

TP 4 : Résolution d'une équation $F(x) = 0$ (Méthode de Newton)

1. Méthode de Newton (ou de la tangente)

La méthode de Newton est une procédure classique pour résoudre des équations du type $f(x) = 0$ par un processus itératif. On part d'une estimation, en réitérant le procédé, on obtient une valeur qui converge vers la solution.

Soit une équation de la forme $f(x) = 0$, considérons la fonction $y = f(x)$ il s'agit de trouver le point r où cette courbe coupe l'axe des x , estimons cette intersection par un premier nombre x_1 ou x_1 constitue notre première approximation de la solution.

Nous cherchons ensuite à améliorer l'approximation, une deuxième approximation x_2 , sera trouvée en prenant l'abscisse du point de rencontre de la tangente à la courbe au point $(x_1, f(x_1))$ avec l'axe des x , puis une troisième approximation x_3 , puis x_4 , et ainsi de suite.

La méthode repose sur la propriété suivante:

- Soit une fonction $f: \mathbb{R} \rightarrow \mathbb{R}$ est dérivable sur $[a, b]$

$$x \in [a, b] \quad f(x)$$

- Soit $x_0 \in [a, b]$

Alors la tangente t à $y = f(x)$ au point $(x_0, f(x_0))$ est une bonne approximation de $y = f(x)$ au voisinage du point $(x_0, f(x_0))$

Il est facile, à partir de l'équation de la tangente, d'obtenir x_1 , on a :

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

Ensuite, partant de x_1 , on calcule x_2 , etc.

1.1 Algorithme de la méthode de Newton

L'algorithme de la méthode de Newton est le suivant :

- Fixer x_0 .
- Soit x_n est une approximation de l'équation $f(x) = 0$, calculant l'approximation suivante x_{n+1} par :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad \text{pour } n = 0, 1, 2, \dots$$

à condition que $f'(x_n) \neq 0$

Exemple :

On se propose de résoudre l'équation $x - 0.2\sin(x) - 0.5 = 0$ par la méthode de Newton à l'aide du programme écrit en Matlab. On pourra prendre x_1 égale à 0.

Solution :

```
>> x(1) = 0;  
>> f = inline('x - 0.2*sin(x) - 0.5');
```

```
>> for n = 1:8
    def = 1 - 0.2*cos(x(n));

    x(n+1) = x(n) - f(x(n))/def;
end
>> format long
>> disp( [x' f(x)'] )
```

Exemple 2 :

Calculer numériquement les racines de $f(x) = x^3 - 4x + 1$, en utilisant la méthode de Newton. On pourra prendre x_1 égale à : -2, 0, 2.

Exemple 3 :

Calculer numériquement les racines de $f(x) = 10x - 9\exp(-x)$, en utilisant la méthode de Newton. On pourra prendre x_1 égale à : 0, 1

Exemple 4 :

Dans cet exemple on utilisera les commandes de matlab pour trouver la racine x qui renvoie la fonction f vers 0.

```
function [val] = myfunc(x)
```

```
val = x-x.^2.*sin(x);
```

la racine se trouve dans l'intervalle [0.84, 1.16].

```
>> fzero( ' myfunc ',1)
```

Exemple 5 :

Calculer numériquement les racines de $f(x) = -x^3 - \cos(x)$, en utilisant la méthode de Newton. On pourra prendre x_1 égale à : -1, 0