Analyse Numérique I

Sup'Galilée, Ingénieurs MACS, 1ère année & L3-MIM

François Cuvelier

Laboratoire d'Analyse Géométrie et Applications
Institut Galilée
Université Paris XIII.

2017/09/11

Divers

- Volume horaire : 10×3h cours - 9×3h TD,
- Prérequis des cours : *Equations différentielles*, *Projets numériques*, *Elements Finis*, *Volumes Finis*, ...
- Note finale : $(P_1 + P_2 + CC_1 + CC_2)/4$ si possible.
  - P_1 partiel 1, 5 ou 6ème séance de cours (1h30),
  - P_2 partiel 2, début janvier ? (1h30),
  - CC_1 contrôle continu par B. Delourme,
  - CC_2 contrôle continu par F. Cuvelier,

Objectifs

Outils de base de l'analyse numérique et du calcul scientifique.

- Mathématiques
- Numériques
- Algorithmique

Bibliographie I

- P.G. Ciarlet, *Analyse numérique et equations différentielles*, DUNOD, 2006.
- ———, *Introduction à l'analyse numérique matricielle et à l'optimisation*, DUNOD, 2006.
- M. Crouzeix and A.L. Mignot, *Analyse numérique des équations différentielles*, Mathématiques appliquées pour la maîtrise, Masson, 1992.
- J.P. Demailly, *Analyse numérique et equations différentielles*, PUG, 1994.
- J.-P. Demailly, *Analyse numérique et équations différentielles*, Grenoble Sciences, EDP Sciences, 2006.
- J.P. Demailly, *Analyse numérique et équations différentielles*, Grenoble Sciences, EDP Sciences, 2012.

Bibliographie II

W. Gander, M.J. Gander, and F. Kwok, *Scientific computing : an introduction using maple and matlab*, Springer, Cham, 2014.

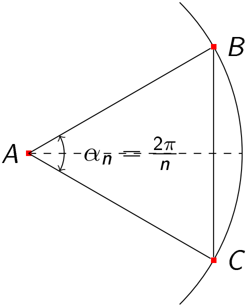
T. Huckle, *Collection of software bugs http://wwwzenger.informatik.tu-muenchen.de/persons/huckle/bugse.html*.

P. Lascaux and R. Théodor, *Analyse numérique matricielle appliquée à l'art de l'ingénieur*, Analyse numérique matricielle appliquée à l'art de l'ingénieur, no. vol. 1 et 2, Dunod, 2004.

A. Quarteroni, R. Sacco, and F. Saleri, *Méthodes Numériques: Algorithmes, analyse et applications (French Edition)*, 1 ed., Springer, September 2007.

Chapitre I

Erreurs : arrondis, bug and Co.



- Aire du cercle : $\mathcal{A} = \pi r^2$.
- Archimède vers 250 avant J-C : $\pi \in]3 + \frac{1}{7}, 3 + \frac{10}{71}[$ avec 96 polygônes.
- Algorithme polygônes inscrits ($r = 1$) : $\mathcal{A} = \lim_{n\to+\infty} \mathcal{A}_n = \lim_{n\to+\infty} \frac{n}{2}\sin(\alpha_n)$ et $\sin\frac{\alpha_n}{2} = \sqrt{\frac{1-\sqrt{1-\sin^2\alpha_n}}{2}}$.

$s \leftarrow 1,\ n \leftarrow 4$ $\qquad\triangleright$ Initialisations
Tantque $s > 1e-10$ **faire** $\qquad\triangleright$ Arrêt si $s = \sin(\alpha)$ est petit
$\quad s \leftarrow sqrt((1 - sqrt(1 - s * s))/2$ $\qquad\triangleright$ nouvelle valeur de $\sin(\alpha/2)$
$\quad n \leftarrow 2 * n$ $\qquad\triangleright$ nouvelle valeur de n
$\quad A \leftarrow (n/2) * s$ $\qquad\triangleright$ nouvelle valeur de l'aire du polygône
Fin Tantque

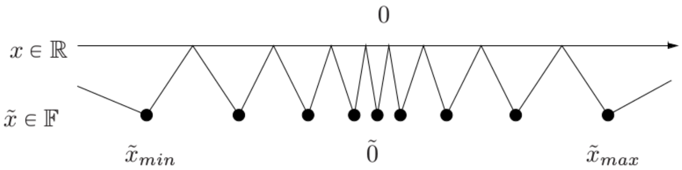
Un exemple : calcul approché de π

n	$\mathcal{A}_n$	$\lvert\mathcal{A}_n - \pi\rvert$	$\sin(\alpha_n)$
4	2.00000000000000	1.141593e+00	1.000000e+00
64	3.13654849054594	5.044163e-03	9.801714e-02
4096	3.14159142150464	1.232085e-06	1.533980e-03
32768	3.14159263346325	2.012654e-08	1.917476e-04
65536	3.14159265480759	1.217796e-09	9.587380e-05
131072	3.14159264532122	8.268578e-09	4.793690e-05
524288	3.14159291093967	2.573499e-07	1.198423e-05
4194304	3.14159655370482	3.900115e-06	1.498030e-06
67108864	3.14245127249413	8.586189e-04	9.365235e-08
2147483648	0.00000000000000	3.141593e+00	0.000000e+00

Nombres flottants en machine

$$\begin{array}{rcll} \tilde{x} & = & \pm m \cdot b^e & \\ m & = & D.D\cdots D, & \text{mantisse} \\ e & = & D\cdots D, & \text{exposant} \end{array}$$

où $D \in \{0, 1, ..., b-1\}$ représente un chiffre et b la base.
Mantisse normalisée : 1er D (avant point) est non nul.



- **Précision machine :** plus petit nombre machine $\mathrm{eps} > 0$ tel que

$$1 + \mathrm{eps} > 1.$$

Matlab/Octave, langage C (double), ... : $\mathrm{eps} = 2.220446049250313e-16$

- **simple précision :** format 32 bits (4 octets), float en langage C.

$$\underbrace{S}_{\text{signe } s}\quad\underbrace{EEEEEEEE}_{\text{exposant } e}\quad\underbrace{FFFFFFFFFFFFFFFFFFFFFFF}_{\text{mantisse } m}$$

0	1	8	9	31

- **double précision :** format 64 bits (8 octets), double en langage C.

$$\underbrace{S}_{\text{signe } s}\quad\underbrace{EEEEEEEEEEE}_{\text{exposant } e}\quad\underbrace{FFFFF..........FFFFFF}_{\text{mantisse } m}$$

0	1	11	12	63

dimension tableau	type	dimension (octets)	dimension total
1000×1000	float	4	$4 * 1000 * 1000 = 4\,Mo$
$10^4 \times 10^4$	float	4	$4 * 10^8 = 400\,Mo$
1000×1000	double	8	$8 * 1000 * 1000 = 8\,Mo$
$10^4 \times 10^4$	double	8	$8 * 10^8 = 800\,Mo$

En langage C (par ex.):

- int a=2147483647,b; alors b=a+1; donne b==-2147483648,
- double a=1/2,b;b=a+1; donne a==0 et b==1,

En Matlab (par ex.) :

- x=eps; alors 1+eps==1 est faux (false=0)
- x=eps/2;y=1; alors (y+x)+x==1 est vrai et y+(x+x)==1 est faux. $(x+y)+z=x+(y+z)$ n'est pas forcément vérifiée sur machine!

Erreurs d'annulation

Il faut être attentifs aux signes.

- $f(x) = \frac{1}{1-\sqrt{1-x^2}}$, si $|x| \leqslant 0.5\sqrt{\mathrm{eps}}$ alors $\sqrt{1-x^2} \equiv 1$!
  $$\Longrightarrow \quad \bar{x} = 0.5\sqrt{\mathrm{eps}}, \quad \boxed{\frac{1}{1-\sqrt{1-\bar{x}^2}} \equiv +\infty}$$
- $$f(x) = \frac{1}{1-\sqrt{1-x^2}} = \frac{1}{1-\sqrt{1-x^2}} \times \frac{1+\sqrt{1-x^2}}{1+\sqrt{1-x^2}} = \frac{1+\sqrt{1-x^2}}{x^2}$$
  $$\Longrightarrow \quad \bar{x} = 0.5\sqrt{\mathrm{eps}}, \quad \boxed{\frac{1+\sqrt{1-x^2}}{x^2} \equiv 3.6029e+16}$$

Erreurs d'annulation : calcul de π, le retour

$$\sin\frac{\alpha_n}{2} = \sqrt{\frac{1-\cos\alpha_n}{2}} = \sqrt{\frac{1-\sqrt{1-\sin^2\alpha_n}}{2}},$$

C'est le calcul de $1-\sqrt{1-\sin^2\alpha_n}$ qui pose problème!

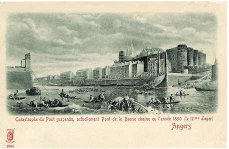
$$\sin\frac{\alpha_n}{2} = \frac{1+\sqrt{1-\sin^2\alpha_n}}{1+\sqrt{1-\sin^2\alpha_n}}\sin\frac{\alpha_n}{2} = \frac{\sin\alpha_n}{\sqrt{2(1+\sqrt{1-\sin^2\alpha_n})}}$$

$s \leftarrow 1,\ n \leftarrow 4,$ ▷ Initialisations
Tantque $s > 1e-10$ **faire** ▷ Arrêt si $s = \sin(\alpha)$ est petit
    $s \leftarrow s/sqrt(2*(1-sqrt(1-s*s)))$ ▷ nouvelle valeur de $\sin(\alpha/2)$
    $n \leftarrow 2*n$ ▷ nouvelle valeur de n
    $A \leftarrow (n/2)*s$ ▷ nouvelle valeur de l'aire du polygône
Fin Tantque

Un exemple : calcul approché de π

n	A_n	$\lvert A_n - \pi \rvert$	$\sin(\alpha_n)$
4	2.00000000000000	1.141593e+00	1.000000e+00
64	3.13654849054594	5.044163e-03	9.801714e-02
4096	3.14159142151120	1.232079e-06	1.533980e-03
32768	3.14159263433856	1.925123e-08	1.917476e-04
65536	3.14159264877699	4.812807e-09	9.587380e-05
131072	3.14159265238659	1.203202e-09	4.793690e-05
524288	3.14159265351459	7.519985e-11	1.198422e-05
4194304	3.14159265358862	1.174172e-12	1.498028e-06
67108864	3.14159265358979	3.552714e-15	9.362676e-08
2147483648	3.14159265358979	1.332268e-15	2.925836e-09

Mais avant de pousuivre ...



(a) Pont de la Basse-Chaîne, *Angers* (1850)

(b) Takoma Narrows Bridge, *Washington* (1940)

(c) Millenium Bridge, *London* (2000)

Figure : Une histoire de ponts

Chapitre II

Langage algorithmique

Plan

Définition

♥ **Definition 1.1: Petit Robert 97**

Algorithmique : Enchaînement d'actions nécessaires à l'accomplissement d'une tâche.

Exemple 1 : permutation

Nous voulons permutter deux voitures sur un parking de trois places numérotées de 1 à 3 et ceci sans géner la circulation.
La première voiture, une Saxo, est sur l'emplacement 2, la seconde, une Clio, est sur l'emplacement 3.
Donner un algorithme permettant de résoudre cette tâche.

Exemple 2 : équation du premier degré

Donner un algorithme permettant de résoudre

$$ax = b$$

Caractéristiques d'un *bon* algorithme

- Il ne souffre d'aucune ambiguité $\Rightarrow$ très clair.
- Combinaison d'opérations (actions) élémentaires.
- Pour toutes les données d'entrée, l'algorithme doit fournir un résultat en un nombre fini d'opérations.

Etape 1 : Définir clairement le problème.

Première approche méthodologique

Etape 1 : Définir clairement le problème.

Etape 2 : Rechercher une méthode de résolution (formules, ...)

Première approche méthodologique

Etape 1 : Définir clairement le problème.

Etape 2 : Rechercher une méthode de résolution (formules, ...)

Etape 3 : Ecrire l'algorithme (par raffinement successif pour des algorithmes *compliqués*).

Plan

Vocabulaire de base

- constantes,variables,
- opérateurs (arithmétiques, relationnels, logiques),
- expressions,
- instructions (simples et composées),
- fonctions.

Données et constantes

- Donnée $\Rightarrow$ introduite par l'utilisateur
- Constante $\Rightarrow$ symbole, identificateur non modifiable

Variables

Definition 2.1

Une variable est un objet dont la valeur est modifiable, qui possède un nom et un type (entier, caractère, réel, complexe, tableau, matrice, vecteur...).

Opérateurs arithmétiques

Nom	Symbole	Exemple
addition	$+$	$a + b$
soustraction	$-$	$a - b$
opposé	$-$	$-a$
produit	$*$	$a * b$
division	$/$	a/b

Opérateurs relationnels

Nom	Symbole	Exemple
identique	$==$	$a == b$
différent	$\sim=$	$a \sim= b$
inférieur	$<$	$a < b$
supérieur	$>$	$a > b$
inférieur ou égal	$<=$	$a <= b$
supérieur ou égal	$>=$	$a >= b$

Opérateurs logiques

Nom	Symbole	Exemple
négation	$\sim$	$\sim a$
ou	$\vert$	$a \vert b$
et	$\&$	$a\&b$

Opérateur d'affectation

Nom	Symbole	Exemple
affectation	$\leftarrow$	$a \leftarrow b$

Expressions

Definition 2.2

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Expressions

Definition 2.3

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Exemple d'expression numérique

$$(b * b - 4 * a * c)/(2 * a)$$

♥ Definition 2.4

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Exemple d'expression numérique

$$(b * b - 4 * a * c)/(2 * a)$$

Opérandes $\Rightarrow$ identifiants a, b, c,
constantes 4 et 2.

Expressions

Definition 2.5

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Exemple d'expression numérique

$$(b * b - 4 * a * c)/(2 * a)$$

Opérandes $\Rightarrow$ identifiants a, b, c,
constantes 4 et 2.

Opérateurs $\Rightarrow$ symboles $*$, $-$ et $/$

Definition 2.6

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Exemple d'expression booléenne

$$(x < 3.14)$$

Definition 2.7

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Exemple d'expression booléenne

$$(x < 3.14)$$

Opérandes $\Rightarrow$ identifiants x et contantes 3.14

Definition 2.8

Une expression est un groupe d'opérandes (i.e. nombres, constantes, variables, ...) liées par certains opérateurs pour former un terme algébrique qui représente une valeur (i.e. un élément de donnée simple)

Exemple d'expression booléenne

$$(x < 3.14)$$

Opérandes $\Rightarrow$ identifiants x et contantes 3.14
Opérateurs $\Rightarrow$ symboles $<$

Instructions

Definition 2.9

Une **instruction** est un ordre ou un groupe d'ordres qui déclenche l'exécution de certaines actions par l'ordinateur. Il y a deux types d'instructions : simple et structuré.

Instructions simples

- affectation d'une valeur a une variable.
- appel d'une fonction (procedure, subroutine, ... suivant les langages).

Instructions structurées

1. les instructions composées, groupe de pulsieurs instructions simples,
2. les instructions répétitives, permettant l'exécution répétée d'instructions simples, (i.e. boucles «pour», «tant que»)
3. les instructions conditionnelles, lesquels ne sont exécutées que si une certaine condition est respectée (i.e. «si»)

Exemple : boucle «pour»

Données : n

$S \leftarrow 0$
Pour $i \leftarrow 1$ **à** n **faire**
    $S \leftarrow S + \cos(i\hat{\ }2)$
Fin Pour

Mais que fait-il?

Exemple : boucle «pour»

Données : n

1: $S \leftarrow 0$
2: **Pour** $i \leftarrow 1$ **à** n **faire**
3: $\quad S \leftarrow S + \cos(i\,\hat{}\,2)$
4: **Fin Pour**

Mais que fait-il?

Calcul de $S = \sum_{i=1}^{n} \cos(i^2)$

Exemple : boucle «tant que»

$i \leftarrow 0, \;\; x \leftarrow 1$
Tantque $i < 1000$ **faire**
    $x \leftarrow x + i * i$
    $i \leftarrow i + 1$
Fin Tantque

Mais que fait-il?

Exemple : boucle «tant que»

$i \leftarrow 0, \;\; x \leftarrow 1$
Tantque $i < 1000$ **faire**
    $x \leftarrow x + i * i$
    $i \leftarrow i + 1$
Fin Tantque

Mais que fait-il?

Calcul de $x = 1 + \sum_{i=0}^{999} i\hat{\ }2$

Exemple : instructions conditionnelles «si»

Données :

age : un réel.

```
Si age >= 18 alors
    affiche('majeur')
Sinon Si age >= 0 alors
    affiche('mineur')
Sinon
    affiche('en devenir')
Fin Si
```

Plan

1. Introduction
2. Pseudo-langage algorithmique
3. Méthodologie de construction
   - Principe
   - Exercices
4. Pseudo-langage algorithmique (suite)

Description du problème

- Spécification d'un ensemble de données
  Origine : énoncé, hypothèses, sources externes, ...
- Spécification d'un ensemble de buts à atteindre
  Origine : résultats, opérations à effectuer, ...
- Spécification des contraintes

Recherche d'une méthode de résolution

- Clarifier l'énoncé.
- Simplifier le problème.
- Ne pas chercher à le traiter directement dans sa globalité.
- S'assurer que le problème est soluble (sinon problème d'indécidabilité!)
- Recherche d'une stratégie de construction de l'algorithme
- Décomposer le problème en sous problèmes partiels plus simples : raffinement.
- Effectuer des raffinements successifs.
- Le niveau de raffinement le plus élémentaire est celui des instructions.

Réalisation d'un algorithme

- Le type des données et des résultats doivent être précisés.
- L'algorithme doit fournir au moins un résultat.
- L'algorithme doit être exécuté en un nombre fini d'opérations.
- L'algorithme doit être spécifié clairement, sans la moindre ambiguïté.

Exercice 3.1

On cherche les solutions réelles de l'équation

$$ax^2 + bx + c = 0, \tag{1}$$

en supposant que $a \in \mathbb{R}^*$, $b \in \mathbb{R}$ et $c \in \mathbb{R}$ sont donnés.
Ecrire un algorithme permettant de résoudre cette équation.

Exercice 3.2

Ecrire un algorithme permettant de calculer

$$S(x) = \sum_{k=1}^{n} k \sin(2 * k * x)$$

Exercices

Exercice 3.3

Ecrire un algorithme permettant de calculer

$$P(z) = \prod_{n=1}^{k} sin(2 * k * z/n)^k$$

Exercice 3.4

Reprendre les quatre exercices précédants en utilisant les boucles «tant que».

Plan

Les fonctions

Les fonctions permettent

- d'automatiser certaines tâches répétitives au sein d'un même algorithme,
- d'ajouter à la clarté de la l'algorithme,
- l'utilisation de portion de code dans un autre algorithme,
- ...

Les fonctions prédéfinies

- les fonctions d'affichage et de lecture : **Affiche**, **Lit**

Les fonctions prédéfinies

- les fonctions d'affichage et de lecture : **Affiche**, **Lit**
- les fonctions mathématiques :

$$\sin, \ \cos, \ \exp, \ \cdots$$

Les fonctions prédéfinies

- les fonctions d'affichage et de lecture : **Affiche**, **Lit**
- les fonctions mathématiques :

$$\sin,\ \cos,\ \exp,\ \cdots$$

- les fonctions de gestion de fichiers
- ...

Ecrire ses propres fonctions

1. Que doit-on calculer/réaliser précisement (but)?
2. Quelles sont les données (avec leurs limitations)?

Syntaxe

Fonction $[args_1, \ldots, args_n] \leftarrow$ NomFonction($arge_1, \ldots, arge_m$)
  instructions
Fin Fonction

Fonction $args \leftarrow$ NomFonction($arge_1, \ldots, arge_m$)
  instructions
Fin Fonction

Ecrire une fonction permettant de résoudre

$$ax + b = 0$$

- But :
- Données :
- Résultats :

Ecrire une fonction permettant de résoudre

$$ax + b = 0$$

- But :
  trouver $x \in \mathbb{R}$ solution de $ax + b = 0$.
- Données :
- Résultats :

équation du premier degré

Ecrire une fonction permettant de résoudre

$$ax + b = 0$$

- But :
  trouver $x \in \mathbb{R}$ solution de $ax + b = 0$.
- Données :
  $a \in \mathbb{R}^*$ et $b \in \mathbb{R}$.
- Résultats :

Ecrire une fonction permettant de résoudre

$$ax + b = 0$$

- But :
  trouver $x \in \mathbb{R}$ solution de $ax + b = 0$.
- Données :
  $a \in \mathbb{R}^*$ et $b \in \mathbb{R}$.
- Résultats :
  $x \in \mathbb{R}$.

Algorithme 1 Exemple de fonction : Résolution de l'équation du premier degré $ax + b = 0$.

Données : a : nombre réel différent de 0
b : nombre réel.
Résultat : x : un réel.

Fonction $x \leftarrow$ REPD(a, b)
    $x \leftarrow -b/a$
Fin Fonction

On cherche les solutions réelles de l'équation

$$ax^2 + bx + c = 0, \tag{2}$$

Pour celà, on pose $\Delta = b^2 - 4ac$

- si $\Delta < 0$ alors les deux solutions sont complexes,
- si $\Delta = 0$ alors la solution est $x = -\frac{b}{2a}$,
- si $\Delta > 0$ alors les deux solutions sont $x_1 = \frac{-b-\sqrt{\Delta}}{2a}$ et $x_2 = \frac{-b-\sqrt{\Delta}}{2a}$.

Exercice

1. Ecrire la fonction `discriminant` permettant de calculer le discriminant de l'équation (2).
2. Ecrire la fonction `RESD` permettant de résoudre l'équation (2) en utilisant la fonction `discriminant`.
3. Ecrire un programme permettant de valider ces deux fonctions.

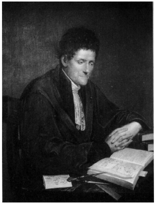
Chapitre III

Résolution de systèmes non-linéaires

Racines/zéros d'un polynôme

- **degré 2** : Babylonniens en 1600 avant J.-C.
- **degré 3** : *Scipio del Ferro* (1465-1526, mathématicien italien) et *Niccolo Fontana* (1499-1557, mathématicien italien)
- **degré 4** : *Ludovico Ferrari* (1522-1565, mathématicien italien)
- **degré 5** : *Paolo Ruffini* (1765-1822, mathématicien italien) en 1799, *Niels Henrick Abel* (1802-1829, mathématicien norvégien) en 1824, montrent qu'il n'existe **pas de solution analytique**.

(a) *Niccolo Fontana* 1499-1557, mathématicien italien



(b) *Paolo Ruffini* 1765-1822, mathématicien italien

(c) *Niels Henrick Abel* 1802-1829, mathématicien norvégien

Plan

5. Recherche des zéros d'une fonction
   - Méthode de dichotomie ou de bissection
   - Points fixes d'une fonction (dimension 1)

6. Résolution de systèmes non linéaires

Soit $f : \mathcal{D} \subset \mathbb{R} \longrightarrow \mathbb{R}$ une fonction continue.

$$\alpha \in \mathcal{D} \text{ tels que } f(\alpha) = 0.$$

Soit $I =]a, b[$, $\bar{I} \subset \mathcal{D}$ on suppose $\exists! \alpha \in I$ tel que $f(\alpha) = 0$.

principe de la méthode de dichotomie : Soit I un intervalle contenant un **unique zéro** de la fonction f, on le divise par son milieu en deux intervalles et on détermine lequel des deux contient le zéro. On itére ce processus sur le nouvel intervalle.

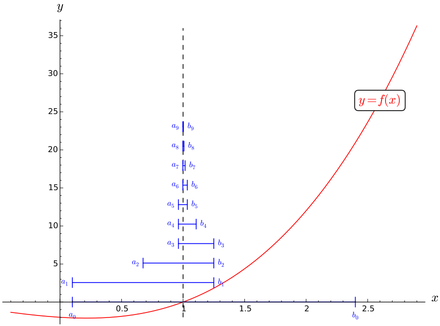


Figure : Méthode de dichotomie: $f(x) = (x+2)(x+1)(x-1)$

principe de la méthode de dichotomie : Soit I un intervalle contenant un **unique zéro** de la fonction f, on le divise par son milieu en deux intervalles et on détermine lequel des deux contient le zéro. On itére ce processus sur le nouvel intervalle.

- $a_0 = a$, $b_0 = b$ et $x_0 = \frac{a+b}{2}$,
- $\forall k \in \mathbb{N}$

$$\begin{cases} a_{k+1} = b_{k+1} = x_k & \text{si } f(x_k) = 0, \\ a_{k+1} = x_k, \ b_{k+1} = b_k & \text{si } f(b_k)f(x_k) < 0, \\ a_{k+1} = a_k, \ b_{k+1} = x_k & \text{sinon (i.e. } f(a_k)f(x_k) < 0.) \end{cases}$$

et

$$x_{k+1} = (a_{k+1} + b_{k+1})/2.$$

Proposition 5.1

Soit $f : [a,b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ une fonction continue vérifiant $f(a)f(b) < 0$ et admettant $\alpha \in]a,b[$ comme **unique** solution de $f(x) = 0$. Alors la suite $(x_k)_{k\in\mathbb{N}}$ définie par la méthode de dichotomie converge vers α et

$$|x_k - \alpha| \leqslant \frac{b-a}{2^{k+1}}, \quad \forall k \in \mathbb{N}.$$

On a alors $\forall \epsilon > 0$, $\forall k \geqslant \frac{\log(\frac{b-a}{\epsilon})}{\log(2)} - 1$

$$|x_k - \alpha| \leqslant \epsilon.$$

Exercice 5.1

On suppose que la fonction f est continue sur $[a, b]$, vérifie $f(a)f(b) < 0$ et qu'il existe un unique $\xi \in]a, b[$ tel que $f(x) = 0$.

Q. 1

1. *Montrer que les suites* (a_k) *et* (b_k) *convergent vers* α.
2. *En déduire que la suite* (x_k) *converge vers* α.

Q. 2

1. *Montrer que pour tout* $k \in \mathbb{N}$, $|x_k - \alpha| \leqslant \frac{b-a}{2^{k+1}}$.
2. *Soit* $\epsilon > 0$. *En déduire que si* $k \geqslant \frac{\log(\frac{b-a}{\epsilon})}{\log(2)} - 1$ *alors* $|x_k - \alpha| \leqslant \epsilon$.

Algorithme

- Que cherche-t'on?

- Quelles sont les données du problèmes?

Algorithme

- Que cherche-t'on?

  Résultat :

  α_ϵ : un réel tel que $|\alpha_\epsilon - \alpha| \leqslant \epsilon$.

- Quelles sont les données du problèmes?

- Que cherche-t'on?

Résultat :

α_ϵ : un réel tel que $|\alpha_\epsilon - \alpha| \leqslant \epsilon$.

- Quelles sont les données du problèmes?

Données : a, b : deux réels $a < b$,

f : $f : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ vérifiant les hypothèses de la proposition ,

ϵ : un réel strictement positif.

Algorithme

Algorithme 2 $\boxed{\mathcal{R}_0}$

1: $k_{\min} \leftarrow \mathrm{E}\left(\frac{\log(\frac{b-a}{\epsilon})}{\log(2)}\right)$
2: $\boxed{\text{Calcul de la suite } (x_k)_{k=0}^{k_{\min}}}$
3: $\alpha_\epsilon \leftarrow x_{k_{\min}}$

Algorithme 2 $\boxed{\mathcal{R}_1}$

1: $k_{\min} \leftarrow \mathrm{E}\left(\frac{\log(\frac{b-a}{\epsilon})}{\log(2)}\right)$
2: Initialisation de x_0
3: **Pour** $k \leftarrow 0$ **à** $k_{\min} - 1$ **faire**
4: Calcul de la suite (x_{k+1})
5: **Fin Pour**
6: $\alpha_\epsilon \leftarrow x_{k_{\min}}$

Algorithme

Algorithme 2 $\boxed{\mathcal{R}_1}$

$k_{\min} \leftarrow \mathrm{E}\left(\frac{\log(\frac{b-a}{\epsilon})}{\log(2)}\right)$

$\boxed{\text{Initialisation de } x_0}$

Pour $k \leftarrow 0$ à $k_{\min} - 1$ **faire**

  $\boxed{\text{Calcul de la suite } (x_{k+1})}$

Fin Pour

$\alpha_\epsilon \leftarrow x_{k_{\min}}$

Algorithme 2 $\boxed{\mathcal{R}_2}$

$k_{\min} \leftarrow \mathrm{E}\left(\frac{\log(\frac{b-a}{\epsilon})}{\log(2)}\right)$

$a_0 \leftarrow a, \;\; b_0 \leftarrow b$

$x_0 \leftarrow \frac{a_0+b_0}{2}$

Pour $k \leftarrow 0$ à $k_{\min} - 1$ **faire**

  Si $f(x_k) == 0$ **alors**

    $a_{k+1} \leftarrow x_k, \;\; b_{k+1} \leftarrow x_k$

  Sinon Si $f(x_k)f(b_k) < 0)$ **alors**

    $a_{k+1} \leftarrow x_k, \;\; b_{k+1} \leftarrow b_k$

  Sinon

    $a_{k+1} \leftarrow a_k, \;\; b_{k+1} \leftarrow x_k$

  Fin Si

  $x_{k+1} \leftarrow \frac{a_{k+1}+b_{k+1}}{2}$

Fin Pour

$\alpha_\epsilon \leftarrow x_{k_{\min}}$

Algorithme 2 Méthode de dichotomie : version 1

Données : $a,\ b$: deux réels $a < b$,
f : $f : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ vérifiant les hypothèses de la proposition 5.1,
eps : un réel strictement positif.

Résultat : x : un réel tel que $|x - \alpha| \leqslant \mathrm{eps}$.

Fonction $x \leftarrow$ DICHOTOMIE1 (f, a, b, eps)
    $\mathrm{kmin} \leftarrow \mathrm{E}(\log((b-a)/\mathrm{eps})/\log(2))$
    $\boldsymbol{A},\ \boldsymbol{B},\ \boldsymbol{X} \in \mathbb{R}^{\mathrm{kmin}+1}$ $\triangleright$ $\boldsymbol{A}(k+1)$ contiendra a_k, ...
    $\boldsymbol{A}(1) \leftarrow a,\ \boldsymbol{B}(1) \leftarrow b,\ \boldsymbol{X}(1) \leftarrow (a+b)/2$
    Pour $k \leftarrow 1$ **à** kmin **faire**
        Si $f(\boldsymbol{X}(k)) == 0$ **alors**
            $\boldsymbol{A}(k+1) \leftarrow \boldsymbol{X}(k),\ \boldsymbol{B}(k+1) \leftarrow \boldsymbol{X}(k)$
        Sinon Si $f(\boldsymbol{B}(k))f(\boldsymbol{X}(k)) < 0$ **alors**
            $\boldsymbol{A}(k+1) \leftarrow \boldsymbol{X}(k),\ \boldsymbol{B}(k+1) \leftarrow \boldsymbol{B}(k)$
        Sinon
            $\boldsymbol{A}(k+1) \leftarrow \boldsymbol{A}(k),\ \boldsymbol{B}(k+1) \leftarrow \boldsymbol{X}(k)$
        Fin Si
        $\boldsymbol{X}(k+1) \leftarrow (\boldsymbol{A}(k+1) + \boldsymbol{B}(k+1))/2$
    Fin Pour
    $x \leftarrow \boldsymbol{X}(\mathrm{kmin}+1)$
Fin Fonction

Algorithme : versions 2, 3 et + si affinités

- $A = a$, $B = b$ et $x_0 = \frac{A+B}{2}$,
- $\forall k \in [\![0, k_{\min} - 1]\!]$,

$$\begin{cases} A = B = x_k & \text{si } f(x_k) = 0, \\ A = x_k,\ B \text{ inchangé} & \text{si } f(B)f(x_k) < 0, \\ B = x_k,\ A \text{ inchangé} & \text{sinon (i.e. } f(A)f(x_k) < 0.) \end{cases}$$

et

$$x_{k+1} = \frac{A+B}{2}$$

Algorithme 3 Méthode de dichotomie : version 2

Données : $a, \ b$: deux réels $a < b$,
f : $f : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ vérifiant les hypothèses de la proposition 5.1,
eps : un réel strictement positif.

Résultat : x : un réel tel que $|x - \alpha| \leqslant \mathrm{eps}$.

Fonction $x \leftarrow$ Dichotomie2 (f, a, b, eps)
  $\mathrm{kmin} \leftarrow \mathrm{E}(\log((b - a)/\mathrm{eps})/\log(2))$
  $\boldsymbol{X} \in \mathbb{R}^{\mathrm{kmin}+1}$ $\rhd$ $\boldsymbol{X}(k + 1)$ contiendra x_k, ...
  $\mathrm{A} \leftarrow a, \ \mathrm{B} \leftarrow b, \ \boldsymbol{X}(1) \leftarrow (\mathrm{A} + \mathrm{B})/2$
  Pour $k \leftarrow 1$ **à** kmin **faire**
    Si $f(\boldsymbol{X}(k)) == 0$ **alors**
      $\mathrm{A} \leftarrow \boldsymbol{X}(k), \ \mathrm{B} \leftarrow \boldsymbol{X}(k)$
    Sinon Si $f(\mathrm{B})f(\boldsymbol{X}(k)) < 0$ **alors**
      $\mathrm{A} \leftarrow \boldsymbol{X}(k)$ $\rhd$ B inchangé
    Sinon
      $\mathrm{B} \leftarrow \boldsymbol{X}(k)$ $\rhd$ A inchangé
    Fin Si
    $\boldsymbol{X}(k + 1) \leftarrow (\mathrm{A} + \mathrm{B})/2$
  Fin Pour
  $x \leftarrow \boldsymbol{X}(\mathrm{kmin} + 1)$
Fin Fonction

Algorithme 4 Méthode de dichotomie : version 3

Données : $a,\ b$: deux réels $a < b$,
f : $f : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ vérifiant les hypothèsesde la proposition 5.1,
eps : un réel strictement positif.

Résultat : x : un réel tel que $|x - \alpha| \leqslant \mathrm{eps}$.

Fonction $x \leftarrow$ DICHOTOMIE2 (f, a, b, eps)
    $\mathrm{kmin} \leftarrow \mathrm{E}(\log((b-a)/\mathrm{eps})/\log(2))$
    $A,\ B \in \mathbb{R}$
    $A \leftarrow a,\ B \leftarrow b,\ x \leftarrow (a+b)/2$
    Pour $k \leftarrow 1$ **à** kmin **faire**
        Si $f(x) == 0$ **alors**
            $A \leftarrow x,\ B \leftarrow x$
        Sinon Si $f(B)f(x) < 0$ **alors**
            $A \leftarrow x$ ▷ B inchangé
        Sinon
            $B \leftarrow x$ ▷ A inchangé
        Fin Si
        $x \leftarrow (A+B)/2$
    Fin Pour
Fin Fonction

Algorithme 5 Méthode de dichotomie : version 4

Données : $a,\ b$: deux réels $a < b$,
f : $f : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ ...,
eps : un réel strictement positif.

Résultat : x : un réel tel que $|x - \alpha| \leqslant \mathrm{eps}$.

Fonction $x \leftarrow$ DICHOTOMIE4 (f, a, b, eps)
  $A,\ B \in \mathbb{R}$
  $A \leftarrow a,\ B \leftarrow b,\ x \leftarrow (a + b)/2$
  Tantque $|x - A| > \mathrm{eps}$ **faire**
    Si $f(x) == 0$ **alors**
      $A \leftarrow x,\ B \leftarrow x$
    Sinon Si $f(B)f(x) < 0$ **alors**
      $A \leftarrow x$ ▷ B inchangé
    Sinon
      $B \leftarrow x$ ▷ A inchangé
    Fin Si
    $x \leftarrow (A + B)/2$
  Fin Tantque
Fin Fonction

Que pensez vous de cet algorithme?

Algorithme 6 Méthode de dichotomie : version 5

Données : a, b : deux réels $a < b$,
f : $f : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$

Résultat : x : un réel tel que $f(x) = 0$.

Fonction $x \leftarrow$ DICHOTOMIE5 (f, a, b)
    $A, \ B \in \mathbb{R}$
    $A \leftarrow a, \ B \leftarrow b, \ x \leftarrow (a + b)/2, \ \mathrm{xp} \leftarrow a$
    Tantque $x \sim= \mathrm{xp}$ **faire**
        Si $f(B)f(x) < 0$ **alors**
            $A \leftarrow x$ ▻ B inchangé
        Sinon
            $B \leftarrow x$ ▻ A inchangé
        Fin Si
        $\mathrm{xp} \leftarrow x$
        $x \leftarrow (A + B)/2$
    Fin Tantque
Fin Fonction

Points fixes

Soit $\Phi : [a, b] \subset \mathbb{R} \longrightarrow \mathbb{R}$ une fonction donnée. Rechercher un **point fixe** de Φ revient à

Trouver $\alpha \in [a, b]$ tel que

$$\alpha = \Phi(\alpha).$$

L'algorithme de la **méthode du point fixe** consiste en la construction, si elle existe, de la suite

$$x^{(k+1)} = \Phi(x^{(k)}), \ \forall k \in \mathbb{N} \tag{3}$$

avec $x^{(0)} \in [a, b]$ donné.

Definition 5.2

On dit qu'une suite $(x_k)_{k\in\mathbb{N}}$, obtenue par une méthode numérique, **converge vers** α **avec un ordre** $p \geqslant 1$ si

$$\exists k_0 \in \mathbb{N},\ \exists C > 0 \ \text{ tels que } \ \frac{|x_{k+1} - \alpha|}{|x_k - \alpha|^p} \leqslant C,\ \forall k \geqslant k_0. \tag{4}$$

où $C < 1$ si $p = 1$.

Théorème 6: Théorème du point fixe dans $\mathbb{R}$

Soient $[a, b]$ un intervalle non vide de $\mathbb{R}$ et Φ une application continue de $[a, b]$ dans lui-même. Alors, il existe **au moins** un point $\alpha \in [a, b]$ vérifiant $\Phi(\alpha) = \alpha$. Le point α est appelé **point fixe de la fonction** Φ.

De plus, si Φ est contractante (lipschitzienne de rapport $L \in [0, 1[$), c'est à dire

$$\exists L < 1 \text{ t.q. } |\Phi(x) - \Phi(y)| \leqslant L|x - y| \ \forall (x, y) \in [a, b]^2, \tag{5}$$

alors Φ admet un **unique** point fixe $\alpha \in [a, b]$, la suite définie en (3) converge vers α avec un ordre 1 pour toute donnée initiale $x^{(0)}$ dans $[a, b]$, et l'on a les deux estimations suivantes :

$$|x_k - \alpha| \leqslant L^k |x_0 - \alpha|, \ \forall k \geqslant 0, \tag{6}$$

$$|x_k - \alpha| \leqslant \frac{L}{1 - L} |x_k - x_{k-1}|, \ \forall k \geqslant 0, \tag{7}$$

Théorème 7: Convergence globale, méthode du point fixe

Soit $\Phi \in \mathcal{C}^1([a,b])$ vérifiant $\Phi([a,b]) \subset [a,b]$ et

$$\exists L < 1 \text{ tel que } \forall x \in [a,b],\ |\Phi'(x)| \leqslant L, \tag{8}$$

Soit $x_0 \in [a,b]$ et $(x_k)_{k\in\mathbb{N}}$ la suite définie par $x_{k+1} = \Phi(x_k)$. On a alors

1. la fonction Φ admet un unique point fixe $\alpha \in [a,b]$,
2. $\forall k \in \mathbb{N},\ x_k \in [a,b]$,
3. la suite (x_k) converge vers α avec un ordre 1 au moins.
4. $$\lim_{k\to+\infty} \frac{x_{k+1}-\alpha}{x_k-\alpha} = \Phi'(\alpha). \tag{9}$$

Théorème 8: Convergence locale du point fixe

Soit α un point fixe d'une fonction Φ de classe $\mathcal{C}^1$ au voisinage de α. Si $|\Phi'(\alpha)| < 1$, alors il existe $\delta > 0$ pour lequel x_k converge vers α pour tout x_0 tel que $|x_0 - \alpha| < \delta$. De plus, on a

$$\lim_{k\to+\infty} \frac{x_{k+1} - \alpha}{x_k - \alpha} = \Phi'(\alpha). \tag{10}$$

Proposition 8.1:

Soit $p \in \mathbb{N}^*$, et $\Phi \in \mathcal{C}^{p+1}(\mathcal{V})$ pour un certain voisinage $\mathcal{V}$ de α point fixe de Φ. Si $\Phi^{(i)}(\alpha) = 0$, pour $1 \leqslant i \leqslant p$ et si $\Phi^{(p+1)}(\alpha) \neq 0$, alors la méthode de point fixe associée à la fonction Φ est d'ordre $p+1$ et

$$\lim_{k\to+\infty} \frac{x_{k+1} - \alpha}{(x_k - \alpha)^{p+1}} = \frac{\Phi^{(p+1)}(\alpha)}{(p+1)!}. \tag{11}$$

On s'interesse ici au point fixe $\alpha = 1$ de la fonction $\Phi\ :\ x \mapsto x^2$.

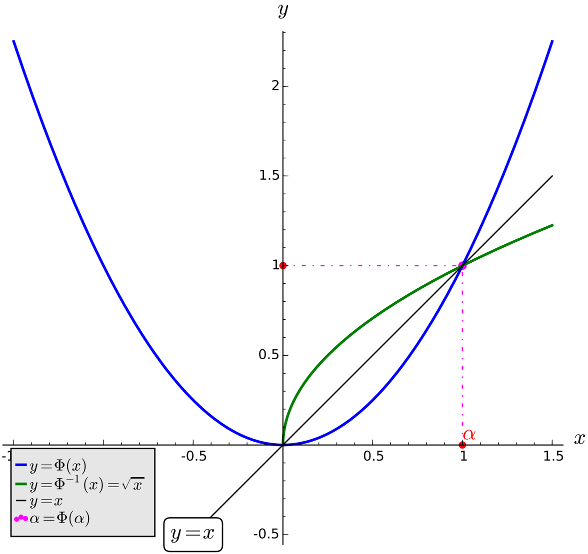
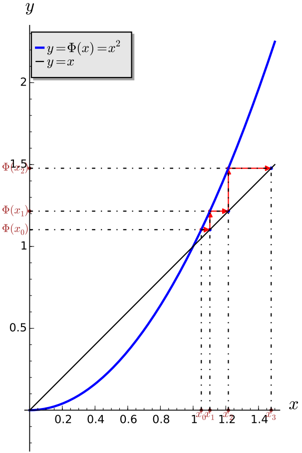
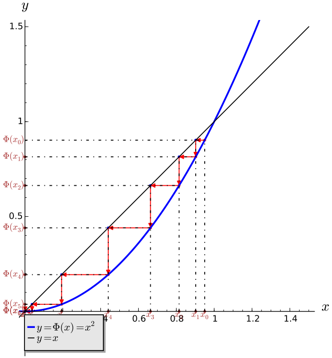


Figure : fonction x^2 et son fonction inverse $\sqrt{x}$ sur $[0, +\infty[$

$\Phi'(\alpha) = 2$: point fixe répulsif



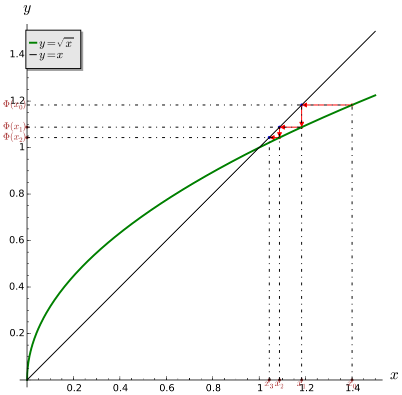
(a) $x_0 = 1.05$



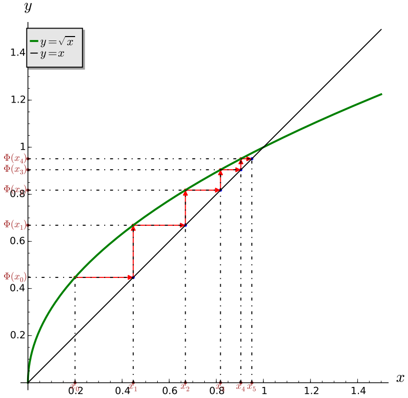
(b) $x_0 = 0.950$

Figure : $\alpha = 1$, point fixe répulsif de $x \mapsto x^2$

$(\Phi^{-1})'(1) = 1/2 < 1,$: **point fixe attractif**



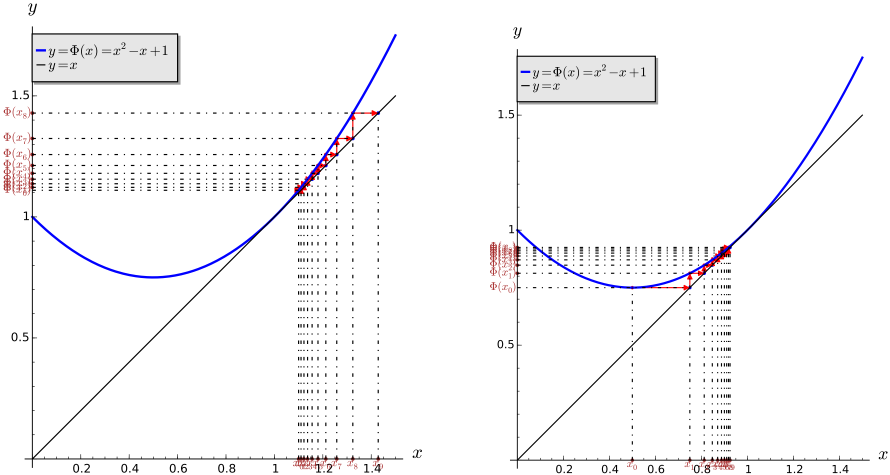
(a) $x_0 = 1.40$



(b) $x_0 = 0.200$

Figure : $\alpha = 1$, point fixe attractif de $\Phi^{-1} : x \mapsto \sqrt{x}$

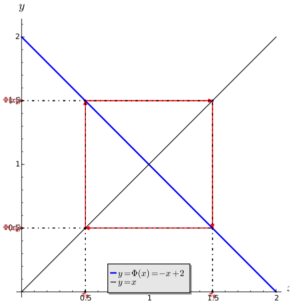
fonction $f : x \mapsto x^2 - x + 1$: **point fixe $\alpha = 1$, $f'(\alpha) = 1$.**



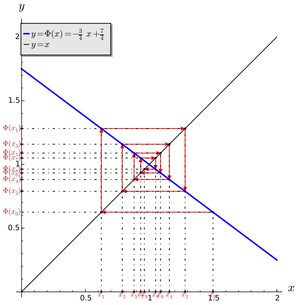
(a) $x_0 = 1.1$, α point répulsif

(b) $x_0 = 0.50$, α point attractif

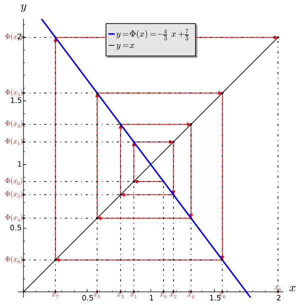
Figure : $\alpha = 1$, point fixe attractif ou répulsif de $x \mapsto x^2 - x + 1$



(a) $x_0 = 1.50$



(b) $x_0 = 1.50$



(c) $x_0 = 1.10$

Figure : $\alpha = 1$, point fixe de fonctions affines particulières

Algorithme générique du point fixe

$$x^{(k+1)} = \Phi(x^{(k)}), \ \forall k \in \mathbb{N}, \text{ avec } x^{(0)} \in [a, b] \text{ donné.}$$

Algorithme 7 Méthode de point fixe : version **Tantque** *formel*

1: $k \leftarrow 0$
2: **Tantque** non convergence **faire**
3: $\quad x_{k+1} \leftarrow \Phi(x_k)$
4: $\quad k \leftarrow k + 1$
5: **Fin Tantque**
6: $\alpha_\epsilon \leftarrow x_k$ ▷ le dernier calculé.

Algorithme 8 Méthode de point fixe : version **Répéter** *formel*

1: $k \leftarrow 0$
2: **Répéter**
3: $\quad k \leftarrow k + 1$
4: $\quad x_k \leftarrow \Phi(x_{k-1})$
5: **jusqu'à** convergence
6: $\alpha_\epsilon \leftarrow x_k$ ▷ le dernier calculé.

Critères d'arrêt?

- On n'est pas sûr de converger $\Longrightarrow$ kmax nb maximum d'itérations
- Si on converge, on s'arrête dès que $|\Phi(x_k) - x_k| = |x_{k+1} - x_k| \leqslant \mathrm{tol}$ (*critère de Poisson? Que mettre à Laplace?*)

Algorithme générique du point fixe

Algorithme 9 Méthode de point fixe : version **Tantque** *formel* avec critères d'arrêt

$k \leftarrow 0$
$\mathrm{err} \leftarrow |\Phi(x_0) - x_0|$ ▷ ou $\frac{|\Phi(x_0)-x_0|}{|x_0|+1}$
Tantque $\mathrm{err} > \epsilon$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k+1$
    $x_k \leftarrow \Phi(x_{k-1})$
    $\mathrm{err} \leftarrow |\Phi(x_k) - x_k|$ ▷ ou $\frac{|\Phi(x_k)-x_k|}{|x_k|+1}$
Fin Tantque
Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow x_k$ ▷ $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$
Fin Si

Algorithme 10 Méthode de point fixe : version **Répéter** *formel* avec critères d'arrêt

$k \leftarrow 0$
Répéter
    $\mathrm{err} \leftarrow |\Phi(x_k) - x_k|$ ▷ ou $\frac{|\Phi(x_k)-x_k|}{|x_k|+1}$
    $x_{k+1} \leftarrow \Phi(x_k)$
    $k \leftarrow k+1$
jusqu'à $\mathrm{err} \leqslant \mathrm{tol}$ ou $k > \mathrm{kmax}$
Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow x_k$ ▷ $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$
Fin Si

Algorithme générique du point fixe

Algorithme 11 Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :
- Φ : $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
- x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
- tol : la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
- kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :
- α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$ (ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
    $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
    $\mathrm{x} \leftarrow x_0$, $\mathrm{fx} \leftarrow \Phi(x_0)$,
    $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
    Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
        $k \leftarrow k + 1$
        $\mathrm{x} \leftarrow \mathrm{fx}$
        $\mathrm{fx} \leftarrow \Phi(\mathrm{x})$
        $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
    Fin Tantque
    Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
        $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
    Fin Si
Fin Fonction

Algorithme 12 Méthode de point fixe : version **Répéter** avec critères d'arrêt

Données :
- Φ : $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
- x_0 : donnée initiale, $x_{\in}\mathbb{R}$,
- tol : la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
- kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :
- α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$ (ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
    $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
    $\mathrm{x} \leftarrow x_0$
    Répéter
        $\mathrm{xp} \leftarrow \mathrm{x}$
        $\mathrm{x} \leftarrow \Phi(\mathrm{xp})$
        $\mathrm{err} \leftarrow |\mathrm{x} - \mathrm{xp}|$ ▷ ou $\frac{|\mathrm{x}-\mathrm{xp}|}{|\mathrm{xp}|+1}$
        $k \leftarrow k + 1$
    jusqu'à $\mathrm{err} \leqslant \mathrm{tol}$ ou $k > \mathrm{kmax}$
    Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
        $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
    Fin Si
Fin Fonction

$$f(x) = 0 \iff \Phi(x) := x + f(x) = x.$$

si $\mathcal{F} \in \mathcal{C}^0$ tel que $\mathcal{F}(0) = 0$ alors

$$f(x) = 0 \iff \Phi(x) := x + \mathcal{F}(f(x)) = x.$$

Objectif : Construire une suite x_{k+1} tel que $|x_{k+1} - \alpha| \leqslant |x_k - \alpha|$.

formule de taylor :

$$f(\alpha) = 0 = f(x_k) + hf'(\xi) \text{ avec } h = \alpha - x_k.$$

Soit $q_k \approx f'(\xi)$ et $\tilde{h}$ solution de

$$f(x_k) + \tilde{h}q_k = 0$$

Si $q_k \neq 0$, on obtient la suite itérative $x_{k+1} = x_k + \tilde{h}$ i.e.

$$x_{k+1} = x_k - \frac{f(x_k)}{q_k}, \ \forall k \in \mathbb{N} \tag{12}$$

Applications à la recherche de racines

$$x_{k+1} = x_k - \frac{f(x_k)}{q_k}, \ \forall k \in \mathbb{N}$$

x_{k+1} : intersection droite de pente q_k passant par $((x_k), f(x_k))$ avec (Ox)

- **Méthode de la corde** :

$$q_k = q = \frac{f(b) - f(a)}{b - a}$$

- **Méthode de la sécante** :

$$q_k = \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

  où x_{-1} et x_0 sont données,

- **Méthode de Newton** : en supposant f' connu, on prend

$$q_k = f'(x_k).$$

$$x_{k+1} = x_k - \frac{b-a}{f(b)-f(a)} f(x_k), \ \forall k \in \mathbb{N}.$$

On pose $\Phi(x) = x - \frac{b-a}{f(b)-f(a)} f(x)$, alors $x_{k+1} = \Phi(x_k)$.

Proposition 8.2: convergence, méthode de la corde

Soit $f \in \mathcal{C}^1([a,b])$ tel que $f(b) \neq f(a)$ et $\lambda = \frac{f(b)-f(a)}{b-a}$. On note $(x_k)_{k\in\mathbb{N}}$ la suite définie par $x_0 \in [a,b]$ et pour tout $k \geqslant 0$

$$x_{k+1} = x_k - \frac{f(x_k)}{\lambda}. \tag{13}$$

On suppose de plus que $\forall x \in [a,b]$

$$\min(\lambda(x-a), \lambda(x-b)) \leqslant f(x) \leqslant \max(\lambda(x-a), \lambda(x-b)) \tag{14}$$

$$\min(0, 2\lambda) < f'(x) < \max(0, 2\lambda) \tag{15}$$

alors la suite (x_k) converge vers l'unique racine $\alpha \in [a,b]$ de f.

Exercice 8.1:

Soit f une fonction de classe $\mathcal{C}^1$ sur $[a, b]$ vérifiant $f(a)f(b) < 0$. et $\lambda = \frac{f(b)-f(a)}{b-a}$. Soit $x_0 \in [a, b]$ donné. La suite obtenue par la méthode de la corde est donnée par

$$x_{k+1} = x_k - \frac{f(x_k)}{\lambda}, \ \forall k \in \mathbb{N}.$$

On note $\Phi(x) = x - \frac{f(x)}{\lambda}$.

Q. 1

Montrer que si pour tout $x \in [a, b]$ on a

$$\min(\lambda(x-a), \lambda(x-b)) \leqslant f(x) \leqslant \max(\lambda(x-a), \lambda(x-b)) \tag{16}$$

alors $\Phi([a, b]) \subset [a, b]$.

Q. 2

Montrer que si pour tout $x \in [a, b]$ on a

$$\min(0, 2\lambda) < f'(x) < \max(0, 2\lambda) \tag{17}$$

alors $|\Phi'(x)| < 1$.

Q. 3

En déduire que sous les deux conditions précédentes la méthode de la corde converge vers l'unique solution $\alpha \in [a, b]$ de $f(x) = 0$.

Méthode de la corde

Proposition 8.3: ordre de convergence de la méthode de la corde

Soit $f \in \mathcal{C}^1([a,b])$ tel que $f(b) \neq f(a)$. Si la suite (x_k) définie par la méthode de la corde en (13) converge vers $\alpha \in]a,b[$ alors la convergence est au moins d'ordre 1.
De plus, si f est de classe $\mathcal{C}^2$ sur un certain voisinage $\mathcal{V}$ de α et si $f'(\alpha) = \frac{f(b)-f(a)}{b-a}$ alors la convergence est au moins d'ordre 2.

Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :

Φ : $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
tol : la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$
(ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
  $\mathrm{x} \leftarrow x_0$, $\mathrm{fx} \leftarrow \Phi(x_0)$,
  $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{x} \leftarrow \mathrm{fx}$
    $\mathrm{fx} \leftarrow \Phi(\mathrm{x})$
    $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Méthode de la corde :

$$\Phi(x) = x - \frac{b-a}{f(b)-f(a)} f(x)$$

Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :

- Φ : $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
- x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
- tol : la tolérance, $\mathrm{tol} \in \mathbb{R}^+$,
- kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

- α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$
  (ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \emptyset$
  $\mathrm{x} \leftarrow x_0$, $\mathrm{fx} \leftarrow \Phi(x_0)$,
  $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{x} \leftarrow \mathrm{fx}$
    $\mathrm{fx} \leftarrow \Phi(\mathrm{x})$
    $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Algorithme 13 Méthode de la corde

Données :

- f : $f : \mathbb{R} \longrightarrow \mathbb{R}$,
- a, b : deux réels tels que $f(a) \neq f(b)$,
- x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
- tol : la tolérance, $\mathrm{tol} \in \mathbb{R}^+$,
- kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

- α_{tol} : un réel tel que $|f(\alpha_{\mathrm{tol}})| \leqslant \mathrm{tol}$

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ Corde ($f, a, b, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \emptyset$
  $q \leftarrow \frac{b-a}{f(b)-f(a)}$
  $\mathrm{x} \leftarrow x_0$,
  $\mathrm{err} \leftarrow \mathrm{tol} + 1$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{xp} \leftarrow \mathrm{x}$
    $\mathrm{x} \leftarrow \mathrm{xp} - q * f(\mathrm{xp})$
    $\mathrm{err} \leftarrow |\mathrm{x} - \mathrm{xp}|$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Plus simple, plus court ... ???

Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :

- Φ : $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
- x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
- tol : la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
- kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

- α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$ (ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

```
Fonction α_tol ← PtFixe ( Φ, x0, tol, kmax )
   k ← 0, α_tol ← ∅
   x ← x0, fx ← Φ(x0),
   err ← |fx − x|                        ▷ ou |fx−x|/(|x|+1)
   Tantque err > tol et k ⩽ kmax faire
      k ← k + 1
      x ← fx
      fx ← Φ(x)
      err ← |fx − x|                     ▷ ou |fx−x|/(|x|+1)
   Fin Tantque
   Si err ⩽ tol alors                    ▷ Convergence
      α_tol ← x
   Fin Si
Fin Fonction
```

Plus simple, plus court ... ???

Algorithme 14 Méthode de la corde utilisant la fonction PtFixe

Données :

- f : $f : \mathbb{R} \longrightarrow \mathbb{R}$,
- a, b : deux réels tels que $f(a) \neq f(b)$,
- x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
- tol : la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
- kmax : nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

- α_{tol} : un réel tel que $|f(\alpha_{\mathrm{tol}})| \leqslant \mathrm{tol}$

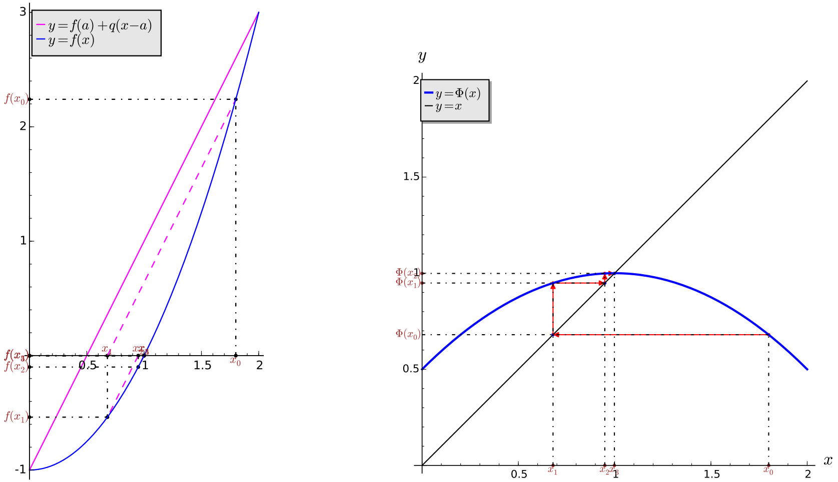
```
Fonction α_tol ← Corde ( f, a, b, x0, tol, kmax )
   q ← (b−a)/(f(b)−f(a))
   Φ ← x ↦ x − q * f(x)
   α_tol ← PtFixe(Φ, x0, tol, kmax)
Fin Fonction
```

$\alpha = 1$, racine de $f : x \mapsto x^2 - 1$

- exemple 1 : $a = 0.000$, $b = 2.000$, $x_0 = 1.800$,
- exemple 2 : $a = 0.5000$, $b = 1.900$, $x_0 = 1.800$.

	exemple 1	exemple 2
k	$\lvert x_k - \alpha\rvert$	$\lvert x_k - \alpha\rvert$
0	8.0000e-01	8.0000e-01
1	3.2000e-01	1.3333e-01
2	5.1200e-02	2.9630e-02
3	1.3107e-03	5.3041e-03
4	8.5899e-07	8.9573e-04
5	3.6893e-13	1.4962e-04
6	0.0000e+00	2.4947e-05
$\vdots$	$\vdots$	$\vdots$
15	0.0000e+00	2.4756e-12

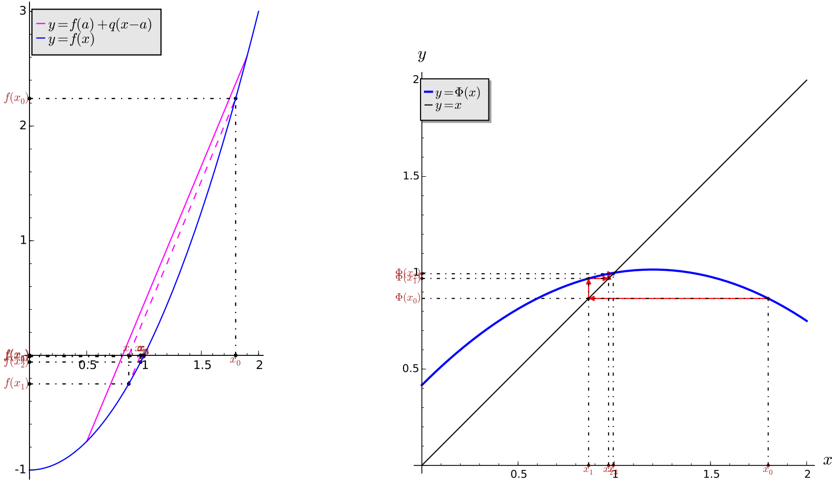
L'exemple 1 converge beaucoup plus rapidement



(a) représentation usuelle

(b) Représentation point fixe

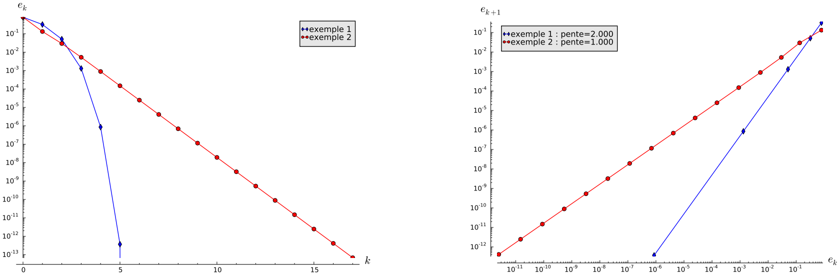
Figure : Exemple 1, méthode de la corde, $\alpha = 1$, racine de $f \ : \ x \mapsto x^2 - 1$ avec $a = 0.00$, $b = 2.00$, $x_0 = 1.80$,



(a) représentation usuelle

(b) Représentation point fixe

Figure : Exemple 2, méthode de la corde, $\alpha = 1$, racine de $f \;:\; x \mapsto x^2 - 1$ avec $a = 0.50$, $b = 1.90$, $x_0 = 1.80$,



(a) Erreurs en fonctions des itérations

(b) Représentation en échelle logarithmique de e_{k+1} en fonction de e_k. Les pentes sont calculées numériquement

Figure : Exemples 1 et 2, méthode de la corde, $\alpha = 1$, racine de $f \;:\; x \mapsto x^2 - 1$

Exemple 1 : $\dfrac{f(b)-f(a)}{b-a} = 2 \;$ et $\; f'(\alpha) = 2 \Rightarrow$ convergence ordre 2.

Exemple 2 : $\dfrac{f(b)-f(a)}{b-a} = 2.400 \neq f'(\alpha) = 2 \Rightarrow$ convergence ordre 1.

Méthode de Newton

Proposition 8.4: convergence de la méthode de Newton

Soit f une fonction de classe $\mathcal{C}^2$ sur un certain voisinage d'une racine simple α de f. Soit x_0 donné dans ce voisinage, la suite $(x_k)_{k\in\mathbb{N}}$ définie par la méthode de Newton

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad \forall k \in \mathbb{N}. \tag{18}$$

est localement convergente d'ordre 2.

Exercice 8.2

En -1700 av. J.-C., les babyloniens ne connaissaient que les nombres rationnels (fractions) et ils utilisaient le système sexagésimal (base 60). Pour approcher la valeur $\sqrt{2}$, ils utilisaient comme approximation (voir tablette YBC 7289)

$$\alpha = 1 + \frac{24}{60} + \frac{51}{60^2} + \frac{10}{60^3} = \frac{30547}{21600}$$

L'erreur commise est $|\alpha - \sqrt{2}| \approx 5.994e-7$.

Q. 1

Comment feriez-vous pour trouver ***à la main*** *une méthode permettant de trouver des nombres rationnels approchant* $\sqrt{2}$.

Q. 2

Généraliser la méthode pour trouver une approximation rationnelle de $\sqrt{a}$ *où* a *est un réel positif.*

Q. 3

Généraliser la méthode pour trouver une approximation rationnelle de $\sqrt[n]{a}$ *où* a *est un réel positif et* $n \in \mathbb{N}^*$.

Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :

Φ : $\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
x_0 : donnée initiale, $x_0 \in \mathbb{R}$,
tol : la tolérence, $\text{tol} \in \mathbb{R}^+$,
kmax : nombre maximum d'itérations, $\text{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que $|\Phi(\alpha_{\text{tol}}) - \alpha_{\text{tol}}| \leqslant \text{tol}$
(ou $\frac{|\Phi(\alpha_{\text{tol}})-\alpha_{\text{tol}}|}{|\alpha_{\text{tol}}|+1} \leqslant \text{tol}$)

Fonction $\alpha_{\text{tol}} \leftarrow \text{PtFixe}\ (\ \Phi, x_0, \text{tol}, \text{kmax}\ )$
  $k \leftarrow 0$, $\alpha_{\text{tol}} \leftarrow \varnothing$
  $\text{x} \leftarrow x_0$, $\text{fx} \leftarrow \Phi(x_0)$,
  $\text{err} \leftarrow |\text{fx} - \text{x}|$ ▷ ou $\frac{|\text{fx}-\text{x}|}{|\text{x}|+1}$
  Tantque $\text{err} > \text{tol}$ et $k \leqslant \text{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\text{x} \leftarrow \text{fx}$
    $\text{fx} \leftarrow \Phi(\text{x})$
    $\text{err} \leftarrow |\text{fx} - \text{x}|$ ▷ ou $\frac{|\text{fx}-\text{x}|}{|\text{x}|+1}$
  Fin Tantque
  Si $\text{err} \leqslant \text{tol}$ **alors** ▷ Convergence
    $\alpha_{\text{tol}} \leftarrow \text{x}$
  Fin Si
Fin Fonction

Méthode de Newton :

$$\Phi(x) = x - \frac{f(x)}{f'(x)}$$

Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :

Φ	:	$\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$ (ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PTFIXE ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \emptyset$
  $\mathrm{x} \leftarrow x_0$, $\mathrm{fx} \leftarrow \Phi(x_0)$,
  $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{x} \leftarrow \mathrm{fx}$
    $\mathrm{fx} \leftarrow \Phi(\mathrm{x})$
    $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Algorithme 15 Méthode de Newton

Données :

f	:	$f : \mathbb{R} \longrightarrow \mathbb{R}$,
df	:	la dérivée de f,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ NEWTON ($f, \mathrm{df}, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \emptyset$
  $\mathrm{x} \leftarrow x_0$,
  $\mathrm{err} \leftarrow \mathrm{tol} + 1$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{xp} \leftarrow \mathrm{x}$
    $\mathrm{x} \leftarrow \mathrm{xp} - f(\mathrm{xp})/\mathrm{df}(\mathrm{xp})$ ▷ $\mathrm{df}(\mathrm{xp}) \neq 0$
    $\mathrm{err} \leftarrow |\mathrm{x} - \mathrm{xp}|$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Plus simple, plus court ???

Méthode de point fixe : version **Tantque** avec critères d'arrêt

Données :

Φ	:	$\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$
(ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
    $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
    $\mathrm{x} \leftarrow x_0$, $\mathrm{fx} \leftarrow \Phi(x_0)$,
    $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
    Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
        $k \leftarrow k + 1$
        $\mathrm{x} \leftarrow \mathrm{fx}$
        $\mathrm{fx} \leftarrow \Phi(\mathrm{x})$
        $\mathrm{err} \leftarrow |\mathrm{fx} - \mathrm{x}|$ ▷ ou $\frac{|\mathrm{fx}-\mathrm{x}|}{|\mathrm{x}|+1}$
    Fin Tantque
    Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
        $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
    Fin Si
Fin Fonction

Algorithme 16 Méthode de Newton scalaire

Données :

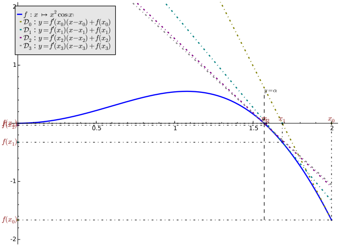
f	:	$f : \mathbb{R} \longrightarrow \mathbb{R}$,
df	:	la dérivée de f,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

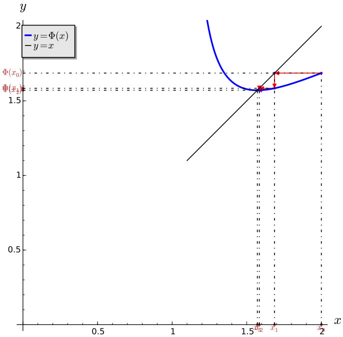
α_{tol} : un réel tel que

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ Newton ($f, \mathrm{df}, x_0, \mathrm{tol}, \mathrm{kmax}$)
    $\Phi \leftarrow x \mapsto x - f(x)/\mathrm{df}(x)$
    $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe$(\Phi, x_0, \mathrm{tol}, \mathrm{kmax})$
Fin Fonction

Plus simple, plus court ... ???

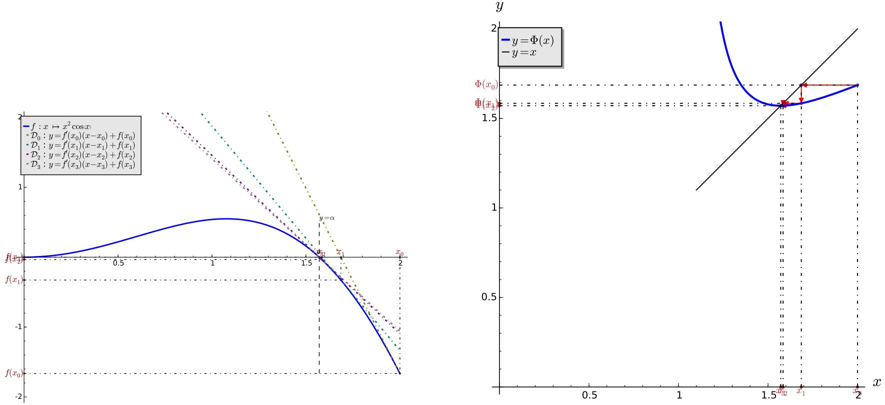


(a) représentation usuelle



(b) Représentation point fixe,
$\Phi : x \mapsto \frac{x^2 \cos(x)}{x^2 \sin(x) - 2\,x \cos(x)} + x$

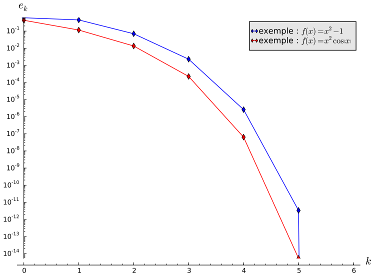
Figure : Exemple 2, méthode de Newton, $\alpha = \frac{1}{2}\pi$, racine de $f : x \mapsto x^2 \cos(x)$ avec $x_0 = 0.40$,



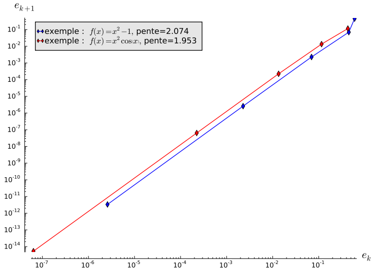
(a) représentation usuelle

(b) Représentation point fixe, $\Phi : x \mapsto \frac{x^2 \cos(x)}{x^2 \sin(x) - 2\,x \cos(x)} + x$

Figure : Exemple 2, méthode de Newton, $\alpha = \frac{1}{2}\pi$, racine de $f : x \mapsto x^2 \cos(x)$ avec $x_0 = 0.40$,



(a) Représentation de la convergence, e_k en fonction de k



(b) Représentation de l'ordre de convergence en échelle logarithmique, e_{k+1} en fonction de e_k . Ordre théorique 2

Figure : Méthode de Newton, convergence et ordre

Alternative à la méthode de Newton lorsque l'on ne connait pas la dérivée de la fonction f :

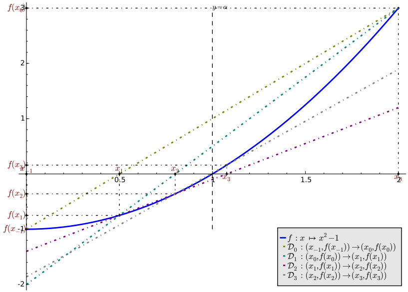
$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}.$$

Proposition 8.5: Convergence méthode de la sécante (Admis)

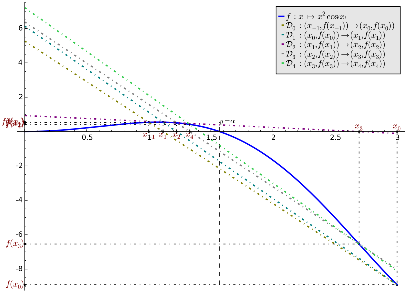
Soit f une fonction de classe $\mathcal{C}^2$ sur un certain voisinage d'une racine simple α de f. Soient x_{-1} et x_0 donnés dans ce voisinage tels que $f(x_{-1}) \neq f(x_0)$, la suite $(x_k)_{k\in\mathbb{N}}$ définie par la méthode de la sécante

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k), \quad \forall k \in \mathbb{N}. \tag{19}$$

est localement convergente d'ordre $\frac{1+\sqrt{5}}{2} \approx 1.618$.

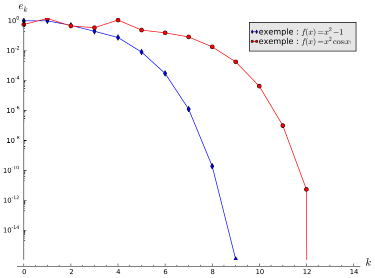


(a) $f(x) = x^2 - 1$, $x_{-1} = 0.000$ et $x_0 = 2.000$

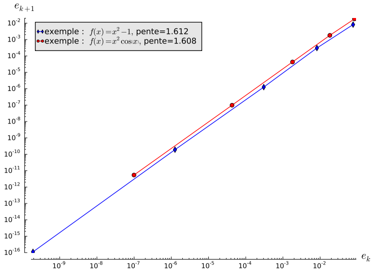


(b) $f(x) = x^2 \cos(x)$, $x_{-1} = 1.000$ et $x_0 = 3.000$

Figure : Méthode de la sécante



(a) Représentation de la convergence, e_k en fonction de k



(b) Représentation de l'ordre de convergence en échelle logarithmique, e_{k+1} en fonction de e_k . Ordre théorique $\frac{1+\sqrt{5}}{2} \approx 1.618$

Figure : Méthode de la sécante, convergence et ordre

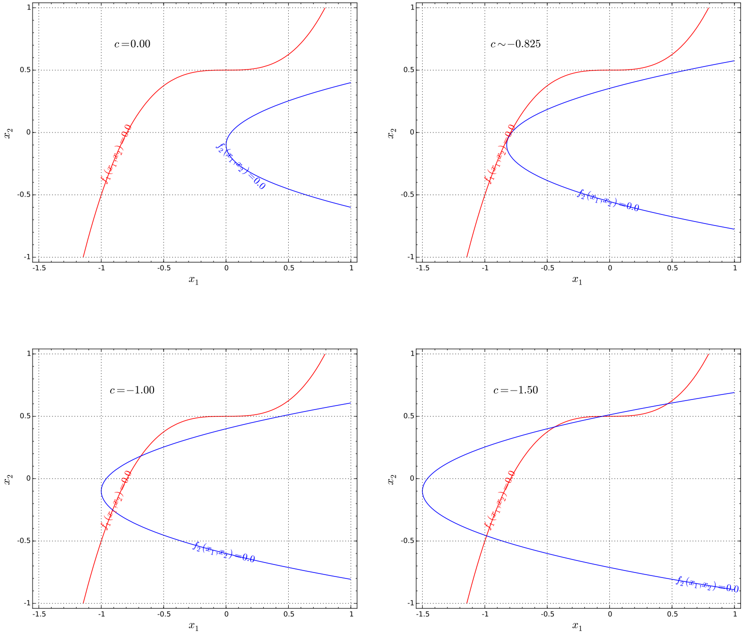
Plan

5 Recherche des zéros d'une fonction

6 Résolution de systèmes non linéaires
- Point fixe
- Méthode de Newton
- Exemples

Soit $c \in \mathbb{R}$ donné.

$$\begin{cases} f_1(x_1, x_2) = -x_1^3 + x_2 - \frac{1}{2} & = 0 \\ f_2(x_1, x_2) = \frac{1}{25}\left(10\,x_2 + 1\right)^2 + c - x_1 & = 0. \end{cases} \tag{20}$$


$c = 0.00$
$c \sim -0.825$
$c = -1.00$
$c = -1.50$
$f_1(x_1, x_2) = 0.0$
$f_2(x_1, x_2) = 0.0$
x_1
x_2

Soient $U \subset \mathbb{R}^N$ un ouvert et $\boldsymbol{f} \in \mathcal{C}^0(U; \mathbb{R}^N)$

Trouver $\boldsymbol{\alpha} \in U \subset \mathbb{R}^N$ tel que

$$\boldsymbol{f}(\boldsymbol{\alpha}) = 0 \iff \begin{cases} \boldsymbol{f}_1(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N) & = & 0 \\ \boldsymbol{f}_2(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N) & = & 0 \\ \vdots & & \\ \boldsymbol{f}_N(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N) & = & 0 \end{cases}$$

On pose, par ex., $\boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{f}(\boldsymbol{x})$, : $\boldsymbol{f}(\boldsymbol{x}) = 0 \Leftrightarrow \boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{x}$ Point fixe

Trouver $\boldsymbol{\alpha} \in U \subset \mathbb{R}^N$ tel que

$$\boldsymbol{\Phi}(\boldsymbol{\alpha}) = \boldsymbol{\alpha} \iff \begin{cases} \boldsymbol{\Phi}_1(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N) & = & \boldsymbol{\alpha}_1 \\ \boldsymbol{\Phi}_2(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N) & = & \boldsymbol{\alpha}_2 \\ \vdots & & \\ \boldsymbol{\Phi}_N(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N) & = & \boldsymbol{\alpha}_N \end{cases}$$

Théorème 9.1: Point fixe de Banach ★★★★★

Soit $\mathcal{B}$ un espace de Banach et $U \subset \mathcal{B}$ un sous-ensemble fermé. On suppose que $\boldsymbol{\Phi} : U \longrightarrow U$ est une application strictement contractante, i.e.

$$\exists L \in]0,1[, \quad \left\|\boldsymbol{\Phi}(\boldsymbol{x}) - \boldsymbol{\Phi}(\boldsymbol{y})\right\| \leqslant L \left\|\boldsymbol{x} - \boldsymbol{y}\right\|, \quad \forall (\boldsymbol{x}, \boldsymbol{y}) \in U \times U. \tag{21}$$

Alors

1. $\boldsymbol{\Phi}$ admet un unique point fixe $\boldsymbol{\alpha} \in U$ (i.e. unique solution de $\boldsymbol{x} = \boldsymbol{\Phi}(\boldsymbol{x})$).
2. La suite des itérés $\boldsymbol{x}^{[k+1]} = \boldsymbol{\Phi}(\boldsymbol{x}^{[k]})$ converge vers $\boldsymbol{\alpha}$ pour toute valeur initiale $\boldsymbol{x}^{[0]} \in U$.
3. Pour tout $k \in \mathbb{N}$,

$$\left\|\boldsymbol{\alpha} - \boldsymbol{x}^{[k]}\right\| \leqslant \frac{L^{k-l}}{1-L} \left\|\boldsymbol{x}^{[l+1]} - \boldsymbol{x}^{[l]}\right\|, \quad 0 \leqslant l \leqslant k \tag{22}$$

$\boldsymbol{f} : \mathbb{R}^N \longrightarrow \mathbb{R}^N$ une fonction suffisament régulière. On défini la **matrice Jacobienne de** $\boldsymbol{f}$, notée $\mathbb{J}_{\boldsymbol{f}}$, par

$$\mathbb{J}_{\boldsymbol{f}} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_N} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_N}{\partial x_1} & \frac{\partial f_N}{\partial x_2} & \cdots & \frac{\partial f_N}{\partial x_N} \end{pmatrix}$$

On a alors $\forall \boldsymbol{h} \in \mathbb{R}^N$ à l'ordre 1

$$\boldsymbol{f}(\boldsymbol{x} + \boldsymbol{h}) \approx \boldsymbol{f}(\boldsymbol{x}) + \mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}).\boldsymbol{h}. \tag{23}$$

On a $\forall \boldsymbol{h} \in \mathbb{R}^N$ à l'ordre 1

$$\boldsymbol{f}(\boldsymbol{x}+\boldsymbol{h}) \approx \boldsymbol{f}(\boldsymbol{x}) + \mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}).\boldsymbol{h}. \tag{24}$$

trouver $\boldsymbol{\alpha}$ tel que $\boldsymbol{f}(\boldsymbol{\alpha}) = 0$.

Si $\boldsymbol{x}^{[k]}$ est proche de $\boldsymbol{\alpha}$, alors avec $\boldsymbol{x} = \boldsymbol{x}^{[k]}$ et $\boldsymbol{\alpha} = \boldsymbol{x}^{[k]} + \boldsymbol{h}$

$$\boldsymbol{f}(\boldsymbol{\alpha}) \approx \boldsymbol{f}(\boldsymbol{x}^{[k]}) + \mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]}).\boldsymbol{h}$$

On résoud le système linéarisé

$$\boldsymbol{f}(\boldsymbol{x}^{[k]}) + \mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]}).\tilde{\boldsymbol{h}} = 0 \;\Leftrightarrow\; \mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]}).\tilde{\boldsymbol{h}} = -\boldsymbol{f}(\boldsymbol{x}^{[k]}).$$

On pose $\Phi(\boldsymbol{x}) = \boldsymbol{x} - ((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}))^{-1}\boldsymbol{f}(\boldsymbol{x})$. la méthode de Newton s'écrit alors

$$\boldsymbol{x}^{[k+1]} = \Phi(\boldsymbol{x}^{[k]}) = \boldsymbol{x}^{[k]} - \left((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]})\right)^{-1}\boldsymbol{f}(\boldsymbol{x}^{[k]}) \tag{25}$$

Théorème 9.2: (Admis)

Soit $\boldsymbol{f} \in \mathcal{C}^3(\mathbb{R}^N;\mathbb{R}^N)$. On suppose que la matrice Jacobienne appliquée en $\boldsymbol{x}$, $\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x})$ est inversible dans un voisinage de $\boldsymbol{\alpha}$, avec $\boldsymbol{f}(\boldsymbol{\alpha}) = 0$. Alors pour tout $\boldsymbol{x}^{[0]}$ suffisament proche de $\boldsymbol{\alpha}$ la suite définie par

$$\boldsymbol{x}^{[k+1]} = \boldsymbol{x}^{[k]} - \left((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]})\right)^{-1} \boldsymbol{f}(\boldsymbol{x}^{[k]})$$

converge vers $\boldsymbol{\alpha}$ et la convergence est d'ordre 2.

Comment fait-on pour calculer $-\left((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]})\right)^{-1} \boldsymbol{f}(\boldsymbol{x}^{[k]})$?

Théorème 9.3: (Admis)

Soit $\boldsymbol{f} \in \mathcal{C}^3(\mathbb{R}^N;\mathbb{R}^N)$. On suppose que la matrice Jacobienne appliquée en $\boldsymbol{x}$, $\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x})$ est inversible dans un voisinage de $\boldsymbol{\alpha}$, avec $\boldsymbol{f}(\boldsymbol{\alpha}) = 0$. Alors pour tout $\boldsymbol{x}^{[0]}$ suffisament proche de $\boldsymbol{\alpha}$ la suite définie par

$$\boldsymbol{x}^{[k+1]} = \boldsymbol{x}^{[k]} - \left((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]})\right)^{-1} \boldsymbol{f}(\boldsymbol{x}^{[k]})$$

converge vers $\boldsymbol{\alpha}$ et la convergence est d'ordre 2.

Comment fait-on pour calculer $-\left((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]})\right)^{-1} \boldsymbol{f}(\boldsymbol{x}^{[k]})$?

On résoud le système linéaire

$$\left((\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x}^{[k]})\right) \boldsymbol{h} = -\boldsymbol{f}(\boldsymbol{x}^{[k]})$$

Remarque : Si l'on ne connait pas explicitement la Jacobienne de f, il est possible de calculer une approximation de celle-ci en utilisant des formules de dérivation numérique.

Méthode de Newton scalaire

Données :

f	:	$f : \mathbb{R} \longrightarrow \mathbb{R}$,
df	:	la dérivée de f,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ Newton ($f, \mathrm{df}, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
  $\mathrm{x} \leftarrow x_0$,
  $\mathrm{err} \leftarrow \mathrm{tol} + 1$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{xp} \leftarrow \mathrm{x}$
    $\mathrm{x} \leftarrow \mathrm{xp} - f(\mathrm{xp})/\mathrm{df}(\mathrm{xp})$ ▷ $\mathrm{x} \leftarrow \Phi(\mathrm{xp})$
    $\mathrm{err} \leftarrow |\mathrm{x} - \mathrm{xp}|$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors**
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Méthode de Newton *vectorielle* :

$$\boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{x} - \left(\left(\mathbb{J}_{\boldsymbol{f}}(\boldsymbol{x})\right)^{-1} \boldsymbol{f}(\boldsymbol{x})\right)$$

Méthode de Newton scalaire

Données :

f	:	$f : \mathbb{R} \longrightarrow \mathbb{R}$,
df	:	la dérivée de f,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol}	:	un réel tel que

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ Newton ($f, \mathrm{df}, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
  $\mathrm{x} \leftarrow x_0$,
  $\mathrm{err} \leftarrow \mathrm{tol} + 1$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{xp} \leftarrow \mathrm{x}$
    $\mathrm{x} \leftarrow \mathrm{xp} - f(\mathrm{xp})/\mathrm{df}(\mathrm{xp})$ ▷ $\mathrm{x} \leftarrow \Phi(\mathrm{xp})$
    $\mathrm{err} \leftarrow |\mathrm{x} - \mathrm{xp}|$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors**
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Algorithme 17 Méthode de Newton vectorielle

Données :

$\boldsymbol{f}$	:	$\boldsymbol{f} : \mathbb{R}^N \longrightarrow \mathbb{R}^N$,
Jf	:	la matrice Jacobienne de $\boldsymbol{f}$,
$\boldsymbol{x0}$	:	donnée initiale, $\boldsymbol{x0} \in \mathbb{R}^N$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

$\boldsymbol{\alpha}_{\mathrm{tol}}$	:	un élément de $\mathbb{R}^N$ proche de $\boldsymbol{\alpha}$.

Fonction $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow$ Newton ($\boldsymbol{f}, \mathrm{Jf}, \boldsymbol{x0}, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
  $\boldsymbol{x} \leftarrow \boldsymbol{x0}$,
  $\mathrm{err} \leftarrow \mathrm{tol} + 1$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\boldsymbol{xp} \leftarrow \boldsymbol{x}$
    $\boldsymbol{h} \leftarrow$ Solve$(\mathrm{Jf}(\boldsymbol{xp}), -\boldsymbol{f}(\boldsymbol{xp}))$
    $\boldsymbol{x} \leftarrow \boldsymbol{xp} + \boldsymbol{h}$
    $\mathrm{err} \leftarrow$ Norm$(\boldsymbol{x} - \boldsymbol{xp})$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow \boldsymbol{x}$
  Fin Si
Fin Fonction

Rappel : Point fixe scalaire. Et le cas vectoriel ... ???

Méthode de point fixe scalaire

Données :

Φ	:	$\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que $|\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}| \leqslant \mathrm{tol}$ (ou $\frac{|\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}|}{|\alpha_{\mathrm{tol}}|+1} \leqslant \mathrm{tol}$)

```
Fonction α_tol ← PtFixe ( Φ, x0, tol, kmax )
    k ← 0, α_tol ← ∅
    x ← x0, fx ← Φ(x0),
    err ← |fx − x|                         ▷ ou |fx−x|/(|x|+1)
    Tantque err > tol et k ⩽ kmax faire
        k ← k + 1
        x ← fx
        fx ← Φ(x)
        err ← |fx − x|                     ▷ ou |fx−x|/(|x|+1)
    Fin Tantque
    Si err ⩽ tol alors                     ▷ Convergence
        α_tol ← x
    Fin Si
Fin Fonction
```

Algorithme 18 Méthode de Newton scalaire

Données :

f	:	$f : \mathbb{R} \longrightarrow \mathbb{R}$,
df	:	la dérivée de f,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol} : un réel tel que

```
Fonction α_tol ← Newton ( f, df, x0, tol, kmax )
    Φ ← x ↦ x − f(x)/df(x)
    α_tol ← PtFixe(Φ, x0, tol, kmax)
Fin Fonction
```

Rappel : Point fixe scalaire. Et le cas vectoriel ... ???

Méthode de point fixe scalaire

Données :

Φ	:	$\Phi : \mathbb{R} \longrightarrow \mathbb{R}$,
x_0	:	donnée initiale, $x_0 \in \mathbb{R}$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

α_{tol}	:	un réel tel que $\lvert\Phi(\alpha_{\mathrm{tol}}) - \alpha_{\mathrm{tol}}\rvert \leqslant \mathrm{tol}$ (ou $\frac{\lvert\Phi(\alpha_{\mathrm{tol}})-\alpha_{\mathrm{tol}}\rvert}{\lvert\alpha_{\mathrm{tol}}\rvert+1} \leqslant \mathrm{tol}$)

Fonction $\alpha_{\mathrm{tol}} \leftarrow$ PtFixe ($\Phi, x_0, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
  $\mathrm{x} \leftarrow x_0$, $\mathrm{fx} \leftarrow \Phi(x_0)$,
  $\mathrm{err} \leftarrow \lvert\mathrm{fx} - \mathrm{x}\rvert$ ▷ ou $\frac{\lvert\mathrm{fx}-\mathrm{x}\rvert}{\lvert\mathrm{x}\rvert+1}$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\mathrm{x} \leftarrow \mathrm{fx}$
    $\mathrm{fx} \leftarrow \Phi(\mathrm{x})$
    $\mathrm{err} \leftarrow \lvert\mathrm{fx} - \mathrm{x}\rvert$ ▷ ou $\frac{\lvert\mathrm{fx}-\mathrm{x}\rvert}{\lvert\mathrm{x}\rvert+1}$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors** ▷ Convergence
    $\alpha_{\mathrm{tol}} \leftarrow \mathrm{x}$
  Fin Si
Fin Fonction

Méthode de point fixe vectorielle

Données :

$\boldsymbol{\Phi}$	:	$\boldsymbol{\Phi} : \mathbb{K}^N \longrightarrow \mathbb{K}^N$,
$\boldsymbol{x0}$	:	donnée initiale, $\boldsymbol{x0} \in \mathbb{K}^N$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

$\boldsymbol{\alpha}_{\mathrm{tol}}$	:	un réel tel que $\lVert\boldsymbol{\Phi}(\boldsymbol{\alpha}_{\mathrm{tol}}) - \boldsymbol{\alpha}_{\mathrm{tol}}\rVert \leqslant \mathrm{tol}$

Fonction $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow$ PtFixe ($\boldsymbol{\Phi}, \boldsymbol{x0}, \mathrm{tol}, \mathrm{kmax}$)
  $k \leftarrow 0$, $\alpha_{\mathrm{tol}} \leftarrow \varnothing$
  $\boldsymbol{x} \leftarrow \boldsymbol{x0}$, $\boldsymbol{fx} \leftarrow \boldsymbol{\Phi}(\boldsymbol{x0})$,
  $\mathrm{err} \leftarrow \lVert\boldsymbol{fx} - \boldsymbol{x}\rVert$
  Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
    $k \leftarrow k + 1$
    $\boldsymbol{x} \leftarrow \boldsymbol{fx}$
    $\boldsymbol{fx} \leftarrow \boldsymbol{\Phi}(\boldsymbol{x})$
    $\mathrm{err} \leftarrow \lVert\boldsymbol{fx} - \boldsymbol{x}\rVert$
  Fin Tantque
  Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors**
    $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow \boldsymbol{x}$
  Fin Si
Fin Fonction

Rappel : Point fixe scalaire. Et le cas vectoriel ... ???

Méthode de Newton vectorielle

Données :

$\boldsymbol{f}$	:	$\boldsymbol{f} : \mathbb{R}^N \longrightarrow \mathbb{R}^N$,
Jf	:	la matrice Jacobienne de f,
$\boldsymbol{x0}$	:	donnée initiale, $\boldsymbol{x0} \in \mathbb{R}^N$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

$\boldsymbol{\alpha}_{\mathrm{tol}}$	:	un élément de $\mathbb{R}^N$ proche de $\boldsymbol{\alpha}$.

Fonction $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow$ NEWTON ($\boldsymbol{f}, \mathrm{Jf}, \boldsymbol{x0}, \mathrm{tol}, \mathrm{kmax}$)
    $\boldsymbol{\Phi} \leftarrow \boldsymbol{x} \mapsto \boldsymbol{x} - \mathrm{SOLVE}(\mathrm{Jf}(\boldsymbol{x}), \boldsymbol{f}(\boldsymbol{x}))$
    $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow \mathrm{PTFIXE}(\boldsymbol{\Phi}, \boldsymbol{x0}, \mathrm{tol}, \mathrm{kmax})$
Fin Fonction

Méthode de point fixe vectorielle

Données :

Φ	:	$\boldsymbol{\Phi} : \mathbb{K}^N \longrightarrow \mathbb{K}^N$,
$\boldsymbol{x0}$	:	donnée initiale, $\boldsymbol{x0} \in \mathbb{K}^N$,
tol	:	la tolérence, $\mathrm{tol} \in \mathbb{R}^+$,
kmax	:	nombre maximum d'itérations, $\mathrm{kmax} \in \mathbb{N}^*$

Résultat :

$\boldsymbol{\alpha}_{\mathrm{tol}}$	:	un réel tel que $\lVert \boldsymbol{\Phi}(\boldsymbol{\alpha}_{\mathrm{tol}}) - \boldsymbol{\alpha}_{\mathrm{tol}} \rVert \leqslant \mathrm{tol}$

Fonction $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow$ PTFIXE ($\boldsymbol{\Phi}, \boldsymbol{x0}, \mathrm{tol}, \mathrm{kmax}$)
    $k \leftarrow 0,\ \alpha_{\mathrm{tol}} \leftarrow \varnothing$
    $\boldsymbol{x} \leftarrow \boldsymbol{x0},\ \boldsymbol{fx} \leftarrow \boldsymbol{\Phi}(\boldsymbol{x0}),$
    $\mathrm{err} \leftarrow \lVert \boldsymbol{fx} - \boldsymbol{x} \rVert$
    Tantque $\mathrm{err} > \mathrm{tol}$ et $k \leqslant \mathrm{kmax}$ **faire**
        $k \leftarrow k + 1$
        $\boldsymbol{x} \leftarrow \boldsymbol{fx}$
        $\boldsymbol{fx} \leftarrow \boldsymbol{\Phi}(\boldsymbol{x})$
        $\mathrm{err} \leftarrow \lVert \boldsymbol{fx} - \boldsymbol{x} \rVert$
    Fin Tantque
    Si $\mathrm{err} \leqslant \mathrm{tol}$ **alors**
        $\boldsymbol{\alpha}_{\mathrm{tol}} \leftarrow \boldsymbol{x}$
    Fin Si
Fin Fonction

Soit $c = 3/2$

$$\begin{cases} f_1(x_1, x_2) = -x_1^3 + x_2 - \frac{1}{2} \\ f_2(x_1, x_2) = \frac{1}{25}\,(10\,x_2 + 1)^2 + c \end{cases}$$

Conclusion?

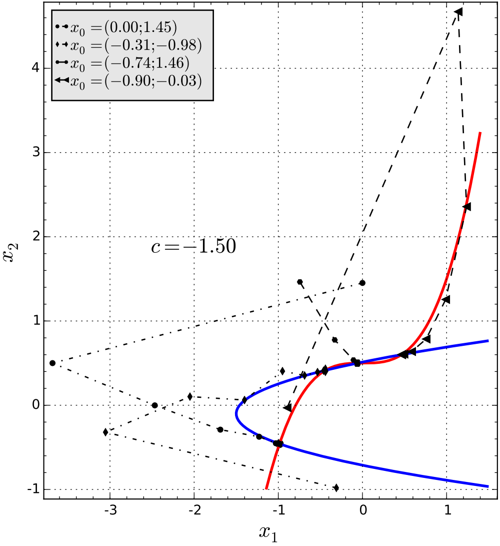


Figure : Représentation de 4 suites de Newton

Soit $c = 3/2$

$$\begin{cases} f_1(x_1, x_2) = -x_1^3 + x_2 - \frac{1}{2} \\ f_2(x_1, x_2) = \frac{1}{25}\,(10\,x_2 + 1)^2 + c \end{cases}$$

Conclusion?

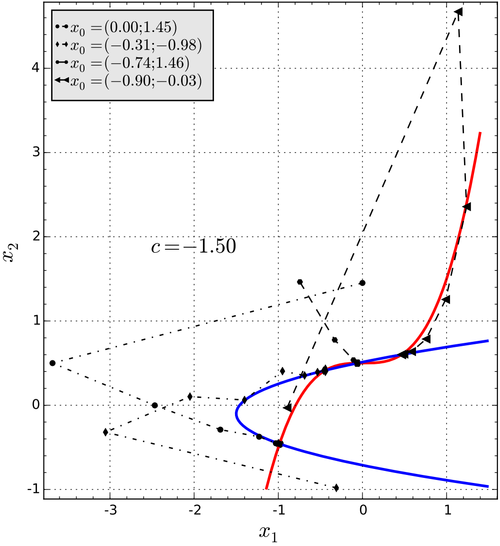
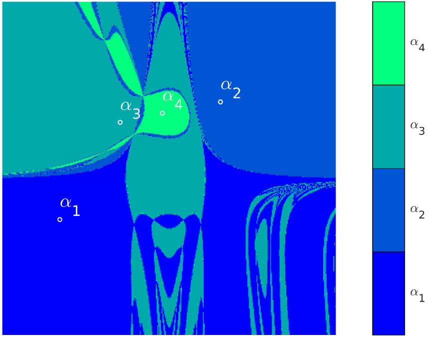
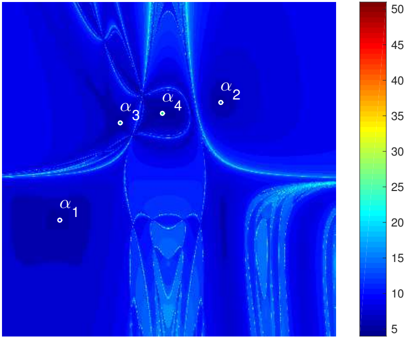


Figure : Représentation de 4 suites de Newton

Très difficile, si l'on n'est pas suffisament proche d'un point fixe, de prédire vers lequel on converge.



(a) Bassin d'attraction des racines



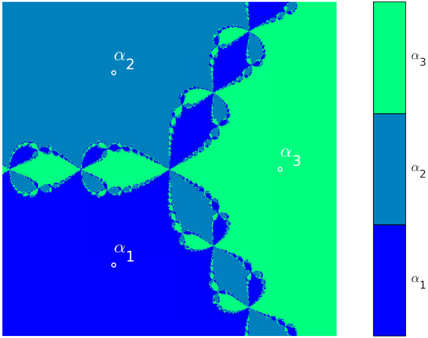
(b) Nombre d'itérations de convergence

Figure : Méthode de Newton

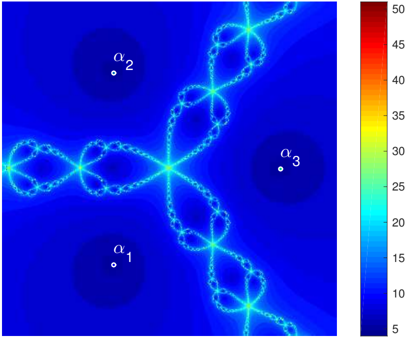
Exemple complexe : $z^3 - 1 = 0$, fractale de Newton

on peut poser $z = x + \imath y$, et le système équivalent devient

$$\begin{cases} f_1(x,y) = x^3 - 3xy^2 - 1 & = 0 \\ f_2(x,y) = 3x^2 y - x^3 & = 0. \end{cases}$$

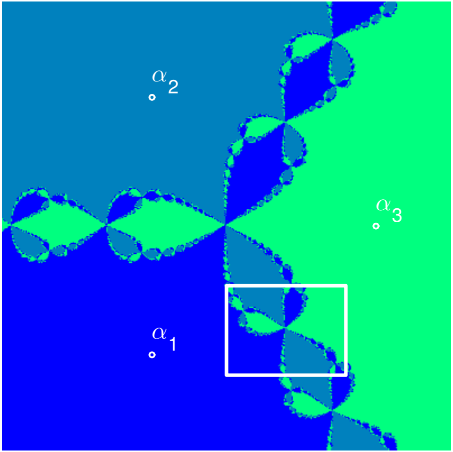


(a) Bassin d'attraction des racines



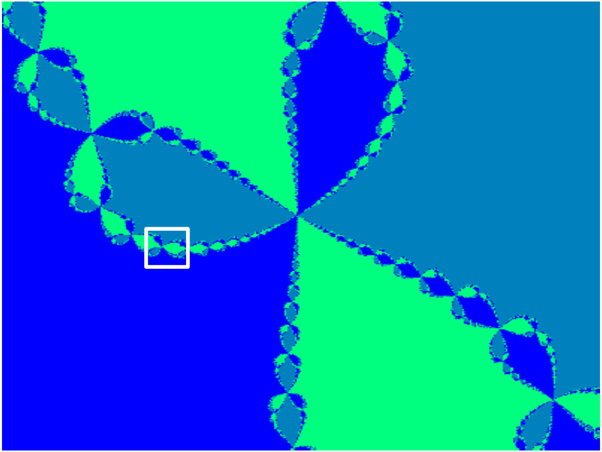
(b) Nombre d'itérations de convergence

Exemple complexe : $z^3 - 1 = 0$, fractale de Newton



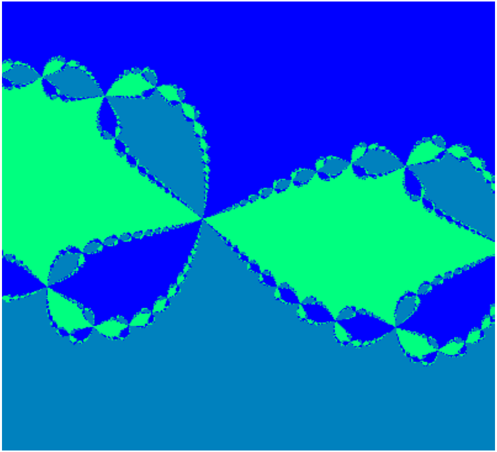
Méthode de Newton, zoom 1 sur les bassins d'attraction

Exemple complexe : $z^3 - 1 = 0$, fractale de Newton



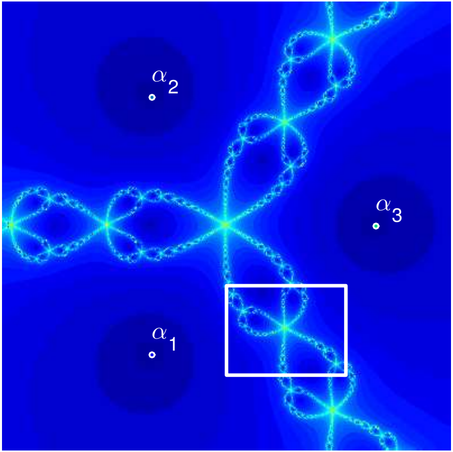
Méthode de Newton, zoom 2 sur les bassins d'attraction

Exemple complexe : $z^3 - 1 = 0$, fractale de Newton



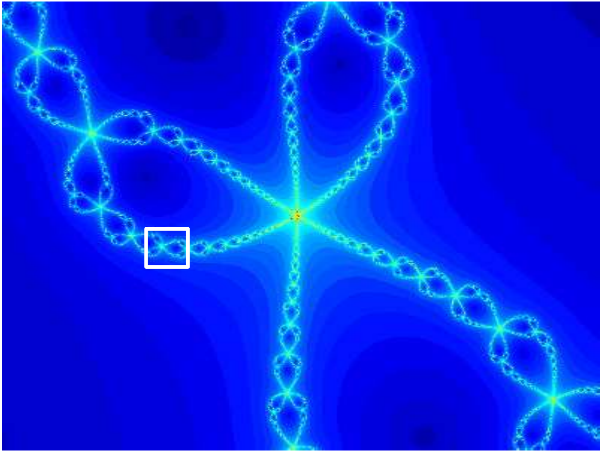
Méthode de Newton, zoom 3 sur les bassins d'attraction

Exemple complexe : $z^3 - 1 = 0$, fractale de Newton



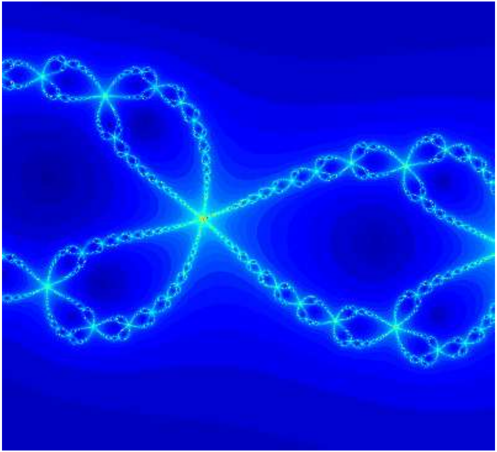
Méthode de Newton, zoom 1 sur les nombres d'itérations

Exemple complexe : $z^3 - 1 = 0$, fractale de Newton



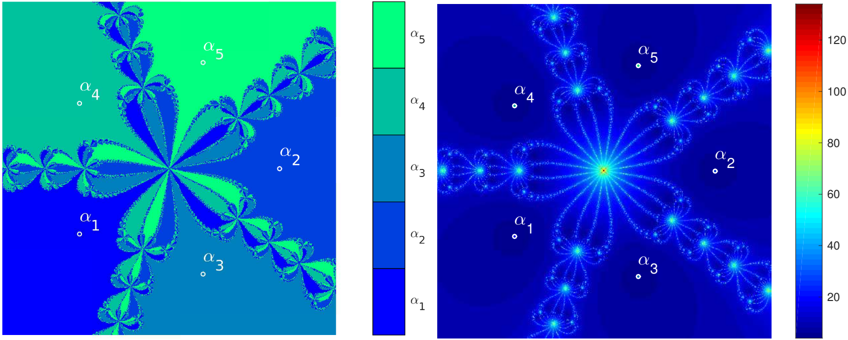
Méthode de Newton, zoom 2 sur les nombres d'itérations

Exemple complexe : $z^3 - 1 = 0$, fractale de Newton



Méthode de Newton, zoom 3 sur les nombres d'itérations

Exemple complexe : $z^5 - 1 = 0$, fractale de Newton

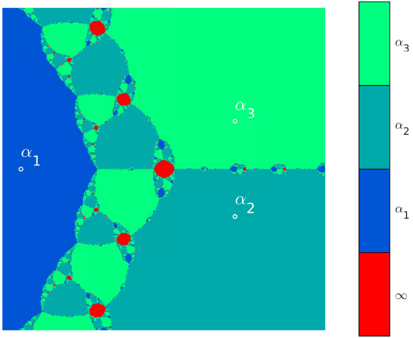


(a) Bassin d'attraction des racines

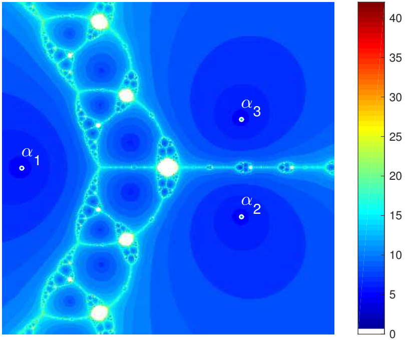
(b) Nombre d'itérations de convergence

$$[-1.5, 1.5] \times [-1.5, 1.5]$$

Exemple complexe : $z^3 - 2z + 2 = 0$, fractale de Newton



(a) Bassin d'attraction des racines. En rouge zône de divergence



(b) Nombre d'itérations de convergence. En blanc zône de divergence

$$[-2, 2] \times [-2, 2]$$

Chapitre IV

Résolution de systèmes linéaires

Chapitre V

Polynômes d'interpolation de Lagrange

Chapitre VI

Dérivation numérique

Chapitre VII

Intégration numérique