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 3 : La méthode du point-fixe

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 :

Principe :

Elle consiste à ré-ecrire l'éq. $f(x)=0$ sous la forme $x=g(x)$

La solution maintenant est un nbre α qui vérifie $\alpha=g(\alpha)$ et on dit que α est un point fixe de $g(x)$

$$\left.\begin{array}{l} f(x)=0 \text{ --- } (1) \\ \text{chercher } \alpha \text{ sachant} \\ \text{que } f(\alpha)=\alpha \end{array}\right\} \Longleftrightarrow \left\{\begin{array}{l} x=g(x) \text{ --- } (2) \\ \text{chercher } \alpha \text{ sachant} \\ \text{que } \alpha=g(\alpha) \end{array}\right.$$

✓ ✓

En utilisant l'éq (2), on peut former une méthode itérative pour résondre l'éq (1), de la manière suivante

$$x_{k+1}=g(x_k) \;;\; k=0,1,2,\dots$$

En possedant une approx. initiale x_0 de la solution, on peut calculer les approx. suivantes comme suit

$$x_1=g(x_0)\;;\; x_2=g(x_1)\;;\;\dots\;;\; x_k=g(x_{k-1})$$

Les critères d'arrêt :

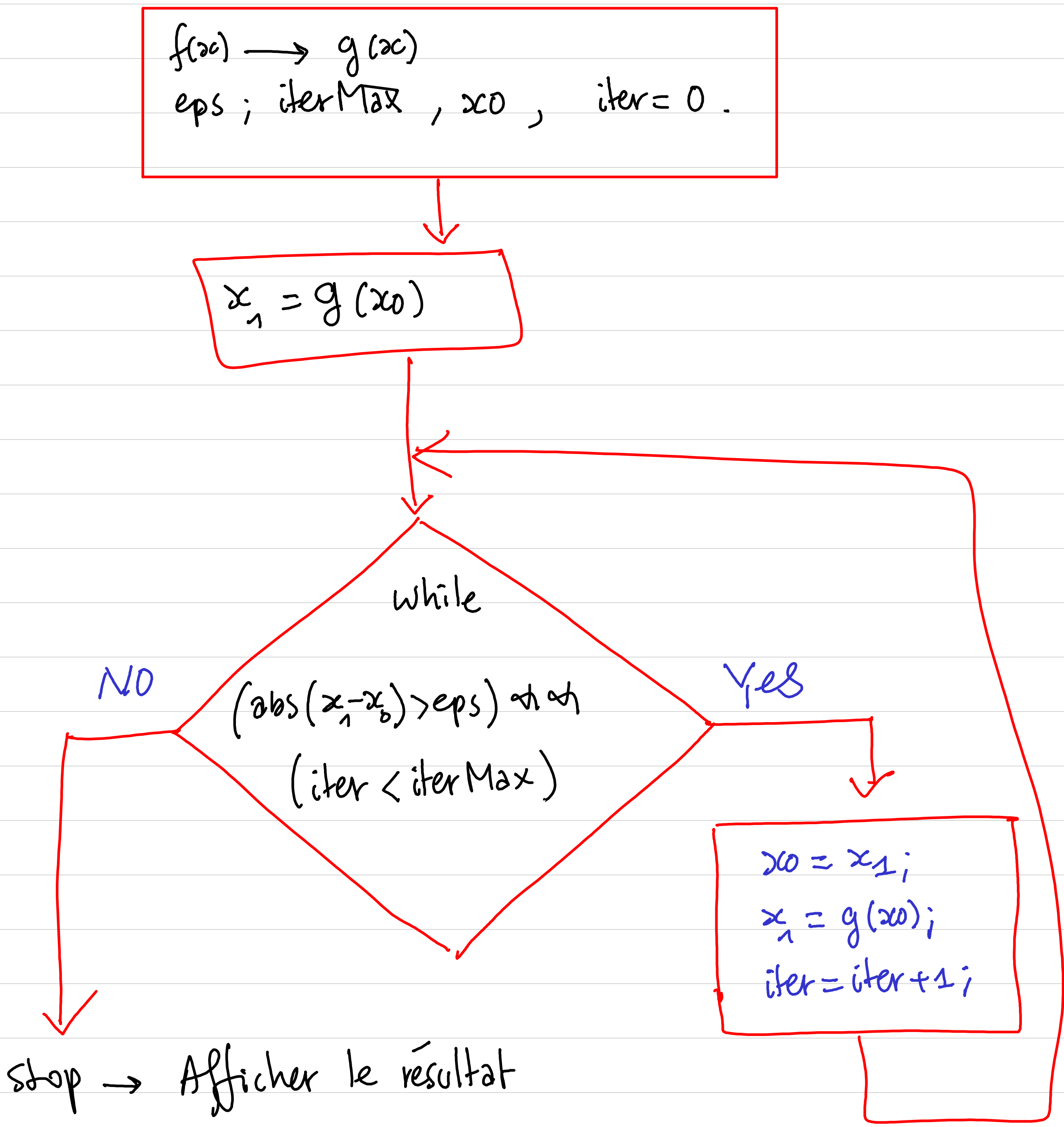
(1) $|x_{k+1}-x_k|<\varepsilon \longrightarrow$ while$(|x_{k+1}-x_k|>\varepsilon)$ ✓

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

(3) les 2 critères combinés

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

Un algorithme pour la méthode:



L'organigramme de la méthode :

$$f(x) = x^3 + 4x^2 - 10 .$$

$\rightarrow \quad x = (10 - 4x^2)^{1/3} = g_1(x)$

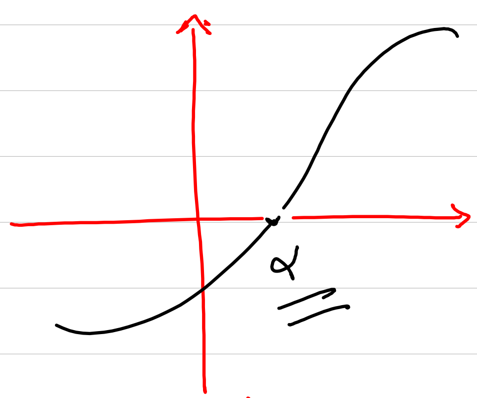
$\rightarrow \quad x = ((10 - x^3)/4)^{1/2} = \sqrt{(10 - x^3)}/2 = g_2(x)$

$\rightarrow \quad x = (10 - x^3)/4x = g_3(x)$

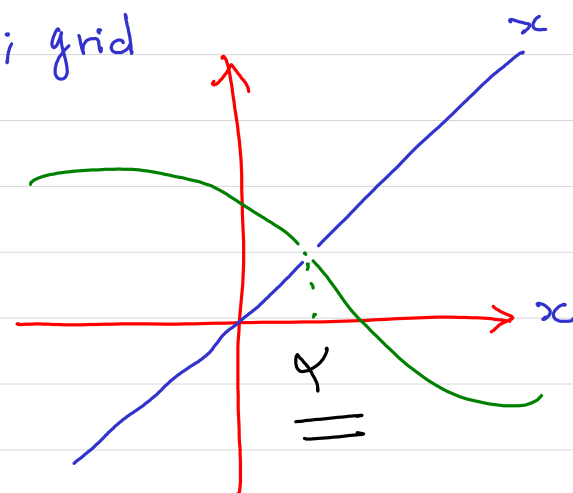
La programmation la méthode en utilisant Matlab:

```
f = inline('x.^3+4*x.^2-10');
g = inline('(10-4*x.^2).^(1/3)'); % g1(x)

% Plots
x = 0:0.01:5;
Figure
Plot(x, f(x)); grid
Figure
Plot(x, g(x), x, x); grid
```



prob#1



prob.#2

L'organigramme de la méthode :

while ((abs(x1-x0)>eps) ~~and~~ (iter < iterMax))

x0 = x1 ;

x1 = g(x0);

iter = iter+1;

fprintf(' - - - -

end

fprintf(' - -

La programmation la méthode
en utilisant Matlab:

```matlab


% Tracer les deux fonctions
figure
f = inline('x.^3 + 4*x.^2 - 10');
x=0.5:0.01:5;
plot(x,f(x)), grid on
figure
g= inline('sqrt(10-x.^3)/2'); % g2
x=0.5:0.01:5;
plot(x,g(x),x,x), grid on

%--------------------------
a=1; b=2;
x0 =1.5;
x1 = g(x0);
eps=10^(-3);
iter=0;
iterMax=5;
if ((f(a)*f(b))< 0)
while ((abs(x1-x0) > eps) && (iter<iterMax))
x0=x1;
x1 = g(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
```