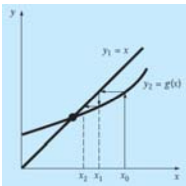
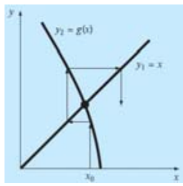
Lecture 06: Roots−Open Methods

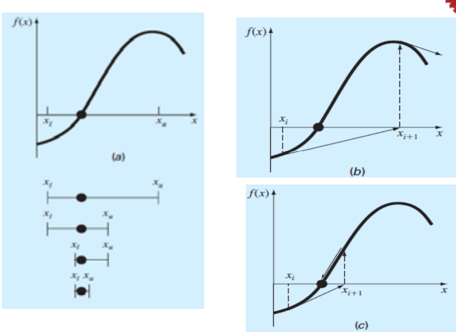




Bracketing Methods vs. Open Methods

- For the bracketing methods, the root is located within an interval prescribed by a lower and an upper bound.
- Repeated application of these methods always results in closer estimates of the true value of the root.
- Such methods are said to be convergent because they move closer to the truth as the computation progresses.
- In contrast, the open methods require only a single starting value or two starting values that do not necessarily bracket the root.
- As such, they sometimes diverge or move away from the true root as the computation progresses.
- However, when the open methods converge they usually do so much more quickly than the bracketing methods.

Bracketing Methods vs. Open Methods



Simple Fixed-point Iteration (FPI)

- Also known as one-point iteration or successive substitution.
- FPI derives an updating formula by rearranging the function as follows:

$$x = g(x)$$

$$x^2 - 2x + 3 = 0 \implies x = \frac{x^2 + 3}{2}$$

$$\sin x = 0 \implies x = \sin x + x$$

Simple Fixed-point Iteration (FPI)

$$x_{i+1} = g(x_i)$$

$$\varepsilon_a = \left| \frac{x_{i+1} - x_i}{x_{i+1}} \right| 100\%$$

Example 1

Problem Statement. Use simple fixed-point iteration to locate the root of $f(x) = e^{-x} - x$.

$$x_{i+1} = e^{-x_i}$$

i	x_i	ε_a (%)	ε_t (%)
0	0		100.0
1	1.000000	100.0	76.3
2	0.367879	171.8	35.1
3	0.692201	46.9	22.1
4	0.500473	38.3	11.8
5	0.606244	17.4	6.89
6	0.545396	11.2	3.83
7	0.579612	5.90	2.20
8	0.560115	3.48	1.24
9	0.571143	1.93	0.705
10	0.564879	1.11	0.399

true value of the root: 0.56714329

Convergence

- Notice that the true percent relative error for each iteration of the previous example is roughly proportional (by a factor of about 0.5 to 0.6) to the error from the previous iteration.
- This property, called linear convergence, is characteristic of fixed-point iteration.
- We must comment at this point about the “possibility” of convergence.
- The two-curve method (an alternative graphical approach) can be used to illustrate the convergence and divergence of fixed-point iteration.

Two-curve Method

- It separates the equation into two component parts:

$$f_1(x) = f_2(x)$$

$$y_1 = f_1(x) \qquad y_2 = f_2(x)$$

- These two equations can be plotted separately.
- The x values corresponding to the intersections of these functions represent the roots of $f(x) = 0$.

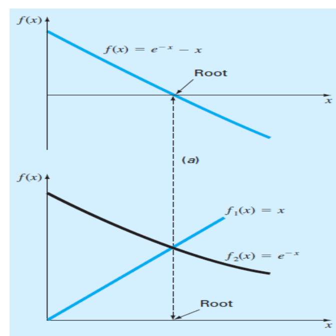
Example 2

Problem Statement. Separate the equation $e^{-x} - x = 0$ into two parts and determine its root graphically.

$$y_1 = x \text{ and } y_2 = e^{-x}$$

x	y_1	y_2
0.0	0.0	1.000
0.2	0.2	0.819
0.4	0.4	0.670
0.6	0.6	0.549
0.8	0.8	0.449
1.0	1.0	0.368

Example 2

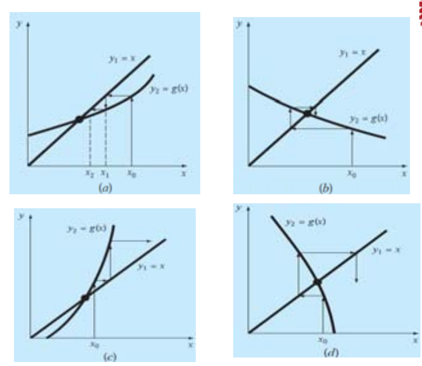


Two-curve Method

$$x = g(x)$$

$$y_1 = x \qquad y_2 = g(x)$$

Two-curve Method



Convergence of FPI

- For the first case (Fig. a), the initial guess of x_0 is used to determine the corresponding point on the y_2 curve [x_0, $g(x_0)$].
- The point (x_1, x_1) is located by moving left horizontally to the y_1 curve.
- These movements are equivalent to the first iteration in the fixed-point method:

$$x_1 = g(x_0)$$

Convergence of FPI

- The next iteration consists of moving to [x_1, $g(x_1)$] and then to (x_2, x_2).
- This iteration is equivalent to the equation

$$x_2 = g(x_1)$$

- Notice that convergence seems to occur only when the absolute value of the slope of $y_2 = g(x)$ is less than the slope of $y_1 = x$, that is, when $|g'(x)| < 1$.

Convergence of FPI: Theoretical Derivation

$$x_{i+1} = g(x_i)$$

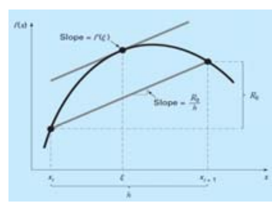
$$x_r = g(x_r)$$

$$x_r - x_{i+1} = g(x_r) - g(x_i)$$

The *derivative mean-value theorem* (recall Sec. 4.1.1) states that if a function $g(x)$ and its first derivative are continuous over an interval $a \le x \le b$, then there exists at least one value of $x = \xi$ within the interval such that

$$g'(\xi) = \frac{g(b) - g(a)}{b - a}$$

Convergence of FPI: Theoretical Derivation



$$g'(\xi) = \frac{g(b) - g(a)}{b - a}$$

$$x_r - x_{i+1} = g(x_r) - g(x_i)$$

let $a = x_i$ and $b = x_r$,

$$g(x_r) - g(x_i) = (x_r - x_i)g'(\xi)$$

$$x_r - x_{i+1} = (x_r - x_i)g'(\xi)$$

If the true error for iteration i is defined as

$$E_{t,i} = x_r - x_i$$

$$E_{t,i+1} = g'(\xi)E_{t,i}$$

Convergence of FPI: Theoretical Derivation

$$E_{t,i+1} = g'(\xi)E_{t,i}$$

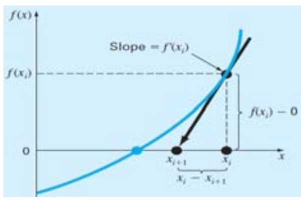
- Consequently, if $|g'(x)|<1$, the errors decrease with each iteration.
- For $|g'(x)|>1$, the errors grow.
- Notice also that if the derivative is positive, the errors will be positive, and hence, the iterative solution will be monotonic (Fig. a and c).
- If the derivative is negative, the errors will oscillate (Fig. b and d).
- When the method converges, the error is roughly proportional to and less than the error of the previous step.
- For this reason, simple fixed point iteration is said to be linearly convergent.

Algorithm for Fixed-Point Iteration

```
FUNCTION Fixpt(x0, es, imax, iter, ea)
  xr = x0
  iter = 0
  DO
    xrold = xr
    xr = g(xrold)
    iter = iter + 1
    IF xr ≠ 0 THEN
      ea = | (xr − xrold) / xr | · 100
    END IF
    IF ea < es OR iter ≥ imax EXIT
  END DO
  Fixpt = xr
END Fixpt
```

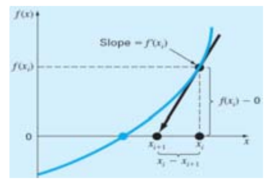
Newton-Raphson Method

- It is perhaps the most widely used of all root-locating method.
- If the initial guess at the root is x_i , a tangent can be extended from the point $[x_i , f(x_i)]$.
- The point where this tangent crosses the x axis usually represents an improved estimate of the root.



Newton-Raphson Method

- The Newton-Raphson method can be derived on the basis of this geometrical interpretation.



$$f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}}$$

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

Newton-Raphson formula

Example 3

Problem Statement. Use the Newton-Raphson method to estimate the root of $f(x) = e^{-x} - x$ employing an initial guess of $x_0 = 0$.

$$f'(x) = -e^{-x} - 1 \qquad x_{i+1} = x_i - \frac{e^{-x_i} - x_i}{-e^{-x_i} - 1}$$

i	x_i	$\lvert \varepsilon_t \rvert$, %
0	0	100
1	0.500000000	11.8
2	0.566311003	0.147
3	0.567143165	0.0000220
4	0.567143290	$< 10^{-8}$

Derivation and Error Analysis of Newton-Raphson Method

- Newton-Raphson method may also be developed from the Taylor series expansion.
- This alternative derivation is useful in that it also provides insight into the rate of convergence of the method.

$$f(x_{i+1}) = f(x_i) + f'(x_i)(x_{i+1} - x_i) + \frac{f''(\xi)}{2!}(x_{i+1} - x_i)^2$$

$$f(x_{i+1}) \cong f(x_i) + f'(x_i)(x_{i+1} - x_i)$$

Derivation and Error Analysis of Newton-Raphson Method

- At the intersection with the x axis, $f(x_{i+1})$ would be equal to zero.

$$f(x_{i+1}) \cong f(x_i) + f'(x_i)(x_{i+1} - x_i)$$

$$0 = f(x_i) + f'(x_i)(x_{i+1} - x_i)$$

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

Derivation and Error Analysis of Newton-Raphson Method

- Taylor series can also be used to estimate the error of the formula.
- This can be done by realizing that if the complete Taylor series were employed, an exact result would be obtained.
- For this situation $x_{i+1} = x_r$, where x is the true value of the root.

$$0 = f(x_i) + f'(x_i)(x_r - x_i) + \frac{f''(\xi)}{2!}(x_r - x_i)^2$$

Derivation and Error Analysis of Newton-Raphson Method

$$0 = f(x_i) + f'(x_i)(x_r - x_i) + \frac{f''(\xi)}{2!}(x_r - x_i)^2$$

$$0 = f(x_i) + f'(x_i)(x_{i+1} - x_i)$$

$$0 = f'(x_i)(x_r - x_{i+1}) + \frac{f''(\xi)}{2!}(x_r - x_i)^2$$

$$E_{t,i+1} = x_r - x_{i+1}$$

$$0 = f'(x_i)E_{t,i+1} + \frac{f''(\xi)}{2!}E_{t,i}^2$$

Bangladesh University of Eng. & Tech. | Slide 25 of 62 | Industrial & Production Engineering

Derivation and Error Analysis of Newton-Raphson Method

$$0 = f'(x_i)E_{t,i+1} + \frac{f''(\xi)}{2!}E_{t,i}^2$$

- If we assume convergence, both x_i and ξ should eventually be approximated by the root x_r.

$$E_{t,i+1} = \frac{-f''(x_r)}{2f'(x_r)}E_{t,i}^2$$

- The error is roughly proportional to the square of the previous error.
- This means that the number of correct decimal places approximately doubles with each iteration.
- Such behavior is referred to as quadratic convergence.

Bangladesh University of Eng. & Tech. | Slide 26 of 62 | Industrial & Production Engineering

Termination Criteria and Error Estimates

$$\varepsilon_a = \left|\frac{x_{i+1} - x_i}{x_{i+1}}\right| 100\%$$

- The Taylor series derivation of the method provides theoretical insight regarding the rate of convergence as expressed by $E_{i+1} = O(E_i^2)$.
- Thus the error should be roughly proportional to the square of the previous error.
- In other words, the number of significant figures of accuracy approximately doubles with each iteration.
- This behavior is called quadratic convergence and is one of the major reasons for the popularity of the method.

Bangladesh University of Eng. & Tech. | Slide 27 of 62 | Industrial & Production Engineering

Example 4

Problem Statement. Use the Newton-Raphson method to estimate the root of $f(x) = e^{-x} - x$ employing an initial guess of $x_0 = 0$.

$$E_{t,i+1} \cong \frac{-f''(x_r)}{2f'(x_r)}E_{t,i}^2$$

Examine this formula and see if it applies to the results of Example 6.3.

Bangladesh University of Eng. & Tech. | Slide 28 of 62 | Industrial & Production Engineering

Example 4

$$f'(x) = -e^{-x} - 1$$

$$x_{i+1} = x_i - \frac{e^{-x_i} - x_i}{-e^{-x_i} - 1}$$

$$x_r = 0.56714329 \qquad f'(0.56714329) = -1.56714329$$

$$f''(x) = e^{-x} \qquad f''(0.56714329) = 0.56714329$$

$$E_{t,i+1} \cong -\frac{0.56714329}{2(-1.56714329)}E_{t,i}^2 = 0.18095E_{t,i}^2$$

$$E_{t,0} = 0.56714329, \qquad E_{t,1} \cong 0.18095(0.56714329)^2 = 0.0582$$

Bangladesh University of Eng. & Tech. | Slide 29 of 62 | Industrial & Production Engineering

Pitfalls of Newton-Raphson Method

- Although the Newton-Raphson method is often very efficient, there are situations where it performs poorly.
- A special case—multiple roots—will be addressed later.
- However, even when dealing with simple roots, difficulties can also arise, as in the following example.

Bangladesh University of Eng. & Tech. | Slide 30 of 62 | Industrial & Production Engineering

Example 5

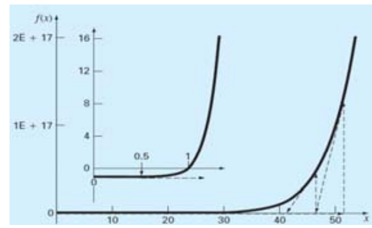
Problem Statement. Determine the positive root of $f(x) = x^{10} - 1$ using the Newton-Raphson method and an initial guess of $x = 0.5$.

$$x_{i+1} = x_i - \frac{x_i^{10} - 1}{10x_i^9}$$

i	x_i	$\lvert\varepsilon_a\rvert$, %
0	0.5	
1	51.65	99.032
2	46.485	11.111
3	41.8365	11.111
4	37.65285	11.111
⋮		
40	1.002316	2.130
41	1.000024	0.229
42	1	0.002

Convergence is very slow

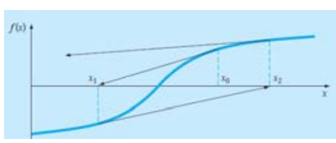
Why does this happen?



- Notice how the first guess is in a region where the slope is near zero.
- Thus, the first iteration flings the solution far away from the initial guess to a new value (x = 51.65) where $f(x)$ has an extremely high value.

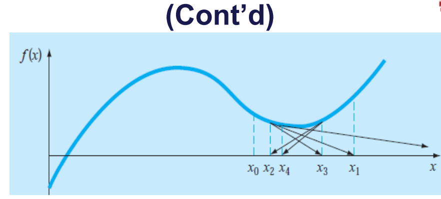
Pitfalls of Newton-Raphson Method (Cont'd)

- Aside from slow convergence due to the nature of the function, other difficulties can arise.



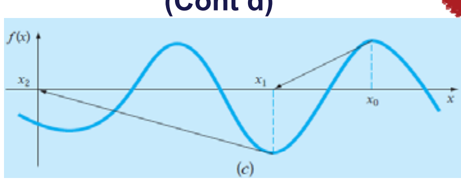
- An inflection point [i.e., $f''(x) = 0$] occurs in the vicinity of a root.
- Notice that iterations beginning at x_0 progressively diverge from the root.

Pitfalls of Newton-Raphson Method (Cont'd)



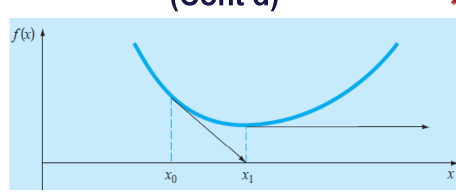
- It illustrates the tendency of the Newton-Raphson technique to oscillate around a local maximum or minimum.
- Such oscillations may persist, or, as in the fig., a near-zero slope is reached whereupon the solution is sent far from the area of interest.

Pitfalls of Newton-Raphson Method (Cont'd)



- It shows how an initial guess that is close to one root can jump to a location several roots away.
- This tendency to move away from the area of interest is because near-zero slopes are encountered.

Pitfalls of Newton-Raphson Method (Cont'd)



- A zero slope [$f(x)$ = 0] is truly a disaster because it causes division by zero in the Newton-Raphson formula.
- Graphically, it means that the solution shoots off horizontally and never hits the x axis.

Pitfalls of Newton-Raphson Method (Cont'd)

- Thus, there is no general convergence criterion for Newton-Raphson.
- Its convergence depends on the nature of the function and on the accuracy of the initial guess.
- The only remedy is to have an initial guess that is "sufficiently" close to the root. (for some functions, no guess will work!)
- Good guesses are usually predicated on knowledge of the physical problem setting or on devices such as graphs that provide insight into the behavior of the solution.
- The lack of a general convergence criterion also suggests that good computer software should be designed to recognize slow convergence or divergence.

Algorithm for Newton-Raphson

```
FUNCTION newtraph(x0, es, imax, iter, ea)
  xr = x0
  iter = 0
  DO
    xrold = xr
    xr = xr - func(xr)/dfunc(xr)
    iter = iter + 1
    IF xr ≠ 0 THEN
      ea = | (xr - xrold) / xr | · 100
    END IF
    IF ea < es OR iter ≥ imax EXIT
  END DO
  newtraph = xr
END newtraph
```


Algorithm for Newton-Raphson

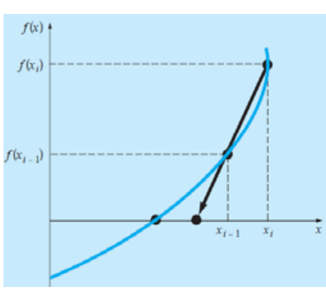
- The program should incorporate several additional features:
  1. A plotting routine should be included in the program.
  2. At the end of the computation, the final root estimate should always be substituted into the original function to compute whether the result is close to zero. This check partially guards against those cases where slow or oscillating convergence may lead to a small value of ε_a while the solution is still far from a root.
  3. The program should always include an upper limit on the number of iterations to guard against oscillating, slowly convergent, or divergent solutions that could persist interminably.
  4. The program should alert the user and take account of the possibility that $f'(x)$ might equal zero at any time during the computation.

Secant Method

- A potential problem in implementing the Newton Raphson method is the evaluation of the derivative.
- There are certain functions whose derivatives may be extremely difficult or inconvenient to evaluate.
- For these cases, the derivative can be approximated by a backward finite divided difference:

$$f'(x_i) \cong \frac{f(x_{i-1}) - f(x_i)}{x_{i-1} - x_i}$$

Secant Method



Secant Method

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

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

- This is the formula for the secant method.
- Notice that the approach requires two initial estimates of *x*.
- However, because *f*(*x*) is not required to change signs between the estimates, it is not classified as a bracketing method.

Difference Between Secant and False-Position Methods

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

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

- A critical difference between the methods is how one of the initial values is replaced by the new estimate.
- Recall that in the false-position method the latest estimate of the root replaces whichever of the original values yielded a function value with the same sign as $f(x_r)$.
- Consequently, the two estimates always bracket the root. Therefore, for all practical purposes, the method always converges because the root is kept within the bracket.

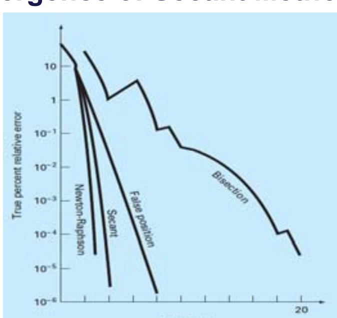
Difference Between Secant and False-Position Methods

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

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

- In contrast, the secant method replaces the values in strict sequence, with the new value x_{i+1} replacing x_i and x_i replacing x_{i-1}.
- As a result, the two values can sometimes lie on the same side of the root.
- For certain cases, this can lead to divergence.

Convergence of Secant Method



Convergence of Secant Method

- Although the secant method may be divergent, when it converges it usually does so at a quicker rate than the false-position method.
- The inferiority of the false-position method is due to one end staying fixed to maintain the bracketing of the root.
- This property, which is an advantage in that it prevents divergence, is a shortcoming with regard to the rate of convergence.

Modified Secant Method

- Rather than using two arbitrary values to estimate the derivative, an alternative approach involves a fractional perturbation of the independent variable to estimate $f'(x)$:

$$f'(x_i) \cong \frac{f(x_i + \delta x_i) - f(x_i)}{\delta x_i}$$

- where δ = a small perturbation fraction.

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

Modified Secant Method

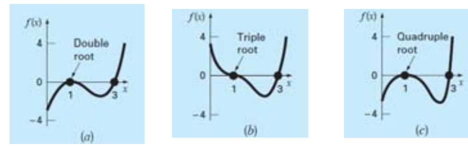
- The choice of a proper value for δ is not automatic.
- If δ is too small, the method can be swamped by round-off error caused by subtractive cancellation in the denominator of the formula.
- If it is too big, the technique can become inefficient and even divergent.
- However, if chosen correctly, it provides a nice alternative for cases where evaluating the derivative is difficult and developing two initial guesses is inconvenient.

Modified Secant Method

- Further, in its most general sense, a univariate function is merely an entity that returns a single value in return for values sent to it.
- Perceived in this sense, functions are not always simple formulas like the one-line equations.
- For example, a function might consist of many lines of code that could take a significant amount of execution time to evaluate.
- In some cases, the function might even represent an independent computer program.
- For such cases, the secant and modified secant methods are valuable.

Multiple Roots

- A multiple root corresponds to a point where a function is tangent to the *x* axis.



- In general, odd multiple roots cross the axis, whereas even ones do not.

Multiple Roots

- Multiple roots pose some difficulties for many of the numerical methods discussed:
  1. The fact that the function does not change sign at even multiple roots precludes the use of the reliable bracketing methods.
  2. Another possible problem is related to the fact that not only *f(x) but also f′(x) goes to* zero at the root.
     - This poses problems for both the Newton-Raphson and secant methods, which both contain the derivative (or its estimate) in the denominator of their respective formulas.
     - This could result in division by zero when the solution converges very close to the root.
     - A simple way to circumvent these problems is based on the fact that it can be demonstrated theoretically (Ralston and Rabinowitz, 1978) that *f(x) will always reach zero before f′(x).*
     - *Therefore, if a zero check for f (x) is incorporated into* the computer program, the computation can be terminated before *f′(x) reaches zero.*

Multiple Roots

3. It can be demonstrated that the Newton-Raphson and secant methods are linearly, rather than quadratically, convergent for multiple roots (Ralston and Rabinowitz, 1978).
   - Modifications have been proposed to alleviate this problem.
   - Ralston and Rabinowitz (1978) have indicated that a slight change in the formulation returns it to quadratic convergence, as in

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

- where *m* is the multiplicity of the root (that is, $m = 2$ for a double root, $m = 3$ for a triple root, etc.).
- This may be an unsatisfactory alternative because it hinges on foreknowledge of the multiplicity of the root.

Multiple Roots

- Another alternative, also suggested by Ralston and Rabinowitz (1978), is to define a new function *u(x),* that is, the ratio of the function to its derivative, as in

$$u(x) = \frac{f(x)}{f'(x)}$$

- It can be shown that this function has roots at all the same locations as the original function.

$$x_{i+1} = x_i - \frac{u(x_i)}{u'(x_i)} \qquad u'(x) = \frac{f'(x)f'(x) - f(x)f''(x)}{[f'(x)]^2}$$

$$x_{i+1} = x_i - \frac{f(x_i)f'(x_i)}{[f'(x_i)]^2 - f(x_i)f''(x_i)}$$

Multiple Roots

- Although it is preferable for multiple roots, it is somewhat less efficient and requires more computational effort than the standard method for simple roots.
- It should be noted that a modified version of the secant method suited for multiple roots can also be developed.

$$u(x) = \frac{f(x)}{f'(x)}$$

$$x_{i+1} = x_i - \frac{u(x_i)(x_{i-1} - x_i)}{u(x_{i-1}) - u(x_i)}$$

Systems of Nonlinear Equations

- Non-linear system of equations:

$$f_1(x_1, x_2, \ldots, x_n) = 0$$
$$f_2(x_1, x_2, \ldots, x_n) = 0$$
$$\vdots$$
$$f_n(x_1, x_2, \ldots, x_n) = 0$$

- Most approaches for determining such solutions are extensions of the open methods for solving single equations.
- In this lecture, we will investigate two of these: fixed-point iteration and Newton-Raphson.

Fixed-Point Iteration

Fixed-Point Iteration for a Nonlinear System

Problem Statement. Use fixed-point iteration to determine the roots of Eq. (6.19). Note that a correct pair of roots is $x = 2$ and $y = 3$. Initiate the computation with guesses of $x = 1.5$ and $y = 3.5$.

$$x^2 + xy = 10$$

and

$$y + 3xy^2 = 57$$

$$x_{i+1} = \frac{10 - x_i^2}{y_i}$$

$$y_{i+1} = 57 - 3x_i y_i^2$$

$$x = \sqrt{10 - xy}$$

$$y = \sqrt{\frac{57 - y}{3x}}$$

Fixed-Point Iteration

- The previous example illustrates the most serious shortcoming of simple fixed-point iteration—that is, convergence often depends on the manner in which the equations are formulated.
- Additionally, even in those instances where convergence is possible, divergence can occur if the initial guesses are insufficiently close to the true solution.
- Using reasoning similar to that in Box 6.1, it can be demonstrated that sufficient conditions for convergence for the two-equation case are

$$\left|\frac{\partial u}{\partial x}\right| + \left|\frac{\partial u}{\partial y}\right| < 1 \qquad \left|\frac{\partial v}{\partial x}\right| + \left|\frac{\partial v}{\partial y}\right| < 1$$

- These criteria are so restrictive that fixed-point iteration has limited utility for solving nonlinear systems.
- However, it can be very useful for solving linear systems.

Newton-Raphson

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

- This is the single-equation form of the Newton-Raphson method.
- The multiequation form is derived in an identical fashion.
- However, a multivariable Taylor series must be used to account for the fact that more than one independent variable contributes to the determination of the root.

Newton-Raphson

$$u_{i+1} = u_i + (x_{i+1} - x_i)\frac{\partial u_i}{\partial x} + (y_{i+1} - y_i)\frac{\partial u_i}{\partial y}$$

and

$$v_{i+1} = v_i + (x_{i+1} - x_i)\frac{\partial v_i}{\partial x} + (y_{i+1} - y_i)\frac{\partial v_i}{\partial y}$$

- Just as for the single-equation version, the root estimate corresponds to the values of *x and y,* where u_{i+1} and v_{i+1} equal zero.

$$\frac{\partial u_i}{\partial x}x_{i+1} + \frac{\partial u_i}{\partial y}y_{i+1} = -u_i + x_i\frac{\partial u_i}{\partial x} + y_i\frac{\partial u_i}{\partial y}$$
$$\frac{\partial v_i}{\partial x}x_{i+1} + \frac{\partial v_i}{\partial y}y_{i+1} = -v_i + x_i\frac{\partial v_i}{\partial x} + y_i\frac{\partial v_i}{\partial y}$$

- This is a set of two linear equations with two unknowns.

Newton-Raphson

- Consequently, algebraic manipulations (for example, Cramer's rule) can be employed to solve for

$$x_{i+1} = x_i - \frac{u_i\frac{\partial v_i}{\partial y} - v_i\frac{\partial u_i}{\partial y}}{\frac{\partial u_i}{\partial x}\frac{\partial v_i}{\partial y} - \frac{\partial u_i}{\partial y}\frac{\partial v_i}{\partial x}}$$

$$y_{i+1} = y_i - \frac{v_i\frac{\partial u_i}{\partial x} - u_i\frac{\partial v_i}{\partial x}}{\frac{\partial u_i}{\partial x}\frac{\partial v_i}{\partial y} - \frac{\partial u_i}{\partial y}\frac{\partial v_i}{\partial x}}$$

- The denominator of each of these equations is formally referred to as the determinant of the Jacobian of the system.

Newton-Raphson

- Just as with fixed-point iteration, the Newton-Raphson approach will often diverge if the initial guesses are not sufficiently close to the true roots.
- Whereas graphical methods could be employed to derive good guesses for the single-equation case, no such simple procedure is available for the multiequation version.
- The two-equation Newton-Raphson approach can be generalized to solve *n* simultaneous equations.

Assignment-06

- Problems 6.3, 6.6, 6.12, 6.18, 6.26, 6.30.