Les bases de la mesure spectropolarimétrique

Frédéric Paletou

Université de Toulouse, Observatoire Midi-Pyrénées, IRAP, 14 ave. E. Belin, F–31400 Toulouse, France (`frederic.paletou@univ-tlse3.fr`)

Ce document vous guidera dans la conduite du TP “construire un polarimètre”. Nous décrirons ici non seulement les manipulations à effectuer avec les fonctions Python définies et installées sur les machines mises à votre disposition mais nous ferons aussi les rappels élémentaires sur les matrices de Muller des retardeur, polariseurs et séparateurs de faisceaux (analyseurs), sur les notions de biréfringence et de chromatisme de phase etc.

1 Utilisation des outils numériques

Les programmes que vous allez utiliser sont écrits en langage Python ; nous utiliserons de plus les librairies `numpy/scipy` (manipulation de tableaux, algèbre linéaire etc.) et `matplolib` pour les graphismes. Ainsi, on démarrera toute session par les commandes suivantes, sous Python :

- import numpy as np
- import matplotlib.pyplot as plt
- import polar as pol

à moins de placer ces commandes dans un fichier disons `.python_startup` contenant ces instructions et d’indiquer ce fichier, avec son chemin, par la variable d’environnement PYTHONSTARTUP Les fonctions que nous allons utiliser se trouvent dans le fichier bibliothèque `polar.py`.

2 Matrice de Muller d’un analyseur

L’*analyseur* ou, le plus souvent, le séparateur de faisceaux constitue en général, derrière un ou une combinaison de *modulateurs*, la dernière pièce optique du polarimètre, avant la “chaîne” optique de transfert – spectrographe – détecteur(s). C’est, soit un polariseur linéaire simple rejetant l’un des faisceaux, soit plutôt un *séparateur de faisceaux* (p. ex. Wollaston ou lame de Savart) ajoutant une modulation spatiale à la modulation temporelle permise, s’ils sont orientables, par le(s) retardeur(s) placés en amont.

La matrice de Mueller d’un polariseur linéaire parfait d’azimut ψ s’écrit :

$$\mathcal{M}_{\rm p}(\psi) = \frac{1}{2}\begin{pmatrix} 1 & c & s & 0 \\ c & c^2 & cs & 0 \\ s & cs & s^2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{1}$$

avec les définitions suivantes : $c = \cos 2\psi$ et $s = \sin 2\psi$.

Un analyseur ou “séparateur de faisceaux” est représenté par une matrice de Mueller de la forme :

$$\mathcal{M}_{\pm} = \frac{1}{2}\begin{pmatrix} 1 & \pm 1 & 0 & 0 \\ \pm 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} . \tag{2}$$

3 Fonctions de base

On trouvera dans le fichier `polar.py` :

- deux fonctions permettant de manipuler les matrices de Muller de déphaseurs de retard δ quelconque et d’azimut ψ quelconque ;
- trois fonctions permettant de simuler des polarimètres simples à un puis deux modulateurs ;
- une fonction permettant de simuler le chromatisme de phase de lames cristallines composites “achromatiques”.

3.1 matphase

Calcul de la matrice de Muller d’un retardeur de phase δ. Un retardeur d’azimut nul est caractérisé par une matrice de Mueller de la forme :

$$\mathcal{M}_{\delta} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\delta & \sin\delta \\ 0 & 0 & -\sin\delta & \cos\delta \end{pmatrix} , \tag{3}$$

où δ est le retard de phase en radians.

Usage : `pol.matphase(phase)` où `phase` est donné en degrés.

3.2 matrot

Maintenant, si on a la possibilité de faire tourner la lame d’un azimut ψ quelconque, il faudra considérer l’expression générale de la matrice de Mueller du retardeur :

$$\mathcal{M}_{(\delta,\psi)} = \mathcal{R}_{-\psi}\mathcal{M}_{\delta}\mathcal{R}_{\psi} , \tag{4}$$

où les matrices de rotation $\mathcal{R}_{\pm\psi}$ sont définies comme suit :

$$\mathcal{R}_{\pm\psi} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 2\psi & \pm\sin 2\psi & 0 \\ 0 & \mp\sin 2\psi & \cos 2\psi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \tag{5}$$

Usage : `pol.matrot(psi)` où `psi` est donné en degrés.

3.3 l2mod

Cette fonction calcule à partir de l'Eq. (4) les coefficients de modulation x_q, x_u et x_v sur un intervalle $[0, \pi]$ de l'azimut pour une lame demi-onde parfaite ($\delta = 180^o$).

Usage : `pol.l2mod(npts)` où `npts` est le nombre de points calculés. La fonction appellée p. ex. de la façon suivante `x=pol.l2mod(100)` retourne 4 vecteurs x$[k]$ de taille `npts` contenant, respectivement, les 100 valeurs de l'azimut ($k = 0$), x_q ($k = 1$), x_u ($k = 2$) et x_v ($k = 3$).

3.4 l4mod

Idem à `l2mod` mais pour une lame quart d'onde ($\delta = 90^o$).

3.5 th_pol

Cette fonction calcule la matrice de Mueller du polarimètre de THéMIS[1]. Ce dernier est constitué de deux lames quart d'onde "achromatiques" indépendament orientables et supposées de retards identiques.

Usage : `pol.th_pol(theta1, theta2, retard)` où `theta1` est l'azimut de la première lame quart d'onde dans le chemin optique.

3.6 phase_400700

Cette fonction calcule le retard en fonction de la longueur d'onde d'une lame quart d'onde composite (MgF_2 et SiO_2) dite "achromatique" telle que celles utilisées au TBL avec MUSiCoS où encore à THéMIS.

Usage : `pol.phase_400700(londe)` où `londe` est donné en Å.

[1] `www.themis.iac.es`

4 EXPÉRIENCES

Si l'on considère k modulateurs orientables empilés suivis par un séparateur de faisceaux, on aura affaire à un polarimètre ayant une matrice de Muller complète de la forme :

$$\mathcal{M}_{\text{polarim.}} = \mathcal{M}_{\pm}\mathcal{M}_{(\delta_k,\psi_k)} \cdots \mathcal{M}_{(\delta_1,\psi_1)} \,. \tag{6}$$

4.1 Effet de l'analyseur

Les détecteurs sont uniquement sensibles à *Stokes I* c-à-d à l'énergie de l'onde électromagnétique détectée. Vérifiez qu'il suffit de calculer les coefficients $i = 2$ et $j = 2, 3, 4$ (i.e., x_q, x_u et x_v) de la combinaison de matrices décrivant les modulateurs orientables pour déterminer quelle combinaison de paramètres de Stokes l'on va mesurer.

4.2 Polarimètre constitué d'une lame demi-onde orientable

Déterminez les périodes de modulation de Q et U et constatez qu'un tel polarimètre ne permet pas la mesure de V (cf. fonction `l2mod`).

4.3 Polarimètre constitué d'une lame quart-onde orientable

Déterminez les périodes de modulation de Q, U et V ; constatez que l'on n'a pas accès à une mesure parfaite de U avec un tel dispositif (cf. fonction `l4mod`).

4.4 Le polarimètre de THéMIS

Il est constitué de deux lames quart d'onde (à peu près) identiques orientables indépendament l'une de l'autre (cf. fonction `th_pol`).

Quelles combinaisons d'azimuts permettent l'analyse successive de Q, U puis V ?

Quels couples d'azimuts doit-on successivement assigner au polarimètre pour effectuer une mesure précise de V par "échange de voies" ?

4.5 Chromatisme de phase

Visualisez les écarts à la phase d'une lame quart d'onde parfaite pour une lame cristalline composite "achromatique" (i.e., quart d'onde à 400 et 700 nm).

4.6 Cross-talk

En tenant compte du chromatisme de phase calculé précédemment (et supposé identique pour les deux lames), quelle combinaison de paramètres de Stokes mesure-t-on à 854.2 nm avec le polarimètre de THéMIS et le couple d'azimuts (-45°, 0°) ?

5 Manuel super-résumé de l'utilisateur

- `machin=pol.nom_de_la_fonction(`*arguments*`)` permet appeller les fonctions du *package* `polar`
- avec la bibliothèque `numpy`, la multiplication de deux matrices a $\times$ b s'effectue avec la commande : `np.dot(a,b)`
- la commande `lambda=4000.  + np.arange(100)*5000./(100 - 1)` permet de créer p. ex. un tableau de 100 valeurs comprises entre 4000 et 9000 Å
- pour faire un graphique, taper `plt.plot(x,y)` puis activer le GUI de matplotlib avec la commande `plt.show()`