Bare Essentials

At the end of this chapter you should be able to

1. Write any scalar equation in the form $f(x) = 0$
2. Give a graphical interpretation of the location of a root on the x axis (when equation is written as $f(x) = 0$).
3. Explain the role of bracketing.
4. Write a simple equation that expresses the condition for finding a root in a bracket interval.
5. Manually perform a few steps of the bisection method
6. Identify the one situation where bisection will return an incorrect value for x as a root.
7. Manually perform a few steps of Secant method
8. Identify situations that cause Newton's method to fail.
9. Manually perform a few steps of Newton's method
10. Identify situations that cause the secant method to fail.
11. Describe in words two complimentary convergence criteria.
12. Write expressions for two complimentary convergence criteria.
13. List the methods used by the built in `fzero` command.
14. List reasons why simple root-finding schemes are not recommended to search for roots of polynomials.
15. Name the procedure used by `roots` to find the roots of a polynomial.

To perform basic root-finding with MATLAB you will need to

16. Plot any $f(x)$ as a means of graphically identifying the location of roots.
17. Write an m-file that evaluates $y = f(x)$ for use with `bisect`, `secant`, and `fzero`
18. Write an m-file that evaluates $f(x)$ and $f'(x)$ for use with the `newton` function
19. Find zeros of a function with the `bisect`, `newton`, and `fzero`.
20. Find roots of polynomials with the `roots` command.

An Expanded Core of Knowledge

After mastering the bare essentials you should move on to a deeper understanding of the fundamentals. Doing so involves being able to

1. Qualitatively compare the convergence rates of bisection, secant and Newton's method
2. Describe the `fzero` command, and how it relates to bisection, secant and reverse interpolation.
3. Specify convergence tolerance for any function so that excessive (unnecessary) iterations of a root-finder are not performed.

To perform more advanced root-finding with MATLAB you will need to

4. Describe the role of global variables in finding the roots of $f(x, a, b, \ldots) = 0$ where a, b, ... are parameters, and the method returns the value of x that gives $f = 0$ for fixed values of a, b, ...
5. Write m-files that use pass-through parameters a, b, ..., to evaluates $y = f(x, a, b, \ldots)$ for use with the `fzero` command

Developing Mastery

Working toward mastery of root-finding you will need to

1. Analyze the convergence rate of bisection.
2. Identify the behavior of Newton's method for repeated roots.
3. Connect the problem of root-finding of a scalar equation to the the solution of nonlinear systems of equations.