MT94 - Problèmes non linéaires I

S. Mottelet

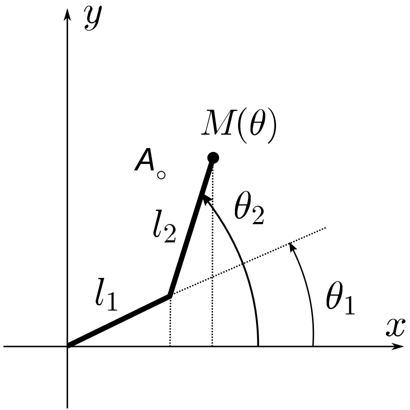
Université de Technologie de Compiègne

Racines de polynômes : trouver $z \in \mathbb{C}$ tel que $P(z) = \sum_{k=0}^{n} a_k z^k = 0$.

- n solutions,
- Pas de formules par radicaux pour $n \geq 5 \Rightarrow$ méthodes itératives
  - On construit (z_k) telle que $\lim_{k\to\infty}(z_k) = z^*$ où $P(z^*) = 0$.

Introduction

Problème de cinématique inverse



On cherche $\theta \in \mathbb{R}^2$ tel que $M(\theta) = A$, soit

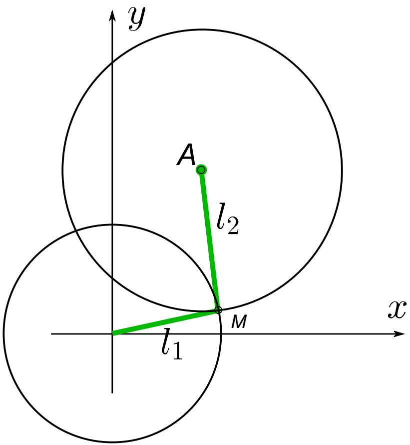
$$\ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 = x_A,$$
$$\ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 = y_A,$$

$\iff f(\theta) = \vec{0}$ où $f : \mathbb{R}^2 \to \mathbb{R}^2$ est définie par

$$f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - x_A \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - y_A \end{pmatrix}.$$

Introduction

Problème de cinématique inverse

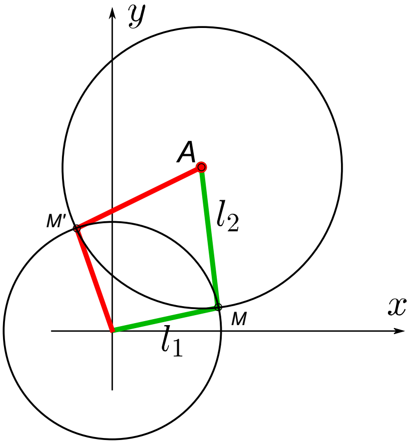


Trouver $M(x, y)$ tel que

$$\begin{aligned}(x - x_A)^2 + (y - y_A)^2 &= \ell_2^2,\\ x^2 + y^2 &= \ell_1^2.\end{aligned}$$

Introduction

Problème de cinématique inverse



Trouver $M(x, y)$ tel que

$$\begin{aligned}(x - x_A)^2 + (y - y_A)^2 &= \ell_2^2,\\ x^2 + y^2 &= \ell_1^2.\end{aligned}$$

Sommaire

Méthodes numériques ($n = 1$)
Bissection ou dichotomie

On suppose que $f : \mathbb{R} \to \mathbb{R}$ est continue et qu'il existe $I = [a, b]$ tel que $f(a)f(b) < 0$.

$$\Longrightarrow \exists x^* \in]a, b[,\ f(x^*) = 0. \tag{TVI}$$

Idée : construire deux suites (a_k), (b_k) telles que

$$\forall k,\ f(a_k)f(b_k) < 0 \quad \text{et} \quad \lim_{k\to\infty} (b_k - a_k) = 0.$$

Puisque $a_k < x^* < b_k$,

$$\lim_{k\to\infty} (a_k) = \lim_{k\to\infty} (b_k) = x^*$$

Méthodes numériques ($n = 1$)

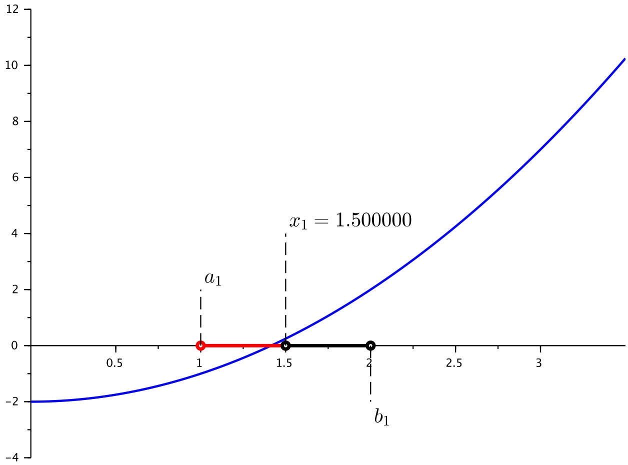
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

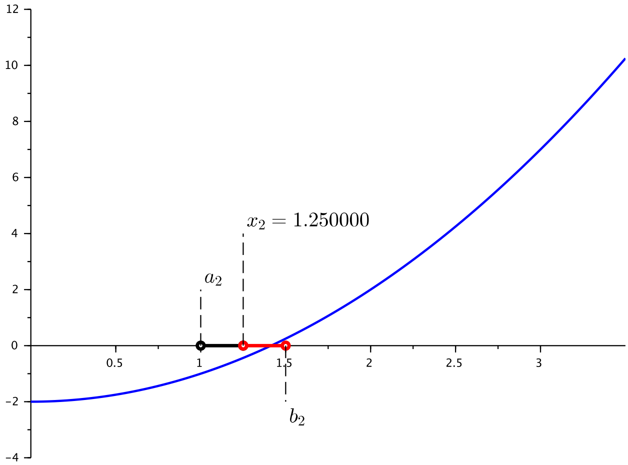


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

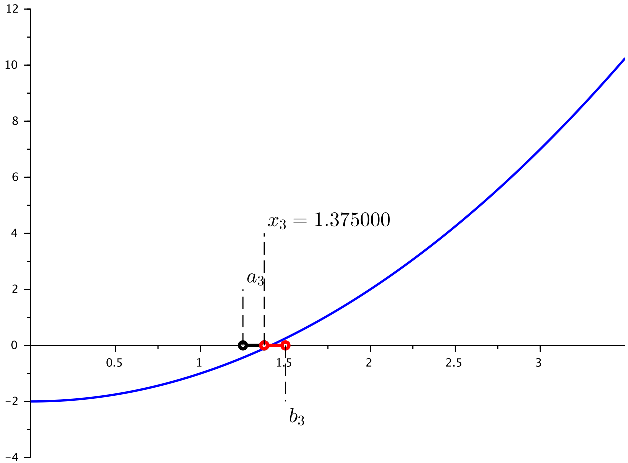


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

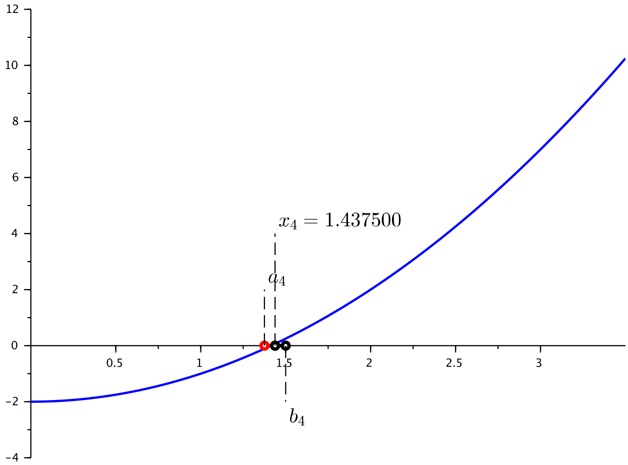
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

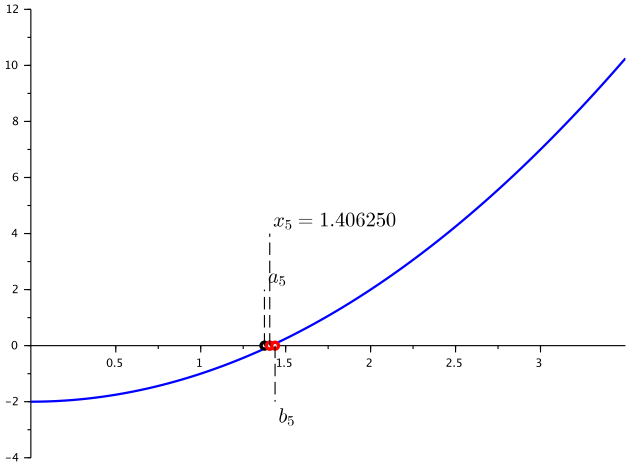
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

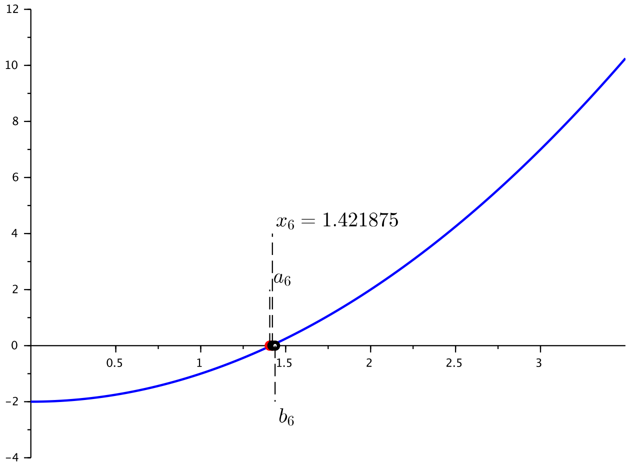
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k + b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

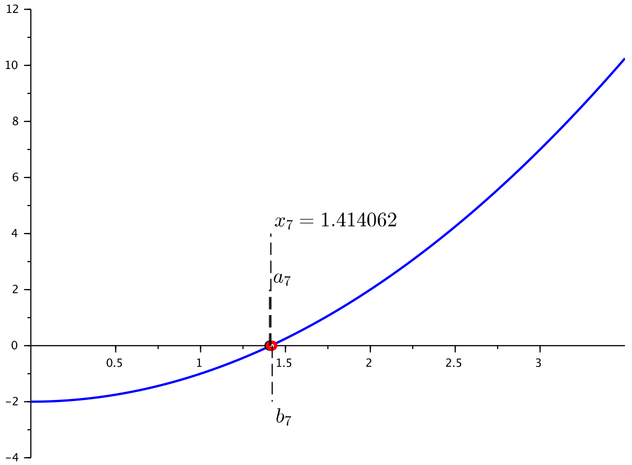
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

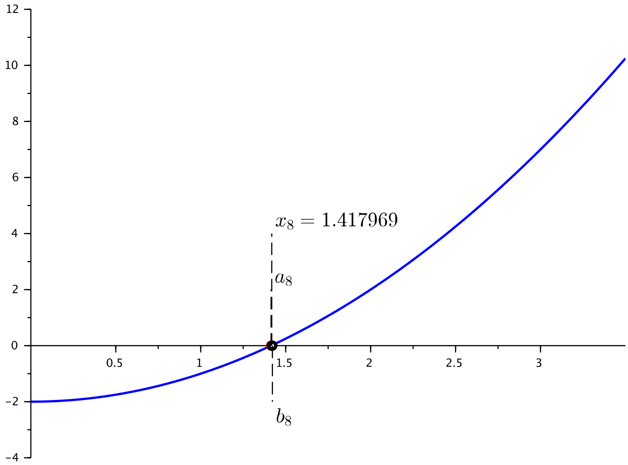


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k + b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

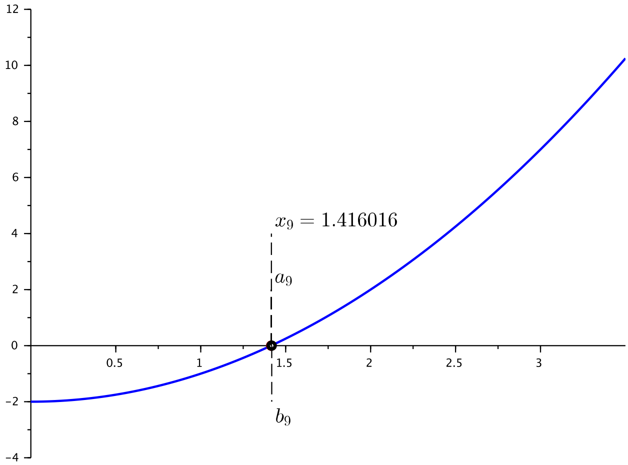
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k + b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

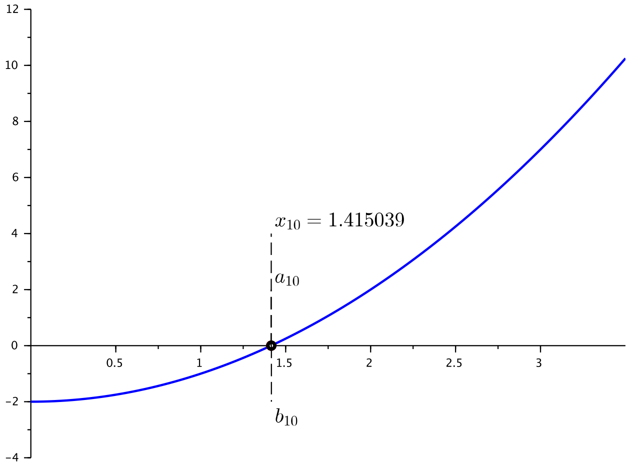
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

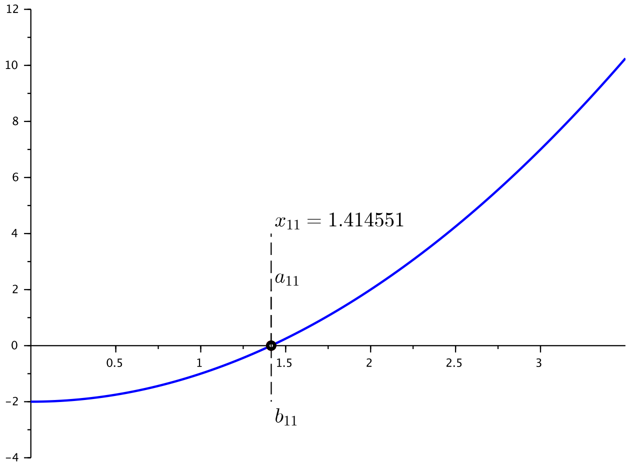


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

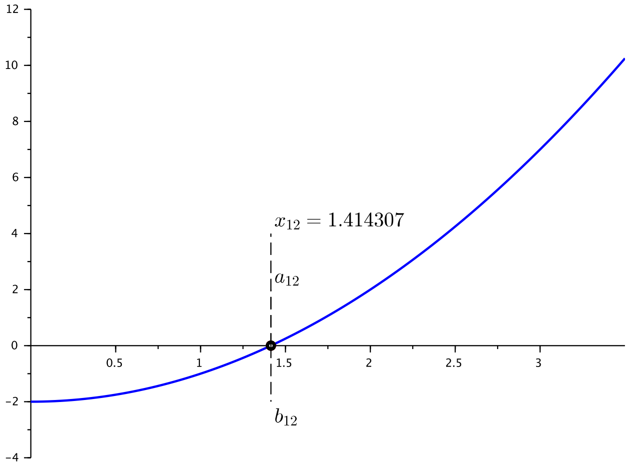
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k + b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

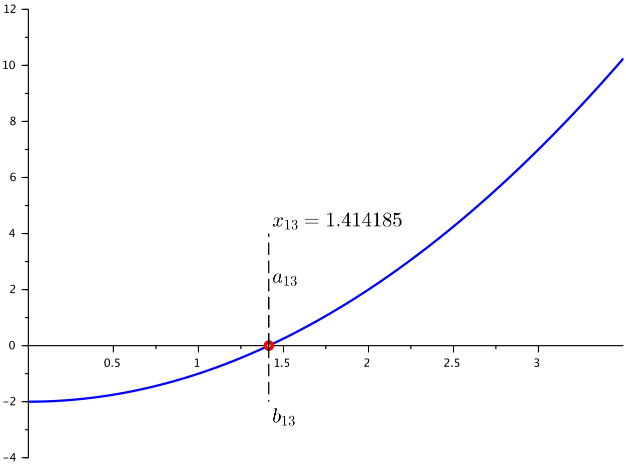
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k + b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

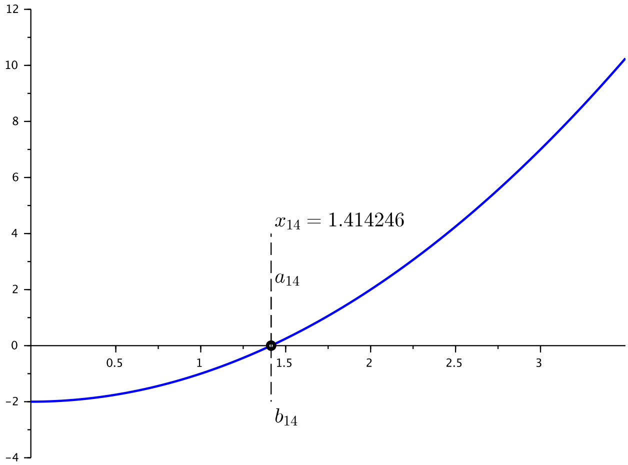
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

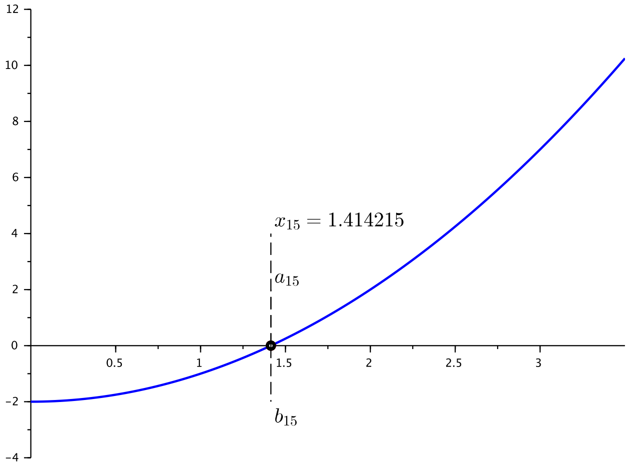


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

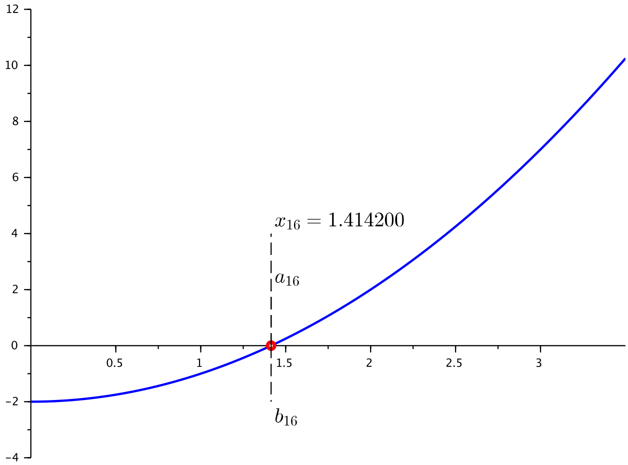
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

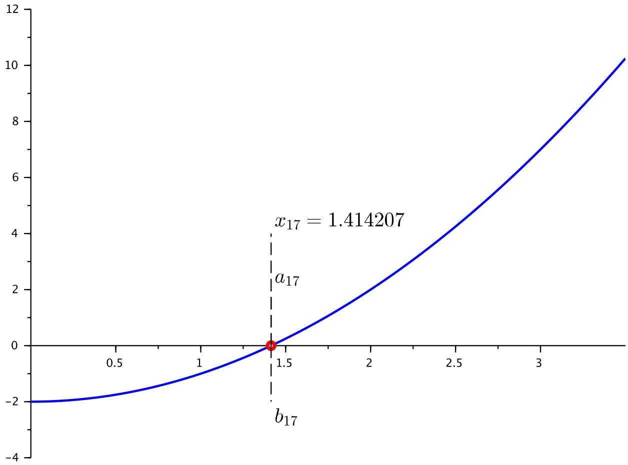
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

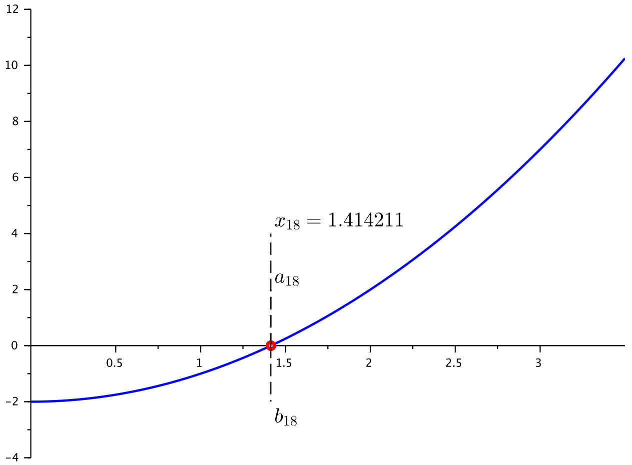


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

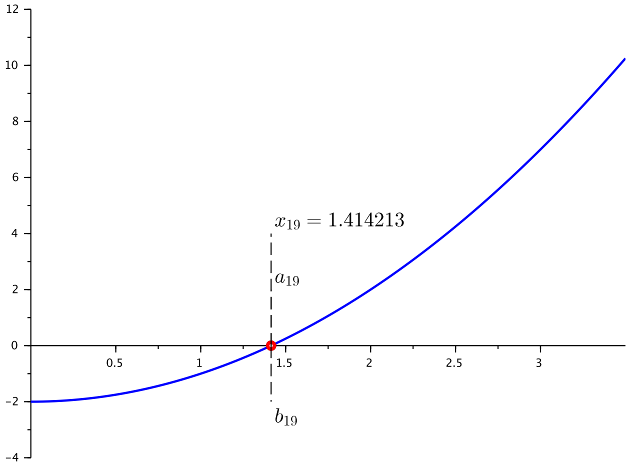
Bissection ou dichotomie

Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin

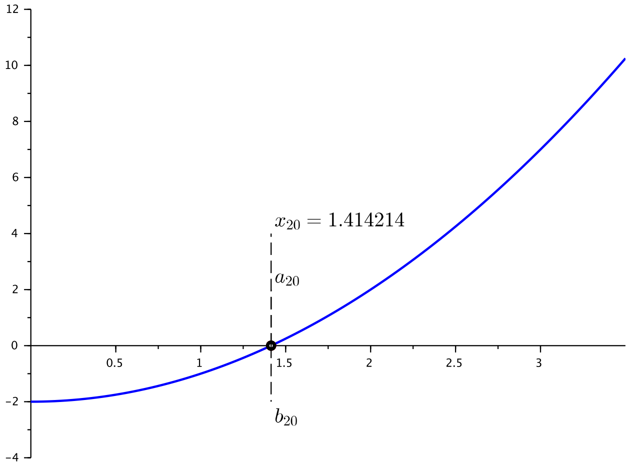


Exemple, $f(x) = x^2 - 2$, $a_0 = 1$, $b_0 = 3$

Algorithme

On définit $x_k = \frac{a_k+b_k}{2}$ puis

- tant que $|f(x_k)| > \epsilon$
  - si $f(x_k)f(a_k) > 0$
    - $a_{k+1} = x_k$
  - sinon
    - $b_{k+1} = x_k$
  - fin
- fin



Méthodes numériques ($n = 1$)

Bissection ou dichotomie, convergence

Puisque $b_{k+1} - a_{k+1} = (b_k - a_k)/2$, $b_k - a_k = \left(\frac{1}{2}\right)^k (b_0 - a_0)$, et comme $|x_k - x^*| \leq (b_k - a_k)/2$, on a

$$|x_k - x^*| \leq \left(\tfrac{1}{2}\right)^{k+1} (b_0 - a_0).$$

Définition

Si (x_k) converge vers x^* et s'il existe $p > 0$ et $\alpha > 0$ tels que

$$\lim_{k \to \infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^p} = \alpha$$

la convergence est dite d'ordre p.

La dichotomie est une méthode d'ordre 1 ($p = 1$ et $\alpha = \frac{1}{2}$).

On suppose que f est continûment dérivable et qu'il existe x^* tel que $f(x^*) = 0$.

On reformule l'équation $f(x) = 0$ sous la forme $g(x) = x$, où x^* est un point fixe de g :

$$g(x^*) = x^*.$$

Exemple : si $f(x) = x^2 - 2$ et $g(x) = \frac{x+2}{x+1}$ alors

$$g(x) = x \Leftrightarrow x^2 + x = x + 2 \Leftrightarrow x^2 = 2 \Leftrightarrow f(x) = 0$$

Méthodes numériques ($n = 1$)

Point fixe, convergence

La méthode de point fixe (appelée aussi *méthode des approximations successives*) est basée sur le théorème suivant :

Théorème

Soit $g : \mathbb{R} \to \mathbb{R}$ continûment dérivable et possédant un point fixe x^* vérifiant

$$|g'(x^*)| < 1.$$

Alors il existe $\alpha > 0$ tel que la suite (x_k) définie par x_0 vérifiant $|x_0 - x^*| < \alpha$ et

$$x_{k+1} = g(x_k), \quad k \geq 0$$

converge vers x^*.

Méthodes numériques ($n = 1$)

Point fixe, convergence

Démonstration

Puisque $|g'(x^*)| < 1$, il existe $L \in]0,1[$ tel que $|g'(x^*)| < L < 1$ et grâce à la continuité de g' il existe $\alpha > 0$ tel que

$$|x - x^*| < \alpha \Longrightarrow |g'(x)| < L.$$

La formule des accroissements finis donne $x_{k+1} - x^* = g(x_k) - g(x^*) = g'(\xi_k)(x_k - x^*)$, où $\xi_k \in \mathrm{Int}(x_k, x^*)$. On en déduit la majoration

$$|x_k - x^*| < \alpha \Longrightarrow |x_{k+1} - x^*| < L|x_k - x^*|.$$

Puis on montre par récurrence que

- $|x_0 - x^*| < \alpha \Longrightarrow \forall k > 0,\ |x_k - x^*| < \alpha$,
- $\forall k \geq 0,\ |x_k - x^*| < L^k |x_0 - x^*|$,

d'où le résultat.

Lorsque la méthode de point fixe converge, si $g'(x^*) \neq 0$ alors elle est d'ordre 1 : on a en effet

$$\forall k > 0,\ x_{k+1} - x^* = g(x_k) - g(x^*) = g'(\xi_k)(x_k - x^*),$$

et comme $\xi_k \in \mathrm{Int}(x_k, x^*)$ et $x_k \to x^*$ on a $\xi_k \to x^*$, donc

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^1} = |g'(x^*)|.$$

Remarque

$f(x) = x^2 - 2 = 0 \Leftrightarrow x = g(x) = \frac{x+2}{x+1}$: deux points fixes : $x_1^* = \sqrt{2}$ et $x_2^* = -\sqrt{2}$

$$g'(x_1^*) \approx 0.5147186, \quad g'(x_2^*) \approx 17.485281$$

Si $x_0 \neq x_2^*$ il n'y a aucune garantie que la méthode converge vers x_2^* !

Méthodes numériques ($n = 1$)

Point fixe, convergence

Illustration avec Scilab :

```
x = 1.5;
for i = 1:30
    x = (x+2)/(x+1);
    printf("%.16f\n",x)
end

x = -sqrt(2);
for i = 1:30
    x = (x+2)/(x+1);
    printf("%.16f\n",x)
end
```

On suppose que f est dérivable en x_k et dans un voisinage suffisamment large. On peut donc écrire que pour tout x

$$f(x) = f(x_k) + f'(x_k)(x - x_k) + (x - x_k)\varepsilon(x - x_k),$$

où $\varepsilon(x - x_k) \to 0$ quand $x \to x_k$. L'idée de la méthode de Newton consiste, si $f'(x_k) \neq 0$, à définir x_{k+1} comme la solution de l'équation linéaire

$$f(x_k) + f'(x_k)(x - x_k) = 0,$$

soit

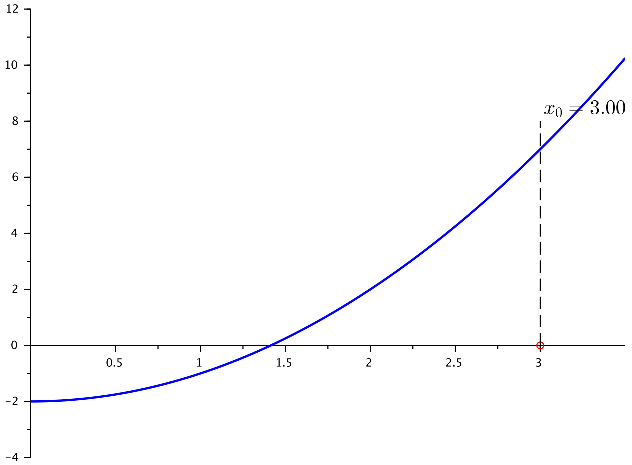
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$$

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

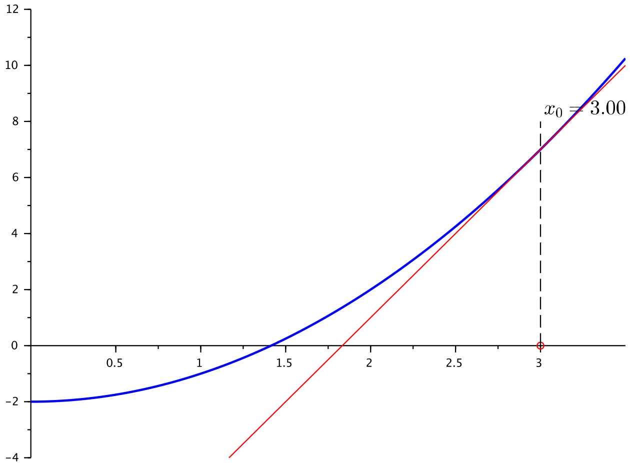
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

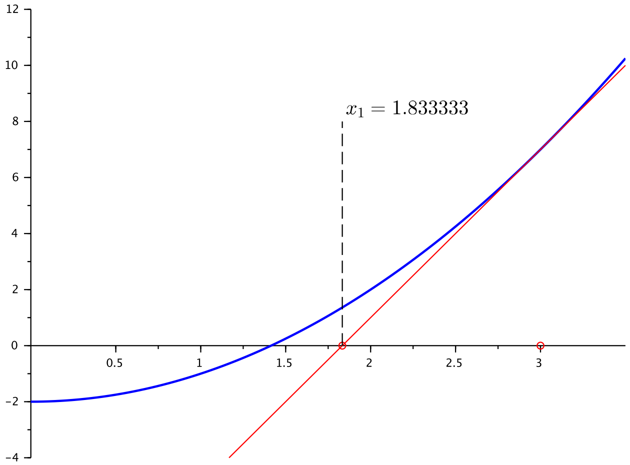
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

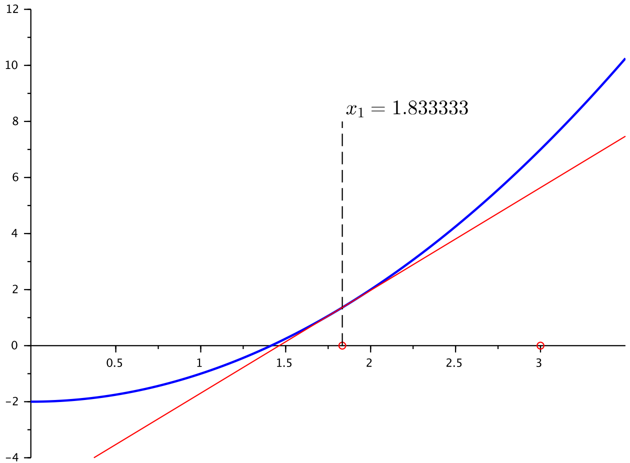
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin

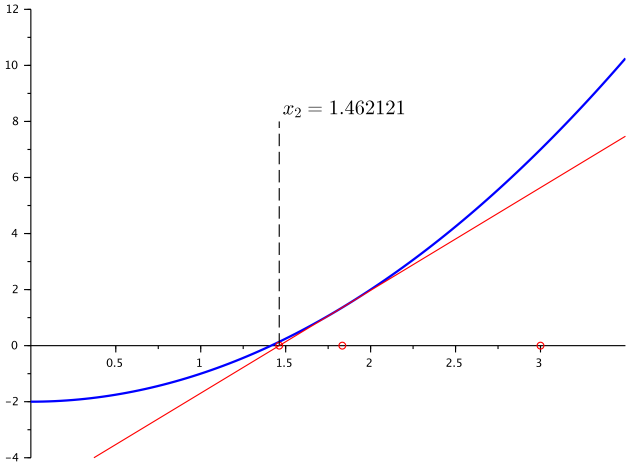


Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

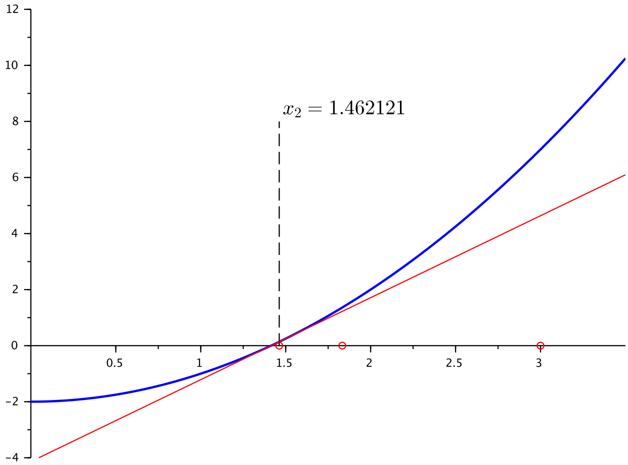
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin

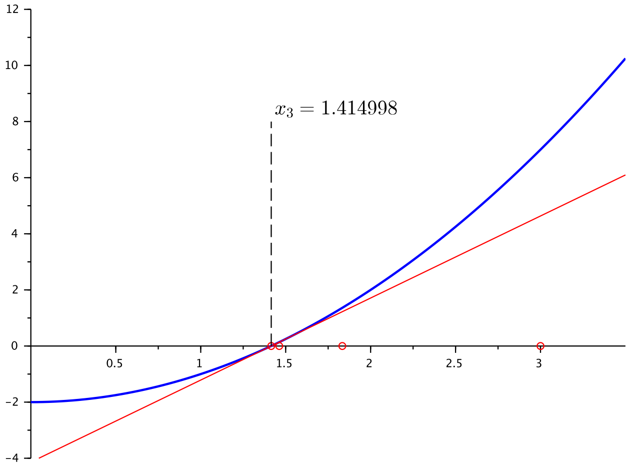


Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

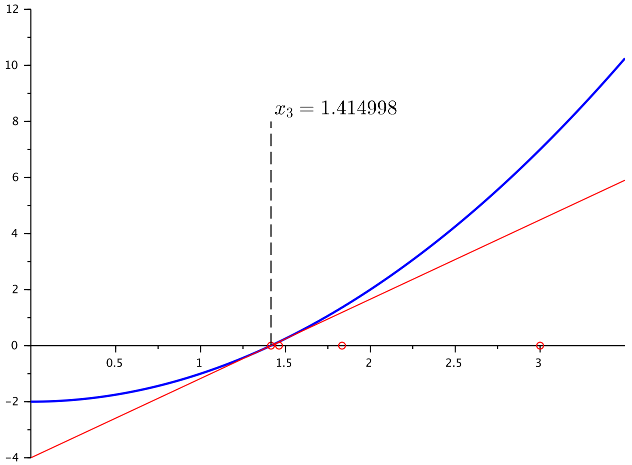
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

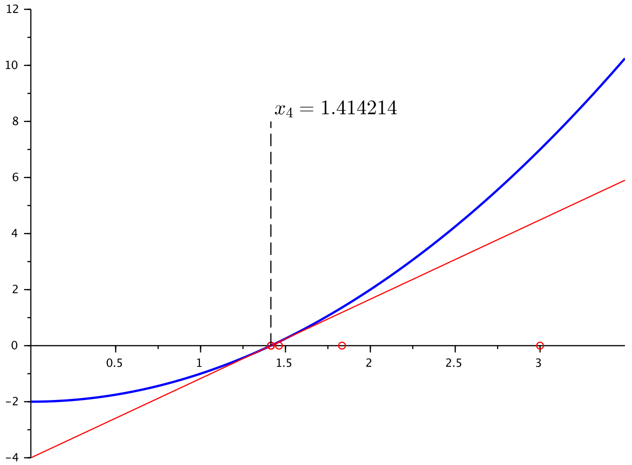
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

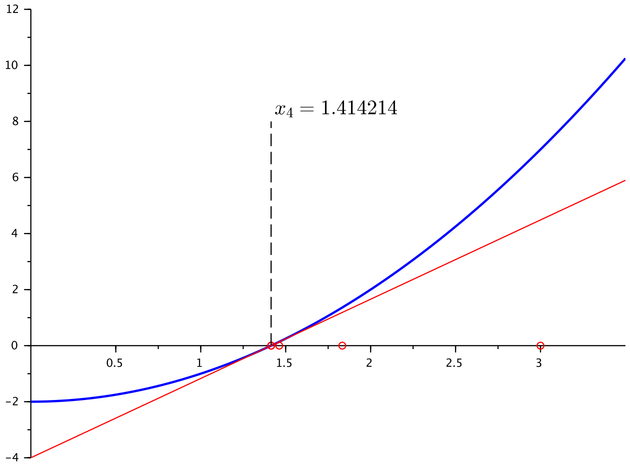
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n = 1$)

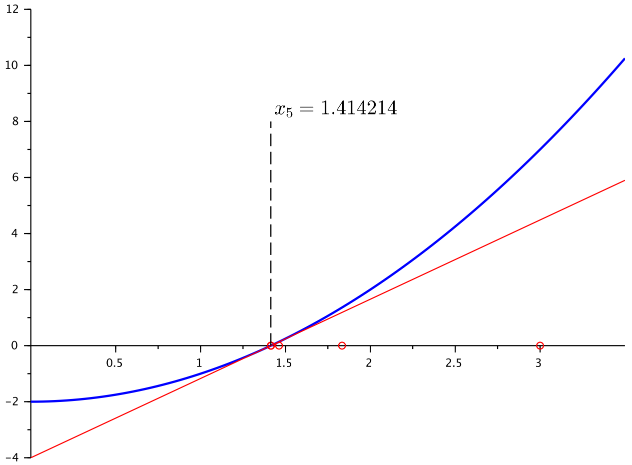
Méthode de Newton

Exemple, $f(x) = x^2 - 2$, $x_0 = 3$

Algorithme

On définit x_0 puis

- tant que $|f(x_k)| > \epsilon$ et $|f'(x_k)| > \epsilon$
  - $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
- fin



Méthodes numériques ($n > 1$)

Méthode de Newton, interprétation

La méthode de Newton est une méthode de point fixe. On a en effet

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

avec $g(x) = x - \dfrac{f(x)}{f'(x)}$. Si f est dérivable deux fois on a donc , si $f(x^*) = 0$ et $f'(x^*) \neq 0$

$$\begin{aligned} g'(x) &= 1 - \frac{f'(x)^2 - f(x)f''(x)}{f'(x)^2}, \\ g'(x^*) &= 1 - \frac{f'(x^*)^2}{f'(x^*)^2} = 0, \end{aligned}$$

Donc pour tout f, il existe nécessairement $\alpha > 0$ tel que pour $|x_0 - x^*| < \alpha$ la méthode de Newton converge.

Théorème

Si f est dérivable 3 fois et qu'il existe $C > 0$ tel que $\forall x$, $\frac{1}{2}|g''(x)| \leq C$ alors pour tout x_0 tel que $|x_0 - x^*| \leq \frac{1}{C}$ la méthode de Newton converge et la convergence est d'ordre 2.

Démonstration : Pour tout k, il existe $\xi_k \in \text{Int}(x_k, x^*)$ tel que

$$\begin{aligned} x_{k+1} - x^* &= g(x_k) - g(x^*) = g'(x^*)(x_k - x^*) + \frac{1}{2}g''(\xi_k)(x_k - x^*)^2, \\ &= \frac{1}{2}g''(\xi_k)(x_k - x^*)^2, \end{aligned}$$

d'où la majoration $|x_{k+1} - x^*| \leq C|x_k - x^*|^2$. Ensuite on a par récurrence

$$C|x_k - x^*| \leq (C|x_0 - x^*|)^{2^k},$$

donc $x_k \to x^*$ si $C|x_0 - x^*| < 1$, d'où le premier résultat. Et pour l'ordre, on a

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^{2}} = \frac{1}{2}\lim_{k\to\infty} |g''(\xi_k)| = \frac{1}{2}|g''(x^*)|.$$

Méthodes numériques ($n > 1$)

Rappels de calcul différentiel, différentiabilité

- Definition : soit $f : \mathbb{R}^n \longrightarrow \mathbb{R}^m$,

$$f(x) = \begin{pmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{pmatrix}, \quad f_i : \mathbb{R}^n \longrightarrow \mathbb{R},$$

f est differentiable en $a \in \mathbb{R}^n$ s'il existe $f'(a)$, une matrice $m \times n$ telle que

$$f(a+h) = f(a) + f'(a)h + \|h\|\varepsilon(h), \quad \lim_{h \to \overrightarrow{0}} \varepsilon(h) = \overrightarrow{0}$$

$$\left[f'(a)\right]_{ij} = \frac{\partial f_i}{\partial x_j}(a)$$

Méthodes numériques ($n > 1$)

Rappels de calcul différentiel, jacobienne, dérivée

- Exemple :

$$f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - x_A \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - y_A \end{pmatrix}, \quad f'(\theta) = \begin{pmatrix} -\ell_1 \sin\theta_1 & -\ell_2 \sin\theta_2 \\ \ell_1 \cos\theta_1 & \ell_2 \cos\theta_2 \end{pmatrix}.$$

$$g(x) = \begin{pmatrix} (x_1 - a_1)^2 + (x_2 - a_2)^2 - \ell_2^2 \\ x_1^2 + x_2^2 - \ell_1^2 \end{pmatrix}, \quad g'(x) = \begin{pmatrix} 2(x_1 - a_1) & 2(x_2 - a_2) \\ 2x_1 & 2x_2 \end{pmatrix}.$$

On suppose que f est différentiable en x_k et dans un voisinage suffisamment large. On peut donc écrire que pour tout x

$$f(x) = f(x_k) + f'(x_k)(x - x_k) + (x - x_k)\varepsilon(x - x_k),$$

où $\varepsilon(x - x_k) \to \overrightarrow{0}$ quand $x \to x_k$. L'idée de la méthode de Newton consiste, si $f'(x_k)$ est inversible, à définir x_{k+1} comme la solution du système d'équations linéaire

$$f(x_k) + f'(x_k)(x - x_k) = 0,$$

soit

$$x_{k+1} = x_k - f'(x_k)^{-1} f(x_k).$$

Notation Scilab : $x_{k+1} = x_k - f'(x_k) \backslash f(x_k)$

Méthodes numériques ($n > 1$)

Méthode de Newton

Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

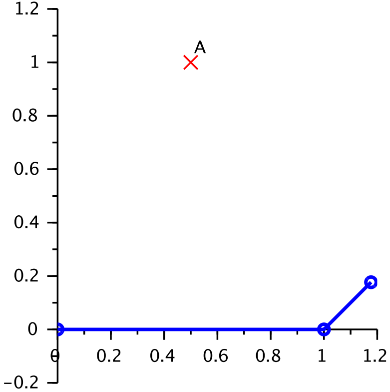
Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$\theta_0 = (0.000000, 0.785398)$

$\|f(\theta_0)\| = 1.065703e+00$



Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

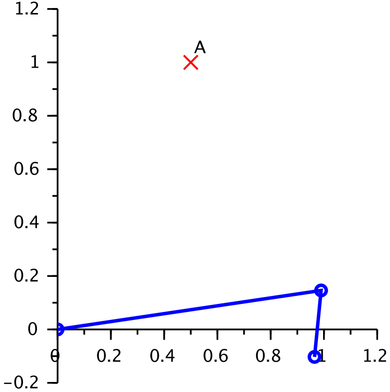
Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$\theta_1 = (0.146447, 4.613825)$

$\|f(\theta_1)\| = 1.196766e+00$



Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

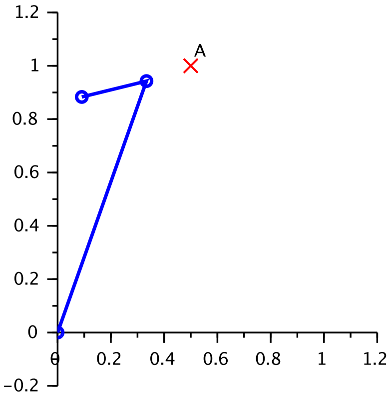
Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$\theta_2 = (1.230608, 3.381886)$

$\|f(\theta_2)\| = 4.254983e-01$



Méthodes numériques ($n > 1$)

Méthode de Newton

Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

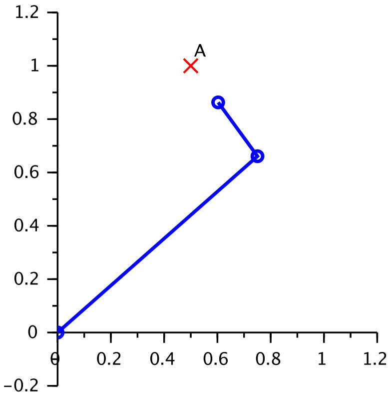
Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$\theta_3 = (0.722122, 2.202113)$

$\|f(\theta_3)\| = 1.714783e-01$

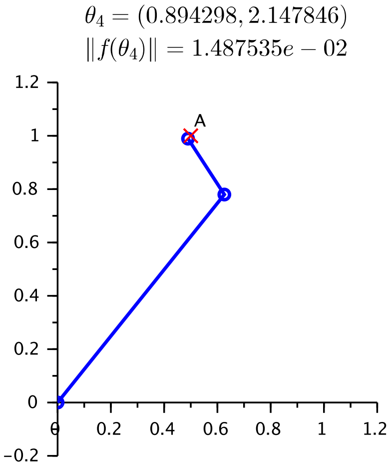


Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin



Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

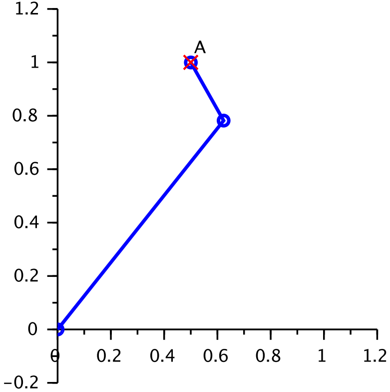
Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$\theta_5 = (0.897819, 2.085475)$

$\|f(\theta_5)\| = 4.883133e-04$



Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

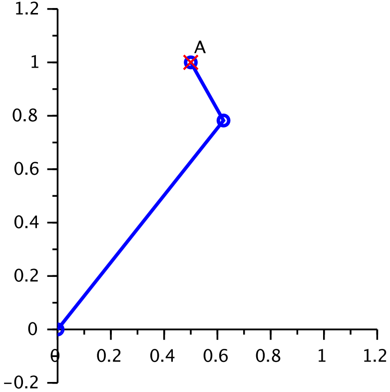
Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$\theta_6 = (0.898346, 2.084746)$

$\|f(\theta_6)\| = 1.745769e-07$



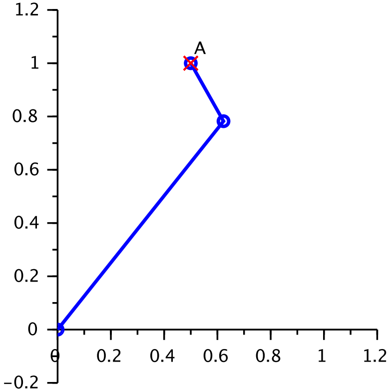
Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin

$$\theta_7 = (0.898346, 2.084745)$$
$$\|f(\theta_7)\| = 6.557237e-14$$



Méthodes numériques ($n > 1$)

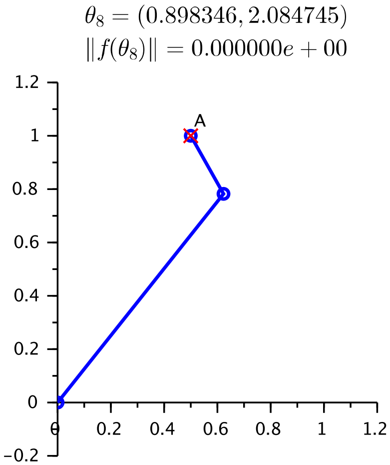
Méthode de Newton

Exemple, $f(\theta) = \begin{pmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 - a_1 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 - a_2 \end{pmatrix}$

Algorithme

On définit x_0 puis

- tant que $\|f(x_k)\| > \epsilon$
  - Résoudre $f'(x_k)h = f(x_k)$
  - si $f'(x_k)$ est singulière
    - STOP
  - sinon
    - $x_{k+1} = x_k - h$
  - fin
- fin



Méthodes numériques ($n > 1$)

Méthode de Newton, propriétés

- Lorsque la méthode converge, elle est d'ordre 2
- La convergence reste locale, et les méthodes modernes qui s'inspirent de la méthode de Newton ont amélioré cet aspect
  - C'est le cas de la fonction fsolve dans Scilab (méthode hybride de Powell)
- Le calcul de la jacobienne peut poser problème, mais il est toujours possible de l'approcher avec précision si on choisit bien la méthode :
  - Pour $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ la $j$ème colonne de $f'(x)$ est donnée par

$$\begin{aligned} f'(x)e_j &= \lim_{h\to 0} \frac{f(x+he_j)-f(x)}{h}, \\ &= \lim_{h\to 0} \frac{\operatorname{Im} f(x+ihe_j)}{h}, \quad \text{(pas complexe)} \end{aligned}$$