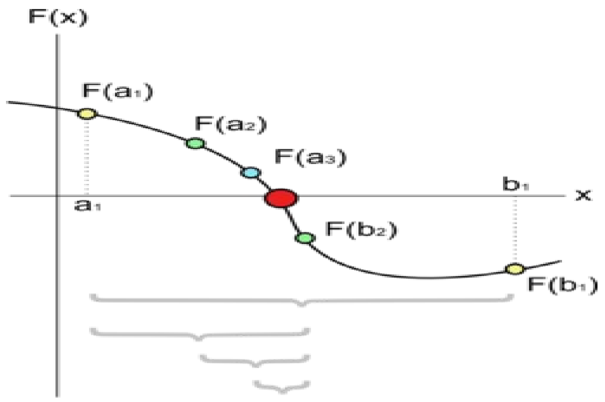
TP 1 SCILAB : Résolution d'équation

En annexe sont présentés quelques rappels sur les théorèmes du point fixe et la méthode de Newton.
On utilisera par défaut la fonction $f(x)=x^2-2$ sur [1,2].

Activité 1 : Méthode par dichotomie

On se place dans le cas d'une fonction f continue sur un intervalle $[a,b]$ de $\mathbb{R}$ sur lequel f ne s'annule qu'une fois en changeant de signe.

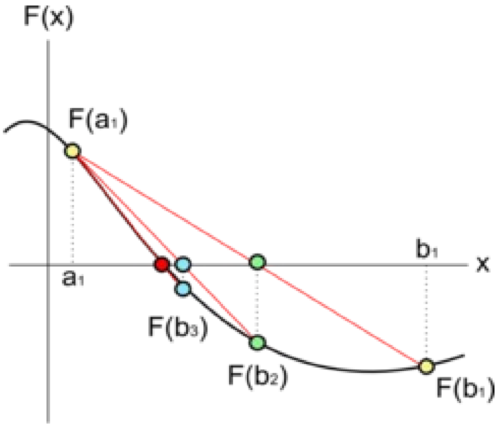
1. Définir une suite x_n convergeant vers $\bar{x}$ avec $f(\bar{x})=0$, l'unique solution de l'équation $f(x)=0$ sur $[a,b]$.
2. Déterminer l'ordre de cette méthode.
3. Définir une condition d'existence et un critère d'arrêt pour le programme.
4. Ecrire une fonction scilab [*x0,n*]=dicho(*f,a,b,e*) permettant le calcul de $\bar{x}$ avec une précision donnée *e* ayant en arguments de sortie la valeur approchée *x0* de $\bar{x}$ et le nombre d'itérations n.



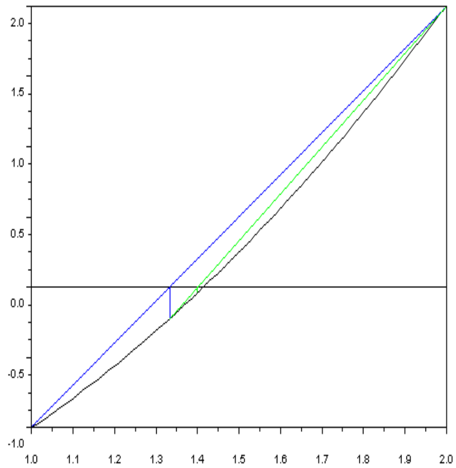
Activité 2 : Méthode de la fausse position (200 ans av J.-C.)

On reprend les mêmes hypothèses que pour la méthode par dichotomie.

1. Définir une suite x_n convergeant vers $\bar{x}$ avec $f(\bar{x})=0$ l'unique solution de l'équation $f(x)=0$ sur $[a,b]$.
2. Définir une condition d'existence et un critère d'arrêt pour le programme.
3. Ecrire une fonction scilab [*x0,n*]=regulafalsi(*f,a,b,e*) permettant le calcul de $\bar{x}$ avec une précision donnée *e* ayant en arguments de sortie la valeur approchée x de $\bar{x}$ et le nombre d'itérations n.
4. Utiliser la fonction scilab graphe1(f,a,b) décrivant la construction graphique pas à pas des termes x_n.



```
function graphe1(f,a,b)
x=linspace(a,b,50)
y=feval(x,f)
clf
plot2d(x,y)
plot2d(x,0*x,1)
n=0
i=1
while n==0
i=i+1
plot2d([a b],[f(a) f(b)],i)
c=a-(f(a)*(b-a))/(f(b)-f(a))
plot2d([c c],[0 f(c)],i)
      if ((f(a)*f(x))<0) then
                                    a=a
                                    b=c
                               else
                                    a=c
                                    b=b
      end
n=input("Pour continuer tapez 0
sinon 1")
end
endfunction
```



Activité 3 : Méthode de Newton (1643-1727)

Partie A : Etude d'une équation historique (Newton, vers 1670) : $x^3 - 2x - 5 = 0$

a. Etudier les solutions de l'équation

b. On propose une racine approximative $u_0 = 2$. Quel est le résultat ?

c. $u_0=2$ semble proche de la solution exacte, notée α. Soit e l'écart. On pose $\alpha= 2 + e$.

Remplacer x par u_0+e dans l'équation, puis négliger les termes e^2 *et* e^3.

En déduire une nouvelle approximation de α, que nous noterons u_1:

d. Reprendre le **c.** mais en remplaçant x par u_1+e dans l'équation.

e. En déduire une relation de récurrence entre u_n et u_{n+1}.

f. Proposer une méthode d'approximations successives de α faisant intervenir $f(u_n)$ et $f'(u_n)$

g. Pourrait-on envisager une autre équation du type g(x)=x

Partie B : Programmation

1. Définir une suite x_n convergeant vers $\bar{x}$ avec $f(\bar{x})=0$ l'unique solution de l'équation $f(x)=0$ sur $[a,b]$.
2. Appliquer cette méthode à l'exemple $f(x)=x^2-2$ et retrouver la méthode de Héron (Ier s. ap. J.-C.)
3. Ecrire une fonction scilab [*x,n*]=newton(*f,df,x0,e*) permettant le calcul de $\bar{x}$ avec une précision donnée *e* ayant en arguments de sortie la valeur approchée x de $\bar{x}$ et le nombre d'itérations n.
4. Construire la fonction scilab graphe2(f,a,b) décrivant la construction graphique pas à pas des termes x_n.

5. On donne ci-dessous le programme correspondant à la méthode de la sécante. Qu'elles sont les différences avec la méthode de fausse position et de Newton?

```
function x=secante(f,x0,eps)
  df = (f(x0+0.001)-f(x0))/0.001;
  x = x0 - f(x0)/df;
  while (abs(f(x))>eps)
    df = (f(x)-f(x0))/(x-x0);
    x1 = x - f(x)/df;
    x0 = x;
    x  = x1;
  end
endfunction
```

Activité 4 : Vitesse de convergence

On reprend la fonction $f(x)=x^2-2$ sur [1,2]

1. Décrire graphiquement l'évolution de $|x_n-\sqrt{(2)}|$ en fonction de n pour les 4 méthodes abordées.
2. Déterminer graphiquement l'ordre de convergence.
3. Reprendre avec une autre fonction.

Activité 5 : La methode de Newton : un exemple en dimension 2 Gregory Vial

Notre problème consiste à déterminer les points d'intersection d'un cercle avec une hyperbole. Precisément, on recherche les solutions du système d'équations polynomiales

$$x^2+y^2=2$$
$$x^2-y^2=1$$

1. Illustrer ce problème à l'aide d'une figure.
2. Calculer les solutions exactes.
3. Construire une fonction [x,n,xx]=newton2(f,df,x0,eps) qui a pour arguments de sorties la solution approchée, le nombre d'itérations et la suite (x_n).
4. Illustrer graphiquement la convergence de la suite.

Le but est d'étudier le comportement de la méthode de Newton pour la résolution du système.
La programmation de la méthode de Newton à l'aide de scilab ne présente pas de difficulté excepté :

- Les fonctions *f* et *df* sont des arguments d'entrée de la fonction newton. Il est donc nécessaire d'utiliser la commande feval pour les appeler.
- On n'inverse pas explicitement la jacobienne, mais on résout un systeme linéaire (a l'aide de la commande \) ;
- La fonction *df* est définie explicitement, mais on pourrait utiliser une dérivation symbolique ou numérique.

Activité 6 : Utilisation de fsolve:

1. Soit $f(x_1,x_2)=(x_1^3+x_2^3-3,\ x_1^2+x_2^2-2x_2)$: on peut montrer que ce système d'équations admet une unique solution dans le ¼ de plan ($x_1>0,x_2>0$). Afin d'en obtenir une approximation, on peut taper dans Scilab :

```
deff('y=f2(x)','y=[x(1)^3+x(2)^3-3,x(1)^2+x(2)^2-2*x(2)]');
x=fsolve([1,1],f2)
```

On obtient alors $x\approx$ (*0.9587068, 1.2843962*) et $f(x)\approx$ (*- 0.4440892 E-15 ,0*) ce qui confirme la validité de la solution obtenue.

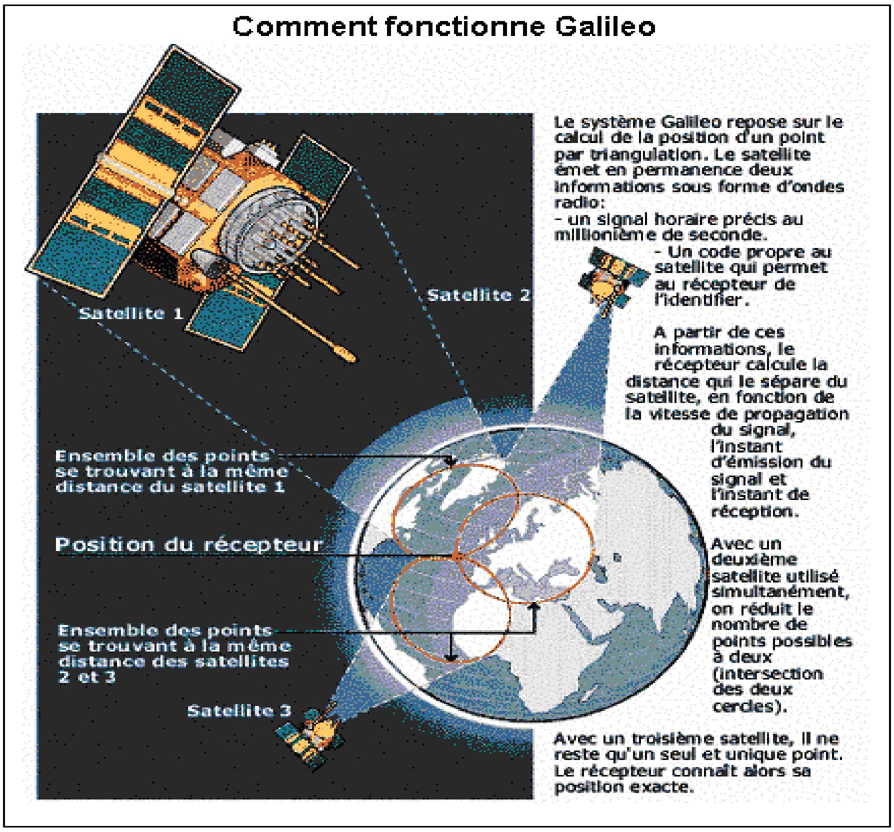
2. Montrer graphiquement avec Scilab puis mathématiquement que les équations suivantes ont une unique solution :

(i) $2\cos(x) - x = 0$ avec $x > 0$

(ii) $\exp(x) - y = 0$ et $x^2 + y^2 = 2$ avec $x > 0$ et $y > 0$

Utiliser l'instruction `fsolve` de Scilab pour en obtenir une valeur approchée.

3. Calcul d'une position par GPS (Global Positioning System)



Comme expliqué sur la figure ci-dessus, le GPS est un système de positionnement basé sur le connaissance avec une précision extrême de la distance du récepteur à trois satellites (situés à des orbites de l'ordre de 28 000km).

On suppose que les trois satellites au moment du calcul de distance ont les positions suivantes dans un repère cartésien d'origine le centre de la terre :

S1=(-11 716.227778 , -10 118.754628 , 21 741.083973) km

S2=(-12 082.643974, -20 428.242179 , 11 741.374154) km

S3=(14 373.286650, -10 448.439349, 19 596.404858) km

Sachant que les trois distances respectives au récepteur ont été calculées et valent

(d1,d2,d3)= (22 163.847742, 21 492.777482, 21 492.469326) km

calculer avec l'instruction `fsolve` de Scilab (et l'instruction `norm`) la position exacte du récepteur. Vérifier que celui-ci se trouve bien à la surface de la terre.

CORRECTIONS TP 1

Activité 1:

```
function y=f(x)
  y=x^2-1;
endfunction

function y=df(x)
  y=2*x;
endfunction

function [x,n]=dicho(f,a,b,eps)
  if (f(a)*f(b)>0) then
    disp("Il ne semble pas y avoir de
zero dans l''intervalle.");
    return;
  end
n=0
  while (abs(b-a))>eps)
    if (f(a)*f((a+b)/2)>0) then
      a = (a+b)/2;
    else
      b = (a+b)/2;
    end
    n=n+1
  end
  x = (a+b)/2;
endfunction
```

Activité 2

```
function [x,n]=regulafalsi(f,a,b,eps)
  if (f(a)*f(b)>0) then
    disp("Il ne semble pas y avoir de
zero dans l''intervalle.");
    return;
  end
n=0
 m = a-f(a)*(b-a)/(f(b)-f(a));
  while (abs(f(m))>eps)
    m = a-f(a)*(b-a)/(f(b)-f(a));
    if (f(a)*f(m)>0) then
      a = m;
    else
      b = m;
    end
    n=n+1
  end
  x = m;
endfunction
```

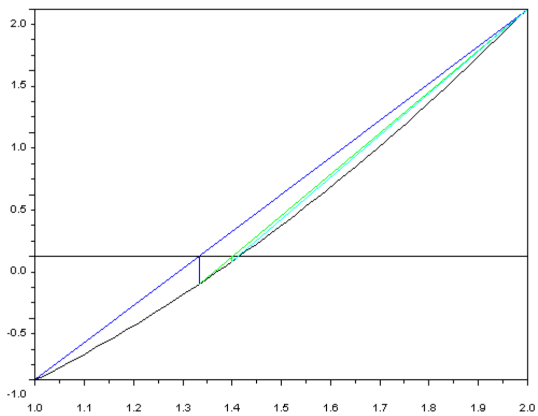
Activité 3

```
function [x,n]=newton(f,df,x0,eps)
  x = x0,  n=0
  while (abs(f(x))>eps)
    x = x - f(x)/df(x), n=n+1
   end
endfunction
```

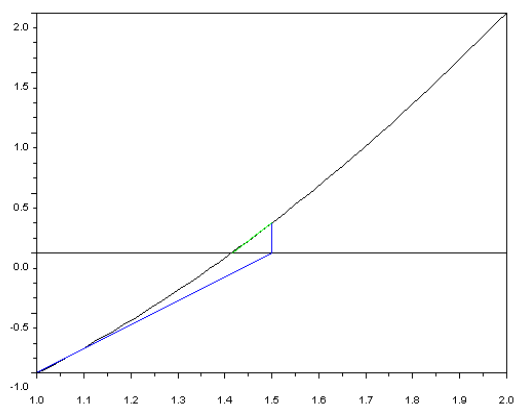
```
function graphe2(f,df,a,b)
x=linspace(a,b,50), y=feval(x,f)
clf(), plot2d(x,y), plot2d(x,0*x,1)
n=0, i=1
while n==0
i=i+1
c=a-f(a)/df(a)
plot2d([a c],[f(a) 0],i)
plot2d([c c],[0 f(c)],i)
a=c
n=input("Pour continuer tapez 0 sinon
1")
end
endfunction
```

```
-->[x,n]=dicho(f,1,2,0.00001)
 n  =   17.
 x  =   1.4142075

-->[x,n]=regulafalsi(f,1,2,0.00001)
 n  =    8.
 x  =    1.4142132
```



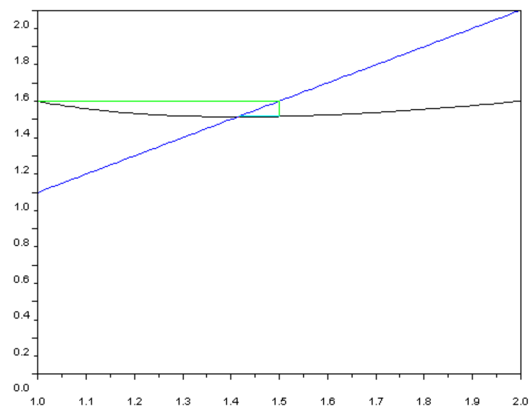

```
-->[x,n]=newton(f,df,1,0.00001)
 n  =    3.
 x  =    1.4142157
```



```
function colimacon(g,a,b)
clf
x=linspace(a,b,100)
y=feval(x,g)
ymin=min(0,a,min(y))
ymax=max(0,b,max(y))
plot2d(x,feval(x,g),1,rect=[a ymin b
ymax])
plot2d(x,x,2)
u0=input('entrer u0')
plot2d([u0 u0],[0 g(u0)],3)
n=1
i=3
while n==1
plot2d([u0 g(u0)],[g(u0) g(u0)],i)
u0=g(u0)
plot2d([u0 u0],[u0 g(u0)],i)
i=i+1
n=input("pour continuer taper 1")
end
endfunction
```

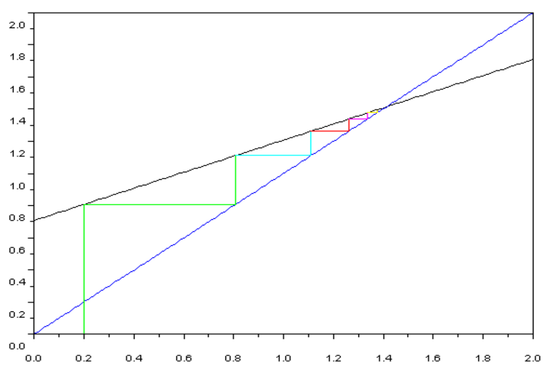
Avec la méthode de Newton:

$g(x)=\frac{1}{2}\left(x+\frac{2}{x}\right)$



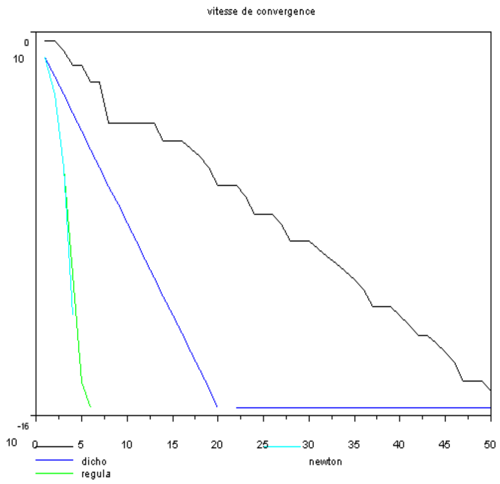
Avec une autre méthode :

$g(x)=0,5x+0,5\sqrt{(2)}$



Activité 4

```
function y=convergence(f,df,a,b,x0)
clf
edicho=[], eregula=[], esecante=[]
enewton=[], adicho=a, bdicho=b
aregula=a, bregula=b, xsecante=a
xnewton=a
for n=1:50
// dichotomie 14
 edicho=[edicho abs(adicho-x0)]
 if f(adicho)*f((adicho+bdicho)/2)<0
then bdicho=(adicho+bdicho)/2
else adicho=(adicho+bdicho)/2
 end
// regula falsi 19
 cregula = aregula-
f(aregula)*(bregula-aregula)/
(f(bregula)-f(aregula))
eregula=[eregula abs(cregula-x0)]
 if f(aregula)*f(cregula)<0 then
bregula=cregula
else aregula=cregula
 end
// secante  25
xsecante=xsecante-
f(xsecante)*(0.0001)/
(f(xsecante+0.0001)-f(xsecante))
esecante=[esecante abs(xsecante-x0)]
// newton 28
xnewton=xnewton-
f(xnewton)/df(xnewton)
enewton=[enewton abs(xnewton-x0)]
end
emax=max(max(esecante),max(edicho),m
ax(eregula),max(enewton))
y=[edicho' eregula' esecante'
enewton']
plot2d(1:50,[edicho' eregula'
esecante'
enewton'],leg="dicho@regula@secante@
newton",logflag="nl")
title("vitesse de convergence")
endfunction
```




```
regress(1:4,log(y(1:4,4)))
```

Activité 5

```
function y=g(x)
y=[x(1)^2+x(2)^2-2;x(1)^2-x(2)^2-1]
endfunction

function y=dg(x)
y=[2*x(1),2*x(2);2*x(1),-2*x(2)];
endfunction

function
[x,nb,xx]=newton2(f,df,x0,eps)
err=1;
xt=x0;x=x0;xx=[x0];
nb=0;
while (err>eps)
        nb=nb+1;
        x=x-df(x)\f(x);
        err=norm(x-xt);
        xt=x;
        xx=[xx,x];
end
endfunction
```

```
t=linspace(0,2*%pi,100);
plot2d(sqrt(2)*cos(t),sqrt(2)*sin(t),fr
ameflag=4);
t=linspace(-2,2,100);
plot(cosh(t),sinh(t));
plot(-cosh(t),sinh(t));
title('Methode de Newton pour
l''intersection de deux courbes');
[x n xx]=newton2(h,dh,[0,5;1,5],0,0001)
plot(xx(1,:),xx(2,:),'r+');
```

Activité 6

2 i Tracer 2cos(x)-x sur [0 2] et montrer que 2cosx-x<2-x

2 ii Tracer la sphère et l'exponentielle, puis utiliser la fonction exp(x)-sqrt(2-x^2) avec fsolve

3.

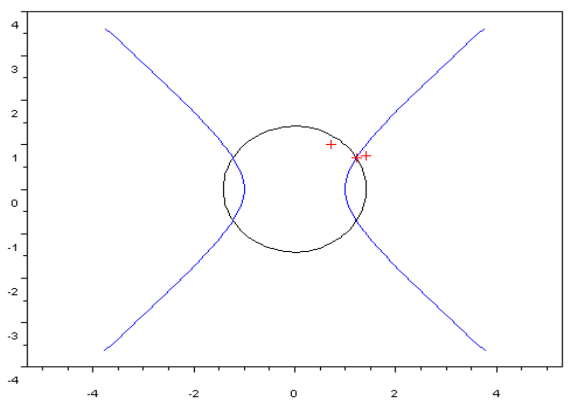
```
deff('y=f2(x)','y=[sqrt((x(1)+11716.
227778)^2+(x(2)+10118.754628)^2+
(x(3)-21741.083973)^2)-22163.847742,
sqrt((x(1)+12082.643974)^2+
(x(2)+20428.242179)^2+
(x(3)-11741.374154)^2)-21492.777482,
sqrt((x(1)-14373.286650)^2+
(x(2)+10448.439349)^2+
(x(3)-19596.404858)^2)-21492.469326]
');
-->fsolve([2000 -5000 1000],f2)
 ans  =

    595.02505  - 4856.0251    4078.33

-->norm(fsolve([2000 -5000 1000],f2))
 ans  =

    6369.2864
```

Construction des courbes et solutions



Quelques sites pour se préparer :

http://www.ann.jussieu.fr/~seguin/enseignement-agreg.html
http://cermics.enpc.fr/scilab_new/site/Initiation/index.htm
http://www.ann.jussieu.fr/~dumas/aide.html
http://www.math.bretagne.ens-cachan.fr/people/gregory.vial/
http://www-fourier.ujf-grenoble.fr/~rjoly/archives.html
http://www.cmi.univ-mrs.fr/~melot/tp.html

Quelques rappels théoriques tirés de http://www.math-linux.com/

L'objectif ce méthode est la résolution d'équation du type : $f(x) = 0 \quad (E)$

Soit x_k une solution de (E). L'idée générale est de se ramener à une équation du type $g(x) = x$

où $x = x_k$ est un point fixe de l'application g.

On introduit alors une suite d'itérée $(x_n)_{n \geq 0}$ qui converge vers le point fixe x_k de g, qui est en l'occurence la solution de l'équation (E).

Théorème du point fixe

Existence.

Si $g \in \mathcal{C}[a,b]$ et $g(x) \in [a,b], \forall x \in [a,b]$, alors g a un point fixe x_k en [a, b].

Unicité. Si $g \in \mathcal{C}^1[a,b]$ et s'il existe une constante τ dans]0,1[telle que $|g'(x)| \leq \tau$

sur [a, b] alors:

- le point fixe x_* est unique
- la suite définie par $x_{n+1} = g(x_n)$ converge vers x_* le point fixe de g $\forall x_0 \in [a,b]$.

On généralise le théorème dans $\mathbb{R}^n$:

Théorème du point fixe pour une application contractante (Théorème de Banach ou de Picard)

Soient E un espace métrique complet (non vide) et f une application contractante de E dans E. Il existe un point fixe unique x^* de f dans E, c'est-à-dire tel que $f(x^*) = x^*$. De plus toute suite d'éléments de E vérifiant la récurrence $x_{n+1} = f(x_n)$ converge vers x^*

Preuve de l'existence. On définit l'application h sur [a,b], comme suit :

$$h(x) = x - g(x)$$

Clairement : $h(a) = a - g(a) \leq 0$ et $h(b) = b - g(b) \geq 0$ puisque par hypothèse $g(x) \in [a,b], \forall x \in [a,b]$. On conclut donc en utilisant le théorème des valeurs intermédiaires :

$$\exists x_* \in [a,b] | / h(x_*) = 0$$

c'est à dire :

$$\exists x_* \in [a,b] | / g(x_*) = x_*.$$

Preuve de l'unicité. Supposons qu'il existe deux point fixes x_1, x_2 pour l'application g avec $x_1 \neq x_2$. En utilisant le théorème de la moyenne, on montre l'existence d'un élément $\xi \in]a,b[$ telle que :

$$g(x_1) - g(x_2) = g'(\xi)(x_1 - x_2)$$

donc $$|g(x_1) - g(x_2)| = |x_1 - x_2| = |g'(\xi)||x_1 - x_2| < \tau|x_1 - x_2| < |x_1 - x_2|$$

ce qui est contradictoire donc $$x_1 = x_2.$$

La suite définie par $x_{n+1} = g(x_n)$ est bien définie puisque $g(x) \in [a,b], \forall x \in [a,b]$ et par voie de conséquence $g(x_n) \in [a,b], \forall x_n$ à condition évidemment que $x_0 \in [a,b]$. Comme précédemment, en utilisant le théorème de la moyenne, on montre l'existence pour chaque n, d'un élément $\xi_{n-1} \in]x_{n-1}, x_*[$ tel que

$$\begin{array}{rcl} |x_n - x_*| & = & |g(x_{n-1}) - g(x_*)| = |g'(\xi_{n-1})||x_{n-1} - x_*| \leq \tau|x_{n-1} - x_*| \\ |x_n - x_*| & \leq & \tau|x_{n-1} - x_*| \leq \tau^2|x_{n-2} - x_*| \leq \ldots \leq \tau^n|x_0 - x_*| \end{array}$$

Par passage à la limite on voit clairement que :

$$\lim_{n\to\infty} |x_n - x_*| = \lim_{n\to\infty} \tau^n|x_0 - x_*| = 0$$

puisque τ est dans]0,1[.

Corollaire

- $|x_n - x_*| \leq \tau^n \sup(x_0 - a, x_0 - b)$
- $|x_n - x_*| \leq \dfrac{\tau^n}{1-\tau}|x_1 - x_0|$

Preuve du corollaire. La première inégalité est évidente. Pour démontrer la seconde inégalité, on utilise le fait que :

$$|x_{n+1} - x_n| \leq \tau|x_n - x_{n-1}| \leq \ldots \leq \tau^n|x_1 - x_0|$$

Soit deux entiers $n, m \in \mathbf{N}$:

$$\begin{array}{rcl} |x_n - x_{n+m}| & \leq & |x_n - x_{n+1}| + |x_{n+1} - x_{n+2}| + \ldots + |x_{n+m-1} - x_{n+m}| \\ & \leq & \tau^n|x_1 - x_0| + \tau^{n+1}|x_1 - x_0| \ldots + \tau^{n+m-1}|x_1 - x_0| \\ & \leq & (\tau^n + \tau^{n+1} + \ldots + \tau^{n+m-1})|x_1 - x_0| \\ & \leq & \tau^n \dfrac{1-\tau^m}{1-\tau}|x_1 - x_0| \\ & \leq & \dfrac{\tau^n}{1-\tau}|x_1 - x_0| \end{array}$$

En faisant tendre m vers l'infini :

$$|x_n - x_*| \leq \frac{\tau^n}{1-\tau}|x_1 - x_0|$$

Vitesse de convergence-Ordre de convergence d'une suite

Supposons qu'une suite $(x_n)_{n\geq 0}$ converge vers un élélment x_* :

$$\lim_{n\to\infty} |x_n - x_*| = \lim_{n\to\infty} |e_n| = 0$$

où $e_n = x_n - x_*$ représente l'erreur.

S'il existe deux constantes $C > 0$ et p telles que : $\lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|^p} = \lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^p} = C$

on dit alors que la convergence de la suite $(x_n)_{n\geq 0}$ vers x_* est d'ordre p avec une constante d'erreur asymptotique C.

Cas particuliers.
Si p=1 et C<1 on dit que la convergence est linéaire.
Si p=2, la convergence est dite quadratique.
Si p=3, la convergence est dite cubique.

Ordre de convergence d'une méthode de point fixe

Évidemment plusieurs cas peuvent se présenter, on peut construire plusieurs fonctions g et cela dépend aussi de la nature de f.

Si $g'(x_*) \neq 0$ alors $\lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|} = \lim_{n\to\infty} |g'(\xi_{n-1})| = |g'(x_*)|$

donc puisque $g'(x_*) \neq 0$, la constante d'erreur asymptotique est $C = |g'(x_*)|$ et la convergence est linéaire, c'est à dire d'ordre 1 et $C < 1$ puisque $|g'(x)| \leq \tau < 1$ sur [a, b].

Si $g'(x_*) = 0$, on doit faire une étude plus poussée en introduisant un développement de Taylor au voisinage de x_* de de la fonction g, en utilisant le fait évidemment que e_n tend vers 0. Par exemple à l'ordre 3 cela donne

$$\begin{aligned} x_{n+1} &= g(x_n) \\ &= g(e_n + x_*) \\ &= g(x_*) + e_n g'(x_*) + \frac{e_n^2}{2!} g''(x_*) + \frac{e_n^3}{3!} g^{(3)}(\xi_n) \\ &= x_* + 0 + \frac{e_n^2}{2!} g''(x_*) + \frac{e_n^3}{3!} g^{(3)}(\xi_n) \\ &= x_* + \frac{e_n^2}{2!} g''(x_*) + \frac{e_n^3}{3!} g^{(3)}(\xi_n) \end{aligned}$$

avec $\xi_n \in]x_*, e_n + x_*[=]x_*, x_n[$, d'où $e_{n+1} = x_{n+1} - x_* = \frac{e_n^2}{2!} g''(x_*) + \frac{e_n^3}{3!} g^{(3)}(\xi_n)$

et $\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^2} = |\frac{g''(x_*)}{2}|$

la constante d'erreur asymptotique est $C = |\frac{g''(x_*)}{2}|$ et la convergence est quadratique, c'est à dire d'ordre 2. On peut alors citer le théorème suivant.

Théorème. Si

$$\begin{array}{rcl} g'(x_*) & = & g''(x_*) = g^{(3)}(x_*) = \ldots = g^{(k-1)}(x_*) = 0 \\ g^{(k)}(x_*) & \neq & 0 \end{array}$$

alors la méthode du point fixe est d'ordre k.

Preuve. On introduit un développement de Taylor au voisinage de x_* de de la fonction g à l'ordre k, en utilisant le fait évidemment que e_n tend vers 0

$$\begin{array}{rcl} x_{n+1} & = & g(x_n) \\ & = & g(e_n + x_*) \\ & = & g(x_*) + e_n g'(x_*) + \frac{e_n^2}{2!} g''(x_*) + \ldots \frac{e_n^k}{k!} g^{(k)}(\xi_n) \\ & = & x_* + \frac{e_n^k}{k!} g^{(k)}(\xi_n) \end{array}$$

avec $\xi_n \in]x_*, e_n + x_*[=]x_*, x_n[$, d'où $e_{n+1} = x_{n+1} - x_* = \frac{e_n^k}{k!} g^{(k)}(\xi_n)$

et $\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^k} = |\frac{g^{(k)}(x_*)}{k!}|$ puisque $\lim_{n\to\infty} \xi_n = x_*$.

Méthode de Newton

Définition

La méthode de Newton est une méthode de point fixe avec pour application g :

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

On voit clairement que rechercher un point fixe de l'application g revient à chercher une solution de l'équation

$$f(x) = 0 \quad (E).$$

On rappelle que la recherche d'un point fixe se fait via un algorithme itératif définit par la suite

$$x_{n+1} = g(x_n)$$

le schéma numérique de la méthode de Newton est donc donné par

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

Méthode de Newton multidimensionnelle

Celle-ci généralise la méthode de Newton-Raphson pour une fonction $f: R^n \longrightarrow R^p$. Le schéma est donné par l'équation suivante : $x_{n+1} = x_n - [Df(x_n)]^{-1} \cdot f(x_n)$, où Df est la matrice jacobienne de f. La suite $(x_n)_n$ converge vers le zéro de f (sous réserve d'existence et d'unicité).

Interprétation géométrique

L'équation de la tangente à la courbe de f au point $(x_n, f(x_n))$ est donné par

$$y = f(x_n) + f'(x_n)(x - x_n)$$

x_{n+1} n'est rien d'autre que l'abscisse du point d'intersection de cette tangente avec l'axe (Ox), en effet

$$\begin{aligned} y &= 0 \\ x &= x_n - \frac{f(x_n)}{f'(x_n)} \end{aligned}$$

On prend alors pour x_{n+1} : $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$

Convergence de la méthode de Newton

Théorème. Soit $f \in \mathcal{C}^2([a,b])$ et soit $x_* \in]a,b[$ tel que $f(x_*) = 0$ et $f'(x_*) \neq 0$ alors il existe $\delta > 0$ tel que la méthode de Newton converge pour tout élément $x_0 \in I_\delta = [x_* - \delta, x_* - \delta]$

Preuve. Par hypothèse, f' est continue et $f'(x_*) \neq 0$, il existe donc un $\eta > 0$ tel que :

$$f'(x) \neq 0 \quad \forall x \in [x_* - \eta, x_* - \eta] \subset [a,b]$$

La dérivée de g est définie par : $g'(x) = (x - \dfrac{f(x)}{f'(x)})' = \dfrac{f(x)f''(x)}{f'(x)^2}$

Par hypothèse, $f(x_*) = 0$ et $f'(x_*) \neq 0$, par conséquent : $g'(x_*) = 0$

De plus, g' est continue sur $[x_* - \eta, x_* - \eta]$ vu que $f'(x) \neq 0 \quad \forall x \in [x_* - \eta, x_* - \eta] \subset [a,b]$

écrivons la continuité de g' en x_* dans l'intervalle $[x_* - \eta, x_* - \eta]$

$$\forall \varepsilon > 0 \quad |x - x*| \leq \eta \Longrightarrow |g'(x) - g'(x_*)| \leq \varepsilon$$

c'est à dire

$$\forall \varepsilon > 0 \quad |x - x*| \leq \eta \Longrightarrow |g'(x)| \leq \varepsilon$$

Il existe alors $\delta < \eta$ tel que $|g'(x)| \leq \tau, \quad \forall x \in I_\delta = [x_* - \delta, x_* - \delta], \quad \tau \in]0,1[$.

On a donc montré l'une des hypothèses du point fixe. Il reste à montrer à présent que g est stable dans l'intervalle I_δ, c'est à dire : $g(I_\delta) \subset I_\delta$

En utilisant le théorème de la moyenne on montre qu'il existe un élément $\xi \in]x, x_*[$ tel que

$$\begin{array}{rcl} |g(x) - g(x_*)| & = & |g'(\xi)||x - x_*| \leq \tau|x - x_*| < |x - x_*| \\ |g(x) - x_*)| & < & |x - x_*| \end{array}$$

d'où $g(I_\delta) \subset I_\delta$

On a donc montré que :

- $g \in \mathcal{C}^1(I_\delta)$ et $g(I_\delta) \subset I_\delta$
- il existe une constante τ dans]0,1[telle que $|g'(x)| \leq \tau$. On conclut alors à l'aide du théorème du point fixe : la suite définie par $x_{n+1} = g(x_n)$ converge vers x_* le point fixe de g $\forall x_0 \in I_\delta$.

Ordre de convergence de la méthode de Newton

Théorème. Si

- $g'(x_*) = 0$
- g'' est continue sur un ouvert $\mathcal{O}$ contenant x_*
- $g''(x_*) \neq 0$

alors il existe $\delta > 0$ telle que la suite définie par $x_{n+1} = g(x_n)$ converge vers x_* le point fixe de g $\forall x_0 \in I_\delta = [x_* - \delta, x_* - \delta]$. L'ordre de convergence de la méthode de Newton est alors 2 : la convergence est **quadratique**.

Preuve. Comme précédemment, on montre qu'il existe $\delta > 0$ et $\tau \in]0,1[$ tels que

$$|g'(x)| \leq \tau, \quad \forall x \in I_\delta = [x_* - \delta, x_* - \delta] \subset \mathcal{O}$$

Puisque $g'(x_*) = 0$, on introduit un développement de Taylor au voisinage de x_* de de la fonction g d'ordre 2, en utilisant le fait évidemment que $e_n = x_n - x_*$ tend vers 0.

$$\begin{aligned} x_{n+1} &= g(x_n) \\ &= g(e_n + x_*) \\ &= g(x_*) + e_n g'(x_*) + \frac{e_n^2}{2!} g''(\xi_n) \\ &= x_* + 0 + \frac{e_n^2}{2!} g''(\xi_n) \\ &= x_* + \frac{e_n^2}{2!} g''(\xi_n) \end{aligned}$$

avec $\xi_n \in]x_*, e_n + x_*[=]x_*, x_n[$, d'où $e_{n+1} = x_{n+1} - x_* = \frac{e_n^2}{2!} g''(\xi_n)$ et $\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^2} = |\frac{g''(x_*)}{2}|$

la constante d'erreur asymptotique est $C = |\frac{g''(x_*)}{2}|$ et la convergence est quadratique, c'est à dire d'ordre 2 puisque par hypothèse $g''(x_*) \neq 0$.

Multiplicité d'une racine

Soit $p \in \mathbf{N}$, on dit que f possède une racine de multiplicité ou d'ordre p si

$$\begin{aligned} f(x_*) &= f'(x_*) = f''(x_*) = f^{(3)}(x_*) = \ldots = f^{(p-1)}(x_*) = 0 \\ f^{(p)}(x_*) &\neq 0 \end{aligned}$$

et f peut s'écrire comme suit : $f(x) = (x - x_*)^p h(x)$ avec $h(x_*) \neq 0$.

Convergence de la méthode de Newton : racine simple

Théorème. Si $f \in \mathcal{C}^2([a, b])$ et f possède une racine simple alors la méthode de Newton converge au moins avec un ordre quadratique (ordre 2).

Preuve. Si f possède une racine simple alors $\begin{aligned} f(x_*) &= 0 \\ f'(x_*) &\neq 0 \end{aligned}$

La méthode de Newton converge.
La dérivée de g est définie par :

La dérivée seconde de g est définie par : $g'(x) = (x - \frac{f(x)}{f'(x)})' = \frac{f(x)f''(x)}{f'(x)^2}$

$$g''(x) = \frac{f'(x)^3 f''(x) + f'(x)^2 f(x) f^{(3)}(x) - 2f'(x)f''^2(x)f(x)}{f'(x)^4}$$

et $g''(x_*) = \frac{f''(x_*)}{f'(x_*)}$

Si $f''(x_*) \neq 0$ alors

- $g'(x_*) = 0$
- g'' est continue sur un ouvert $\mathcal{O}$ contenant x_*
- $g''(x_*) \neq 0$

Si $f''(x_*) = 0$, alors $g''(x_*) = 0$, on doit faire une étude plus poussée en introduisant un développement de Taylor au voisinage de x_* de de la fonction g à l'ordre 3, en utilisant le fait évidemment que e_n tend vers 0. Cela donne

$$\begin{aligned} x_{n+1} &= g(x_n) \\ &= g(e_n + x_*) \\ &= g(x_*) + e_n g'(x_*) + \frac{e_n^2}{2!} g''(x_*) + \frac{e_n^3}{3!} g^{(3)}(\xi_n) \\ &= x_* + 0 + 0 + \frac{e_n^3}{3!} g^{(3)}(\xi_n) \\ &= x_* + \frac{e_n^3}{3!} g^{(3)}(\xi_n) \end{aligned}$$

avec $\xi_n \in]x_*, e_n + x_*[=]x_*, x_n[$, d'où $e_{n+1} = x_{n+1} - x_* = \frac{e_n^3}{3!} g^{(3)}(\xi_n)$ et $\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^3} = |\frac{g^{(3)}(x_*)}{3!}|$

la constante d'erreur asymptotique est $C = |\frac{g^{(3)}(x_*)}{3!}|$ et la convergence est cubique si $g^{(3)}(x_*) \neq 0$ c'est à dire d'ordre 3. Si $g^{(3)}(x_*) = 0$ il faudra faire un développement de Taylor d'ordre 4 de g au voisinage de x_*.

On a donc montré que la convergence de la méthode de Newton est au moins quadratique.

Compléments non développés :

Convergence de la méthode de Newton : racine multiple

Méthode de la sécante : On pose $x_{n+1} = x_n - \dfrac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$

Accélération de la convergence : Procédé Δ^2 d'Aitken et Méthode de Stellensen