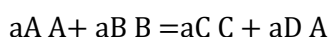


PYTHON POUR LA PHYSIQUE –CHIMIE en 1ère

FICHE n°2 : Réaction et évolution des quantités de matière

Capacité numérique mise en œuvre : Déterminer la composition de l'état final d'un système siège d'une transformation chimique totale à l'aide d'un langage de programmation.

On utilise ici le langage Python pour tracer l'évolution des quantités de matière de diverses espèces engagées dans une réaction chimique unique et considérée comme totale. Cela permet de déterminer graphiquement l'avancement maximal et d'identifier le réactif limitant à partir de la donnée des quantités de matière initiales pour une équation de réaction. On peut réutiliser le script précédent pour un système siège d'une réaction du type :

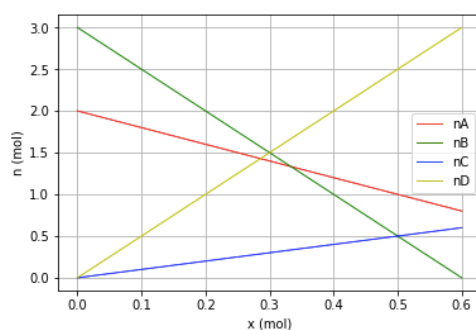


La syntaxe débute par « **def nom_procedure(arguments)** : ». On reprend **les mêmes idées** que précédemment, mais on utilise la fonction elle-même pour faire tracer les évolutions des quantités de matière des différentes espèces, grâce à des commandes **plt.plot**. Ici encore, on peut adapter le script de façon à demander à l'utilisateur de saisir les données, nombres stœchiométriques et quantités initiales. Après avoir appelé **matplotlib.pyplot**, on entre :

```
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 D  
    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()
```

Si l'on choisit un jeu de coefficients stœchiométriques et de concentrations initiales, le script renvoie le graphe ci-après :

In [5]: `evol_qt(2,5,1,5,2,3,0,0)`



Analyse : sur l'exemple choisi, le réactif limitant est donc B, et l'avancement maximal est 0.6mol. Les concentrations des autres espèces peuvent être lues directement sur le graphe en $x=0.6\text{mol}$.