

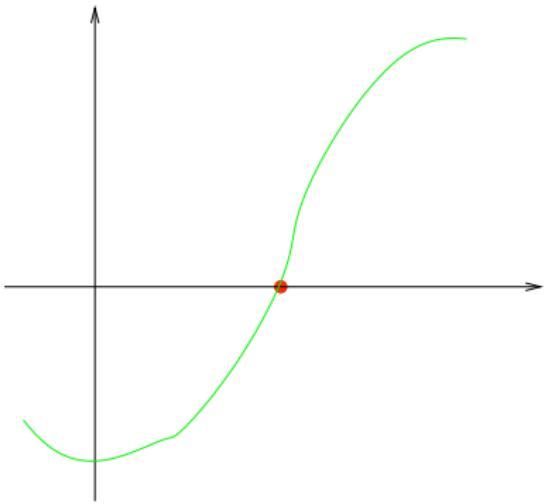
Les équations algébriques

La méthode de la dichotomie

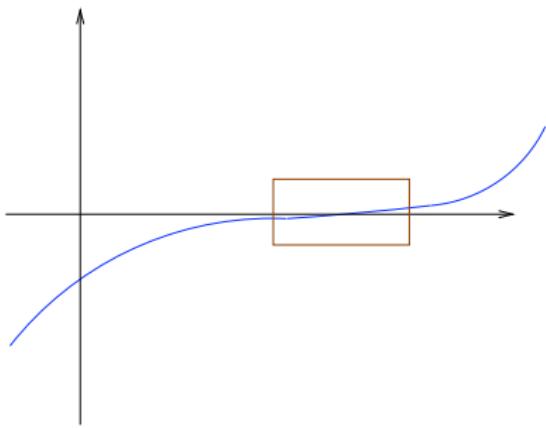
Soit f une fonction C^∞ de $[a, b] \subset \mathbb{R}$ dans \mathbb{R} . On cherche une racine de l'équation

$$f(x) = 0$$

- ▶ On cherche une valeur numérique, i.e. un nombre flottant..
- ▶ Une équation de ce type peut avoir aucune, une ou plusieurs solutions.
- ▶ L'existence d'une solution est un problème mathématique (i.e. théorique).
- ▶ On ne cherche à calculer que des racines dont on est sur de l'existence.



(a) Situation favorable



(b) Situation défavorable

Méthode de la dichotomie

Soit f une fonction C^∞ de $[a, b] \subset \mathbb{R}$ dans \mathbb{R} telle que $f(a) * f(b) < 0$.

Nous cherchons à résoudre $f(x) = 0$.

D'après le théorème des valeurs intermédiaires, $\exists e \in [a, b]$ tel que $f(e) = 0$.

On se fixe $\varepsilon > 0$ et on pose $a_0 = a$ et $b_0 = b$.

On considère le milieu m de l'intervalle : $m = \frac{b_0 + a_0}{2}$

Si $f(m)$ est du signe de $f(a_0)$ alors on pose $a_1 = m$ et $b_1 = b_0$,
sinon $a_1 = a_0$ et $b_1 = m$.

On réitère l'opération jusqu'à ce que $b_k - a_k < \varepsilon$.

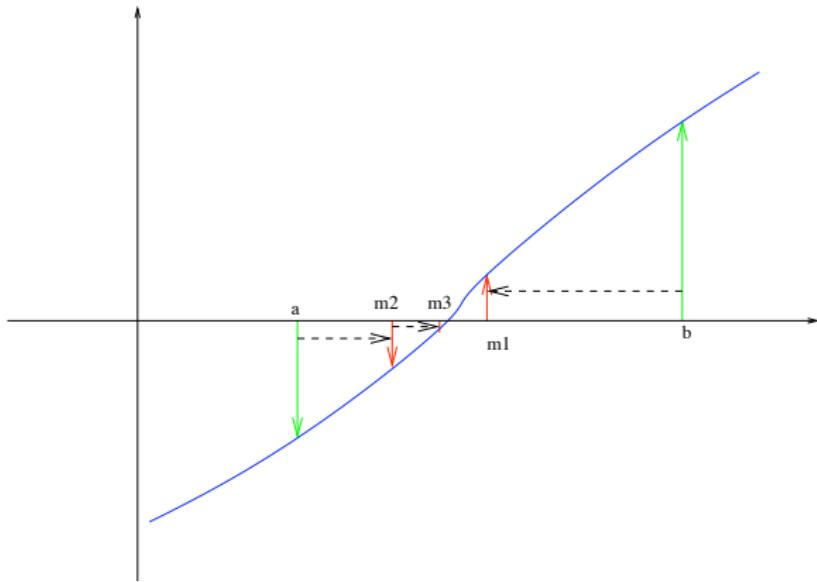


Figure – Méthode de la dichotomie avec m_i, \dots les milieux successifs

Caractéristiques de la méthode de dichotomie

$$b_{n+1} - a_{n+1} = \frac{b_n - a_n}{2}.$$

La convergence est linéaire en $\frac{1}{2^n}$

On gagne un bit de précision à chaque itération

Le nombre d'itérations est connu et vaut $\left\lceil \log_2 \frac{b-a}{\varepsilon} \right\rceil$.

Il faut que la fonction change de signe au voisinage de la racine donc les racines d'ordre pair échappent à la méthode.

Le passage d'une fonction en paramètre en C

Soit la déclaration d'une fonction réelle de variable réelle :

```
float f(float x1, float x2)
{
    float y;
    .....
    return y;
}
```

Le passage en paramètre de la fonction *f* dans une autre fonction ou procédure s'écrit :

```
float exemple1(float (*f) (float, float), .... )
{
    .....
}
```

Un code en C

```
float bissec(float (*f) (float), float a, float b, float eps)
{
    float a1, b1, m;
    if (b < a)
        { a1 = b; b1 = a;}
    else
        { a1 = a; b1 = b;};
    if (f(a1).f(b1) > 0) exit(0);
    do
    {
        m = (a1+b1)/2;;
        if (f(a1).f(m) < 0)
            b1 = m;
        else
            a1 = m;
    } while(b1-a1) > eps);
    return a1;
}
```

Le code optimisé en C

```
float bissec(float (*f) (float), float a, float b, float eps)
{
    float a1, b1, m, fa1, fb1, fm;
    if (b < a)
        { a1 = b; b1 = a;}
    else
        { a1 = a; b1 = b;};
    fa1 = f(a1); fb1 = f(b1);
    if (fa1.fb1 > 0) exit(0);
    do
    {
        m = (a1+b1)/2;
        fm = f(m);
        if (fa1.fm <0)
            { b1 = m; fb1 = fm; }
        else
            { a1 = m; fa1 = fm; };
    } while (b1-a1)>eps;
    retrun a1;
}
```

Fin

La méthode de Newton

Méthode de Newton

Soit $f(x)$ une fonction C^∞ de \mathbb{R} dans \mathbb{R} et $x_0 \in \mathbb{R}$. La tangente à f en x_0 est définie par

$$y = f(x_0) + f'(x_0).(x - x_0)$$

Soit x_1 l'intersection de la tangente avec l'axe des abscisses. Alors

$$0 = f(x_0) + f'(x_0).(x_1 - x_0) \Rightarrow x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Au lieu de résoudre $f(x) = 0$, on résout $T_{x_0}(f)(x) = 0$ et on recommence.

L'itération de Newton est définie par la suite récurrente

$$x_{k+1} = \phi(x_k) = x_k - \frac{f(x_k)}{f'(x_k)}$$

Interprétation géométrique

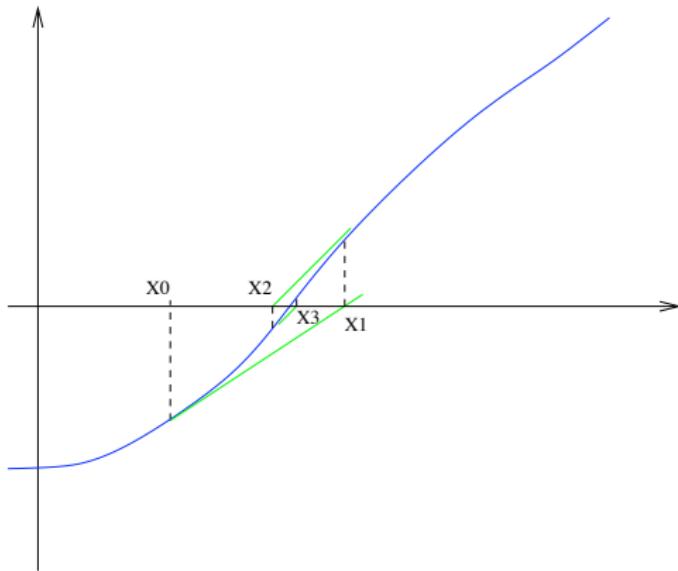


Figure – Méthode de Newton via les tangentes

Etude théorique

Soit a une racine de $f(x)$ où f est une fonction C^∞ sur un intervalle I ouvert contenant a .

Théorème : $\exists \alpha \in \mathbb{R}$ tel que $\forall x_0 \in [a - \alpha, a + \alpha]$,

$$\lim_{n \rightarrow \infty} x_n = a.$$

De plus :

- ▶ si $f'(a) \neq 0$, la convergence est exponentielle (au moins quadratique).
- ▶ si $f'(a) = 0$ alors la convergence est linéaire en $\left(\frac{p-1}{p}\right)^n$ avec p tel que $f'(a) = \dots = f^{(p-1)}(a) = 0$ et $f^{(p)}(a) \neq 0$.

Démonstration

Soit $\phi(x) = x - \frac{f(x)}{f'(x)}$. Si $f'(a) \neq 0$ alors $\phi(x)$ est continue et dérivable au voisinage de a et

$$\phi(a) = a - \frac{f(a)}{f'(a)} = a$$

donc a est un point fixe pour ϕ et

$$\phi'(x) = 1 - \frac{f'^2(x) - f(x).f''(x)}{f'^2(x)}$$

donc

$$\phi'(a) = 1 - \frac{f'^2(a) - f(a).f''(a)}{f'^2(a)} = 0$$

donc a est une limite possible pour ϕ et la convergence est au moins quadratique.

Démonstration suite

Si $f(a) = f'(a) = 0$, soit $f^{(p)}(a)$ la première dérivée non nulle de f en a alors les développements de Taylor-Young de f et f' au voisinage de a donnent

$$f(x) = \frac{(x-a)^p}{p!} f^{(p)}(a) + \mathcal{O}((x-a)^{p+1})$$

et

$$f'(x) = \frac{(x-a)^{p-1}}{(p-1)!} f^{(p)}(a) + \mathcal{O}((x-a)^p)$$

donc

$$\phi(x) = x - \frac{\frac{(x-a)^p}{p!} f^{(p)}(a) + \mathcal{O}((x-a)^{p+1})}{\frac{(x-a)^{p-1}}{(p-1)!} f^{(p)}(a) + \mathcal{O}((x-a)^p)} = x - \frac{1}{p} \cdot (x-a) + \mathcal{O}((x-a)^2)$$

donc $\phi(a) = a$ et $\exists \phi'(a) = \lim_{x \rightarrow a} \frac{\phi(x) - \phi(a)}{x - a} = \frac{p-1}{p}$ cqfd.

Exemple de la racine carrée

Pour les calculs à la main, la méthode de Newton est imbattable !

Soit $a \in \mathbb{R}$, la méthode de Newton appliquée à $f(x) = x^2 - a$ donne

$$\phi(x) = x - \frac{x^2 - a}{2x} = \frac{1}{2} \cdot \left(x + \frac{a}{x} \right).$$

$a = m \times 10^k = m \times 10^\delta \times 10^{2l}$ avec $\delta = 0$ ou 1 .

```
double sqrt_newton(double a)
{
    double x, y eps = 1.0e-8;
    int i = 0;
    y = round(10*sqrt(a))/10;
    i = 0;
    do
    {
        x = y;
        y = (x + a/x)/2;
        i++;
    } while((fabs(x-y) > eps)&&(i< IMAX));
}
```

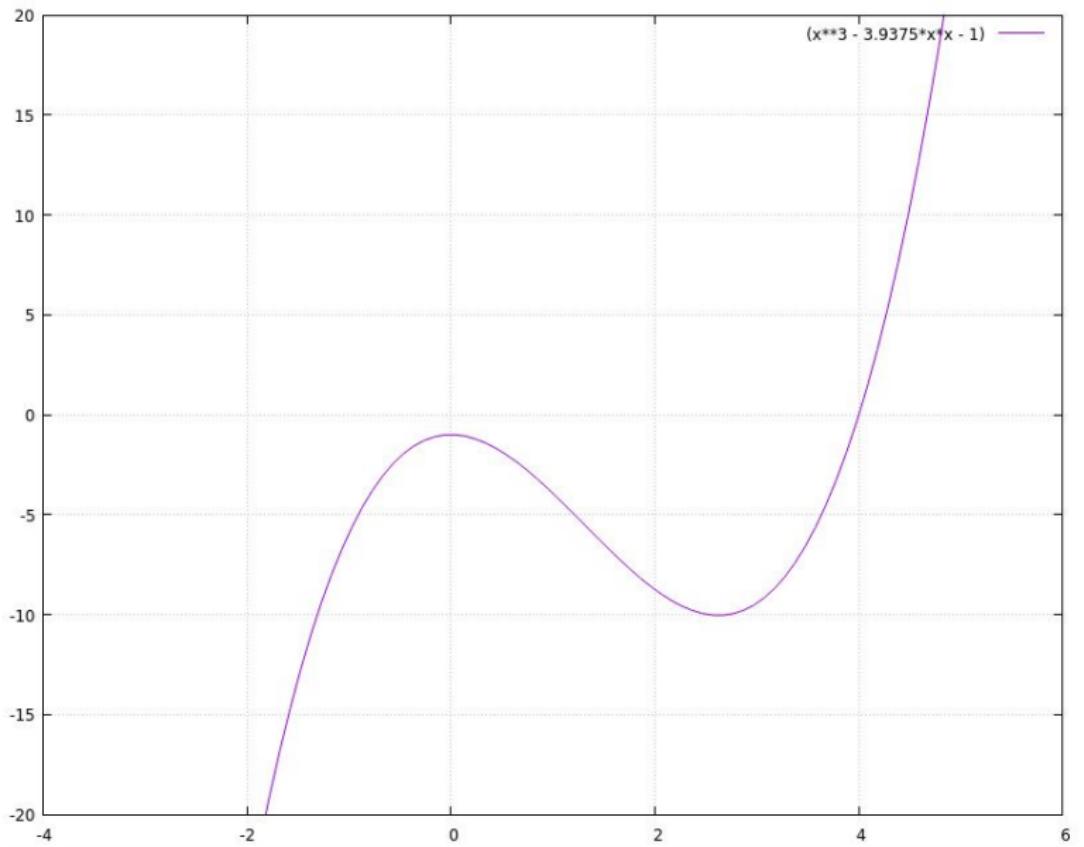
Quel que soit $1 < a < 100$, on obtient 8 chiffres exacts en 3 itérations !

les trois phases de la méthode de Newton

- ▶ la phase de recherche
- ▶ la phase de convergence
- ▶ la phase stationnaire

Soit $f(x) = x^3 - \frac{63}{16} \cdot x - 1$.

On applique la méthode de Newton avec $x_0 = -10$.



Les résultats :

1	$-6.317491749174918e + 00$	25	$1.821675339593982e - 01$
2	$-3.896685622851384e + 00$	26	$-6.602359880042916e - 01$
3	$-2.323268942304249e + 00$	27	$-1.985542159676725e - 01$
4	$-1.314439510479102e + 00$	28	$4.929665109633391e - 01$
5	$-6.659430836488338e - 01$	29	$-8.966583473140366e - 02$
6	$-2.033339574091531e - 01$	30	$1.324089389430439e + 00$
7	$4.755097169532684e - 01$	31	$2.439164590287561e - 01$
8	$-1.059019934402428e - 01$	32	$-4.561415985273706e - 01$
9	$1.098937682293482e + 00$	33	$-2.151384860475836e - 03$
10	$2.188135541117587e - 01$	34	$5.897495452413800e + 01$
11	$-5.270139097861306e - 01$	35	$3.977461710449570e + 01$
12	$-7.752942392319240e - 02$	36	$2.698505083711078e + 01$
13	$1.551760477951325e + 00$	37	$1.847518527054536e + 01$
14	$2.017898006452707e - 01$	38	$1.282788425505722e + 01$
15	$-5.835981217441562e - 01$	39	$9.104529754509855e + 00$
16	$-1.314783152541896e - 01$	40	$6.690077755470120e + 00$
17	$8.529661391543262e - 01$	41	$5.192321729057257e + 00$
18	$1.375203635471723e - 01$	42	$4.371382409769733e + 00$
19	$-9.069404391186823e - 01$	43	$4.053028052588934e + 00$
20	$-3.882239552313449e - 01$	44	$4.001322875732497e + 00$
21	$8.249915645488815e - 02$	45	$4.000000854289640e + 00$
22	$-1.548358431213166e + 00$	46	$4.0000000000000357e + 00$
23	$-8.183372268651269e - 01$	47	$4.000000000000000e + 00$
24	$-3.232875678353185e - 01$	48	$4.000000000000000e + 00$

Fin

La méthode regula-falsi

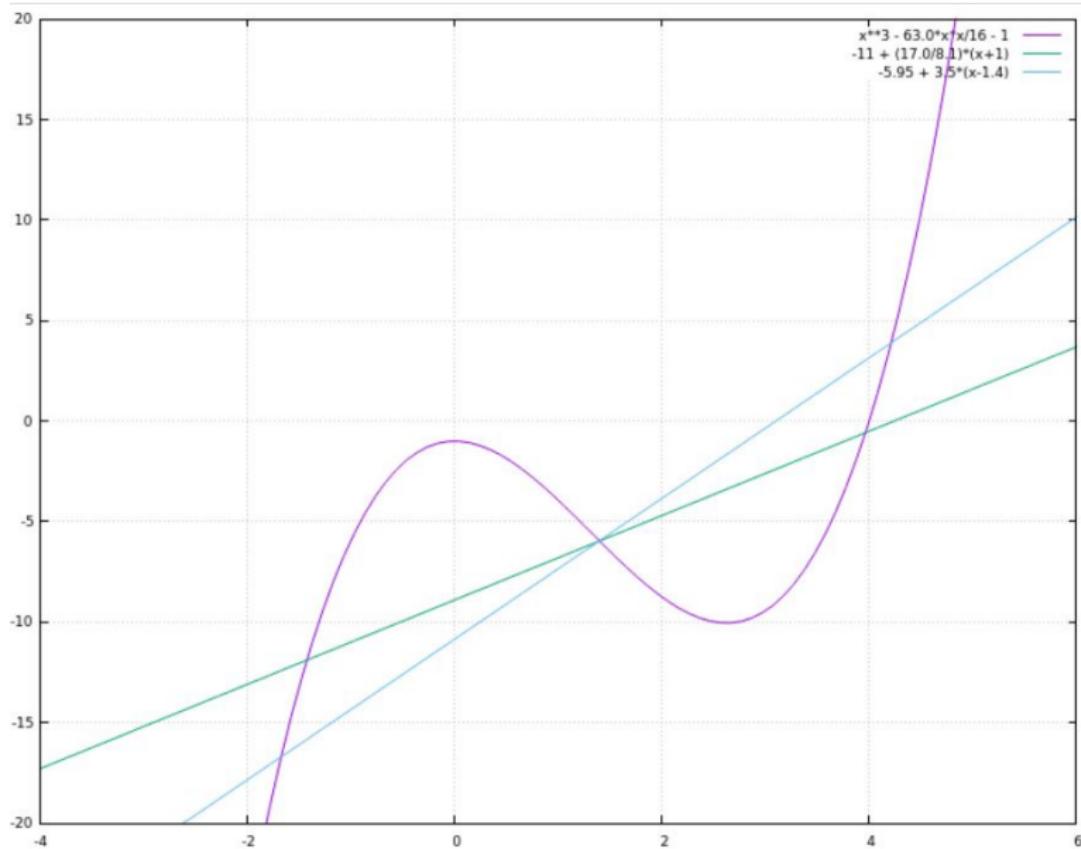
La méthode de Newton nécessite la connaissance de la dérivée de la fonction f .

Pour éviter f' , on part de 2 points x_0 et x_1 et on cherche l'intersection de la droite passant par $(x_0, f(x_0))$ et $(x_1, f(x_1))$ avec l'axe des abscisses que l'on appelle x_2 et on recommence.

L'équation de la droite est $y = f(x_1) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_1)$ d'où
 $x_2 = x_1 - \frac{f(x_1).(x_1 - x_0)}{f(x_1) - f(x_0)}$.

La méthode *regula falsi* ou méthode des cordes : si x_0 et x_1 sont deux points alors

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



Etude théorique

Théorème : Si f est C^∞ sur I et $a \in I$ tel que

$$f(a) = 0 \text{ et } f'(a) \neq 0,$$

alors $\exists \alpha > 0, A > 1, K > 0$ tel que $\forall (x_0, x_1) \in [a - \alpha, a + \alpha]$, la suite x_n définie par la méthode régula falsi vérifie

$$\forall n \geq 1, |x_n - a| \leq \frac{K}{A^{(\sigma)^n}}$$

où $\sigma = \frac{1 + \sqrt{5}}{2}$ (le nombre d'or).

La convergence est exponentielle.

Lemme : Si f est C^∞ sur I et $a \in I$ tel que $f(a) = 0$ et $f'(a) = 0$ et si $z = y - \frac{f(y).(y-x)}{f(y)-f(x)}$ alors, au voisinage de a ,

$$z - a = \frac{f''(a)}{2.f'(a)}.(x-a)(y-a) + o((x-a)(y-a))$$

On part de

$$f(z+h) = f(z) + h.f'(z) + \frac{h^2}{2}.f''(z) + o(h^2)$$

$$f(z-h) = f(z) - h.f'(z) + \frac{h^2}{2}.f''(z) + o(h^2)$$

avec $h = \frac{(y-a)-(x-a)}{2} = \frac{y-x}{2}$ et $z = \frac{(y-a)+(x-a)}{2} = \frac{x+y}{2} - a$
d'où $z+h = y-a$ et $z-h = x-a$

$$\Rightarrow f'(z) = f'\left(\frac{x+y}{2} - a\right) = f'(a) + \frac{x+y}{2}.f''(a) + o\left(\frac{x+y}{2} - a\right)$$

et

$$f(y) = f'(a).(y-a) + \frac{f''(a)}{2}.(y-a)^2 + o((y-a)^2)$$

Donc, si $y = x_n$ et $x = x_{n-1}$, alors $z = x_{n+1}$

$$|x_{n+1} - a| \approx K \cdot |x_n - a| \cdot |x_{n-1} - a|$$

Soit v_n définie par $v_0 = |x_0 - a|$, $v_1 = |x_1 - a|$ et

$$v_{n+1} = 2K \cdot v_n \cdot v_{n-1}$$

alors

$$\forall n, |x_n - a| \leq v_n \text{ et } \log(v_{n+1}) = \log(2K) + \log(v_n) + \log(v_{n-1})$$

donc

$$\log(v_n) = u \cdot \sigma^n + v \cdot \sigma^{-n} + w \approx u \cdot \sigma^n$$

donc

$$|x_n - a| \leq v_n \approx \frac{K'}{A'(\sigma)^n}$$

Attention ! L'absence de la dérivée a un coût.

La formule

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

est, par nature, très instable.

Fin