جامعة محمد بوضياف - المسيلة
Université Mohamed Boudiaf - M'sila

2019-2020, Licence 2ème année
Faculté de Technologie
parcours: Électronique

Module: TPs des Méthodes Numériques

TP 4 : La méthode de Newton

Les objectifs de cette leçon:

- Comprendre la méthode.
- Écrire un algorithme/organigramme pour cette méthode.
- Programmer cette méthode en utilisant l'environnement Matlab.
- Être capable d'appliquer cette méthode pour résoudre 1 Eq. non-lin. $f(x) = 0$.
- Être capable d'utiliser les différent critères d'arrêt pour quitter l'alg. de la méthode.

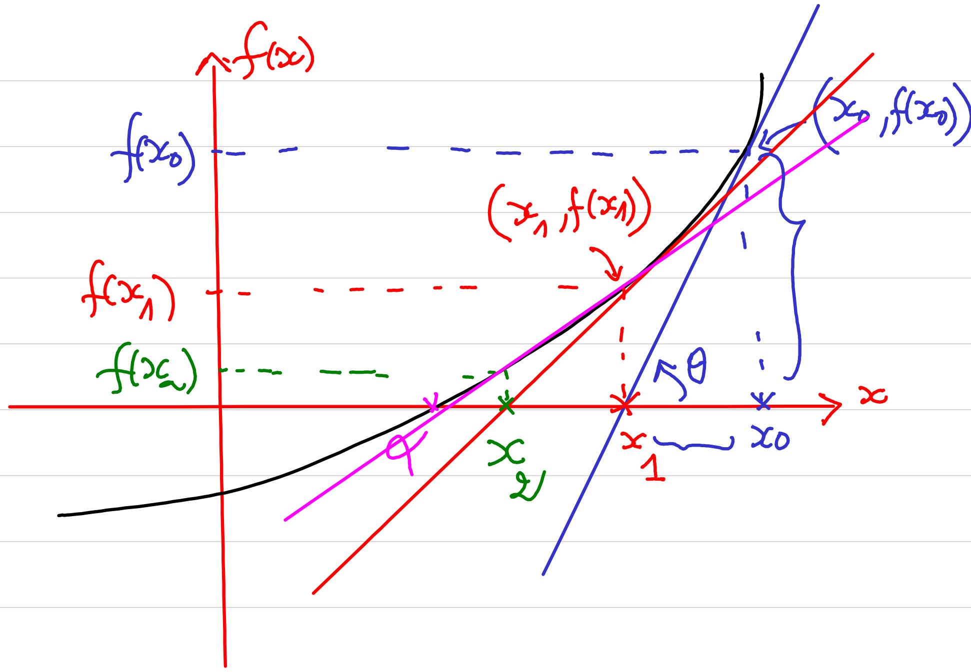
Le principe de la méthode :

1 **Principe :**

$f(x) = 0$

chercher α

s.q $f(\alpha) = 0$.



$$tg\,\theta = \frac{f(x_0) - 0}{x_0 - x_1} = f'(x_0)$$

$$\boxed{x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}} \quad \text{——} \quad (1)$$

En suivant les m̂ demarches, on peut écrire

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

$\vdots$

x_3

Le principe de la méthode:

on peut généraliser l'éq ① pour formuler une séq/ suite des itérées d'1 méthode itérative de la façon suivante

$$\boxed{x_{k+1} = \underbrace{x_k - \frac{f(x_k)}{f'(x_k)}}_{}} \longrightarrow ②$$

$x = g(x)$

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

cette formule concerne la méthode de Newton.

on peut aussi trouver la m̂ formule ds ① et ② à partir d'1 dev. en séries de Taylor de la fonction f au voisinage de x_0 c-à-d :

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + R$$

$\searrow 0$

③ Critères d'arrêt. $\longrightarrow$ ①

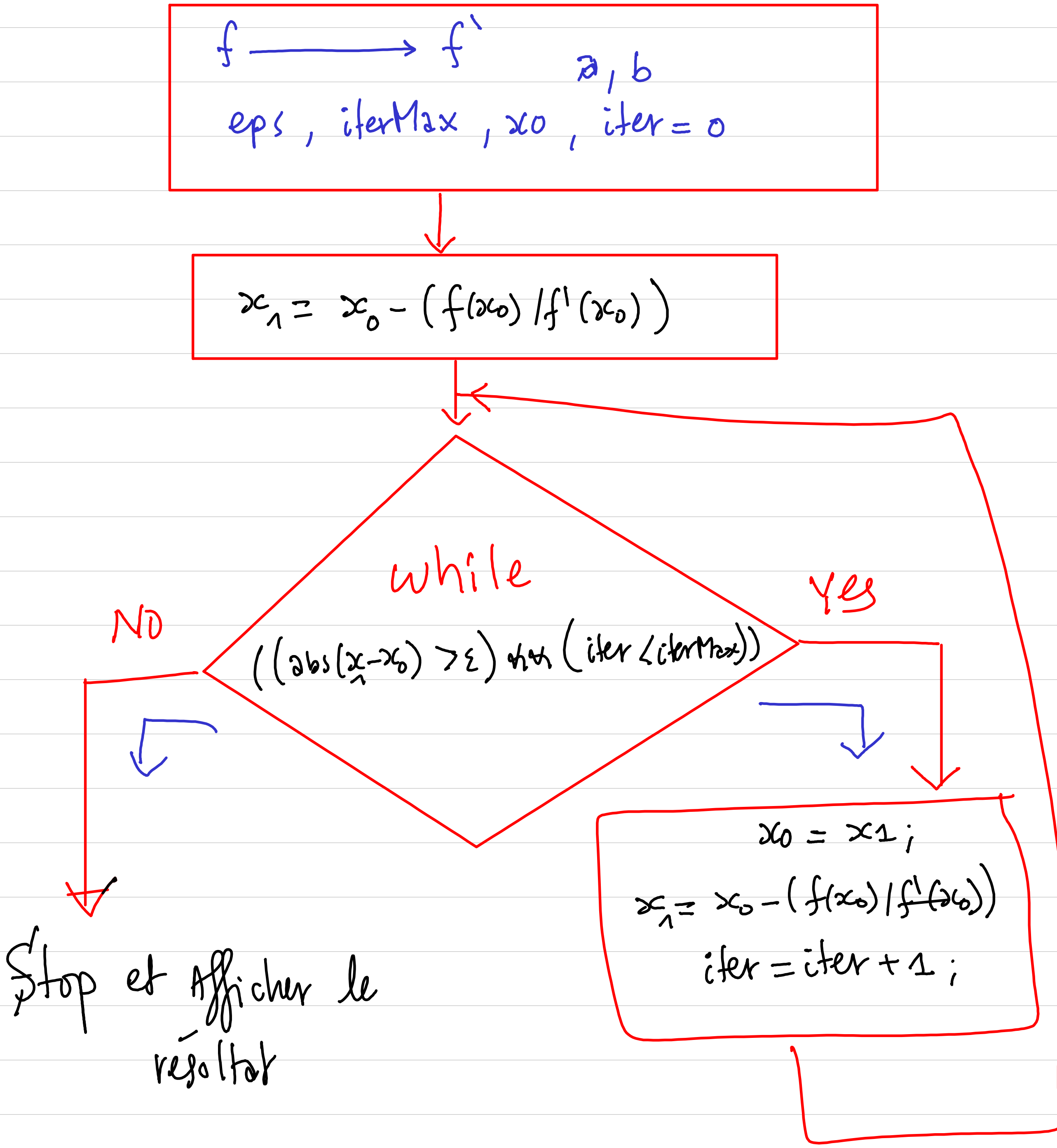
① $|x_{k+1} - x_k| < \varepsilon \longrightarrow$ while

② nbre des itérations (iterMax) $\longrightarrow$ for iter=1: iterMax

③ combinaison de ces 2 critères

while(($|x_{k+1} - x_k| > \varepsilon$) and (iter < iterMax))

L'organigramme de la méthode :



La programmation la méthode en utilisant Matlab:

```
f = inline('x.^3 + 4*x.^2 - 10');
fp = inline('3*x.^2 + 8*x');

% Paramètres
a = 1;  b = 2;
x0 = 1.5;
x1 = x0 - (f(x0)/f'(x0));
eps = 10^(-6);
iter = 0;   iterMax = 50;

if (f(a)*f(b) < 0)
    while ((abs(x1 - x0) > eps) && (iter < iterMax))
        x0 = x1;
        x1 = x0 - (f(x0)/f'(x0));
        iter = iter + 1;
        fprintf(' - - - -
    end
    fprintf(' - - - - - -
else
    disp('Il n y a pas de sol ds [a,b]');
end
```

La programmation la méthode en utilisant Matlab:

```
f = inline('x.^3 + 4*x.^2 - 10');
% Tracer la fonction f
% figure
%   x=1:0.01:5;
%   plot(x,f(x)), grid on
%--------------------------
fp=inline('3*x.^2 + 8.*x');
% Parametres
a=1;
b=2;
x0 =1.5;
x1 = x0-(f(x0)/fp(x0));
eps=1.0e-6;
iter=0;
iterMax=50;
if ((f(a)*f(b))< 0)
while ((abs(x1-x0) > eps) && (iter<iterMax))
x0=x1;
x1 = x0-(f(x0)/fp(x0));
iter=iter+1;
fprintf('pour l iteration=%d\t, la solution est x1=%f\n',iter,x1);
end
fprintf('La solution finale est x1 = %f \n',x1) ;

else
disp('il n y a pas de solution dans [a,b]')
end
```