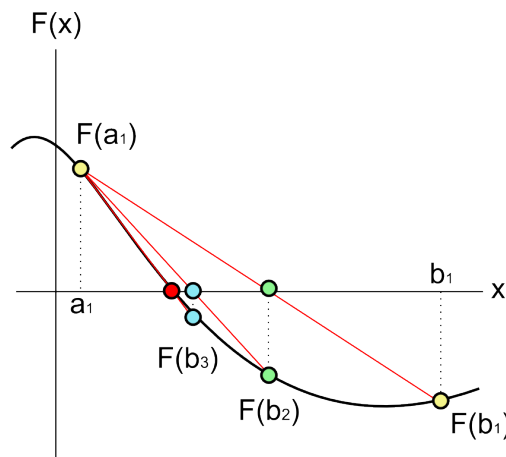


Figure 2: Regula falsi method. Pic From Wiki

1.1.3 Secant Method

Secant method use a succession of roots of secant lines to better approximate a root of a function. We can estimate its convergence by following equation

$$\epsilon_{n+1} = \text{const} \times |\epsilon_n|^\phi \quad (3)$$

where $\phi = 1.618\dots$ is golden ratio.

1.1.4 Newton's Method

From Taylor expansion of arbitrary function $f(x)$, we have

$$f(x + \delta) \approx f(x) + f'(x)\delta + \frac{f''(x)}{2}\delta^2 + \dots \quad (4)$$

We can derive root finding procedure from an initial guess x_0 and derivative function $f'(x)$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (5)$$

In my program, $f'(x)$ is defined

$$f'(x) = \frac{f(x - h) - f(x)}{h} \quad (6)$$

$h = 10^{-14}$.

The convergent rate of newton's method is quadratic. One can derive the recurrence relation for the deviations of the trial solution

$$\epsilon_{n+1} = -\epsilon_n^2 \frac{f''(x)}{2f'(x)} \quad (7)$$

1.2 Description of program

My program can divided to four part.

First, testing function and initial input. At the top of code is where I define testing function and give an initial interval and desired iteration. Initial interval can be assigned either single-float precision or double-float point precision.

Second part is root-finding part. Here I define root-finding function for each method and return $x_n, f(x_n), \epsilon_n (= |a_n - b_n|), \delta_n (= |x_n - x^*|)$ in python list.

Third, main code throw initial value into functions. *if* statement are applied to assure different sign of interval.

Forth, output each method in list. Draw picture of ϵ_n v.s. n and δ_n v.s. n . Observing how different methods convergent and approach to exact solution.

1.3 Sample output

Take $\sin(x)$ root-finding in $[3.0, 3.5]$ as sample (see /out/func sin output.txt). Obviously, newton's method owns the fastest convergence rate. Only two iteration is used we approach the exact solution in machine's limit. Regula falsi and secant are roughly convergent in the same rate. But secant method jumps less step than regular falsi does. Bisection method take over 30 steps to find the root within $1e-10$ level.

Figure 3 illustrate clearly how ϵ_n and δ_n converge. We can see that bisection follows linear convergence while the other follow quadratic or higher order convergence.

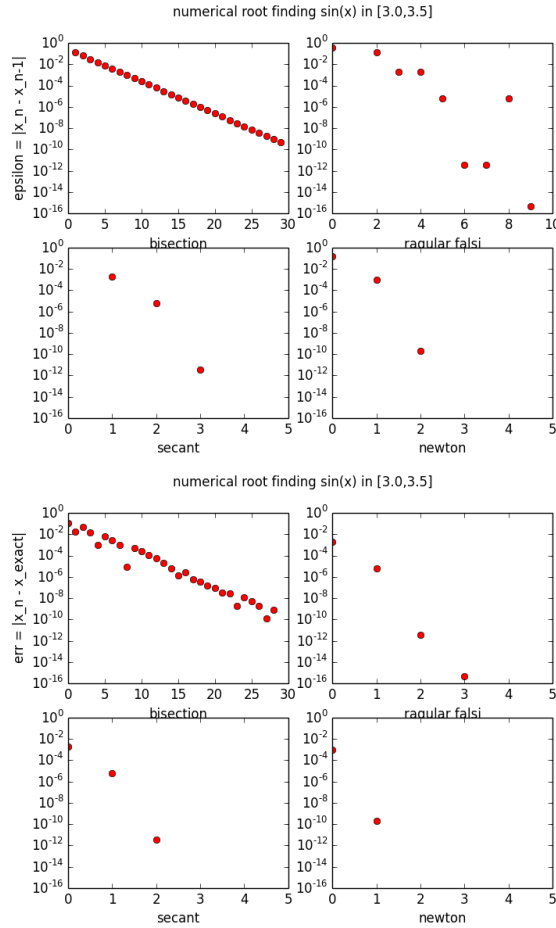


Figure 3: Sample output display. Four different method are applied to find the root of $\sin(x)$. Upper: y axis is ϵ_n , x axis is iteration number. Lower: y axis is δ_n , x axis is iteration number.

1.4 Respond to Question

1.4.1 Question (b)

Figure 3 shows that ϵ_n do follow the way we expect. Although in Regula-falsi(R-F), secant and newton take only few iteration to find the root, we can still see their behavior clearly. Except for small fluctuation, bisection goes linearly, R-F and secant goes quadratic and newton in higher order.

Figure 4 compare the ϵ_n of series iteration with different initial interval $[3.0, 3.5]$ and $[3.0, 4.5]$. Obviously green dot is further than red dot, which implies initial condition or first guess affect efficiency very much, except for bisection method.

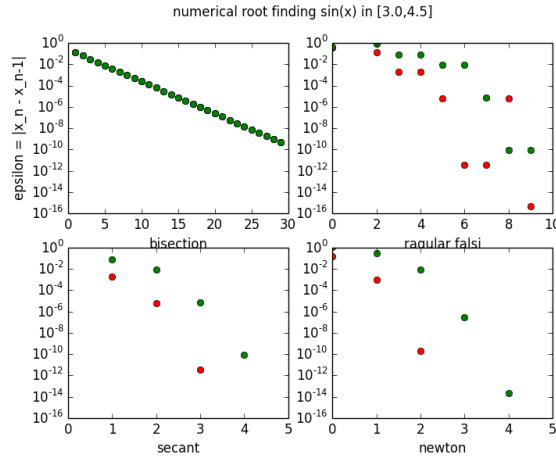


Figure 4: Series after initial interval $[3.0, 3.5]$ are shown in red. Initial interval $[3.0, 4.5]$ are shown in green.

1.4.2 Question (c)

Figure 5 shows ϵ_n and δ_n for $\sin(x)^3$ root-finding. Bisection still works well as it promised, even better than newton's method. Both bisection and newton converge linearly, which is not expected on newton's method. In the other hand, R-F and secant method seem "stuck". The some phenomena keep the same when the shorten the initial interval closer to exact answer.

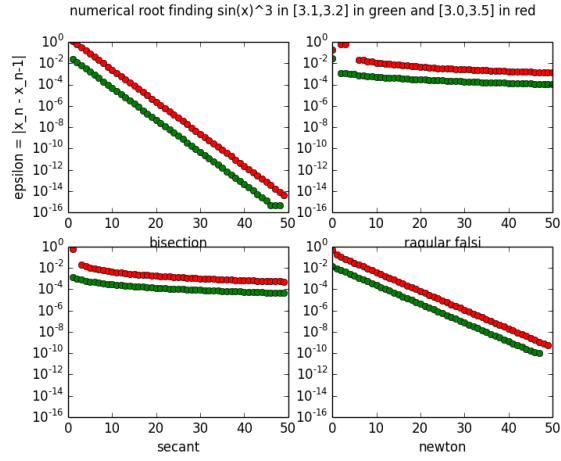


Figure 5:

1.4.3 Question (d)

Again, take $\sin(x)$ as an example. See (`/out/funcsinputautostop.txt`). Iteration series stop automatically. When I try on $f(x) = e^x, 1/x, x^{11}, xe^{-\frac{1}{x^2}}$, the series goes to the left hand side or right hand side due to there's no root for such a function in arbitrary given interval in bisection and R-F method. In the other hand, series blows up in secant and newton method because bracket is broken.