By the similarity of the triangles

$$\frac{(x_1 - x_2)}{f(x_1)} = \frac{(x_0 - x_1)}{f(x_0) - f(x_1)}$$

Solving for x_2

$$x_2 = x_1 - f(x_1)\,\frac{(x_0 - x_1)}{f(x_0) - f(x_1)}$$

This expression can be used to repeatedly generate a sequence of approximations to the root

$$x_{n+1} = x_n - f(x_n)\,\frac{x_{n-1} - x_n}{f(x_{n-1}) - f(x_n)}, \quad n = 1 \ldots$$

ALGORITHM

start with x_0, x_1 "sufficiently close" to the root

if $|f(x_0)| < |f(x_1)|$ swap x_0 and x_1

repeat

set $x_2 := x_1 - f(x_1)\,\dfrac{x_0 - x_1}{f(x_0) - f(x_1)}$

set $x_0 := x_1$

set $x_1 := x_2$

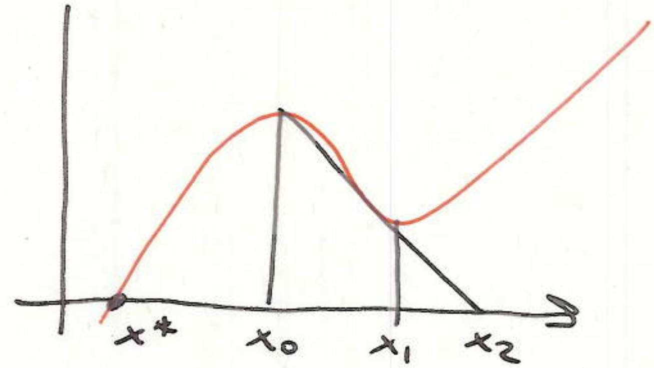
until $|f(x_2)| <$ tolerance (or $|x_2 - x_0| < 2\varepsilon$)

Remarks

* as before, single (new) function evaluation per iteration

* in general, convergence faster than for the bisection method, but no convergence in regions where $f(x)$ is poorly approximated by a linear function

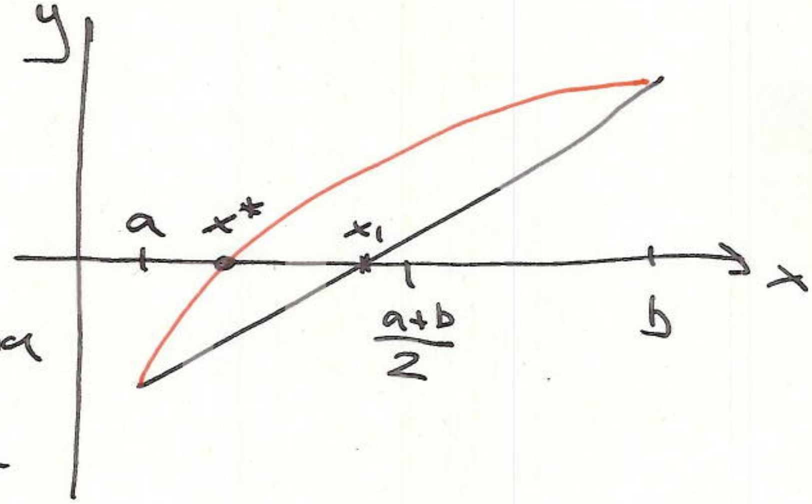
* for linear functions $f(x)$, solution found in a single iteration



REGULA FALSI (false position) METHOD

Lack of guaranteed convergence of the secant method can be remedied (still using linear approximation) by ensuring the root remains bracketed at all times.

Regula Falsi is similar to bisection, but the new iterate is taken at the intersection of the linear interpolant and the Ox axis, rather than at the midpoint $\Rightarrow$ faster convergence



ALGORITHM

Start with x_0 and x_1 such that $f(x_0)\cdot f(x_1)<0$

repeat

  set $x_2 := x_1 - f(x_1)\dfrac{x_0-x_1}{f(x_0)-f(x_1)}$

  if $f(x_2)\cdot f(x_0) < 0$ then

    set $x_1 := x_2$

  else

    set $x_0 := x_2$

  end if

until $|f(x_2)| <$ tolerance

Remarks

* Regula Falsi is more robust, but has slower convergence than the secant method (typical trade-off)

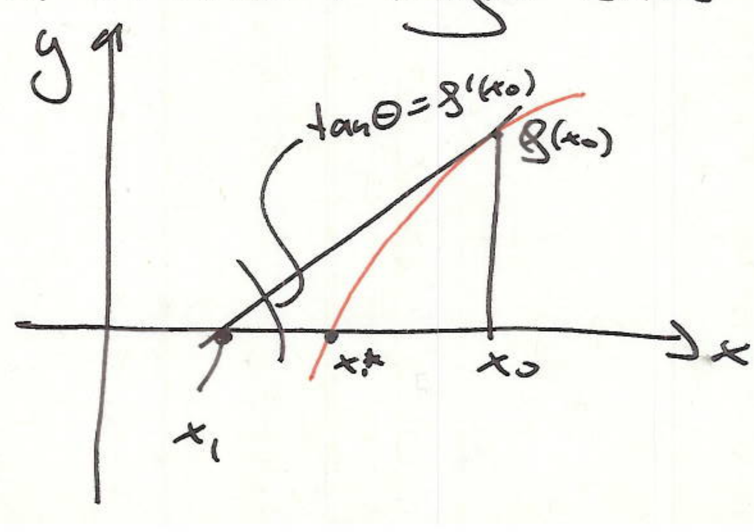
NEWTON'S METHOD

~~Newton~~ Still faster methods can be designed using more information about the function (e.g., its derivative)

Newton's method relies on a truncated Taylor series expansion

$$f(x^*) \cong f(x_0) + f'(x_0)(x^*-x_0)$$

$$\overset{\shortparallel}{0} \qquad x^* = x_0 - \frac{f(x_0)}{f'(x_0)}$$



Using this expression repeatedly

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n=1,2,\ldots$$

REMARKS

* method still faster to converge, but two evaluations (function and its derivative) are required at every iteration

* unlike for the previous methods, the function $f(x)$ must be differentiable; if the function derivative is not available analytically, it can be calculated approximately, e.g., as (we will learn about this more in due course)

$$f'(x_n) \cong \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}},$$

so that we obtain

$$x_{n+1} = x_n - f(x_n) \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

(i.e., the secant method)

* Newton's method will break down when $f'(x_n) = 0$ and $f(x_n) \neq 0$

FIXED-POINT ITERATION

Rearrange the equation $f(x) = 0$ into an equivalent form $x = g(x)$ (often there is more than one way to do this). If r is a root of $f(r) = 0$, then it is also the root of $r = g(r)$, called the fixed point (fixed-point problems play an important role in mathematical analysis). Under certain conditions (to be made precise later), the iterations

$x_{n+1} = g(x_n)$, $n=1,2,\ldots$

converges to the fixed point r, i.e., the root of $f(x)=0$.

Ex. Consider the equation

$$f(x) = x^2 - 2x - 3 = 0$$

where $f(x) = (x+1)(x-3)$

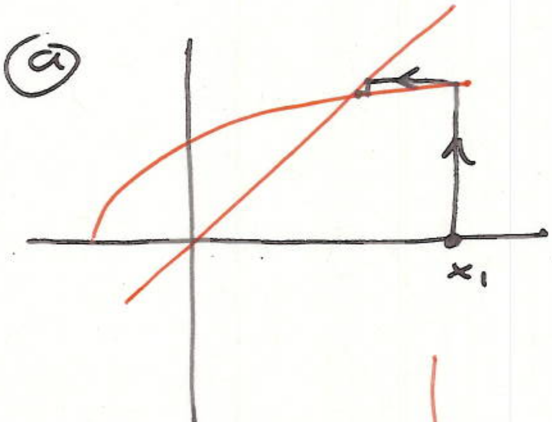
Possible rearrangements

(a) $x = \sqrt{2x+3} \triangleq g_1(x)$

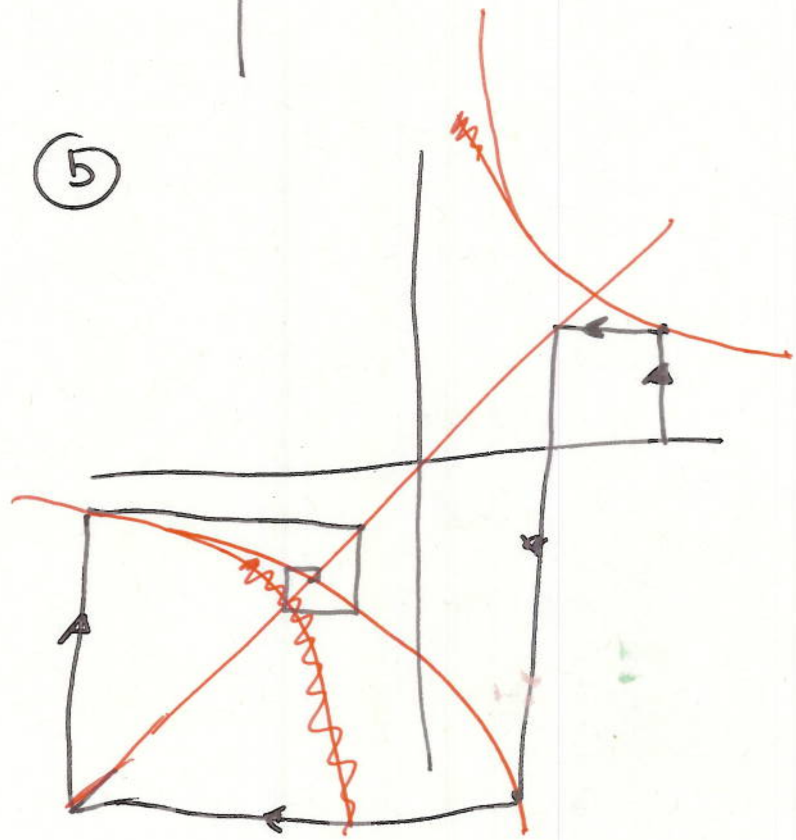
(b) $x = \dfrac{3}{x-2} \triangleq g_2(x)$

(c) $x = \dfrac{x^2-3}{2} \triangleq g_3(x)$

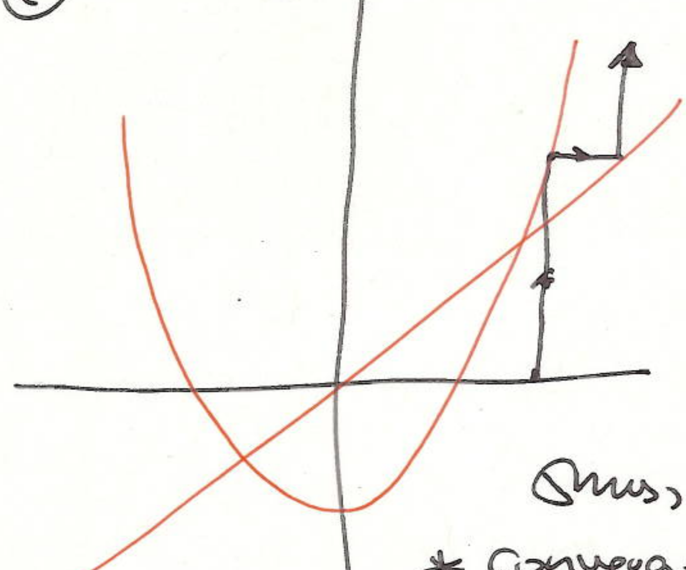
(a)



(b)



(c)



Thus, we may have:

* convergence to a root / one of the roots
* no convergence at all

How to choose the best rearrangement