Lec 21: Roots of Nonlinear Equations

The Rootfinding Problem

Problem Statement

Rootfinding Problem

Given a continuous scalar function of a scalar variable, find a real number r such that $f(r) = 0$.

- r is a **root** of the function f.
- The formulation $f(x) = 0$ is general enough; *e.g.*, to solve $g(x) = h(x)$, set $f = g - h$ and find a root of f.

Iterative Methods

- Unlike the earlier linear problems, the root cannot be produced in a finite number of operations.
- Rather, a sequence of approximations that formally converge to the root is pursued.

Iteration Strategy for Rootfinding. To find the root of f:

1. Start with an initial iterate, say x_0.
2. Generate a sequence of iterates $x_1, x_2, \ldots$ using an *iteration algorithm* of the form
$$x_{k+1} = g(x_k), \quad k = 0, 1, \ldots$$
3. Continue the iteration process until you find an x_i such that $f(x_i) = 0$. (In practice, continue until some member of the sequence seems to be "good enough".)

MATLAB's FZERO

`fzero` is MATLAB's general purpose rootfinding tool.

Syntax:

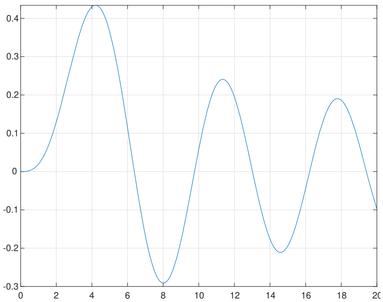
```
x_zero = fzero( <function>, <initial iterate> )
x_zero = fzero( <function>, <initial interval> )
[x_zero, fx_zero] = ....
```

Example

The roots of J_m, a Bessel function of the first kind, is found by

- Plot the function.
- Find approximate locations of roots.

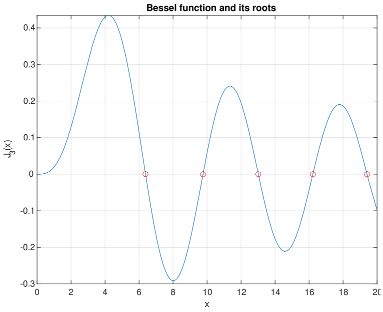
```
J3 = @(x) besselj(3,x);
fplot(J3,[0 20])
grid on
guess = [6,10,13,16,19];
```



Example (cont')

- Then use `fzero` to locate the roots:

```
omega = zeros(size(guess));
for j = 1:length(guess)
  omega(j) = fzero(J3,guess(j));
end
hold on
plot(omega,J3(omega),'ro')
```



Conditioning

- Sensitivity of the rootfinding problem can be measured in terms of the condition number:

$$\text{(absolute condition number)} = \frac{|\text{abs. error in output}|}{|\text{abs. error in input}|},$$

  where, in the context of finding roots of f,

  - input: f (function)
  - output: r (root)

- Denote the changes by:
  - error/change in input: ϵg, where $\epsilon > 0$ is small $\quad (f \mapsto f + \epsilon g)$
  - error/change in output: Δr $\quad (r \mapsto r + \Delta r)$

Conditioning (cont')

- The *perturbed equation*

$$f(r+\Delta r)+\epsilon g(r+\Delta r)=0$$

is linearized to (Taylor expansion)

$$f(r)+f'(r)\Delta r+g(r)\epsilon+g'(r)\epsilon\Delta r\approx 0,$$

ignoring $O((\Delta r)^2)$ terms[1].

- Since $f(r)=0$, we solve for Δr to get

$$\Delta r\approx -\epsilon\frac{g(r)}{f'(r)+\epsilon g'(r)}\approx -\epsilon\frac{g(r)}{f'(r)},$$

for small ϵ compared with $f'(r)$.

[1] That is, terms involving $(\Delta r)^2$ and higher powers of Δr

Conditioning (cont')

- Therefore, the absolute condition number of the rootfinding problem is

$$\kappa_{f \mapsto r} = \frac{1}{|f'(r)|},$$

  which implies that the problem is highly sensitive whenever $f'(r) \approx 0$.

- In other words, if $|f'|$ is small at the root, a computed *root estimate* may involve large errors.

Residual and Backward Error

- Without knowing the exact root, we cannot compute the error.
- But the **residual** of a root estimate $\tilde{r}$ can be computed:

$$\text{(residual)} = f(\tilde{r}).$$

- Small residual *might* be associated with a small error.
- The residual $f(\tilde{r})$ is the *backward error* of the estimate.

Multiple Roots

Definition 1 (Multiplicity of Roots)

Assume that r is a root of the differentiable function f. Then if

$$0 = f(r) = f'(r) = \cdots = f^{(m-1)}(r) \quad \text{but} \quad f^{(m)}(r) \neq 0,$$

we say that f has a root of **multiplicity** m at r.

- We say that f has a **multiple root** at r if the multiplicity is greater than 1.
- A root is called **simple** if its multiplicity is 1.
- If r is a multiple root, the condition number is infinite.
- Even if r is a simple root, we expect difficulty in numerical computation if $f'(r) \approx 0$.

Fixed Point Iteration

Fixed Point

Definition 2 (Fixed Point)

The real number r is a **fixed point** of the function g if $g(r) = r$.

- The rootfinding problem $f(x) = 0$ can always be written as a fixed point problem $g(x) = x$ by, *e.g.*, setting[2]

$$g(x) = x - f(x).$$

- The fixed point problem is true at, and only at, a root of f.

[2] This is not the only way to transform the rootfinding problem. More on this later.

Fixed Point Iteration

A fixed point problem $g(x) = x$ naturally provides an iteration scheme:

$$\begin{cases} x_0 = \text{initial guess} \\ x_{k+1} = g(x_k), \quad k = 0, 1, 2, \ldots. \end{cases} \qquad \text{(fixed point iteration)}$$

- The sequence $\{x_k\}$ may or may not converge as $k \to \infty$.
- If g is continuous and $\{x_k\}$ converges to a number r, then r is a fixed point of g.

$$g(r) = g\left(\lim_{k\to\infty} x_k\right) = \lim_{k\to\infty} g(x_k) = \lim_{k\to\infty} x_{k+1} = r.$$

Fixed Point Iteration Algorithm

```
function x = fpi(g, x0, n)
% FPI x = fpi(g, x0, n)
% Computes approximate solution of g(x)=x
% Input:
%   g    function handle
%   x0   initial guess
%   n    number of iteration steps
    x = x0;
    for k = 1:n
        x = g(x);
    end
end
```

Examples

- To find a fixed point of $g(x) = 0.3\cos(2x)$ near 0.5 using `fpi`:

```
g = @(x) 0.3*cos(2*x);
xc = fpi(g,0.5,20)
```

```
xc = 0.260266319627758
```

Not All Fixed Point Problems Are The Same

The rootfinding problem $f(x) = x^3 + x - 1 = 0$ can be transformed to various fixed point problems:

- $g_1(x) = x - f(x) = 1 - x^3$
- $g_2(x) = \sqrt[3]{1 - x}$
- $g_3(x) = \dfrac{1 + 2x^3}{1 + 3x^2}$

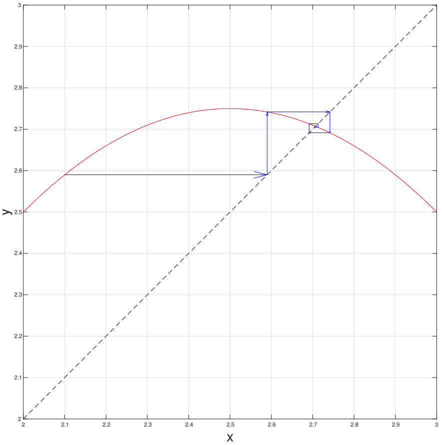
Note that all $g_j(x) = x$ are equivalent to $f(x) = 0$. However, not all these find a fixed point of g, that is, a root of f on the computer.

Exercise. Run `fpi` with g_j and $x_0 = 0.5$. Which fixed point iterations converge?

Geometry of Fixed Point Iteration

The following script[3] finds a root of $f(x) = x^2 - 4x + 3.5$ via FPI.

```
f = @(x) x.^2 - 4*x + 3.5;
g = @(x) x - f(x);
fplot(g, [2 3], 'r');
hold on
plot([2 3], [2 3], 'k--')
x = 2.1;
y = g(x);
for k = 1:5
    arrow([x y], [y y], 'b');
    x = y; y = g(x);
    arrow([x x], [x y], 'b');
end
```



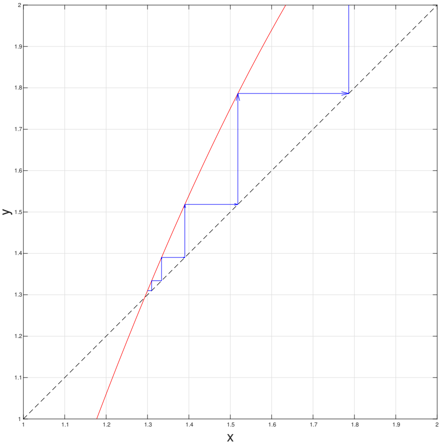
Note the line segments spiral in towards the fixed point.

[3]Modified from FNC.

Geometry of Fixed Point Iteration (cont')

However, with a different starting point, the process does not converge.

```
clf
fplot(g, [1 2], 'r');
hold on
plot([1 2], [1 2], 'k--'),
ylim([1 2])
x = 1.3; y = g(x);
for k = 1:5
    arrow([x y], [y y], 'b');
    x = y; y = g(x);
    arrow([x x], [x y], 'b');
end
```



Custom function: `arrow = @(p1, p2, varargin) quiver(p1(1), p1(2), p2(1)-p1(1), p2(2)-p1(2), 0, varargin{:})`

Series Analysis

To understand the difference of the two cases, use Taylor series expansions.

- Suppose r is a fixed point of g, the limit of $\{x_k\}$ generated by fixed point iteration:

$$x_{k+1} = g(x_k), \quad k = 1, 2, \dots$$

- Let $\epsilon_k = x_k - r$ for $k = 1, 2, \dots$.

Definition 3 (Linear Convergnece)

Let ϵ_k denote the error at step k of an iteration method. If

$$\lim_{k \to \infty} \frac{\varepsilon_{k+1}}{\varepsilon_k} = \sigma < 1,$$

the method is said to obey **linear convergence** with rate σ.

Series Analysis (cont')

Theorem 4 (Convergence of FPI)

Assume that g is continuously differentiable, that $g(r) = r$, and that $\sigma = \left|g'(r)\right| < 1$. Then the fixed point iterates x_k generated by

$$x_{k+1} = x_k, \quad k = 1, 2, \ldots,$$

converge linearly with rate σ to the fixed point r for x_0 sufficiently close to r.

In the previous example with $g(x) = x - f(x) = -x^2 + 5x - 3.5$:

- For the first fixed point, near 2.71, we get $g'(r) \approx -0.42$ (convergence);
- For the second fixed point, near 1.29, we get $g'(r) \approx 2.42$ (divergence).

Contraction Maps

Lipschitz Condition

A function g is said to satisfy a **Lipschitz condition** with constant L on the interval $S \subset \mathbb{R}$ if

$$\left|g(s) - g(t)\right| \leqslant L\,|s - t| \quad \text{for all } s, t \in S.$$

- A function satisfying the Lipschitz condition is continuous on S.
- If $L < 1$, g is called a **contraction map**.

When Does FPI Succeed?

Contraction Mapping Theorem

Suppose that g satisfies Lipschitz condition on S with $L < 1$, *i.e.*, g is a contraction map on S. Then S contains exactly one fixed point r of g. If $x_1, x_2, \ldots$ are generated by the fixed point iteration $x_{k+1} = g(x_k)$, and $x_1, x_2, \ldots$ all lie in S, then

$$|x_k - r| \leqslant L^{k-1} |x_1 - r|, \quad k > 1.$$