MPSI 2
Année scolaire 2014/15

INFORMATIQUE
Calcul approchée de racine p^{eme}

1 Recherche du zéro d'une fonction

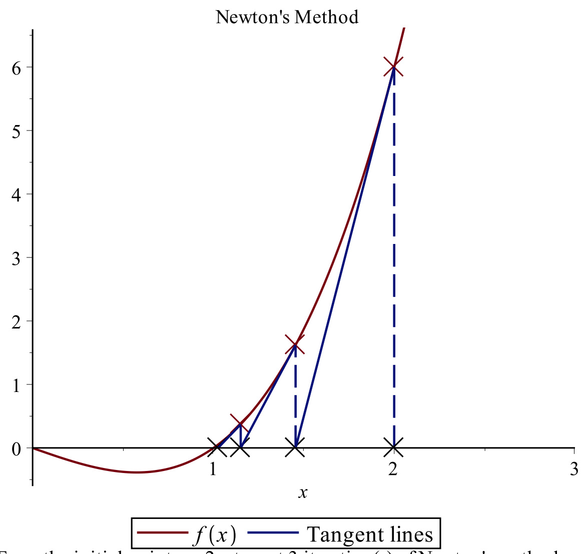
Soit $[a, b]$ un segment réel et soit f une fonction de classe C^2 sur $[a, b]$ telle que l'équation $f(x) = 0$ admet une unique solution α sur $[a, b]$. Nous étudierons dans ce TP deux méthodes permettant d'obtenir une valeur approchée de α.

1. **Par dichotomie :**
   Il s'agit d'une méthode itérative assez intuitive :
   On pose $c = (a + b)/2$ le milieu de $[a, b]$, puis :
   - Si $\alpha > c$ on pose alors $a = c$.
   - Sinon on pose $b = c$.

   On itère ensuite ce procédé le nombre de fois souhaité, puis on retourne b si on souhaite une valeur approchée supérieure de α, a si on souhaite une valeur approchée inférieure de α.
   On remarque alors qu'en n étapes on obtient une valeur approchée de α à $\dfrac{b-a}{2^n}$ près.

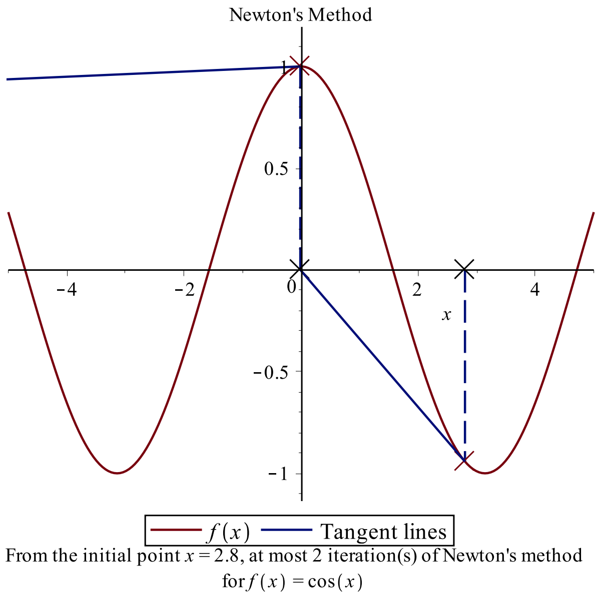
2. **La méthode de Newton :**
   On construit une suite $(x_n)_{n\in\mathbb{N}}$ de valeurs approchées de la façon suivante :
   - On choisit $x_0 \in [a, b]$
   - On pose x_1 le point d'intersection de l'axe des abscisses avec la tangente à la courbe de f en $(x_0, f(x_0))$.
   - On construit de même les termes suivants de la suite.

   On a donc la relation de récurrence : $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$



From the initial point $x = 2$, at most 3 iteration(s) of Newton's method for $f(x) = x^3 - x$

Contrairement à la méthode de dichotomie, on ne peut pas garantir la convergence de cette méthode en toute généralité comme on peut le voir dans l'exemple ci-dessous :



On devine que pour que cette méthode converge, il ne faut pas que f' soit "trop petite" sur $[a, b]$ ni que ses variations soient "trop grandes".
La proposition suivante donne une condition suffisante assez simple de convergence :

Proposition 1 *Si $f'(a) > 0$ et si f est convexe sur $[a, +\infty]$ alors pour tout $x_0 \in [a, b]$ la suite $(x_n)_{n\in\mathbb{N}}$ est bien définie et elle converge vers α en décroissant au moins à partir de x_1.*

Il est facile de s'en convaincre graphiquement, la courbe représentative d'une fonction convexe étant au-dessus de ses tangentes, nous en laissons la démonstration à la discrétion du lecteur. Remarquons que cette proposition ne nous donne pas d'informations sur la vitesse de convergence.

2 Recherche de valeurs approchées de la racine p^{eme} d'un réel strictement supérieur à 1

Soit A un réel strictement supérieur à 1, pour tout entier p supérieur ou égal à 2 on pose $f_p(x) = x^p - A$. On remarque alors que l'équation $f_p(x) = 0$ admet une unique solution $\alpha = A^{1/p}$ sur $[1, +\infty[$ et que f_p vérifie les hypothèses de la proposition 1. Nous pouvons donc obtenir des valeurs approchées de α par les deux méthodes.

Exercice 1 *Calcul de valeurs approchées par dichotomie*

1. Programmer une fonction $dicho(p, A, n)$ qui retourne la liste `L` des encadrements de α obtenus par dichotomie en n étapes à partir du segment $[1, A]$. `L[0]` sera la liste des bornes inférieures, `L[1]` sera la liste des bornes supérieures.
2. Écrire un programme permettant à l'utilisateur de saisir p, A et n et qui affiche la valeur de l'encadrement de α ainsi que le graphique correspondant (courbe de la fonction f_p et les différents encadrements).

p	A	n	borne inf	borne sup
3	156	18		
20	543576	39		

Exercice 2 *Calcul de valeurs approchées par la méthode de Newton.*

1. Programmer une fonction $newton(p, A, x, n)$ qui retourne la liste `L` des coordonnées des points construits par une méthode de Newton en n étapes avec $x_0 = x$.
2. Écrire un programme permettant à l'utilisateur de saisir p, A, x et n et qui affiche la valeur approchée de α ainsi que le graphique correspondant (courbe de la fonction f_p et les points de la méthode de Newton). Complétez le tableau suivant :

p	A	x	n	Valeur approchée
3	156	1	10	
3	156	156	10	
3	156	156	11	
20	543576	543576	100	
20	543576	1	100	
20	543576	543576	200	
20	543576	1	200	

3. Commentez les résultats précédents.

Exercice 3 *Comparaison des deux méthodes.*

1. Programmer une fonction $compareNewtonDicho(p, A, n)$ qui retourne le nombre d'étapes nécessaires par dichotomie pour obtenir une valeur approchée supérieur de α au moins aussi fine qu'avec une méthode de Newton en n étapes initialisées avec $x_0 = 1$.
2. Compléter les tableaux suivants :

- Pour $p = 2$ et $A = 15$:

n	3	4	5	6
$compareNewtonDicho(p, A, n)$:				

- Pour $p = 5$ et $A = 7384$:

n	27	28	29	30
$compareNewtonDicho(p, A, n)$:				

3 Fusion des deux méthodes

On constate que si la méthode de Newton peut être plus lente sur les premières itérations, elle semble être beaucoup plus rapide que la dichotomie à partir d'un certain rang. Cette observation sera formalisée par la proposition 3 de la section suivante.
Pour optimiser notre recherche de racine, on va donc fusionner les deux méthodes, c'est à dire appliquer à chaque itération les deux méthodes et conserver la valeur approchée supérieure la plus fine.
Pour cela, on construit trois suites $(a_n)_{n\in\mathbb{N}}$, $(b_n)_{n\in\mathbb{N}}$ et $(x_n)_{n\in\mathbb{N}}$ de la façon suivante :

1. **Initialisation :**
   On pose $a_0 = 1$, $b_0 = A$, $x_0 = 1$.

2. **Hérédité :**
   Pour tout $n \in \mathbb{N}$:
   - On pose $c = \dfrac{a_n + b_n}{2}$ et $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$.
   - Si $\alpha > c$ on pose alors $a_{n+1} = c$, sinon on pose $b_{n+1} = c$.
   - Si $x_{n+1} < b_{n+1}$ on pose $b_{n+1} = x_{n+1}$, sinon on pose $x_{n+1} = b_{n+1}$.

Exercice 4 *Fusion des deux méthodes.*

1. Programmer la fonction $fusion(p, A, n)$ qui retourne une valeur approchée de α en n étapes en fusionnant la recherche par dichotomie et la méthode de Newton.
2. Programmer une fonction $compareFusionDicho(p, A, n)$ qui retourne le nombre d'étapes nécessaires par dichotomie pour obtenir une valeur approchée supérieure de α au moins aussi fine qu'avec la méthode de fusion en n étapes.
3. Programmer une fonction $compareFusionNewton(p, A, n)$ qui retourne le nombre d'étapes nécessaires avec la méthode de Newton pour obtenir une valeur approchée supérieure de α au moins aussi fine qu'avec la méthode de fusion en n étapes.
4. Compléter le tableau suivant pour $p = 7$ et $A = 10^{10} + 1$:

n	10	20	25	27	30	31	32
$compareFusionDicho(p, A, n)$							
$compareFusionNewton(p, A, n)$							

4 Vitesse de convergence de l'approximation de la racine carrée

On cherche dans cette section à établir le nombre d'étapes à effectuer pour obtenir une valeur approchée de α avec une erreur inférieure à une quantité donnée.
On remarque qu'avec la recherche par dichotomie initialisée sur $[1, A]$, on obtient en n étapes un encadrement de $a_n \leq \alpha \leq b_n$ avec $b_n - a_n = \dfrac{A-1}{2^n}$. **La vitesse de convergence de cette méthode est donc indépendante de p.**

Exercice 5 *Vitesse de convergence de l'approximation par dichotomie*

1. Programmer une fonction $vitesseDicho(A, n)$ qui retourne le nombre d'étape nécessaires pour obtenir par dichotomie à partir du segment $[1, A]$, une valeur approchée supérieur de α avec une erreur maximale de 10^{-n} .
2. Compléter le tableau suivant :

n	0	2	4	6	8	10
$vitesseDicho(4000, n)$						

Étudions à présent la vitesse de convergence de la méthode de Newton.

Proposition 2 *Pour tout $n \in \mathbb{N}$ et pour $p = 2$, $|x_n - \alpha| \leq \dfrac{A-1}{2^n}$.*
La méthode de Newton converge donc au moins aussi vite que la méthode par dichotomie.

Cela se prouve simplement par récurrence en le vérifiant pour $n = 0$ et $n = 1$ puis en remarquant grâce à la définition de la suite que pour tout $n \in \mathbb{N}^*$ (on rappelle qu'alors $\alpha \leq x_n$) on a $x_{n+1} - \alpha \leq \dfrac{x_n - \alpha}{2}$.

La proposition suivante permet de confirmer l'observation de la section précédente : la méthode de Newton converge plus vite que la dichotomie à partir d'un certain rang.

Proposition 3

- *Pour tout $n \in \mathbb{N}^*$ et pour tout $p \in [\![2, +\infty[\![$,*

$$|x_{n+1} - \alpha| < \frac{(p-1)A^{p-3+1/p}}{2}|x_n - \alpha|^2$$

La méthode de Newton est donc plus rapide que la dichotomie dès que $|x_n - \alpha| \leq \left(\dfrac{1}{(p-1)A^{p-3+1/p}}\right)^{1/2}$.

- *En particulier, pour $p = 2$ et pour tout $n \in \mathbb{N}^*$,*

$$|x_{n+1} - \alpha| < \frac{A^{-1/2}}{2}|x_n - \alpha|^2 \leq \frac{1}{2}|x_n - \alpha|^2$$

La méthode de Newton est alors plus rapide que la dichotomie dès que $|x_n - \alpha| \leq 1$

La démonstration de cette proposition sera faite dans la section suivante.

Exercice 6 *Vitesse de convergence de l'approximation par la méthode de Newton.*

Soit A un réel strictement supérieur à 1 et soit $(x_n)_{n\in\mathbb{N}}$ la suite d'approximations de $\alpha = \sqrt{A}$ obtenue par la méthode de Newton initialisée par $x_0 = 1$. On a donc :

- Pour tout $n \in \mathbb{N}$, $|x_n - \alpha| \leq \dfrac{A-1}{2^n}$
- Pour tout $n \in \mathbb{N}$, $|x_{n+1} - \alpha| < \dfrac{1}{2}|x_n - \alpha|^2$

1. A l'aide des inégalités précédentes, programmer une fonction $VitesseNewton(A, n)$ qui retourne le nombre d'étape nécessaires pour obtenir par la méthode de Newton une valeur approchée supérieur de α avec une erreur maximale de 10^{-n} .
2. Compléter le tableau suivant :

n	0	2	4	6	8	10
$vitesseNewton(4000, n)$						

5 Preuve de la proposition 3

Pour prouver cette proposition, nous aurons besoin du théorème suivant qui généralise l'égalité des accroissements finis :

Théorème 1 ***Formule de Taylor-Lagrange***
Soient a et b deux réels distincts et soit I l'intervalle fermé d'extrémités a et b (i.e. $I = [a, b]$ si $a < b$, $I = [b, a]$ sinon) et soit J l'intervalle ouvert d'extrémités a et b.
Soit $f : I \to \mathbb{R}$ une fonction de classe C^n sur I, $n+1$ fois dérivable sur J. Alors il existe $c \in J$ tel que :

$$f(b) = f(a) + \sum_{k=1}^{n} \frac{(b-a)^k}{k!} f^{(k)}(a) + \frac{(b-a)^{n+1}}{(n+1)!} f^{(n+1)}(c)$$

<u>Démonstration :</u>
Soit $\phi : I \to \mathbb{R}$ l'application définie pour tout $x \in I$ par :

$$\phi(x) = f(b) - f(x) - \sum_{k=1}^{n} \frac{(b-x)^k}{k!} f^{(k)}(x) - C\frac{(b-x)^{n+1}}{(n+1)!}$$

où C est une constante telle que $\phi(a) = 0$. Remarquons que l'on a aussi $\phi(b) = 0$.
ϕ étant continue sur I, dérivable sur J, il existe d'après le théorème de Rolle $c \in J$ tel que $\phi'(c) = 0$. De plus, on a pour tout $x \in J$:

$$\phi'(x) = -\frac{(b-x)^n}{n!} f^{(n+1)}(x) + C\frac{(b-x)^n}{n!}$$

On a donc $C = f^{(n+1)}(c)$. En écrivant alors $\phi(a) = 0$, on obtient bien le résultat souhaité.$\square$

Soient A un réel strictement supérieur à 1 et $p \in [\![2, +\infty[\![$. On pose $f(x) = x^p - A$ et $\alpha = A^{1/p}$, alors d'après la proposition 1, la suite $(x_n)_{n\in\mathbb{N}}$ définie par :

- $x_0 = 1$
- Pour tout $n \in \mathbb{N}$, $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$

est convergente vers α en décroissant à partir de x_1 et on a pour tout $n \in \mathbb{N}^*$, $1 < \alpha < x_n < A$.
Soit $n \in \mathbb{N}^*$, par définition de la suite $(x_n)_{n\in\mathbb{N}}$ et puisque $f(\alpha) = 0$ on a :

$$x_{n+1} - \alpha = x_n - \alpha - \frac{f(x_n)}{f'(x_n)} = x_n - \alpha + \frac{f(\alpha) - f(x_n)}{f'(x_n)}$$

f étant de classe C^2 sur $[\alpha, x_n]$, on obtient en appliquant la formule de Taylor-Lagrange à f :

$$f(\alpha) - f(x_n) = (\alpha - x_n)f'(x_n) + \frac{(\alpha - x_n)^2}{2} f''(\theta) \ , \ \theta \in]\alpha, x_n[$$

En reportant ceci dans l'égalité précédente on a donc :

$$x_{n+1} - \alpha = \frac{f''(\theta)}{2f'(x_n)}(\alpha - x_n)^2$$

En remarquant que $f''(\theta) < p(p-1)A^{p-2}$ et que $f'(x_n) > p\alpha^{p-1} = pA^{1-1/p}$ on obtient bien l'inégalité souhaitée.