# TP d'initiation à MATLAB $n^0$ 9 : Calcul approché des solutions d'une équation

On suppose dans tout ce TP que l'on veut résoudre une équation de type

$$f(x) = 0 \quad \text{pour} \quad x \in [0,1]. \tag{1}$$

Ce type d'équations recoupe un grand nombre de problèmes mathématiques, comme celui de trouver la solution de $y_0 = g(x)$ avec $x \in [a,b]$, de déterminer un extremum local à une fonction,... Dans la plupart des cas (à part les rares cas d'école habituels de type polynomiaux...), on ne peut pas trouver une solution théorique à cette équation (voir l'exemple ci-dessous). Aussi est-on amené à calculer de façon approchée l'ensemble des solutions de cette équation. Par la suite, nous étudions différentes méthodes numériques classiques de résolution de (1).

## Un exemple

Dans toute la suite nous considérerons l'exemple de fonction $f$ suivant :

$$f(x) = \sin(2x) - x.$$

Étudier théoriquement les solutions de (1) dans ce cas (utiliser par exemple un tableau de variations...). En déduire que (1) possède alors 2 solutions, une que l'on peut donner théoriquement et notée $x_0$, l'autre notée $x_1$ a priori inconnue. Vérifier en traçant le graphe de $f$ sur $[0,1]$. Déterminer une valeur approchée à 0.1 près de $x_1$.

## Méthode de dichotomie

La première méthode, la plus naturelle, pour obtenir une valeur approchée de $x_1$ est une méthode d'approximation successive dite par "dichotomie". Elle consiste à d'abord fixer un intervalle $[a_0,b_0]$ dans lequel (1) admet une unique solution. Ainsi, $f(a_0).f(b_0) < 0$. Ensuite, on considère le milieu du segment $[a_0,b_0]$, soit $\frac{1}{2}(a_0+b_0)$ et on calcule numériquement $f\left(\frac{1}{2}(a_0+b_0)\right)$. Si $f(a_0).f\left(\frac{1}{2}(a_0+b_0)\right) < 0$ alors $x_1$ est dans $[a_0, \frac{1}{2}(a_0+b_0)]$, on pose $a_1 = a_0$, $b_1 = \frac{1}{2}(a_0+b_0)$ et on recommence le méme procédé. Sinon, $x_1$ est dans $[\frac{1}{2}(a_0+b_0), b_0]$, on pose $b_1 = b_0$, $a_1 = \frac{1}{2}(a_0+b_0)$ et on recommence le méme procédé.
Programmer cette méthode (fichier *dichotomie.m*) pour une fonction $f$ quelconque définie dans un fichier *f.m* et d'abord pour 10 itérations. Introduire ensuite dans le programme le calcul du nombre d'itérations nécessaires pour une approximation $Delta$ donnée (par exemple $Delta = 0.0001$) en fonction de $Delta$, $a_0$ et $b_0$. Appliquée cette méthode pour obtenir une valeur approchée à $10^{-10}$ près de $x_1$.

## Méthode de la sécante et méthode de Newton

On se place encore dans le cadre où l'on sait que (1) admet une unique solution dans $[a_0,b_0]$.

**Méthode de la sécante** On donne pour valeur approchée $\tilde{x_1}$ de $x_1$ le point d'intersection entre la droite $(a_0,b_0)$ et l'axe des abscisses. Montrer qu'alors $\tilde{x_1} = a_0 - f(a_0).\frac{b_0 - a_0}{f(b_0) - f(a_0)}$. En déduire que si $f$ est de classe $\mathcal{C}^2$ sur $[a_0,b_0]$, si $m_1 = \inf_{x\in[a_0,b_0]} |f'(x)|$ et $M_2 = \sup_{x\in[a_0,b_0]} |f''(x)|$ alors

$$|\tilde{x_1} - x_1| \leq \frac{M_2}{m_1} \times \frac{(b_0-a_0)^2}{8}.$$

**Méthode de Newton** On donne pour valeur approchée $\hat{x_1}$ de $x_1$ le point d'intersection entre la tangente à $f$ en $a_0$ et l'axe des abscisses. Montrer qu'alors $\hat{x_1} = a_0 - \dfrac{f(a_0)}{f'(a_0)}$. En déduire que si $f$ est de classe $\mathcal{C}^2$ sur $[a_0, b_0]$, si $m_1 = \inf_{x \in [a_0,b_0]} |f'(x)|$ et $M_2 = \sup_{x \in [a_0,b_0]} |f''(x)|$ alors

$$|\hat{x_1} - x_1| \leq \frac{M_2}{m_1} \times \frac{(b_0 - a_0)^2}{2}.$$

Pour ces 2 méthodes, on peut alors utiliser la méthode itérative de type dichotomie en remplaçant $\frac{1}{2}(a_0 + b_0)$ par $\tilde{x_1}$ ou $\hat{x_1}$. Dans quel cas gagne-t-on en rapidité de convergence par rapport à la méthode de dichotomie ?

## Méthode de Newton itérée

L'objectif de cette méthode est de déterminer une suite $(u_n)$ convergent vers $x_1$. Pour cela, on considère une fonction $\phi$ telle que pour $x \in [a, b]$, $f(x) = 0 \Longleftrightarrow \phi(x) = x$ et si $\phi$ vérifie des propriétés de type $\phi([a_0, b_0]) \subset [a_0, b_0]$ et $\sup_{x \in [a_0,b_0]} |\phi'(x)| < 1$. On définit alors $(u_n)$ par $u_{n+1} = \phi(u_n)$ avec $u_0 \in [a_0, b_0]$ et on a bien $(u_n)$ qui converge vers $x_1$, unique solution de $x = \phi(x)$. Cette méthode est appelée méthode de Newton itérée lorsque la fonction $\phi$ considérée est :

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

Expliquer le choix d'une telle fonction $\phi$ à partir de considérations sur $\phi'$. En déduire, en fonction de $a_0$ et de $b_0$, et de $N_1 = \sup_{x \in [a_0,b_0]} |\phi'(x)|$, la vitesse de convergence de $(u_n)$ vers $x_1$. Appliquer cette méthode au cas particulier de $f$ précédent. Combien faut-il calculer de terme de $(u_n)$ pour obtenir une approximation à $10^{-10}$ près de $x_1$.

## Conclusions

Comparez les différentes méthodes et mettez en évidence les avantages et inconvénients de chacune d'entre elles (on examinera en particulier les problèmes du choix de $[a_0, b_0]$, d'unicité de la solution de [1], de la régularité de la fonction). On pourra à titre d'exemple traîter le cas de $f(x) = \cos(1/(x + 0.1)) - \sqrt{x}$. Comparez avec les performances des commandes Matlab *fzero* ou *fmin*.