

## PROGRAMMER EN PYTHON

### FICHE N°2 : ÉVOLUTION DES QUANTITÉS DE MATIÈRE LORS D'UNE TRANSFORMATION CHIMIQUE

#### Capacité numérique

Déterminer la composition de l'état final d'un système siège d'une transformation chimique totale à l'aide d'un langage de programmation.

#### Présentation du script

Le langage Python est utilisé pour tracer l'évolution des quantités de matière des réactifs et produits lors d'une transformation chimique modélisée par une unique réaction dont l'équation est de la forme  $a_A A + a_B B = C + a_D D$ . La transformation est considérée totale. La détermination de la valeur maximale de l'avancement et l'identification du réactif limitant s'opèrent graphiquement à partir de la donnée des quantités de matière initiales. Le script de la fiche précédente est repris.

Remarque : si dans la rédaction de l'activité, des notations générales (espèces A, B, C et D) ont été retenues, il serait préférable de proposer aux élèves de modéliser des cas concrets. Par exemple, le cas de différents mélanges initiaux mettant en présence des ions thiosulfate et du diiode en phase aqueuse permet de confronter la prévision du réactif limitant par le code à l'observation expérimentale, la coloration prise par la solution en présence de diiode pouvant servir d'indicateur concernant la nature du réactif limitant.

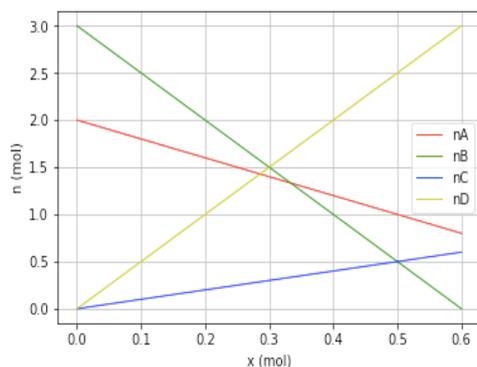
Sur le modèle du script précédent, il est proposé de créer une procédure pour tracer les évolutions des quantités de matière des différentes espèces. De fait, la syntaxe débute par « `def nom_procedure(arguments) :` ». Ce script peut être adapté pour demander à l'utilisateur de saisir les données, nombres stœchiométriques et quantités initiales au moyen de la commande `input`. Il est également possible de ne pas définir une procédure. Le script peut également être écrit sans définir une procédure. Dans ce cas, préalablement entrer les valeurs des variables ( $a_A, \dots, a_D, n_A, \dots, n_D$ ).

```
def evol_qt(aA, aB, aC, aD, nA, nB, nC, nD) :
    x=0 # Initialisation de l'avancement
    dx=0.001 # Incrément d'avancement
    X=[x] # Liste stockant les valeurs successives d'avancement
    NA=[nA] # Liste stockant les quantités des matières du réactif A
    NB=[nB] # Idem pour le réactif B
    NC=[nC] # Idem pour le produit C
    ND=[nD] # Idem pour le produit C
    while NA[-1]>0 and NB[-1]>0 :
        x=x+dx
        X.append(x)
        NA.append(nA-aA*x)
        NB.append(nB-aB*x)
        NC.append(nC+aC*x)
        ND.append(nD+aD*x)
    plt.figure(1)
    plt.plot(X, NA, 'r-', lw=1, label='nA')
    plt.plot(X, NB, 'g-', lw=1, label='nB')
    plt.plot(X, NC, 'b-', lw=1, label='nC')
    plt.plot(X, ND, 'y-', lw=1, label='nD')
    plt.grid(True)
    plt.xlabel('x (mol)')
    plt.ylabel('n (mol)')
    plt.legend()
```

### Analyse d'un exemple de courbe

Si l'on choisit un jeu de nombres stœchiométriques  $2 A + 5 B = C + 5 D$ , et pour des quantités de matière initiales ( $n_{A0} = 2 \text{ mol}$ ,  $n_{B0} = 3 \text{ mol}$ ,  $n_{C0} = 0 \text{ mol}$ ,  $n_{D0} = 0 \text{ mol}$ ) le script génère le graphique suivant :

```
evol_qt(2,5,1,5,2,3,0,0)
```



Sur l'exemple choisi, le réactif limitant est B : sa quantité de matière s'annule ce qui arrête l'évolution du système. La valeur maximale de l'avancement est 0,6 mol. Les quantités de matière finales des espèces dans l'hypothèse d'une transformation totale peuvent être lues directement sur le graphe à l'abscisse  $x = 0,6 \text{ mol}$ .

Retrouvez éducol sur :

