Résolution d'équations non linéaires

Compétences

— Problème stationnaire à une dimension, linéaire ou non conduisant à la résolution approchée d'une équation algébrique ou transcendante à l'aide de la méthode de dichotomie ou de Newton

Pré-conditions

Pour résoudre numériquement une équation non linéaire de la forme $f(x) = 0$, il convient de :

- localiser grossièrement le ou les zéros de f, on notera x_0 cette solution grossière
- construire à partir de x_0 une suite telle que $\lim_{n\to\infty} x_n = x_r$ où $f(x_r) = 0$ (x_r est un zéro de f)

Localisation des zéros

Pour localiser le ou les zéros de f, il faut commencer par une étude des variations de la fonction f (je laisse cette partie au prof de math...) On supposera dans la suite que cette étude est déjà réalisée et que l'on dispose d'un intervalle $[a, b]$ sur lequel la fonction est strictement monotone et tel que $f(a) \times f(b) < 0$, on admet alors que f admet un zéro unique dans $]a, b[$

Construction d'une suite convergente

Une fois les zéros de f encadrés, tout réside dans la construction de la suite permettant d'approcher au plus juste et en moins d'itérations possibles la valeur de x_r cherchée. Pour cela il existe plusieurs méthodes numériques possédant chacune ses défauts et ses qualités.

Méthode de dichotomie et de Lagrange (ou Regula falsi)

Dans ces deux méthodes, à chaque itération on divise en deux un intervalle donné et on choisit le sous intervalle où f change de signe.

- Pour la méthode de dichotomie, on découpe l'intervalle $[a_i, b_i]$ en deux intervalles de même longueur $[a_i, c_i]$ et $[c_i, b_i]$:

$$c_i = \frac{a_i + b_i}{2}$$

- Pour la méthode de Lagrange (variante de la dichotomie), on considère le point d'intersection, de la corde joignant les points $(a_i, f(a_i))$ et $(b_i, f(b_i))$ avec l'axe des abscisses pour découper l'intervalle en deux sous intervalles de longueurs différentes. On obtient :

$$c_i = \frac{a_i f(b_i) - b_i f(a_i)}{f(b_i) - f(a_i)}$$

Données : *fonction :* f, *réels :* a, b, ϵ
g, d ← a, b
$f_g, f_d \leftarrow f(g), f(d)$
tant que (| d - g | > ϵ) **faire**
  m ← (d + g) / 2
  $f_m \leftarrow f(m)$
  si ($f_m \times f_g <= 0$) **alors**
    $d, f_d \leftarrow m, f_m$
  sinon
    $g, f_g \leftarrow m, f_m$
  fin si
fin tant que
affiche g

1. Implémenter l'algorithme ci-contre à l'aide d'une fonction dont le prototype est : `dichotomie(f, a, b, epsilon)`
2. Pour la définition d'une fonction on peut utiliser la méthode suivante : `def g(x): return x**2` mais Python offre la possibilité de définir des fonctions *anonymes* via l'opérateur **lambda** `lambda x: x**2`. Ainsi l'appel de la fonction dichotomie avec $f(x) = x^2 - 2$ peut se faire de la manière suivante : `dichotomie(lambda x: x**2 - 2, 1, 3, 1e-6)`
3. Déterminer sur votre feuille les 4 premiers itérés dans l'intervalle $[1, 3]$ avec la fonction de la question précédente.
4. Combien d'itérations doit-on effectuer pour améliorer d'un ordre de grandeur la précision de l'approximation de la racine ?
5. Discuter de la condition d'arrêt pour la fonction suivante : `dichotomie(lambda x : 0.0001*x-10000, 1e5, 1e15, 1e-8)`
6. Qu'advient-il si f ne s'annule pas sur [a, b] ? Placer dans votre code la gestion d'une exception de type **AssertionError** pour éviter cette anomalie.
7. Quel est l'invariant de boucle de cet algorithme ?
8. Proposer un algorithme dans le langage naturel de la méthode Lagrange ainsi qu'une implémentation.
9. Ajouter à chaque algorithme un compteur permettant d'afficher le nombre d'étapes nécessaires au calcul.
10. Vérifier que pour la dichotomie le nombre d'itérations est de l'ordre de : $\log_2 \frac{|b-a|}{\epsilon}$

Une méthode du point fixe particulièrement connue : la méthode de Newton

Le principe en est le suivant : partant d'une valeur x_0, on trace la tangente à la courbe représentative de f au point $M_0(x_0, f(x_0))$; cette droite coupe l'axe des abscisses en un point d'abscisse :

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Approcher les zéros de f se ramène au problème de la détermination des points fixes d'une fonction ϕ, en construisant la suite récurrente :

$$\begin{cases} x_{k+1} = \phi(x_k) \\ x_0 = \text{ valeur connue} \end{cases}$$

Méthode de Newton :

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

Imaginer que le seul outil de calcul à votre disposition soit une calculatrice possédant uniquement les opérations suivantes : addition, soustraction, multiplication, la division est absente. Pour un $a > 0$ on veut déterminer $1/a$, ce qui équivaut à chercher le zéro de la fonction $f: \begin{array}{l} \mathbb{R}_+^* \longrightarrow \mathbb{R} \\ x \longmapsto \frac{1}{x} - a \end{array}$

1. Appliquer la formule de Newton pour déterminer $\phi(x_k)$
2. Pour $a = 7$ et $x_0 = 0.2$ déterminer x_6 et comparer à la valeur donnée par une machine à calculer de type TI-83 : $1/7 \simeq 0.1428571429$

Newton en langage naturel :

```
Données : fonction : f, g (g est la dérivée de f)
          réels : x0, ε
x ← x0
y ← x - f(x) / g(x)
tant que ( | y - x | > ε) faire
    x ← y
    y ← y - f(y) / g(y)
fin tant que
affiche y
```

3. Implémenter l'algorithme ci-contre à l'aide d'une fonction dont le prototype est :
   `newton(f, g, a, epsilon)`
4. Tester votre implémentation avec $f(x) = 1/x - a$
5. Ajouter un compteur permettant d'afficher le nombre d'étapes nécessaires au calcul et comparer le nombre d'itérations obtenues avec Newton, dichotomie et LAGRANGE pour la fonction $f(x) = x^3 - x^2 + 8x - 8$ sachant que $x_r = 1$ est l'unique zéro réel de f (cf utilisation de `numpy/scipy`)
6. Donner une solution approchée de $f: \begin{array}{l} \mathbb{R}_+^* \longrightarrow \mathbb{R} \\ x \longmapsto 1 - x\ln(x) \end{array}$

Conclusion

Il n'y a pas vraiment de méthode universelle de résolution pour une équation du type $f(x) = 0$, chaque méthode a ses avantages et ses inconvénients dont il faut être conscient. Le choix sera fait en fonction des besoins : rapidité, connaissance de f' et/ou $f"$, d'une bonne approximation d'un zéro, etc. On peut pratiquer des méthodes mixtes avec une première phase de dichotomie pour approcher grossièrement un zéro puis Newton pour affiner le résultat.

Utilisation de `numpy/scipy`

numpy.roots détermine les racines d'un polynôme donné par la liste de ces coefficients :

```
>>> numpy.roots([1, -1, 8, -8])
>>> array([ -5.76741346e-16 + 2.82842712j,
            -5.76741346e-16 - 2.82842712j,
             1.00000000e+00 + 0.j])
```

scipy.optimize.bisect une implémentation de la méthode de dichotomie

```
>>> scipy.optimize.bisect(lambda x: 1e-4*x - 1e4), 1e5, 1e15|
>>> 100000000.04012266
```

scipy.optimize.newton une implémentation de la méthode de Newton, à noter que si on ne donne pas la dérivée c'est la méthode de la sécante (méthode non étudiée en TD mais que je conseille de regarder tout seul) qui est appliquée.

```
>>> scipy.optimize.newton(lambda x:1/x-7, 0.2, lambda x:-1/x**2)
>>> 0.14285714285714288
```

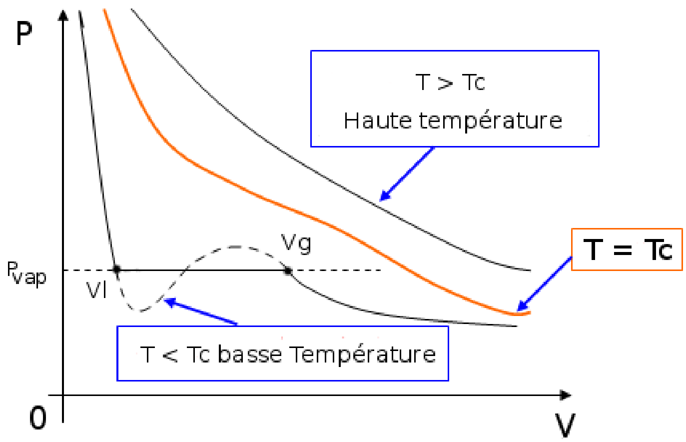
L'équation d'état de Van der Waals

L'équation d'état des gaz parfait décrit un gaz dont les molécules sont ponctuelles et non interagissantes. Cette équation décrit correctement les gaz monoatomiques et relativement bien les gaz polyatomiques à basse densité, mais à haute densité, l'écart par rapport à l'expérience devient important. Une meilleure approximation est obtenue vis-à-vis des gaz réels avec l'équation de Van der Waals qui tient compte du volume des molécules et introduit un terme d'interaction simple.

$$\left(P + \frac{an^2}{V^2}\right)(V - nb) = nRT$$

Isothermes des gaz de Van Der Waals

À haute température, l'isotherme ressemble à une portion d'hyperbole. En fait, la courbe de VAN DER WAALS rejoint celle de BOYLE-MARIOTTE pour les gaz parfaits. Au dessus de la température critique, les courbes tendent à se rapprocher de celles d'un gaz parfait, c'est pourquoi un gaz dont la température critique est très basse comme l'azote ou l'oxygène se comportent comme un gaz parfait à la température ambiante.
Au fur et à mesure que la température décroît, cette portion d'hyperbole se déforme peu à peu. Il apparaît progressivement une sorte de bosse qui fait place à un point d'inflexion horizontal à la température T = Tc.
À température plus basse on observe une déformation en forme de S (en pointillée) qui devient de plus en plus importante lorsque la température diminue fortement. Cette déformation découle directement de l'équation de Van Der Waals et n'est que théorique. Dans la pratique on observe un plateau représenté en trait plein.



Décrivons cette isotherme en partant des basses pressions et des grands volumes (gaz). Lorsque la pression augmente, il y a diminution du volume, jusqu'à une valeur Vg correspondant au début de la liquéfaction (équilibre entre le gaz et une première goutte de liquide : on dit que le gaz forme une vapeur saturante). On observe alors de B à A un plateau pour une valeur de $P = P^* = P_{vap}$ jusqu'à V_l où tout le gaz est liquéfié.

Problèmatique : Comment déterminer une valeur approchée de V_g et V_l à l'aide des méthodes de dichotomie ou de Newton ?

Données : Pour l'exemple on prend une isotherme de CO_2 à $T = 280\ K$ avec $a = 3.658\ bar.L^2.mol^{-2}$; $b = 0.0429\ L.mol^{-1}$; $P_{vap} = 41.6\ bar$; $R = 0.08315\ bar.L.mol^{-1}.K^{-1}$; $n = 1.00\ mol$