Recap of last lecture

① Calculate $\sqrt{a}$: $x_{t+1} = \frac{1}{2}\left(x_t + \frac{a}{x_t}\right)$

②. Computer Arithmetic : $x \in \mathbb{R} \to fl(x)$

- Float-point numbers, roundoff errors $\begin{cases} \varepsilon_m = 2^{-52} \\ \left|\frac{fl(x) - x}{x}\right| \le \frac{1}{2} \cdot \varepsilon_m \end{cases}$

double: $x = (-1)^s \cdot 1.b_1 b_2 \cdots b_{52} \times 2^m$, $m = (e_{11})_2 - 1023$

$\boxed{s \mid e_{1:11} \mid b_{1:52}}$ $e_{1:11} = \begin{cases} 0 & \{ \pm 0.\, b_{1:52} \times 2^{-1022} \to \{^{+0}_{-0} \\ 1\text{-}2046 & \to m \in [-1022, 1023] \\ 2047 & : \infty \text{ if } b_{1:52} = 0,\ \text{NaN if } b_{1:52} \ne 0 \end{cases}$

- loss of significant

③. Solve $f(x) = 0$

- Bisect (f, a, b, ε) : $f(a) \cdot f(b) < 0$.

$\begin{cases} 1.\ n \text{ iterations} : n+2 \text{ oracles} \\ 2.\ |r - c_n| \le \dfrac{b-a}{2^{n+1}} \end{cases}$

Today's lecture :

①. $\circ \in \{+, -, \times, \div\}$, $x \circ y$ v.s. $fl(x \circ y)$

$\sum_{i=1}^{n} x_i$: naive sum, pairwise sum

②. FPI, Newton's, Secant, Sensitivity

① IEEE 754, $x \to fl(x)$, $y \to fl(y)$

$x+y$ will have the following steps:

Streaming SIMD Ext. 2

1. load x and y into registers ← SSE2/AVX ← x86

Advanced Vector

MOVSD xmm0, [x]

MOVSD xmm1, [y]

2. align the exponents : $e_{\min} \to \min(e_x, e_y)$

3. +/- mantissas ← $(b_x + b_y)$
   - extra bits of precision x87 FPU uses 80 bits $\begin{cases} 1: \pm 1 \\ 15: \text{exp} \\ 64: \text{mantissa} \end{cases}$
   - Guard bits /extra bits in SSE2/AVX.

4. normalized → rounding → save

$$\begin{array}{ccl} x & & y \\ \downarrow & & \downarrow \\ fl(x) & & fl(y) \quad : \text{rounding (may use extra bits)} \end{array}$$

Reg. → SSE2/AVX : $fl(x) + fl(y) \to s$

$s \to fl(s)$: rounding, $fl(x \circ y) := fl(s)$

$$fl(x \circ y) = x \circ y (1+\delta), \quad |\delta| \le \begin{cases} 2^{-24} & : \text{single} \\ 2^{-53} & : \text{double} \end{cases}$$

- We assume $fl(x)$, $fl(y)$, $fl(x \circ y)$ are representable!

Rel. error does not make sense when it is overflow or underflow.

Summation : $\sum_{i=1}^{n} x_i$

- **Naive sum:**

Let $\{x_i\}_{i=1}^{n}$ be positive positive machine numbers in a computer whose unit roundoff error is ε.

$$s = \sum_{i=1}^{n} x_i \qquad \left| \frac{s - fl(s)}{s} \right| \le (1+\varepsilon)^n - 1 \approx n \cdot \varepsilon.$$

```
ŝ = 0
for i = 1, 2, ... n:
    ŝ += x_i
return ŝ
```

$$s_k = \sum_{i=1}^{k} x_i \Leftrightarrow \hat{s}_k$$

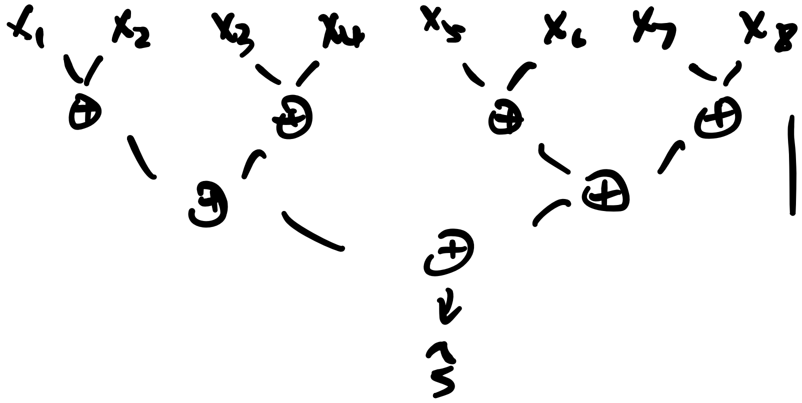
$$\begin{cases} s_1 = x_1 \\ s_{k+1} = s_k + x_{k+1} \end{cases} \qquad \begin{cases} \hat{s}_1 = x_1 \\ \hat{s}_{k+1} = fl(\hat{s}_k + x_{k+1}) \end{cases}$$

def. $\rho_k = \dfrac{\hat{s}_k - s_k}{s_k}$ $\qquad |\rho_k| \le (1+\varepsilon)^n - 1.$

$$= 1 + \binom{n}{1} \cdot \varepsilon + \binom{n}{2} \cdot \varepsilon^2 + \cdots$$

$$= O(n \cdot \varepsilon).$$

- **Pairwise sum:**



$$\left| \frac{s - \hat{s}}{s} \right| \le \frac{\varepsilon \cdot \log_2 n}{1 - \varepsilon \log_2 n} \cdot \left(\frac{\sum_{i=1}^{n} |x_i|}{\left|\sum_{i=1}^{n} x_i\right|} \right)$$

$$= O(\varepsilon \cdot \log_2 n).$$

If we assume rounding errors is random

- naive sum: $O(\varepsilon \sqrt{n})$
- pairwise sum: $O(\varepsilon \sqrt{\log n})$.

Further reading: Handbook of floating-point Arithmetic, Muller, 2018.

Lecture-02. Solving nonlinear equations

09/11/2024

In this lecture, we want to solve

$$f(x)=0, \quad x\in\mathbb{R}, \quad f\in C(\mathbb{R}) \text{ is continuous.}$$

We will assume $f(a)\cdot f(b)<0$ and root $r\in[a,b]$.

② Fixed Point Iteration (FPI)

(1). FPI: $x_{t+1}=g(x_t)$ $\to$ find fix point $x=g(x)$

$$g(x)=\begin{cases} f(x)+x \\ x-\dfrac{f(x)}{f'(x)} & \text{many ways} \\ \vdots \end{cases}$$

Main idea: Guess x_0, compute $g(x_0)$. If x_0 is near to r, then $g(x_0)$ is near to $g(r)$.

$$x_{t+1}=g(x_t) \quad \text{until} \begin{cases} |x_{t+1}-x_t|\le tol \\ |g(x_t)-x_t|\le tol \end{cases}$$

Convergence analysis: $e_t=x_t-r$. (r is a fix point)

$$e_{t+1}=x_{t+1}-r=g(x_t)-g(r) \quad \text{v.s.} \quad x_t-r=e_t.$$

By mean value thm. $g\in C'[a,b]$, $\exists\, \xi\in(a,b)$ s.t.

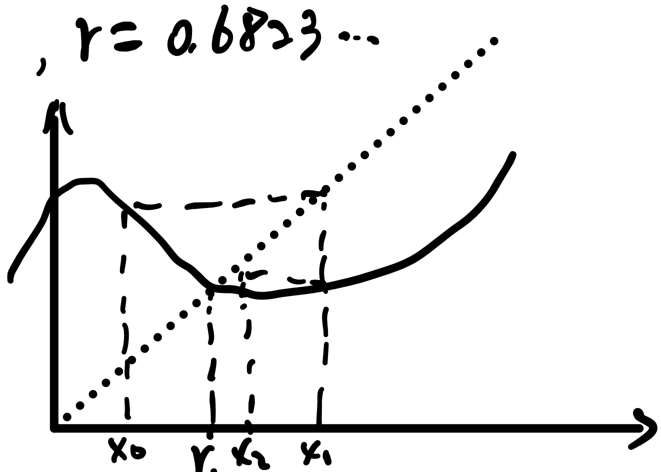
$$g'(\xi)=\frac{g(b)-g(a)}{b-a}. \text{ let } \begin{cases} b=x_t \\ a=r \\ \xi=\xi_t \end{cases}$$

$|g(x_t) - g(r)| = |g'(\xi_t)| \cdot |x_t - r|, \ \xi_t \in \text{conv}(r, x_t)$

$|e_{t+1}| = |g'(\xi_t)| \cdot |e_t|, \ t \geq 0.$

1. $|g'(\xi_t)| > 1$, $|e_{t+1}| > |e_t|$. diverge.
2. $|g'(\xi_t)| < 1$, $|e_{t+1}| < |e_t|$, converge.
3. $|g'(\xi_t)| = 1$. potentially stuck.

- Example: $f(x) = x^3 + x - 1 = 0$, $r = 0.6823\ldots$



- $g_1(x) = 1 - x^3 \Rightarrow g_1'(r) > 1$
- $g_2(x) = (1-x)^{\frac{1}{3}} \Rightarrow g_2'(r) < 1$
- $g_3(x) = \dfrac{1+2x^3}{1+3x^2} \Rightarrow g_3'(r) = 0$

Convergence: $\exists I \in [r-c, r+c]$, for some $c > 0$, s.t. $|g'(x)| < 1$ on I and $x_0 \in I$, then FPI will converge!

$$\lim_{t\to\infty} \left|\frac{e_{t+1}}{e_t}\right| = \lim_{t\to\infty} |g'(\xi_t)| = g'(r) \neq 0 < 1.$$

$|e_{t+1}| = c \cdot |e_t|$: linear convergence!

Iteration complexity:

We want to find t, s.t.

$|e_t| = |x_t - r| < tol.$ Assume $|g'(x)| \leq m < 1$ and $x_0 \in I = [x-c, x+c]$.

$|e_{t+1}| \leq m \cdot |e_t| \cdots \leq m^{t+1} \cdot |e_0|$

But $e_0 = x_0 - r$ is unknown!

Note that $|e_0| = |x_0 - r|$ and $|e_1| = |x_1 - r| < |x_0 - r|$

$$\Rightarrow |e_0| = |x_0 - r| = |x_0 - x_1 + x_1 - r|$$

$$\le |x_0 - x_1| + |e_1|$$

$$\le |x_0 - x_1| + m \cdot |e_0|$$

$\Rightarrow |e_0| \le |x_0 - x_1| / (1-m)$. So,

$\Rightarrow |e_t| \le m^t \cdot \dfrac{|x_0 - x_1|}{1-m}$. Given $|e_t| \le \varepsilon$.

$\Rightarrow m^t \cdot \dfrac{|x_0 - x_1|}{1-m} \le \varepsilon \Rightarrow k \ge \left\lceil \ln \dfrac{\varepsilon(1-m)}{|x_0 - x_1|} / \ln m \right\rceil$.

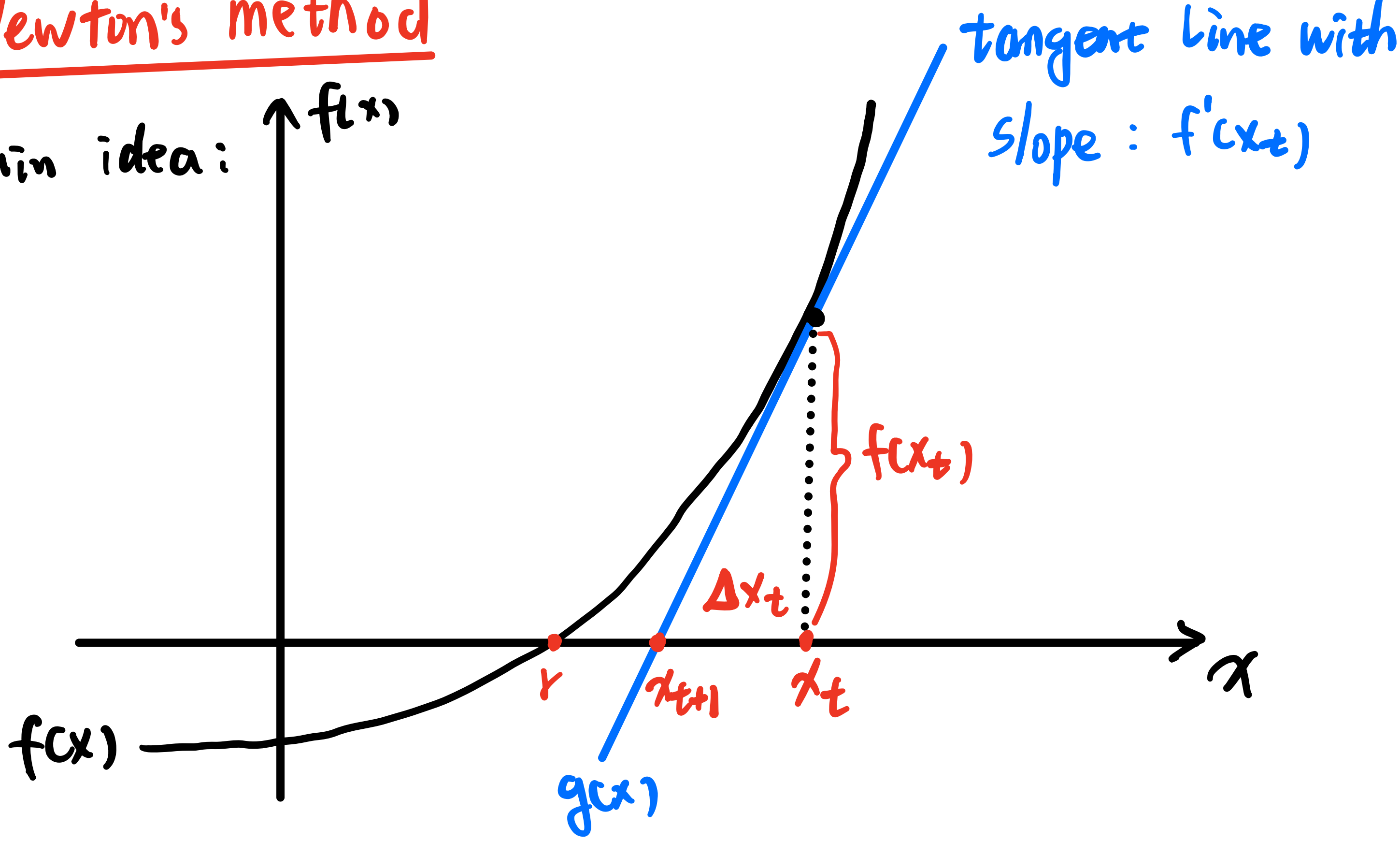
. Example : $x = g(x) = \cos x$ $|g'(x)| = |\sin x|$.

$m \approx 0.845$, $x_1 = \cos 1 \approx 0.5403$.

$\varepsilon = 0.5 \times 10^{-5} \Rightarrow k \ge 77$. (In worse case).

③ Newton's method

Main idea:



Approximate f(x) by linearization at x_t to obtain x_{t+1}.

To do ~~this~~ this, draw the tangent line at x_t and use the intersection point of this line and x-axis as an approximation root. To measure the slope of g(x):

$$\frac{f(x_t)}{\Delta x_t} = f'(x_t), \quad \text{where } \Delta x_t := x_t - x_{t+1}.$$

Then we have the following updates:

(2). $$x_{t+1} = x_t - \Delta x_t = x_t - \frac{f(x_t)}{f'(x_t)}$$ Newton's

An alternative way: the tangent line of g(x):

$y - f(x_t) = f'(x_t)\cdot(x - x_t)$. letting $y=0$, we will get the intersection point $\Rightarrow x = x_t - \frac{f(x_t)}{f'(x_t)} := x_{t+1}$.

Newton's algo: Given f and f', with $t=0,1,2,\cdots$, it generates $\{x_t\} \to r$, defined by (2).

Remark: Newton's algo. can be viewed as "optimal" fixed point iteration. Recall that FPI: $x_{t+1} = g(x_t)$, with

$$g(x) = x - \frac{f(x)}{f'(x)}, \quad g'(x) = 1 - \frac{f'(x)^2 - f(x)\cdot f''(x)}{f'(x)^2} = \frac{f(x)\cdot f''(x)}{f'(x)^2}.$$

So, assume that $f'(r) \neq 0$. $\Rightarrow g'(r) = \frac{f(r)\cdot f''(r)}{f'(r)^2} = 0.$

Note r is the root of $f(x)$: $f(r)=0$.

Error analysis: Recall $f(r)=0$, $e_t = x_t - r$, $g(x) = x - \frac{f(x)}{f'(x)}$.

We assume f'' is continuous and r is a simple root of f, i.e., $f(r) = 0 \neq f'(r)$.

From Newton's Iteration (2), we have

$$e_{t+1} = x_{t+1} - r = x_t - \frac{f(x_t)}{f'(x_t)} - r$$

We hope this part is small!

$$= e_t - \frac{f(x_t)}{f'(x_t)} = \frac{e_t\cdot f'(x_t) - f(x_t)}{f'(x_t)}$$

$$= \frac{f''(\xi_t)}{2f'(x_t)}\cdot e_t^2,$$ where the last eqn. follows by Taylor's theorem. Recall if f is two times differentiable, then

$$f(x) = f(a) + f'(a)\cdot(x-a) + \frac{f''(\xi)}{2}(x-a)^2,$$

$\xi \in \mathrm{conv}(x,a)$. Here $\begin{cases} x := r & e_t = x_t - r \\ a := x_t & \xi_t \in \mathrm{conv}(r, x_t). \end{cases}$

$$0 = f(r) = f(x_t) + f'(x_t)\cdot(-e_t) + \frac{f''(\xi_t)}{2}(-e_t)^2.$$

It then leads to $e_{t+1} = \frac{f''(\xi_t)}{2f'(x_t)}\cdot e_t^2$.

If we assume further that

$$\begin{cases} 1. & f'(x) \neq 0 \text{ for all } x \in [r-|e_0|, r+|e_0|] := I \\ 2. & f''(x) \text{ is continuous for all } x \in I. \\ 3. & \exists M = \frac{1}{2}\left(\sup_{x\in I}|f''(x)|\right)\left(\sup_{x\in I}\frac{1}{|f'(x)|}\right) \text{ s.t. } M|e_0|<1. \end{cases}$$

If the above conditions hold, then

$$|e_{t+1}| \le M\cdot|e_t|^2.$$

- To estimate M, given $\delta>0$, let's define $M(\delta)$:

$$M(\delta) = \frac{1}{2}\cdot\frac{\max_{|x-r|\le\delta}|f''(x)|}{\min_{|x-r|\le\delta}|f'(x)|}. \quad \text{Since } f''\in C(\mathbb{R}),\ f'(r)\neq 0.$$

One can choose δ s.t. $\delta\cdot M(\delta)<1$. This is possible as $\delta\to 0$, $M(\delta)\to\frac{|f''(r)|}{|2f'(r)|}$. Hence, $\delta\cdot M(\delta)\to 0$.

Assume we start a point x_0 satisfy $|x_0-r|\le\delta$. Then $|e_0|\le\delta$.

$$|e_1| = |x_1-r| = \frac{1}{2}\frac{|f''(\xi_0)|}{|f'(\xi_0)|}\cdot e_0^2 \le M(\delta)\cdot e_0^2$$
$$= |e_0|\cdot|e_0|\cdot M(\delta)$$
$$\le |e_0|\cdot\delta\cdot M(\delta) \le |e_0| \le \delta.$$

$\Rightarrow |x_1-r|\le\delta$, $x_1\in(r-\delta, r+\delta)$.

$\Rightarrow |e_t| \le [\delta\cdot M(\delta)]^t\cdot|e_0|. \Rightarrow \lim_{t\to\infty}|e_t| = 0.$

Convergence of Newton's method: Let's assume $f'' \in C(\mathbb{R})$, and let r be a simple root of f. Then there is a neighbor of r and a constant M s.t. if Newton's starts in that neighbor, the points $\{x_t\}$ become steadily closer to r and satisfy $|e_{t+1}| \le M \cdot |e_t|^2$. $(t \geq 0)$.

Compute $\sqrt{a}$ in our last lecture:

$$x_{t+1} = \frac{1}{2}\left(x_t + \frac{a}{x_t}\right).$$

Define $f(x) = x^2 - a$, to find $\sqrt{a}$ is to find $f(r) = 0$.

f', f'' are continuous and $f'(r) = 2r = 2\sqrt{a} \neq 0$.

Newton's: $x_{t+1} = x_t - \dfrac{f(x_t)}{f'(x_t)} = x_t - \dfrac{x_t^2 - a}{2x_t} = \dfrac{1}{2}\left(x_t + \dfrac{a}{x_t}\right)$.

To estimate M: $M = \left|\dfrac{f''(r)}{2f'(r)}\right| = \dfrac{1}{2r} = \dfrac{1}{2\sqrt{a}}$.

Handle multiple roots:

Def. $r \in \mathbb{R}$ is a root of multiplicity m of $f(x)$ if there is a polynomial $s(x)$ s.t. $s(r) \neq 0$ and $f(x) = (x-r)^m \cdot s(x)$.

$$\begin{cases} m = 1: & r \text{ is a simple root.} \\ m \geq 2: & r \text{ is a multiple root.} \end{cases}$$

$\Leftrightarrow$ $f(r) = 0$, $f^{(i)}(r) = 0$, $i = 1, 2, \cdots, m-1$, $f^{(m)}(r) \neq 0$.

Consider $f(x) = x^m$ $(m \geq 2)$, f has multiple roots at $x=0$.
$f'(r) = m \cdot x^{m-1} = 0$. Above analysis for simple root does not work for this case!

$$x_{t+1} = x_t - \frac{f(x_t)}{f'(x_t)} = x_t - \frac{x_t^m}{m \cdot x_t^{m-1}} = \frac{m-1}{m} \cdot x_t$$

$\Rightarrow e_{t+1} = \frac{m-1}{m} \cdot e_t \quad \Rightarrow e_t = \left(\frac{m-1}{m}\right)^t \cdot e_0$. It is slow.

Q: In the multiple case, can we fix Newton's so that it converges faster?

If $f(x) = (x-r)^m \cdot s(x)$, then we have

$$|e_{t+1}| \lessapprox \frac{m-1}{m} \cdot |e_t|.$$ To show this,

$$x_{t+1} = x_t - \frac{f(x_t)}{f'(x_t)} \Leftrightarrow e_{t+1} = e_t - \frac{f(x_t)}{f'(x_t)}$$

$$e_{t+1} = e_t - \frac{(x_t-r)^m \cdot s(x_t)}{m(x_t-r)^{m-1} \cdot s(x_t) + (x_t-r)^m \cdot s'(x_t)}$$

$$= e_t - \frac{e_t \cdot s(x_t)}{m \cdot s(x_t) + s'(x_t) \cdot e_t}$$

$$= \left[\frac{(m-1) \cdot s(x_t) + s'(x_t) \cdot e_t}{m \cdot s(x_t) + s'(x_t) \cdot e_t}\right] \cdot e_t$$

$$= \frac{(m-1) \cdot f(x_t)}{f'(x_t)} + \frac{s'(x_t)}{m \cdot s(x_t) + e_t \cdot s'(x_t)} \cdot e_t^2$$

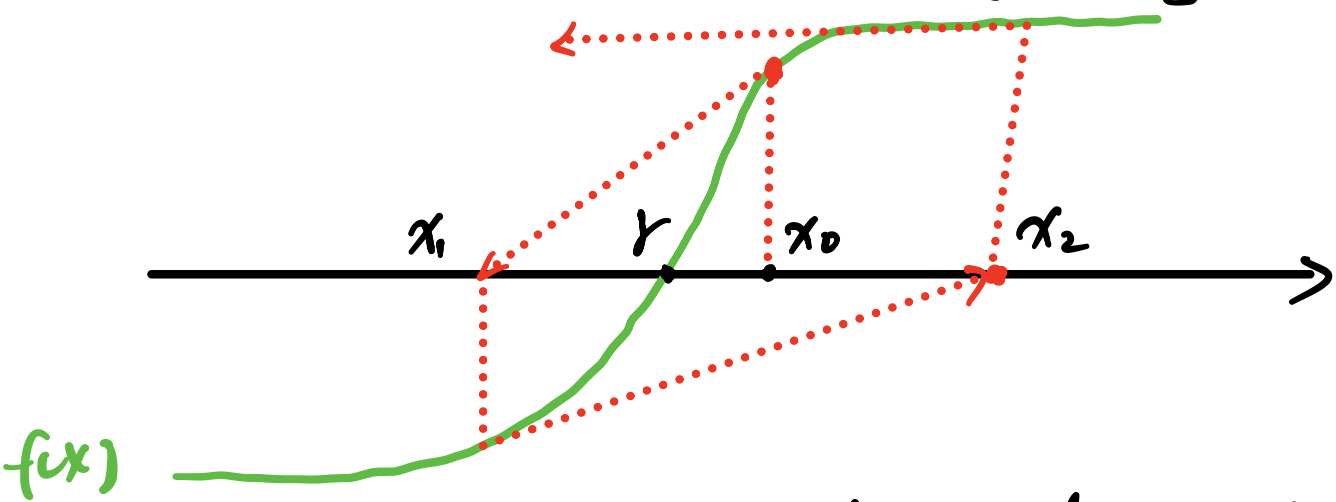
Fixing it: Just need to add $-\frac{(m-1) \cdot f(x_t)}{f'(x_t)}$ on right side

$\Rightarrow e_{t+1} = O(e_t^2)$. So, a fixed version is:

$$x_{t+1} = x_t - \frac{m \cdot f(x_t)}{f'(x_t)}.$$

Summary of Newton's:

- For simple root, it converges quadratically when x_0 is close enough to r.
- For multiple roots, it converges linearly but can be fixed.
- When x_0 is far from r, it may diverge!

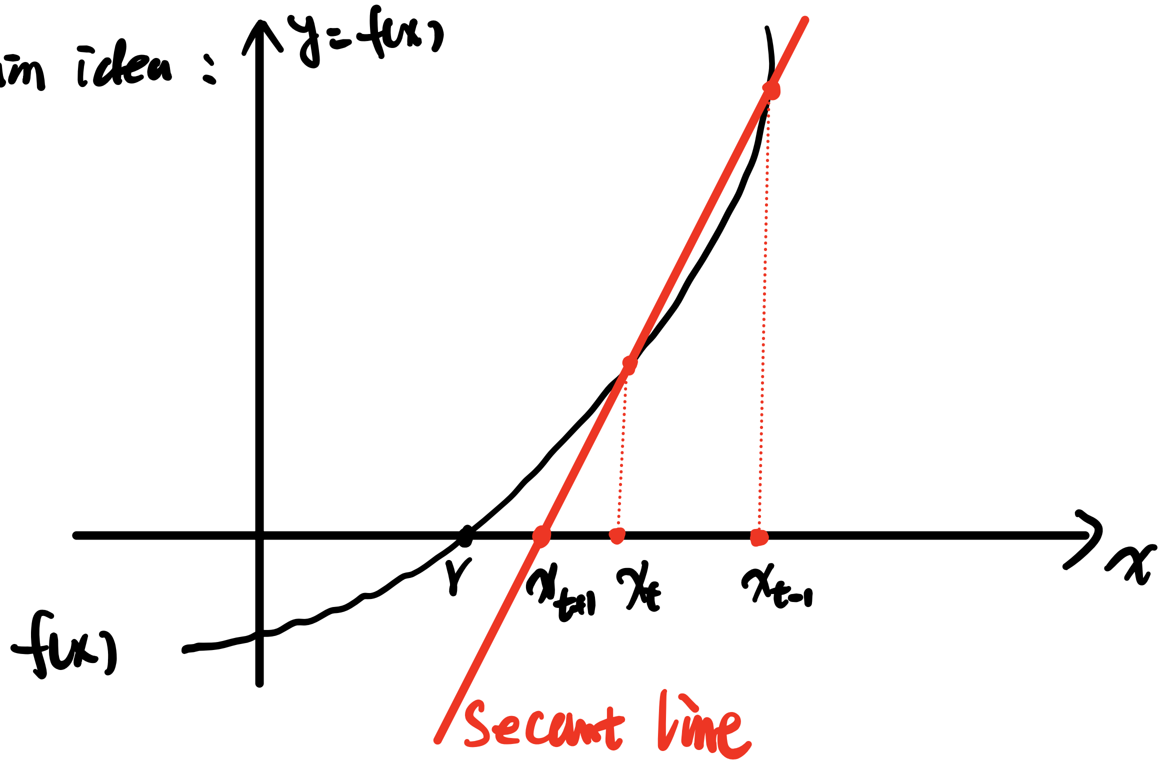


- One can use Bisect to find a good x_0 and then apply Newton's.
- One needs to have $f'(x)$ and $f'(x) \neq 0$!

Q: If $f'(x)$ is hard to obtain, can we find a good method s.t. it is superlinear?

④ Qusi-Newton method: Secant method

Main idea:



To approximate the derivative at the point x_t with previous point x_{t-1}.

$$f'(x_t) \approx \frac{f(x_t)-f(x_{t-1})}{x_t - x_{t-1}} \quad : \text{difference quotient.}$$

(Note $f'(x) = \lim_{u \to x} \frac{f(x)-f(u)}{x-u}$). Hence, we have

$$x_{t+1} = x_t - f(x_t)\cdot\left[\frac{x_t - x_{t-1}}{f(x_t)-f(x_{t-1})}\right] \quad \text{Secant}$$

$$= \frac{x_{t-1}\cdot f(x_t) - x_t \cdot f(x_{t-1})}{f(x_t) - f(x_{t-1})}.$$

Per-iteration only requires 1 function evaluation.

Error analysis: Consider $e_t = x_t - r$ and $f(r)=0$, r is simple.

$$e_{t+1} = x_{t+1} - r = \frac{x_{t-1}\cdot f(x_t) - x_t\cdot f(x_{t-1})}{f(x_t)-f(x_{t-1})} - r = \frac{e_{t-1}\cdot f(x_t) - e_t \cdot f(x_{t-1})}{f(x_t)-f(x_{t-1})}$$

It leads to $e_{t+1} = \dfrac{f(x_t)/e_t - f(x_{t-1})/e_{t-1}}{f(x_t) - f(x_{t-1})} \cdot e_t e_{t-1}$

Def. $h(x) = \begin{cases} \dfrac{f(x)-f(r)}{x-r}, & x \neq r \\ f'(r), & x = r \end{cases} \Rightarrow h'(x) = \begin{cases} \dfrac{f'(x)(x-r) - f(x)}{(x-r)^2} & x \neq r \\ \frac{1}{2} f''(r) & x = r. \end{cases}$

$$= \frac{h(x_t) - h(x_{t-1})}{f(x_t) - f(x_{t-1})} \cdot e_t e_{t-1}$$

$$\frac{h(x_t)-h(x_{t-1})}{f(x_t)-f(x_{t-1})} = \frac{h(x_t)-h(x_{t-1})}{x_t - x_{t-1}} \cdot \frac{x_t - x_{t-1}}{f(x_t)-f(x_{t-1})} = \frac{h'(\eta_t)}{f'(\xi_t)}$$

$\eta_t, \xi_t \in \mathrm{conv}(x_t, x_{t-1})$. We assume $f(x) \in C^2[\mathbb{R}]$.

$$f(x) = f(a) + f'(a)\cdot(x-a) + \frac{f''(\xi)}{2}(x-a)^2,$$

$\xi \in \mathrm{conv}(x, a)$. Here $\begin{cases} x := r \\ a := \eta_t, \ \delta_t \in \mathrm{conv}(r, \eta_t). \end{cases}$

$$\Rightarrow f(r) = f(\eta_t) + f'(\eta_t)(r-\eta_t) + \frac{(r-\eta_t)^2}{2} \cdot f''(\delta_t)$$

$$\frac{1}{2} f''(\delta_t) = \frac{f'(\eta_t)(\eta_t - r) - f(\eta_t)}{(\eta_t - r)^2} = h'(\eta_t).$$

$$\Rightarrow e_{t+1} = \frac{f''(\delta_t)}{2f'(\xi_t)} \cdot e_t \cdot e_{t-1}, \quad \begin{cases} \delta_t \in \mathrm{conv}(x_t, x_{t-1}, r) \\ \xi_t \in \mathrm{conv}(x_t, x_{t-1}). \end{cases}$$

Note $x_t - x_{t-1} = e_t - e_{t-1}$.

If x_0, x_1, x_2 are close enough to r and

$\left| \dfrac{f''(\delta_t)}{2f'(\xi_t)} \right| \leq M$ is properly bounded. Suppose that

$|e_1| \le \frac{1}{2M}$ and $|e_0| \le \frac{1}{2M}$. Then $|e_2| \le M \cdot |e_1| \cdot |e_0|$

$\le \frac{1}{2}|e_0|$ and $\le \frac{1}{2}|e_1|$ $\Rightarrow$

$$|e_2| \le \frac{1}{2}\min\{|e_0|, |e_1|\} \le \frac{1}{2^2 \cdot M}.$$

By induction, $|e_{t+1}| \le \cdots \le \frac{1}{2^{t+1} \cdot M}$, $\{e_t\} \searrow$.

$$\left|\frac{f''(\xi_t)}{2f'(\zeta_t)}\right| \to \left|\frac{f''(r)}{2f'(r)}\right| \triangleq C.$$

$|e_{t+1}| \sim C \cdot |e_t| \cdot |e_{t-1}|$. We assume that $|e_{t+1}| \sim A |e_t|^\alpha$,

$A > 0$, meaning $\lim\limits_{t\to\infty} \frac{|e_{t+1}|}{A \cdot |e_t|^\alpha} = 1.$ $\Rightarrow$

$\Leftrightarrow$ $A|e_t|^\alpha \sim C \cdot |e_t| \cdot |e_{t-1}|$

$\Rightarrow$ $|e_t|^{\alpha-1} \sim C \cdot A^{-1} \cdot |e_{t-1}|$

$\Rightarrow$ $|e_t| \sim (C \cdot A^{-1})^{\frac{1}{\alpha-1}} \cdot |e_{t-1}|^{\frac{1}{\alpha-1}}$

$\Rightarrow$ $A^{1+\frac{1}{\alpha-1}} = C^{\frac{1}{\alpha-1}}$, $\alpha = \frac{1}{\alpha-1}$ ($\alpha > 0$, $\Rightarrow \alpha = \frac{1+\sqrt{5}}{2}$)

$\approx 1.6180\ldots$

$|e_{t+1}| \sim C \cdot |e_t| \cdot |e_{t-1}| \Leftrightarrow A = C^{\frac{1}{\alpha}} = C^{\frac{\sqrt{5}-1}{2}}$

That is $|e_{t+1}| \sim C^{\frac{\sqrt{5}-1}{2}} \cdot |e_t|^{\frac{1+\sqrt{5}}{2}}$

$$\approx \left|\frac{f''(r)}{2f'(r)}\right|^{\frac{\sqrt{5}-1}{2}} \cdot |e_t|^{\frac{1+\sqrt{5}}{2}}$$

Multiple roots:

One can reduce to the following error iteration:

$$e_{t+1} = \frac{e_{t-1} e_t^m - e_t e_{t-1}^{m-1} + O(|e_{t-1}|^{m+2})}{e_t^m - e_{t-1}^m + O(|e_{t-1}|^{m+1})} \quad (m \geq 2)$$

$\Leftrightarrow \begin{cases} 1.\ m=2, \text{ it is a Fibonacci Sequence: } |e_{t+1}| \sim (\alpha-1)\cdot|e_t| \\ 2.\ m \geq 3,\ |e_{t+1}| \sim \lambda |e_t| \text{ with } \\ \qquad \lambda\cdot(\lambda-1)\cdot(\lambda^m + \lambda^{m-1} - 1) = 0,\ \lambda \in (0,1). \end{cases}$

⑤ Stability of solving nonlinear equation:

To design the stop conditions of above methods, one can consider two types of errors:

- Forward error (FE): $|x_t - r|$
- Backward error (BE): $|f(x_t) - f(r)| = |f(x_t)|$.

To get approximate x_t, the forward error comes from the algorithm while the backward error comes from f itself.

- **Example 1:** $f(x) = (x - \frac{2}{3})^3 = x^3 - 2x^2 + \frac{4}{3}x - \frac{8}{27}$

After 16th iteration of Bisect:

BE $\approx 2.0 \times 10^{-16}$ while FE $\approx 10^{-5}$.

why? Note $|f(x_t)| = (x_t - \frac{2}{3})^3 = e_t^3$.

- Example 2: $f(x) = \sin(x) - x$. Let $x_t = 10^{-3}$, $r = 0$

BE : $|f(x_t)| = |\sin(0.001) - 0.001|$

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \cdots$$

$|f(x_t)| = |\sin(x_t) - x_t| = O(x_t^3) \approx 1.67 \times 10^{-10}$.

FE $= |x_t - r| = 10^{-3}$.

Sensitivity: Small floating point errors in the function f translate into large errors in the root! It is called a sensitive numerical problem.

To measure the sensitivity, assume $f(r) = 0$. there is a small change is made on $f(x)$: $\varepsilon \cdot g(x)$ where ε is small value. We actually find:

$$f(x) + \varepsilon \cdot g(x) = 0. \qquad \text{Def. } F(x) = f(x) + \varepsilon \cdot g(x)$$

Let $r + \Delta r$ is the root of $F(x)$. That is

$F(r+\Delta r) = f(r+\Delta r) + \varepsilon \cdot g(r+\Delta r) = 0$,

where Δr is the change of r due to small error ε.

Use Taylor polynomial of f:

$$\begin{cases} f(r+\Delta r) = f(r) + \Delta r \cdot f'(r) + O(|\Delta r|^2) \\ g(r+\Delta r) = g(r) + \Delta r \cdot g'(r) + O(|\Delta r|^2). \end{cases}$$

So, $0 = f(r+\Delta r) + \varepsilon \cdot g(r+\Delta r)$

$= \Delta r \cdot f'(r) + \Delta r \cdot \varepsilon \cdot g'(r) + \varepsilon \cdot g(r) + O(|\Delta r|^2)$

$\Rightarrow \Delta r \cdot (f'(r) + \varepsilon \cdot g'(r)) \approx -\varepsilon \cdot g(r).$

$\Rightarrow \Delta r \approx \dfrac{-\varepsilon g(r)}{f'(r) + \varepsilon \cdot g'(r)} \approx \boxed{-\varepsilon \cdot \dfrac{g(r)}{f'(r)}}$ Sensitivity of roots.

(assume $\varepsilon \cdot g'(r) \ll f'(r)$).

Example: Say we want to estimate polynomial $p(x) = \prod_{i=1}^{6} (x-i)$, there are system errors s.t. the final estimate is $p(x) + \varepsilon \cdot x^7$. Find sensitivity.

Solution: Let $f(x) = \prod_{i=1}^{6} (x-i)$, $\varepsilon = -10^{-6}$ and $g(x) = x^7$

$\Delta r \approx -\varepsilon \cdot \dfrac{g(r)}{f'(r)} = -\dfrac{\varepsilon \cdot 6^7}{5!} = -2332.8 \cdot \varepsilon$

(Note $f'(x) = \sum_j (x-6) \cdot h_j(x) + \prod_{i=1}^{5} (x-i)$, So $f'(6) = 5!$

$10^{-6} \cdot 6^7 \approx 0.2799.$

The estimated root is $r + \Delta r = 6.0023328.$

$\Rightarrow$ 6 digits of $f(x)$ will cause 3 digits change error of root!

Error magnificant Factor (EMF):

Idea: when bad cases happen, quantity FE/BE be large.

So, a reasonable way to measure this:

$$\text{EMF} = \left| \frac{\text{rel. FE}}{\text{rel. BE}} \right| = \left| \frac{\Delta r / r}{\varepsilon \cdot g(r) / g(r)} \right|, \quad \frac{\Delta r}{r} \approx \left| \frac{\varepsilon \cdot g(r)}{r \cdot f'(r)} \right|$$

$$= \left| \frac{g(r)}{r \cdot f'(r)} \right|$$

Note: rel. BE: $\left| \frac{f(r) + \varepsilon \cdot g(r) - f(r)}{g(r)} \right|$ is relative to g.

EMF is highly related to <u>condition number</u>.

- <u>Example</u>: $W(x) = \prod_{i=1}^{20} (x-i)$. use the sensitivity formula to estimate the root change in x^{15} term of $W(x)$ on root $r = 16$. Then find EMF.

$W_\varepsilon(x) = W(x) + \varepsilon \cdot g(x)$, $g(x) = -1{,}672{,}280{,}820 \cdot x^{15}$

$W'(x=16) = 15!\,4!$

$\Delta r \approx \left| \frac{\varepsilon \cdot g(x)}{f'(r)} \right| \approx 6.1432 \times 10^{13} \cdot \varepsilon \quad (\varepsilon_m = \pm 2.22 \times 10^{-16})$

$\Delta r \approx \pm 0.0136.$

$$\left| \frac{g(r)}{r \cdot f'(r)} \right| = \frac{16^{15} \cdot 1{,}672{,}280{,}820}{15!\,4!\,16} \approx 3.8 \times 10^{12}.$$

$\left(W(x) = (x-16) \cdot q(x), \quad q(x) = \prod_{i=1}^{15} (x-i) \cdot \prod_{j=17}^{20} (x-j) \right.$

$\left. W'(x) = q(x) + (x-16) \cdot q'(x), \Rightarrow W'(16) = q(16) \right).$