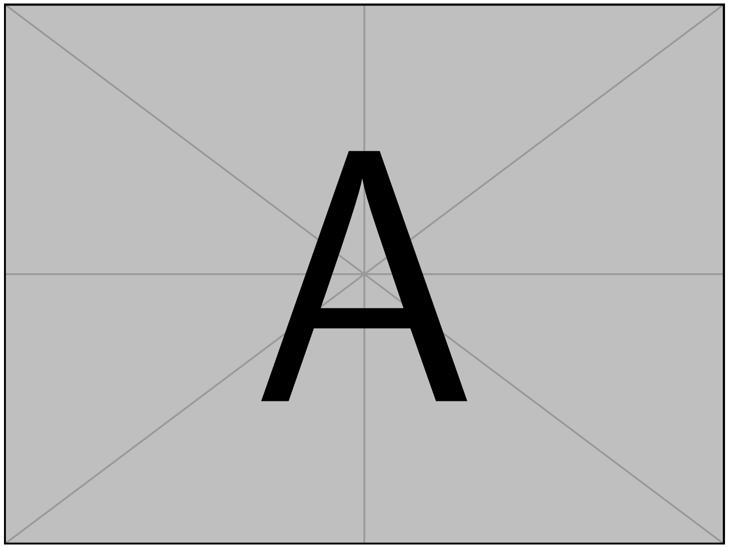
CEE384—Numerical Methods

Root Finding

Solving Non-linear Equations



Arizona State University

Brian Chevalier

Updated: April 20, 2020

Contents

1 Introduction

The roots of simple functions, such as linear and quadratic functions, are very easy to find and have close form solutions. However, most functions have no closed form solution, or difficult to analytically calculate solutions. In fact, analytical formulas do not exist for fifth-degree polynomials or higher (as proven by Abel and Galois in the 19th century).

Numerical root finding algorithms allows us to search for roots of such equations, and several will be discussed here. The general function will take the general form:

$$f(x_r) = 0 = \dots \tag{1}$$

where x_r is the root of the function, that makes the function equal to zero, and where $f(x)$ is a scalar valued function, and is not multidimensional.

In practice, root finding methods are important in many fields, such as non-linear dynamics, where a closed form solution to equations of motion cannot be computed. If the following is some equation of motion with non-linear acceleration:

$$\sum F = ma \tag{2}$$

Then the equation of motion can be rephrased as a root-finding problem:

$$0 = ma - \sum F \tag{3}$$

Which can be read as “what value of acceleration will satisfy Newton’s second law for the given forces”. The acceleration can then be iteratively solved for using the methods discussed here.

2 Stopping Criterion

Before proceeding we must consider the evaluation of a good estimate for the root of an equation. How do we know that a root is “almost right” or “good enough”. The easiet way is to simply evaluate the function with an estimate for the root and look at how close the value is to zero. This value, called the *residual* tells us the true error of the estimate.

$$g = f(x_r) \tag{4}$$

where g is the residual value.

3 Bisection

The bisection method is the simplest root finding method. It is a *bracketing method* that requires an initial range to search for a root for the equation. The range can be defined by x_l as the lower bound of the search range, and x_u is the upper bound of the search range. We can cut the range in half and check if the sign of the function changes in the upper or lower bound of the range.

$$x_r = \frac{x_l + x_u}{2} \tag{5}$$

where x_r is the estimate for the root of the equation,

Whether the root of the equation lies in the upper or lower half of the range can be checked by the following:

- Multiply the function evaluated at the lower bound by the function evaluated at the middle. If this value is less than zero, then the root is within the lower bound and the upper bound should be moved to the midpoint
- Multiply the function evaluated at the lower bound by the function evaluated at the middle. If this value is greater than zero, then the root is within the upper bound and the lower bound should be moved up to the midpoint.
- Otherwise, the midpoint must be the root of the equation

This can be written more concisely as:

$$x_u = x_r \quad (f(x_l)f(x_r) < 0) \tag{6}$$
$$x_l = x_r \quad (f(x_l)f(x_r) > 0) \tag{7}$$
$$x_r = x_r \quad (f(x_l)f(x_r) = 0) \tag{8}$$

3.1 Limitations

- Bisection method is the slowest possible bracketing method. No information about the function is used to converge faster, and therefore will take more iterations than other methods, in general.
- The root you are searching for must be within the initial bracketing range.
- If there are multiple roots in the bracketing range, unexpected behavior will occur with bisection method.

4 False Position

The false position method is another bracketing method, like the bisection method, that takes into about the values of the function when looking for the root. False position uses similar triangles to compute an approximation for the root of the equation. In most instances this will speed up the time of convergence, but it is not guarunteed to.

$$\frac{f(x_l)}{x_r - x_l} = \frac{f(x_u)}{x_r - x_u} \tag{9}$$

$$x_r = x_u - \frac{f(x_u)(x_l - x_u)}{f(x_l) - f(x_u)} \tag{10}$$

5 Newton's Method

Newton's method (also known as the Newton-Raphson method) is an open root finding method (only requires one initial guess, and no bracket) that uses the explicit derivative of a function to search for the root of an equation. Newton's method is one of the fastest root finding methods to converge to a solution with quadratic convergence.

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \tag{11}$$

where x_i is the current approximation of the root, $f(x_i)$ is the function evaluated at x_i, $f'(x_i)$ is the derivative of the function evaluated at the current approximation of the root, and x_{i+1} is the next, improved estimate for the root of the equation.

5.1 Limitations

- The function must be differentiable (not all functions can be analytically differentiated)
- Newton's method does not guaruntee convergence so if an initial guess is bad, may diverge from the nearest root.

6 Secant Method

Secant method is a modification to Newton's method. If an analytical derivative of a function cannot be determined, then an approximation can be made using the finite divided difference.

Take Equation 11 and substitute the approximation of the derivative from the following equation:

$$f'(x_i) = \frac{f(x_{i-1}) - f(x_i)}{x_{i-1} - x_i} \tag{12}$$

This yields the secant method equation:

$$x_{i+1} = x_i - \frac{f(x_i)(x_{i-1} - x_i)}{f(x_{i-1}) - f(x_i)} \tag{13}$$

References

[1] Chapra, Steven C. and Canale, Raymond P., *Numerical Methods for Engineers*, 7th ed. McGraw Hill Education, 2015.

[2] A. K. Kaw, E. E. Kalu, and D. Nguyen. Numerical methods with applications. [Online]. Available: http://nm.mathforcollege.com/topics/textbook_index.html