

TP Informatique : Résolution numérique d'équations

1 Un peu de chimie

En chimie, il est fréquent, lorsque l'on souhaite déterminer l'état d'équilibre d'une réaction, d'obtenir des équations ou des systèmes d'équations non-linéaires, qu'il est difficile, voire impossible, de résoudre à la main sans effectuer quelques hypothèses simplificatrices (qu'on vérifiera une fois le résultat obtenu !)

Ces équations non-linéaires peuvent cependant être résolues numériquement par un ordinateur, éventuellement sans simplification aucune, par des méthodes de recherche de racine telles la méthode de Newton.

Dans cette première partie, nous allons nous pencher sur deux cas concrets de chimie des solutions, et plus précisément ici de calcul de pH, et voir comment on peut efficacement déterminer le pH d'une solution ou la courbe d'un dosage.

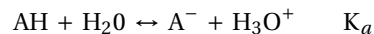


1.1 pH d'une solution d'acide faible

On s'intéresse à un acide faible, par exemple l'acide éthanóïque CH_3COOH (associé à la base faible que constitue l'ion éthanóate CH_3COO^-), dont le $\text{p}K_a$ est égal à 4.8.

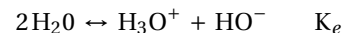
Dans la suite, on notera ce couple acide-base AH et A^- .

L'acide AH réagit avec l'eau selon la réaction suivante



où la constante d'équilibre K_a vérifie $K_a = 10^{-\text{p}K_a}$.

En solution aqueuse, une autre réaction est présente, celle de l'autoprotolyse de l'eau, dont la réaction est



On s'intéresse à une solution aqueuse de concentration c de cet acide faible, et on souhaiterait connaître son pH.

Pour ce faire, on aura besoin des constantes d'équilibres K_e et K_a et des lois d'action de masse qui leur sont associées (soit $Q_{\text{eq}} = K$). Lorsque les concentrations ne sont pas trop élevées, on peut assimiler l'activité d'une espèce chimique dissoute en solution aqueuse à sa concentration (divisée, pour donner une grandeur sans dimension, par une concentration de référence $c_0 = 1 \text{ mol/L}$). L'activité du solvant est quant à elle égale à 1.

Les concentrations des espèces dans la solution doivent donc satisfaire à deux équations :

$$\frac{[\text{A}^-][\text{H}_3\text{O}^+]}{[\text{AH}]c_0} = K_a \quad \text{et} \quad \frac{[\text{H}_3\text{O}^+][\text{HO}^-]}{c_0^2} = K_e$$

Par ailleurs, la conservation de la matière impose que $[\text{AH}] + [\text{A}^-] = c$, et la neutralité électrique que $[\text{H}_3\text{O}^+] = [\text{HO}^-] + [\text{A}^-]$.

Ces quatre équations permettent ainsi de déterminer les quatre inconnues que sont les quatre concentrations. Comme on s'intéresse au pH, on souhaitera tout particulièrement déterminer la concentration en H_3O^+ .

En résolvant le système des quatre équations, on en déduit que $[\text{H}_3\text{O}^+]$ est solution de l'équation

$$x^2 - K_e c_0^2 = K_a c_0 \left(c - x + \frac{K_e c_0^2}{x} \right)$$

1.a Définir les quantités K_a , K_e et c_0 .

1.b Proposer une fonction $f(x, c)$ qui s'annule lorsque x vérifie l'équation précédente, le c étant fourni en paramètre.

1.c Définir de même une fonction $f_{\text{prime}}(x, c)$, dérivée de la fonction précédente par rapport à x .

1.d Écrire une fonction $\text{pH}(c)$ qui, par la méthode de Newton, détermine le pH de la solution en fonction de la concentration c , fournie en paramètre.

2.a Tracer la courbe du pH de la solution en fonction de c pour des concentrations allant de $1 \times 10^{-7} \text{ mol/L}$ à $1,0 \text{ mol/L}$. On utilisera une échelle logarithmique pour l'axe des abscisses grâce à la commande `semilogx` de `matplotlib.pyplot`.

2.b Tracer sur le même graphique la fonction $\frac{1}{2}\text{p}K_a - \frac{1}{2}\log(c)$.

2.c Pour quelles valeurs de c cette dernière formule est-elle une bonne approximation du pH de la solution ?

1.2 Courbe de dosage

On souhaite à présent modéliser un dosage d'une solution de AH par de la soude (une base forte, $(\text{Na}^+, \text{HO}^-)$), et tracer l'évolution du pH au cours du dosage.

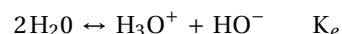
On place dans un bécher une solution de $V_0 = 2,0 \times 10^2 \text{ mL}$ de AH à la concentration $c_1 = 0,050 \text{ mol/L}$, et on remplit une burette avec $V_{\text{max}} = 25 \text{ mL}$ de soude à la concentration $c_2 = 1,0 \text{ mol/L}$. Le dosage consiste à verser progressivement la soude dans le bécher. On notera v le volume de soude versé.

Le volume dans le bécher, pour une quantité v de soude versée, est $V = V_0 + v$. On ne négligera pas ici les effets de dilution, puisque c'est l'ordinateur qui fera les calculs à notre place. On travaillera donc dans la suite avec la quantité de matière des différentes espèces, et non plus avec les concentrations.

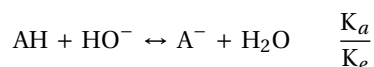
Cette fois encore, on considère les deux réactions (les ions sodium Na^+ étant ici spectateurs) :



et



Bien évidemment, si l'on s'intéresse à la réaction prépondérante, en début de dosage par exemple, il faudrait plutôt considérer celle faisant réagir AH avec HO^- :



Toutefois, cette réaction n'est qu'une combinaison des deux précédentes (c'est d'ailleurs ce qui permet de déterminer la constante d'équilibre de cette dernière réaction), et pour l'ordinateur qui cherchera ici à trouver l'équilibre final en considérant toutes les réactions possibles, cela n'aura pas d'importance.

Notons x le *nombre de moles* d'ions H_3O^+ dans la solution, à l'équilibre, après avoir versé un volume v de soude.

Notons de même y le nombre de moles de A^- . Le nombre de moles de HA est donc $c_1V_0 - y$ par conservation de la quantité de matière de A.

La loi d'action de masse associée à la constante d'équilibre de la première réaction impose donc

$$\frac{\frac{y}{V_0+v} \times \frac{x}{V_0+v}}{\frac{c_1V_0-y}{V_0+v} \times c_0} = K_a$$

où $c_0 = 1 \text{ mol/L}$ est la concentration standard.

Ce qui se réécrit en

$$x \times y = K_a \times (c_1V_0 - y) \times (V_0 + v) c_0 \quad (1)$$

L'écriture de la constante d'équilibre de la seconde réaction nécessite de connaître le nombre de moles de HO^- . Cela peut être obtenu en utilisant la neutralité électrique. En effet, le bécher contient c_2v moles de Na^+ , x moles de H_3O^+ , y moles de A^- . HO^- étant le seul ion restant, il y en a nécessairement $c_2v + x - y$.

La seconde réaction conduit donc à cette équation :

$$\frac{\frac{x}{V_0+v} \times \frac{c_2v+x-y}{V_0+v}}{c_0^2} = K_e$$

Autrement dit,

$$y = x + c_2v - \frac{K_e \times (V_0 + v)^2 \times c_0^2}{x} \quad (2)$$

En remplaçant y dans la première équation par cette expression (et en la multipliant par x pour éviter d'en trouver au dénominateur), on obtient finalement :

$$x^3 + (c_2v + K_a(V_0 + v)c_0)x^2 - K_aK_e(V_0 + v)^3c_0^3 + (K_a(V_0 + v)c_0c_2v - K_e(V_0 + v)^2c_0^2 - K_a(V_0 + v)c_0c_1V_0)x = 0 \quad (3)$$

Ce polynôme du troisième degré lie directement le nombre x de moles de H_3O^+ dans le bécher au volume v de soude versé.

3.a Définir les constantes V_0 , c_1 , c_2 , K_a , K_e et c_0 .

3.b Écrire une fonction $f(x, v)$ qui s'annule lorsque x et v respectent l'équation précédente.

3.c Écrire une fonction $f_{\text{prime}}(x, v)$ correspondant à la dérivée par rapport à x de la fonction précédente.

3.d Construire une liste ou un `numpy.array` désigné par `L_v` contenant une centaine de valeurs régulièrement réparties entre 0 et $25 \times 10^{-3} \text{ L}$.

On pourra utiliser, pour ce faire, la fonction `numpy.linspace(a, b, N)` qui construit un `numpy.array` contenant `N+1` valeurs régulièrement réparties entre `a` et `b`.

3.e Construire, avec une boucle ou une compréhension de liste, et en utilisant la méthode de Newton, la liste `L_x` des x correspondants.

On n'essaiera pas de ici de reprogrammer la méthode de Newton, on préférera utiliser la fonction `newton` du module `scipy.optimize` que l'on aura importé avec `from scipy.optimize import newton`. On pourra retrouver grâce à l'aide (`help(newton)`) les arguments attendus par la fonction.

Ici, la fonction f nécessite *deux* arguments, et non un seul. Il faut préciser à la fonction `newton` ce qu'elle doit utiliser comme second argument. Cela se fait, ainsi que le suggère l'aide, en plaçant l'argument `args=(v,)` parmi les arguments de la fonction `newton`, en dernière position.

3.f Construire la liste `L_pH` des pH pour chacun des volumes versés présents dans `L_v`, et tracer la courbe du pH en fonction de v . Que constate-t-on ? Pourquoi ?

3.g Déterminer pH_i et pH_f , le pH dans le bécher au début et à la fin du dosage.

Mais alors, comment les logiciels de simulation numérique tels que Simulitt parviennent à obtenir de belles courbes en évitant les problèmes de précision ?

En fait, ils rusent un peu. Plutôt que de déterminer le pH en fonction du volume versé v , ils déterminent le volume qu'il faut verser pour obtenir chacun des pH !

4.a Construire une liste ou un `numpy.array` désigné par `L_pH` contenant 200 valeurs de pH régulièrement réparties entre pH_i et pH_f .

4.b Proposer une fonction $f_2(v, \text{pH})$ qui s'annule lorsque le volume v versé donne un pH égal à pH .

4.c Utiliser la méthode de Newton pour déterminer, pour chaque pH présent dans L_{pH} , le volume v correspondant. On rangera le résultat dans une liste désignée par L_v . On ne se souciera pas de fournir la dérivée de f_2 à la fonction `newton`, on le laissera utiliser la méthode de la sécante à la place.

4.d Tracer L_{pH} en fonction de L_v . Qu'en pensez-vous?

5.a Construire une liste $L_{\text{der_pH}}$ contenant une estimation numérique de la dérivée de la fonction associant le volume versé v au pH, pour chacune des valeurs v dans L_v (attention, les valeurs de v ne sont pas ici régulièrement espacées), et tracer la courbe correspondante.

5.b Utiliser cette liste pour déterminer le volume équivalent, et le pH à l'équivalence. Est-ce conforme à ce que l'on pouvait attendre?

On remarquera que les valeurs de v sont nettement plus rapprochées au voisinage de l'équivalence, par construction, ce qui joue ici en notre faveur!

On rappelle que les concentrations des différentes espèces sont

$$[A^-] = \frac{y}{V_0 + v} \quad [AH] = \frac{c_1 V_0 - y}{V_0 + v} \quad [H_3O^+] = \frac{x}{V_0 + v} \quad [HO^-] = \frac{c_2 v + x - y}{V_0 + v} \quad [Na^+] = \frac{c_2 v}{V_0 + v}$$

6.a Construire les listes L_A , L_{AH} , L_{H_3O} , L_{OH} , L_{NA} , de même longueur que L_{pH} et L_v et qui indiquent les concentrations des différentes espèces dans la solution pour chaque pH/volume v versé.

6.b Se servir de ces listes pour tracer une courbe indiquant les titres en A^- et en AH en fonction du pH.

6.c De même, s'en servir pour tracer la courbe de la conductivité de la solution en fonction du volume v versé.

On donne les conductivités des différents ions présents dans la solution :

$$\lambda_{A^-} = 4,1 \text{ mS.m}^2/\text{mol} \quad \lambda_{H_3O^+} = 35 \text{ mS.m}^2/\text{mol}$$

$$\lambda_{HO^-} = 20 \text{ mS.m}^2/\text{mol} \quad \lambda_{Na^+} = 5,0 \text{ mS.m}^2/\text{mol}$$

2 Un peu de fractales

Dans le cours, nous avons vu que la recherche des racines de la fonction

$$f : x \mapsto f(x) = 1 - \frac{5}{\sqrt{x^2 + (x^2 - 1)^2}}$$

dont la dérivée est

$$f' : x \mapsto f'(x) = \frac{5(2x^2 - 1)x}{(x^4 - x^2 + 1)^{3/2}}$$

était parfois très sensible aux conditions initiales. Nous essaierons ici d'en savoir un peu plus.

La résolution de l'équation $f(x) = 0$ revient à rechercher les racines d'un polynôme du quatrième degré.

Nous avons vu en cours qu'il avait deux racines réelles x_1 et x_2 , égales à

$$\pm \sqrt{0.5(1 + \sqrt{97})} \approx \pm 2,329040339045$$

Si l'on factorise ce polynôme du quatrième degré en faisant apparaître $(x - x_1) \times (x - x_2)$, le polynôme du second degré qui apparaît a un discriminant négatif, aussi les deux autres racines du polynôme sont complexes (et conjuguées).

La méthode de Newton que nous avons vu en cours fonctionne tout aussi bien sur les complexes. Pour obtenir une racine complexe, il suffit simplement de partir d'une estimation initiale complexe pas trop loin de la racine recherchée.

1. Écrire une fonction `Newton(f, x0, fprime, eps, maxiter)` qui recherche une solution à l'équation $f(x) = 0$, avec les propriétés suivantes :

- On part de x_0 et on applique la méthode de Newton pour déterminer une suite de valeurs (x_k) ;
- L'algorithme s'arrête si $|x_{k+1} - x_k|$ est inférieur à la tolérance `eps` passée en paramètre, et retourne alors le couple (tuple) $x_{k+1}, k + 1$;
- L'algorithme s'arrête aussi si k dépasse la valeur `maxiter` (échec de convergence), auquel cas il retourne le couple `None, maxiter` ;
- L'algorithme s'arrête également si, pour un k quelconque, $|x_k|$ dépasse 10^{20} (divergence), auquel cas il retourne le couple `None, k`

Python manipule réels et complexes exactement de la même façon, il n'y a rien de particulier à changer dans l'algorithme.

2. Définir les fonctions $f(x)$ et $fprime(x)$. Comme on va travailler avec des complexes, on prendra soin d'utiliser la fonction `sqrt` du module `cmath` et non celle du module `math`.

3. Utiliser la fonction `Newton` pour trouver des estimations des racines complexes de l'équation $f(x) = 0$. On rappelle que le nombre complexe i s'écrit en Python `1j` (sans multiplication entre `1` et `j`), et vérifier qu'elles sont bien conjuguées (à la précision de l'algorithme près).

4. Écrire une fonction qui prend un argument complexe x_0 , utilise l'algorithme de Newton pour chercher une racine à partir de x_0 , et retourne 1, 2, 3 ou 4 si l'algorithme converge vers une des quatre racines, et 0 si l'algorithme ne converge pas (ou diverge).

5.a Construire une image couleur de taille 500, 500 représentant la partie du plan complexe $-3 \leq \operatorname{Re}(x) \leq 3$ et $-3 \leq \operatorname{Im}(x) \leq 3$, et pour laquelle la couleur d'un point donnée est noire, verte, jaune, rouge ou bleue selon que la fonction précédente, en prenant pour point de départ le complexe x correspondant à la position dans l'image, retourne 0, 1, 2, 3 ou 4.

5.b Afficher le résultat, et justifier que l'on obtient une fractale. Lorsque la fonction précédente marchera correctement, on augmentera la résolution de l'image produite afin qu'elle soit plus nette. On pourra également essayer de zoomer sur certaines zones intéressantes de la courbe en changeant les bornes des intervalles, pour vérifier l'auto-similarité de la figure obtenue.

Ce n'est pas un cas isolé, bien au contraire. On peut le voir en essayant bien des équations, par exemple $f(z) = z^3 - 1$ (dont les racines sont 1, j et $j^2 = -1$) de dérivée $f'(z) = 3z^2$.

De nombreux sites discutent de ce genre de phénomène, on pourra par exemple se reporter à la page image math.cnrs.fr/La-methode-de-newton-et-son-891.html