

TP de physique n° 7

Polarisation : loi de Malus

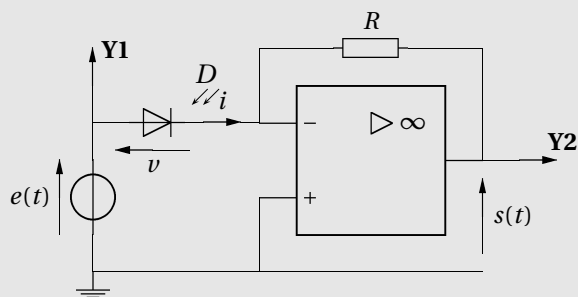
1 — Étude de la photodiode

Sur le site <https://cahier-de-prepa.fr/psi-perrin/>, dans la rubrique TP/TP-PHY-7, télécharger les fichiers caractéristique.py et malus.py sur le répertoire de travail de votre clef USB.

Les données de l'oscilloscope seront enregistrées sur ce même répertoire.

1.1 Caractéristique de la photodiode

Réaliser le montage suivant :



- Le générateur est un GBF délivrant une tension sinusoïdale comprise entre -10 V et $0,5\text{ V}$ (utiliser l'offset), de fréquence $f \approx 100\text{ Hz}$.
- La borne rouge de la diode est reliée au GBF et la borne noire à l'ALI.
- On utilisera une boîte de résistances à décade, réglée sur la valeur $R = 50\text{ k}\Omega$.

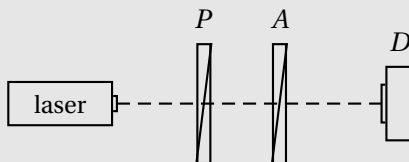
1. Indiquer les réglages de l'oscilloscope à effectuer afin d'observer directement la caractéristique de la photodiode sur l'écran de l'oscilloscope. Réaliser l'observation.

2. La loi $i = f(v)$ pour la photodiode est

$$i(v) = I_0(e^{v/a} - 1) - I_p.$$

Représenter la courbe $i(v)$.

Réaliser la partie optique du montage :



- L'analyseur A est un polariseur de la même nature que le polariseur P ; c'est sa position en aval du polariseur qui lui confère la dénomination d'analyseur.
- On veillera à ce que le faisceau laser éclaire bien la surface sensible de la photodiode.

3. Le polariseur P étant fixe, faire varier l'orientation de l'analyseur A , et observer les conséquences sur la caractéristique de la photodiode.

En déduire le paramètre qui dépend de l'éclairement lumineux dans l'expression de $i(v)$.

On admettra pour la suite que cette dépendance est linéaire.

Effectuer les réglages suivants :

- Régler la fréquence du GBF à la valeur $f = 20\text{ Hz}$.
- L'oscilloscope étant en mode bi-courbe, régler la base de temps afin d'observer une période pour chacun des signaux sur l'écran.

4. Pour des angles entre le polariseur et l'analyseur de 0° , 30° , 60° et 90° , enregistrer les courbes des voies I (tension $e(t)$) et II (tension $s(t)$) sur votre clef USB au format CSV.

- Les signaux étant bruités, on fera une moyenne sur 128 mesures (cf. fiche méthode).

Renommer les fichiers comme indiqué :

Angle θ	0°	30°	60°	90°
Voie I	F0001CH1.CSV	F0002CH1.CSV	F0003CH1.CSV	F0004CH1.CSV
Voie II	F0001CH2.CSV	F0002CH2.CSV	F0003CH2.CSV	F0004CH2.CSV

Ouvrir le fichier `caracteristique.py` et l'utiliser pour tracer les caractéristiques, aux différents éclairage, de la photodiode et pour réaliser l'ajustement des courbes sur la loi fournie.

5. Commenter les valeurs des paramètres I_0 , a et I_p retournés par le programme, au vu de la question 3.

2 — Loi de Malus

L'intensité lumineuse à la sortie de l'analyseur dont l'axe fait un angle θ avec l'axe du polariseur est donnée par la loi de Malus :

$$I(\theta) = I_0 \cos^2 \theta.$$

Dans la pratique :

- il existe toujours une luminosité ambiante tombant sur le récepteur;
- l'angle nul n'est pas repéré avec précision (défaut sur l'origine des graduations des supports).

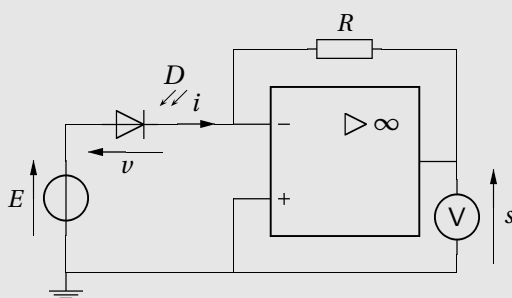
Nous utilisons donc la loi de Malus sous la forme

$$I(\theta) = I_1 + I_0 \cos^2(\theta - \theta_0).$$

2.1 Étude expérimentale

6. La diode étant alimentée par une générateur de tension continue E , quelle valeur de E choisir pour que la mesure de l'intensité dans la photodiode permettent de déterminer l'intensité lumineuse qu'elle reçoit?

Réaliser le montage suivant :



- Le générateur délivre une tension $E \approx -7$ V continue.
- On utilisera une boîte de résistances à décade, avec $R = 50$ k Ω .
- La tension continue s est mesurée au multimètre.

7. Relever la valeur de s pour diverses valeurs de θ (on fera varier de -90° à 90° par pas de 5°).

On pourra entrer les valeurs directement dans le fichier `malus.py`.

8. Compléter les ligne 6, 19 et 20 du code de la procédure `malus`.

Exécuter le code et commenter la courbe obtenue.

2.2 Exploitation des mesures

La tension s mesurée étant proportionnelle au flux lumineux Φ reçu par la diode, la loi de Malus conduit à

$$s = s_0 + s_0 \cos^2(\theta - \theta_0). \quad (1)$$

9. On pose $x = \cos^2(\theta - \theta_0)$ et $y = \frac{s - s_1}{s_0}$. À quelle relation se ramène la loi de Malus?

10. À partir de la relation $x = \cos^2(\theta - \theta_0)$, relier l'incertitude type $u(x)$ à l'incertitude type $u(\theta)$.

De même, relier l'incertitude type $u(y)$ à l'incertitude type $u(s)$.

11. Compléter les lignes du code indiquées¹ « à compléter ». Tracer la courbe $y(x)$ et discuter du résultat obtenu. Expliquer la ligne 49 du code.

- Les courbes sont enregistrées au format pdf sur votre clef USB par le programme.

1. Lignes 6, 19, 20, 40, 41, 43 et 44 du code page 4.

Code fichier caracteristique.py

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from scipy.optimize import curve_fit
4
5 def diode(v,I0,Ip,a):
6     i =
7     return i
8
9 R = 50e3
10
11 fichiers = [{"F0001CH1.CSV","F0001CH2.CSV"}, {"F0002CH1.CSV","F0002CH2.CSV"},
12             {"F0003CH1.CSV","F0003CH2.CSV"}, {"F0004CH1.CSV","F0004CH2.CSV"}]
13
14 plt.close()
15
16 for fichier in fichiers:
17     data_CH1 = np.genfromtxt(fichier[0],delimiter=",")
18     data_CH2 = np.genfromtxt(fichier[1],delimiter=",")
19     v = data_CH1[:,4]
20     i = data_CH2[:,4]/R
21     # tracé caractéristique, i en micro ampère
22     plt.plot(v,1e6*i,"+")
23     # ajustement de la courbe
24     popt,pcov = curve_fit(diode,v,i,p0=None)
25     plt.plot(v,1e6*diode(v,*popt))
26     print(popt)
27
28 plt.grid()
29 plt.xlabel("$v$ (V)")
30 plt.ylabel("$i$ ($\mu$A)")
31
32 plt.savefig("caracteristique-photodiode.pdf")
33
34 plt.show()
```

Code fichier malus.py

```

1 import numpy as np
2 from scipy.optimize import curve_fit
3 import matplotlib.pyplot as plt
4
5 def malus(a,a0,s_0,s_1):
6     s = # à compléter
7     return s
8
9 # angles entre le polariseur et l'analyseur, en degrés
10 theta_deg = np.array([-90,-85,-80,-75,-70,-65,-60,-55,-50,-45,-40,-35,-30,-25,-20,-15,
11     -10,-5,0,5,10,15,20,25,30,35,40,45,50,55,60,65,70,75,80,85,90])
12
13 # tensions mesurées
14 s = np.array([
15     ]) # à compléter
16
17 # angles en radian
18 theta = theta_deg*np.pi/180
19 u_theta = *np.pi/180 # incertitude type sur les angles - à compléter
20 u_s = # incertitude type sur les tensions - à compléter
21
22 popt, pcov = curve_fit(malus, theta, s ,p0=None)
23 theta_0, s_0, s_1 = popt
24 theta_cont = np.linspace(np.min(theta), np.max(theta), 500)
25
26 # tracé de la courbe s=f(\theta)
27 plt.figure(figsize=(11.69,8.27),tight_layout=True)
28 plt.grid()
29 plt.errorbar(theta, s, xerr=u_theta, yerr=u_s, fmt='+',label="Mesures")
30 plt.plot(theta_cont, malus(theta_cont,*popt), label="Ajustement")
31 plt.xlabel(r"$\theta$")
32 plt.ylabel(r"s/V")
33 plt.legend()
34 plt.savefig("malus-courbe.pdf")
35 plt.show()
36
37 # Décommentez les lignes suivantes après avoir tracé la première courbe
38
39 ## tracé de la courbe y=f(x)
40 #y = # à compléter
41 #u_y = # à compléter
42
43 #x = # à compléter
44 #u_x = # à compléter
45
46 #plt.figure(figsize=(11.69,8.27),tight_layout=True)
47 #plt.grid()
48 #plt.errorbar(x, y, xerr=u_x, yerr=u_y,fmt='o')
49 #plt.plot([0,1],[0,1],color='orange')
50 #plt.xlabel(r"$\cos^2(\theta-\theta_0)$")
51 #plt.ylabel(r"$\frac{s-s_1}{s_0}$")
52 #plt.savefig("malus-test.pdf")
53 #plt.show()

```