
[O] TP N°1 – LOIS DE SNELL-DESCARTES (2H)

Capacités exigibles dans ce TP

- Procéder à l'évaluation d'une incertitude-type par une approche statistique (évaluation de type A)
- Utiliser la fonction `np.random.normal` pour simuler un processus aléatoire
- Utiliser la fonction `np.polyfit` pour effectuer une régression affine

Dans ce TP, nous allons vérifier la validité des lois de Snell-Descartes, mesurer l'indice optique du plexiglas et observer le phénomène de réflexion total.

I) Rappels théoriques

Le rayon réfléchi et le rayon réfracté appartiennent au plan d'incidence. Les angles d'incidence et de réfraction sont donnés par :

$$i_1 = i'_1 \quad \text{et} \quad n_1 \sin(i_1) = n_2 \sin(i_2)$$

Le phénomène de réflexion totale se produit lorsque :

$$n_1 > n_2 \quad \text{et} \quad i_1 > i_{1,\text{lim}} = \arcsin\left(\frac{n_2}{n_1}\right)$$

II) Mesures expérimentales

1) Détermination de l'indice du plexiglas

- 🔧 Réaliser le montage tel qu'illustré ci-contre.
- 🔧 Faire varier l'angle d'incidence i de 5° à 75° par pas de 5° . Mesurer l'angle de réfraction r correspondant. Sauvegarder toutes les mesures dans un fichier Python.

Traitement des incertitudes de type A

- 📏 Calculer l'indice optique n pour l'ensemble des mesures.
- 📏 Calculer la valeur moyenne de cette série de mesure et l'incertitude-type sur cette valeur moyenne. Exprimer le résultat avec un nombre adéquat de chiffres significatifs.

Régression linéaire

On pose : $y = \sin(i)$ et $x = \sin(r)$.

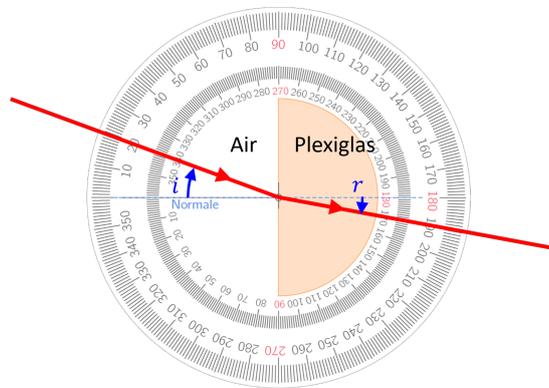
- 📏 Utiliser la fonction `polyfit` de la bibliothèque `numpy` pour effectuer une régression affine $y = ax + b$. L'indice n obtenu est-il dans l'intervalle précédemment trouvé ?

2) Détermination de l'angle limite entre de plexiglas et l'air

Nous allons nous servir des mesures précédentes pour prédire la valeur de l'angle limite i_{lim} et de son incertitude-type associée, d'un rayon lumineux se propageant du plexiglas vers l'air.

Simulation de Monte Carlo

- 📏 Sur Python, générer un tirage aléatoire (avec une loi normale) d'un grand nombre de valeurs de n de valeur moyenne et d'écart-type déterminés précédemment.



- ☐ Calculer les angles limite correspondant. Calculer la valeur moyenne et l'incertitude-type. Exprimer le résultat avec un nombre adéquat de chiffres significatifs.

Vérification expérimentale

- ⚙️ Inverser le sens de passage de la lumière : du plexiglas vers l'air. Observer la réflexion totale lorsque i est suffisamment grand.
- ⚙️ Observer la cas de la réfraction limite. Mesurer i_{lim} . La valeur obtenue est-elle dans l'intervalle précédemment trouvé ?

AIDE POUR PYTHON

`import numpy as np` permet d'importer l'ensemble des fonctions du module numpy.

`len(u)` renvoie le nombre d'éléments contenus dans `u`.

`np.array(u)` crée un tableau numpy contenant les éléments de la liste `u`.

`np.arange(a, b, p)` crée un tableau numpy contenant les valeurs comprises entre `a` (inclus) et `b` (exclu), régulièrement espacées du pas `p`.

`np.linspace(a, b, N)` crée un tableau numpy contenant `N` valeurs régulièrement espacées et comprises entre `a` (inclus) et `b` (inclus).

`a, b = np.polyfit(x, y, 1)` stocke dans les variables `a` et `b` le résultat de la régression affine $y = ax + b$.

`np.random.normal(moy, std, N)` crée un tableau numpy contenant `N` valeurs tirées aléatoirement selon une loi normale de valeur moyenne `moy` et d'écart-type `std`.

`np.mean(u)` et `np.average(u)` calculent la valeur moyenne du tableau numpy `u`.

`np.std(u, ddof=1)` calcule l'écart-type du tableau numpy `u`.

`np.sqrt(x)` calcule la racine carrée du nombre `x`.

`np.sin(x)` renvoient le sinus de l'angle `x` exprimé en radians.

`np.arcsin(x)` renvoient l'arcsinus du nombre `x`.