Partie II : Calcul numérique

Recherche du zéro d'une fonction

Méthode 'Dekker'

Dans cette partie, on suppose que les modules **numpy** et **matplotlib.pyplot** sont importés :

```
import numpy as np
import matplotlib.pyplot as pl
```

Calculer les zéros d'une fonction réelle f (c'est-à-dire les racines de l'équation $(x) = 0$) est un problème que l'on rencontre très souvent en calcul scientifique. En général, cette tâche ne peut être effectuée en un nombre fini d'opérations. Par exemple, il n'existait pas de formule explicite donnant les racines d'un polynôme quelconque de très grand degré. La situation est bien sûr encore plus complexe quand f n'est pas un polynôme.

Pour résoudre ce problème, on utilise donc des méthodes itératives ou récursives : partant d'une ou plusieurs valeurs initiales, on construit une suite de valeurs x_n qui converge vers un zéro de la fonction f considérée.

Un problème simple et concret qui donne lieu à une équation non linéaire est celui de l'**équation d'état d'un gaz** : Nous voulons déterminer le volume ***V*** occupé par un gaz dont la température est ***T*** et dont la pression est ***p***. L'équation d'état, (liant ***p***, ***V*** et ***T***) est donnée par :

$$\left(\mathbf{p} + \mathbf{a}\frac{\mathbf{N^2}}{\mathbf{V^2}}\right) * (\mathbf{V} - \mathbf{N.b}) = \mathbf{k.N.T}$$

Avec :

- ***a*** et ***b*** sont deux coefficients qui dépendent du gaz considéré ;
- ***N*** est le nombre de molécules contenues dans le volume ***V*** ;
- ***p*** est la pression ;
- ***T*** est la température ;
- ***k*** est la constante de Boltzmann.

Nous devons donc résoudre une équation non linéaire dont la solution est ***V***. Pour cela, on doit calculer les zéros de la fonction g définit par :

$$g(v) = \left(p + \frac{a.N^2}{v^2}\right) * (v - N.b) - k.N.T$$

On suppose que les valeurs des paramètres sont les suivantes, (*pour le dioxyde de carbone CO_2*) :

a=0.40 , b=42.7 * 10^{-6} , N=10^3 , T=300 , p=3.5*10^7 et k=1.38*10^{-23}

Q.1- Écrire la fonction $g(v)$ en langage Python.

Q.2- Écrire le programme python qui permet de tracer la représentation graphique ci-dessous (Figure 1), de la fonction g. Le nombre de points générés dans la courbe est **500**.

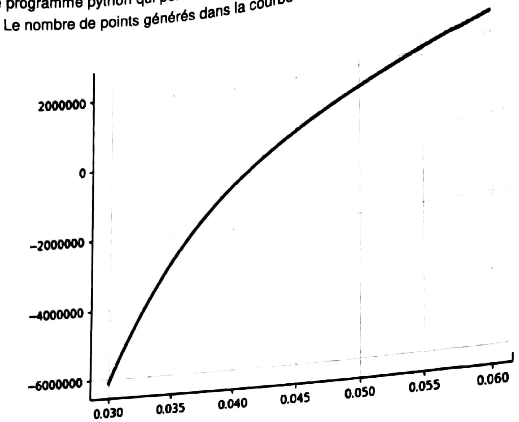

Figure 1

À partir de cette représentation graphique, on voit bien que la fonction g admet un zéro dans l'intervalle : **[0.040 , 0.045]**.

La méthode de dichotomie :

La méthode de la dichotomie (ou de la bissection) est un algorithme de recherche d'un zéro d'une fonction. Cette méthode consiste à répéter de partager un intervalle en deux parties, et sélectionner le sous-intervalle dans lequel existe le zéro de la fonction.

La méthode de la sécante :

La méthode de la sécante est un algorithme de recherche d'un zéro d'une fonction. Cette méthode est basée sur la relation de récurrence suivante :

Étant donnés deux réels différents a et b proches du zéro d'une fonction f :

$$c = b - f(b)\frac{b - a}{f(b) - f(a)}$$

Le nombre c est la valeur de l'abscisse du point d'intersection entre l'axe des abscisses et la droite passante par les deux points de coordonnées $(a, f(a))$ et $(b, f(b))$.

La méthode de 'Dekker' :

La méthode de **Dekker** est un algorithme de recherche d'un zéro d'une fonction, combinant les deux méthodes précédentes : la dichotomie et la sécante. L'idée d'allier ces deux méthodes différentes est due à *Theodorus Dekker (1969)*.

La méthode de Dekker est basée sur le principe suivant : À chaque itération, elle décide laquelle des deux méthodes (la dichotomie et la sécante) est susceptible d'approcher au mieux le zéro de la fonction, et effectue une itération en utilisant cette méthode. *L'idée principale est d'utiliser la méthode de la sécante parce qu'elle converge vite, et de revenir à la méthode de dichotomie si besoin est.*

Algorithme de la méthode de 'Dekker' :

Entrée : f est une fonction continue ayant un zéro dans un intervalle $[a\ ;\ b]$, telle que $f(a)$ *et* $f(b)$ ont des signes opposés, et **eps** est la précision désirée.

Tant que $f(b) \neq 0$ *et* $|b - a| > eps$ *(condition de convergence)*

- $m \leftarrow$ la valeur du milieu de a et b
- $c \leftarrow$ la valeur de la formule de la sécante passante par les deux points $(a, f(a))$ et $(b, f(b))$
- Si *d est compris* entre b et m, alors
  - $d \leftarrow c$
- Sinon
  - $d \leftarrow m$
- Fin si
- Si $f(b)$ *et* $f(d)$ ne sont pas de même signe, alors
  - $a \leftarrow b$.
- Fin si
- $b \leftarrow d$
- Si $|f(a)| < |f(b)|$ alors
  - échanger les valeurs de a et b
- Fin si

Fin tant que

Sortie : retourner b

Q.3- Écrire la fonction ***secante*** (f, a, b), qui reçoit en paramètres une fonction f et deux réels a et b tels que $f(a)$ *et* $f(b)$ ont des signes opposés. La fonction retourne la valeur c de l'abscisse du point d'intersection entre l'axe des abscisses et la droite passante par les deux points de coordonnées $(a, f(a))$ *et* $(b, f(b))$.

Q.4- Écrire la fonction ***dekker_racine*** (f, a, b, eps), qui reçoit en paramètres une fonction f continue ayant un zéro dans un intervalle $[a\ ;\ b]$ tellee que $f(a)$ *et* $f(b)$ ont des signes opposés, et un réel **eps** qui représente la précision. En utilisant l'algorithme de la méthode de Dekker (ci-dessus), la fonction retourne la valeur du zéro de la fonction f.

Q.5- Écrire le programme python qui permet d'afficher la valeur du zéro de la fonction g (voir Figure1), en utilisant la méthode de Dekker, sachant que la valeur de la précision choisie est : **eps** $= 10^{-12}$