1985

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

2019-2020, Bachelor 2nd Year (L2)
Faculty of Technology
Depart: Electronic Engineering

Class: Numerical Methods Lab

Lab 5 : Secant method

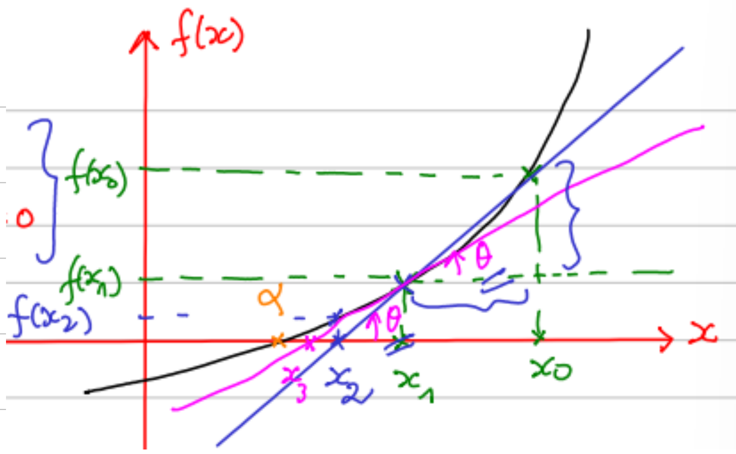
The objectives of this lesson:

- **Understand the bisection method.**
- **Write an algorithm/flowchart for this method.**
- **Write a Matlab script for this method.**
- **Be able to apply this method to solve a non-linear equation $f(x) = 0$.**
- **Be able to use various stopping criteria to exit the algorithm of the bisection method.**

Basic ideas and fundamental concepts:

$$f(x) = 0$$

$\xrightarrow{?}$ α s.t $f(\alpha) = 0$



$$tg\theta = \frac{f(x_1) - 0}{x_1 - x_2} = \frac{f(x_0) - f(x_1)}{x_0 - x_1}$$

$$\hookrightarrow \quad x_2 = x_1 - f(x_1) \cdot \frac{x_1 - x_0}{f(x_1) - f(x_0)} \longrightarrow ①$$

By following the same steps, we can get

$x_3 = \ ---$

$x_4 = \ - \ -$

$\vdots$

Now, one can generilize this formula to get

$$x_{k+1} = x_k - f(x_k) \frac{(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})} \text{ ——— } ②$$

$\nearrow$ This formula concern the secant method.

An algorithm for the method

Comparison b/w Newton's method & secant method formulas

Newton $\rightarrow$ $$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad \text{——— (3)}$$

Secant $\rightarrow$ $$x_{k+1} = x_k - f(x_k)\frac{(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})} \quad \text{——— (4)}$$

By comparing (3) & (4), we conclude that $f'(x_k)$ is estimated by

$$f'(x_k) = \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \quad \text{——— (5)}$$

Stopping Criteria :

Criteria # 1. The length of the intrvl. $[a,b] \lll$ $\rightarrow$ $|a-b| < \varepsilon$

$\rightarrow$ Stop if abs(a-b) < epsilon $\rightarrow$ while (abs(a-b) > eps) $\hookleftarrow$

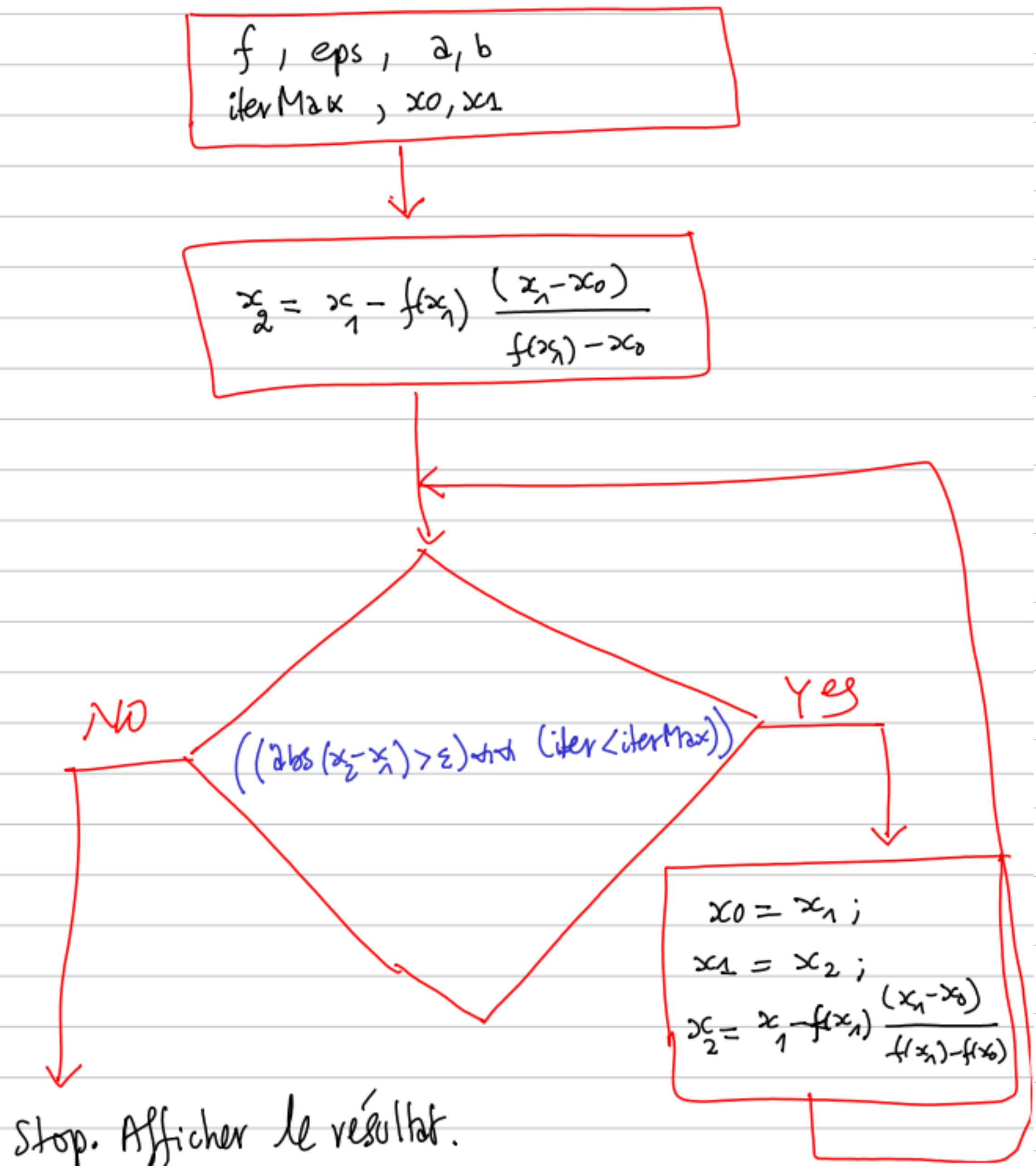
Criteria # 2. The number of iterations (MaxIter)

$\rightarrow$ for iter = 1 : MaxIter $\hookleftarrow$

Criteria # 3. Both criteria combined $\rightarrow$

while ((abs(a-b) > eps) && (iter < MaxIter)) $\hookleftarrow$

A flowchart for the method



Matlab script(s) for the method:

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

a=1;
b=2;
x0 =0;
x1=1;
x2 = x1-((f(x1)*(x1-x0))/(f(x1)-f(x0)));

iterMax=5;
if ((f(a)*f(b))< 0)
for iter=1:iterMax

x0=x1;
x1=x2;
x2 = x1-((f(x1)*(x1-x0))/(f(x1)-f(x0)));

fprintf('For iteration =%d \t , the solution is x0=%f\n',iter,x0)
end
fprintf('The final solution is x0 = %f \n',x0) ;
else
disp('There is no solution in [a,b]')
end
```

```
%f = inline('cos(x)-x.^3');

f = inline('x.^3 + 4*x.^2 - 10');
a=1; b=2;
x0 =0; x1=1;
x2 = x1-((f(x1)*(x1-x0))/(f(x1)-f(x0)));
iter=0;
eps=10^(-6);
if ((f(a)*f(b))< 0)
while (abs(x2-x1)>eps)
iter=iter+1;
x0=x1;
x1=x2;
x2 = x1-((f(x1)*(x1-x0))/(f(x1)-f(x0)));
fprintf('For iteration =%d \t , the solution is x0=%f\n',iter,x0)
end
fprintf('The final solution is x0 = %f \n',x0) ;
else
disp('There is no solution in [a,b]')
end
```