Introduction to Iterative Algorithms

CS 210 – University of Massachusetts, Boston
15 July 2014

Shiva Iyer
shiva.iyer@gmail.com

Direct vs. Iterative Algorithms

Direct Algorithms	**Iterative Algorithms**
Use finite sequence of operations to arrive at solution	Start with initial guess and successively refine
Usually give exact solution (ignoring rounding errors)	May not converge to a solution at all
Usually limited to linear equations	Can be used for nonlinear equations
Can be prohibitively expensive for large systems of equations	Deliver much better performance for large systems

Iterative Algorithms

Needed when:

1. There is no direct formula for unknown variables in terms of known quantities.
2. Data points needed for analysis are collected at different points in time

Other applications exist, but the ones above are frequently encountered.

Example 1 – Planetary Motion

Problem: Given a planet's orbital position around the Sun, find its position at a future time.

Solution: Solve Kepler's equation for E, where M and k are known constants for the planet's orbit.

$$M = E - k\sin(E)$$

Approach: Since Kepler's equation cannot be solved for E directly, use an iterative method such as Newton-Raphson, bisection, or Laguerre's algorithm.

Example 2 – Satellite Tracking

Problem: Determine the trajectory of a satellite using range (distance) and range-rate (speed) data from radar sites. Since measurements are error prone, redundant information is used for accuracy.

Solution: Guess the position and velocity of the satellite using three radar fixes. Refine the initial guess when more measurements are received.

Approach: The Kalman filter is a widely used iterative algorithm for dealing with measurement noise and uncertainty.

The Iterative Process

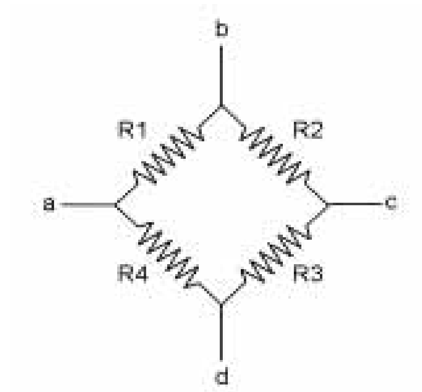
1. Guess
   - Use domain knowledge, past experience, heuristics, etc. to guess a solution.
2. Refine
   - Use governing equations or additional data to improve upon the previous guess
3. Repeat
   - Repeat until successive improvements are within desired tolerance limits. **Warning:** Guard against infinite loops because the process may not converge to a solution.

Caveats

1. Problems can be very sensitive to the initial guess; slight changes can affect whether the process converges or not.
2. Rounding errors can affect intermediate values and the final outcome. The tolerance values must be chosen carefully.
3. Always implement checks to ensure the iteration terminates after a finite number of steps.

A Simple Application

Problem: Determine the values of the resistors in the circuit below.

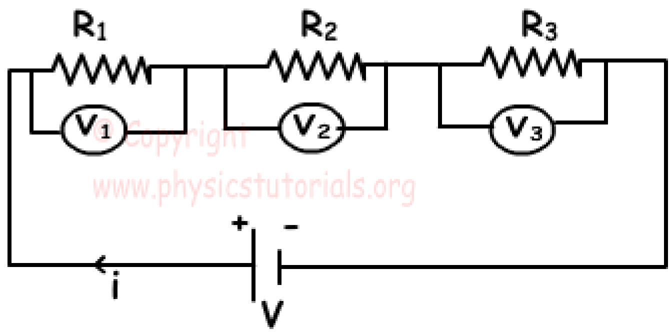


Approach: Using a multimeter, measure R_{m1} across ab, R_{m2} across bc, R_{m3} across cd, R_{m4} across ad. Use Picard iteration to calculate R_1, R_2, R_3, R_4.

Image Source: http://www.daviddarling.info/images/Wheatstone_bridge.jpg

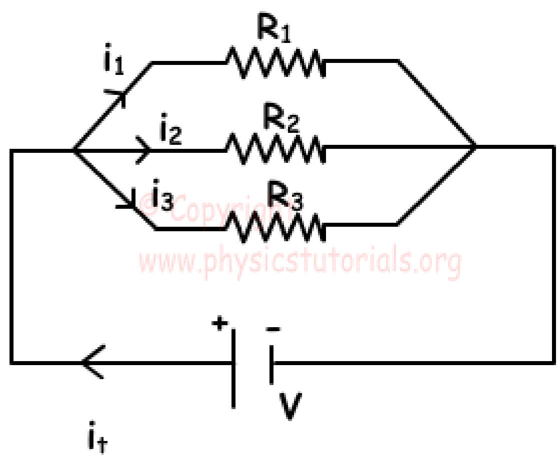
Review of Resistors

- Equivalent resistance - resistors in series



$$R_{eq} = R_1 + R_2 + R_3$$

- Equivalent resistance - resistors in parallel



$$\frac{1}{R_{eq}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}$$

Image Source : http://www.physicstutorials.org/pt/102-Electric_Current_Cheatsheet

Picard Iteration (PicardIteration.java)

Picard iteration was originally formulated to solve systems of ordinary differential equations. We use it to solve the following system of algebraic equations for the unknowns R_1, R_2, R_3, and R_4.

$$\frac{1}{R_{11}} = \frac{1}{R_{m1}} - \frac{1}{R_{20} + R_{30} + R_{40}}$$

$$\frac{1}{R_{21}} = \frac{1}{R_{m2}} - \frac{1}{R_{10} + R_{30} + R_{40}}$$

$$\frac{1}{R_{31}} = \frac{1}{R_{m3}} - \frac{1}{R_{10} + R_{20} + R_{40}}$$

$$\frac{1}{R_{41}} = \frac{1}{R_{m4}} - \frac{1}{R_{10} + R_{20} + R_{30}}$$

Conclusion

- Iterative algorithms are used extensively in numerical analysis, optimization, engineering, etc. We have barely scratched the surface.
- A fertile field of research at the intersection of applied mathematics, computer science, and engineering.
- Merits further study. You are bound to implement an iterative algorithm at least once in your academic or professional lives!