



## SA - EXERCICE 20

# ETUDE D'UN PHOTOCHROME

[1. Ajustement sur une expérience \( \$\lambda\_{irr} = 366\$   
nm\)](#)

[2. Ajustement sur deux expériences \( \$\lambda\_{irr} = 366\$  et  
400 nm\)](#)

### 1. Ajustement sur une expérience ( $\lambda_{irr} = 366$ nm)

[Télécharger hv-k.cpp](#)   [Télécharger hv-k.sac](#)   [Télécharger photochrom\\_1irr.exp](#)

La programmation ne présente rien de nouveau. Vous trouverez les paramètres optimisés dans le fichier *hv-k.sac*. Le fichier expérimental *photochrom\_1irr.exp* est en mode *standard*.

A vous, bien entendu, de trouver des paramètres de départ acceptables pour une optimisation.

Vous pourrez vérifier que l'on retrouve bien toujours la même solution, quel que soit le point de départ.

Observez également les tracés des résiduels et les diagrammes de sensibilité globale : tout indique un ajustement excellent, et il l'est... mais cela ne prouve pas que le modèle est correct !

## 2. Ajustement sur deux expériences ( $\lambda_{\text{irr}} = 366$ et $400$ nm)

### Programmation

[Télécharger hv-hv-k.cpp](#)

Dressons d'abord la liste des paramètres nécessaires :

p[0]  $l$  (trajet optique, supposé identique pour l'irradiation et pour la mesure de l'absorbance)

p[1]  $\varphi_1$

p[2]  $\varphi_2$

p[3]  $k$

p[4]  $\varepsilon_A^{366}$  (exp 1)

p[5]  $\varepsilon_A^{400}$  (exp 2)

p[6]  $\varepsilon_A^{646}$

p[7]  $\varepsilon_B^{366}$  (exp 1)

p[8]  $\varepsilon_B^{400}$  (exp 2)

p[9]  $\varepsilon_B^{646}$

p[10]  $I_{0,1}$  (à 366 nm, exp 1)

p[11]  $I_{0,2}$  (à 400 nm, exp 2)

(seuls les paramètres en rouge sont inconnus et devront être ajustés)

Il y a ici une petite difficulté : les équations différentielles sont les mêmes pour les deux expériences, il paraît donc judicieux de n'écrire qu'une seule fois ces équations dans `eqdiff`, mais avec des paramètres  $I_0$ ,  $\varepsilon_A^{\text{irr}}$  et  $\varepsilon_B^{\text{irr}}$  différents. Il faut donc trouver un moyen de communiquer à `eqdiff` les paramètres corrects en fonction de l'expérience en cours de traitement. Pour cela, on déclare les variables *globales*  $I_0$ ,  $E_{\text{Airr}}$ ,  $E_{\text{Birr}}$  à l'extérieur de toute fonction, et il suffit ensuite de leur affecter les valeurs des paramètres correspondant à l'expérience (indice  $k$ ), dans `fappel` :

```
...
//-----
Sa_data I0, EAirr, EBirr;
//-----
void eqdiff (Sa_data x, Sa_data* y, Sa_data* dy)
{
    Sa_data r1, r2, r3, abs, Q;
    abs = p[0]*(EAirr*y[0] + EBirr*y[1]);
    Q = (1 - pow(10, -abs));
```

```

r1 = I0*p[1]*Q*p[0]*EAirr*y[0]/abs; // A hv-> B
r2 = I0*p[2]*Q*p[0]*EBirr*y[1]/abs; // B hv-> A
r3 = p[3]*y[1]; // B -> A
dy[0] = - r1 + r2 + r3; // dA/dt
dy[1] = r1 - r2 - r3; // dB/dt
}
//-----
void fappel()
{
    for (int i = 0; i < np; ++i)
        p[i] = fabs(p[i]);
    for (int k = 0; k < nexp; k++) {
        I0 = p[10+k];
        EAirr = p[4+k];
        EBirr = p[7+k];
        first_var = k*nv_mod;
        srkvi(n_diff, &ca[first_var], ind, npt, h0, tol, iset, jacob,
            h_compt, c_min);
        for (int j = 0; j < npt; ++j) {
            Ca(k,2,j) = p[0]*(EAirr*Ca(k,0,j)+ EBirr*Ca(k,1,j)); // abs1(irr)
            Ca(k,3,j) = p[0]*(p[6]*Ca(k,0,j) + p[9]*Ca(k,1,j)); // abs2
        }
    }
}
//-----

```

Notez l'emploi des variables locales `abs` et `Q` dans `eqdiff`, qui permettent d'éviter de recalculer plusieurs fois la même chose. Cela est particulièrement conseillé lorsque des fonctions telles que `pow`, `exp`, etc., relativement consommatrices de temps de calcul, sont impliquées.

Au lieu de déclarer de nouvelles variables globales `I0`, `EAirr`, `EBirr`, il aurait été possible d'utiliser exactement de la même manière, par exemple, `p[12]`, `p[13]`, `p[14]`, variables globales également et non utilisées par ailleurs. Cela fonctionnerait pareillement, mais la solution adoptée ici est plus lisible.

## Ajustement

[Télécharger hv-hv-k.sac](#)    [Télécharger photochrom\\_2irr.exp](#)

Le fichier *hv-hv-k.sac* contient les valeurs des paramètres ajustés ([figure VI.11](#)). Le fichier *photochrom\_2irr.exp* contient les données des deux expériences, en mode **multi**.

**A** - Essayez d'abord d'ajuster ces deux expériences sans la réaction de retour photochimique.

Il suffit pour cela de mettre  $\varphi_2$  (ou  $I_{0,2}$ ) à zéro, et d'ajuster  $\varphi_1$ ,  $\varepsilon_B^{366}$ ,  $\varepsilon_B^{400}$  et  $\varepsilon_B^{646}$ .

**B** - Ajustez ensuite avec  $\varphi_2$  ( $I_{0,2}$  étant à sa valeur correcte). Attention : **les valeurs de départ des paramètres ajustables doivent être différentes de zéro**. Essayez par exemple de démarrer avec  $\varphi_2 = 10^{-4}$ . Les paramètres ajustés ci-dessus doivent évidemment être réajustés dans cette nouvelle situation.

**C** - Vous pouvez également refaire, à titre d'exercice, l'ajustement sur une seule expérience avec cette version du programme. Deux méthodes sont possibles :

**1) en gardant le fichier *photochrom\_2irr.exp* :**

- onglet *Données exp.*, mettez *nexp* à 1 et confirmez
- onglet *Variables*, ne cochez les cases *obs* que des variables 2 et 3.

**2) en rechargeant le fichier *photochrom\_1irr.exp*, pour cela :**

- onglet *Variables*, cochez les cases *obs* des variables 2 et 3 (seulement) ; confirmez à chaque fois
- onglet *Données exp.*, mettez *nexp* à 1 (confirmez), et faites lire le fichier *photochrom\_1irr.exp* en mode *standard* (libérez le bouton *Multi*)

Une fois ces opérations effectuées, par l'une ou l'autre méthode, le programme n'effectuera le calcul et ne prendra en compte les données expérimentales que pour la première expérience. Vous ne devez bien sûr plus ajuster  $\varepsilon_B^{400}$ . Par contre, vous pouvez toujours utiliser et ajuster, avec cette version du programme, les deux rendements quantiques  $\varphi_1$  et  $\varphi_2$ .

En fixant  $\varphi_2$  à zéro, vous retrouverez exactement les conditions de l'[exercice 1](#).

En essayant d'ajuster  $\varphi_1$  et  $\varphi_2$  sur une seule expérience, vous observerez que l'optimisation converge "mollement" vers la solution obtenue avec deux expériences : le résultat dépend un peu du point de départ, de sorte qu'il serait difficile dans ces conditions de choisir une solution plutôt qu'une autre.