

Mesures et incertitudes

Sommaire

I Variabilité et incertitude-type	2
I/A Variabilités de mesures	2
I/B Demi-largeur et incertitude-type	2
I/C Comparaison de deux mesures	3
II Méthodes analytiques d'estimation d'incertitudes	4
II/A Estimation pour une mesure invariable : type B	4
II/B Estimation pour une série de mesures : type A	5
II/C Incertitudes composées	6
III Méthode numérique : simulations MONTE-CARLO	7
III/A Principe	7
III/B MONTE-CARLO sur un scalaire	8
III/C MONTE-CARLO sur un vecteur	9

Capacités exigibles

- Identifier les incertitudes liées, par exemple, à l'opérateur-ice, à l'environnement, aux instruments ou à la méthode de mesure.
- Associer un intervalle de confiance à l'écart-type dans l'hypothèse d'une distribution suivant la loi normale.
- Écrire, avec un nombre adapté de chiffres significatifs, le résultat d'une mesure.
- Procéder à l'évaluation d'une incertitude-type par une approche statistique (type A).
- Procéder à l'évaluation d'une incertitude-type par une approche autre que statistique (type B).
- Évaluer l'incertitude-type d'une grandeur s'exprimant en fonction d'autres grandeurs, dont les incertitudes-types sont connues, à l'aide d'une somme, d'une différence, d'un produit ou d'un quotient.
- Comparer deux valeurs dont les incertitudes-types sont connues à l'aide de leur écart normalisé.
- Analyser les causes d'une éventuelle incompatibilité entre le résultat d'une mesure et le résultat attendu par une modélisation
- Simuler, à l'aide d'un langage de programmation ou d'un tableur, une simulation MONTE-CARLO pour évaluer l'incertitude sur les paramètres du modèle.

L'essentiel

Définitions

- N2.1 : Valeurs possibles et demi-largeur 2
- N2.2 : Incertitude-type et relative 2
- N2.3 : Écart normalisé 3
- N2.4 : Écart absolu et relatif 4
- N2.5 : Simulations MONTE-CARLO 7

Applications

- N2.1 : Présenter un résultat numérique 3
- N2.2 : Incertitudes de type B 4
- N2.3 : Incertitude de type A 5
- N2.4 : Incertitude composée à une variable 6
- N2.5 : Incertitude composée à 2 variables 6

Interprétations

- N2.1 : Écart normalisé 3
- N2.2 : Incertitude de type A 5
- N2.3 : Cœur de la simulation 7

Points importants

- N2.1 : Présentation d'un résultat de TP 3
- N2.2 : Incertitude de type B 4
- N2.3 : Incertitude de type A 5
- N2.4 : Incertitudes composées à 1 variable 6
- N2.5 : Incertitudes composées à 2 variables 6
- N2.6 : MONTE-CARLO scalaire 8
- N2.7 : MONTE-CARLO vecteur 9

Erreurs communes

- N2.1 : Valeurs possibles d'une mesure 2
- N2.2 : Écart-types automatiques 5
- N2.3 : Distribution uniforme ou gaussienne ? 7
- N2.4 : Δ ou u pour MONTE-CARLO vecteur 9

I Variabilité et incertitude-type

I/A Variabilités de mesures

Une expérience de mesure en science expérimentale est un processus généralement complexe qui entremêle de très nombreux processus. Cette complexité se traduit systématiquement par une variabilité de la mesure, qui implique que la répétition de l'ensemble de la mesure conduit généralement à une valeur mesurée sensiblement différente de la première. Cette variabilité est naturelle et fait intrinsèquement partie de la mesure. Il ne faut pas chercher à la faire disparaître, bien au contraire, elle renferme généralement une grande richesse d'information sur le processus physique !

Cette variabilité peut provenir de nombreux aspects, dont les principaux sont les suivants :

- ◇ la méthode de mesure : règle graduée ou pied à coulisse pour une longueur ;
- ◇ les variations de l'environnement : célérité du son différente s'il fait chaud ou froid ;
- ◇ les instruments de mesure : deux voltmètres *a priori* identiques peuvent donner des résultats différents ;
- ◇ le processus physique même : expériences de mécanique quantique, par essence probabiliste ;
- ◇ l'expérimentataire.

I/B Demi-largeur et incertitude-type

Définition N2.1 : Valeurs possibles et demi-largeur

Dans beaucoup de cas, il est possible d'estimer le **plus petit intervalle** dans lequel on est certain-e trouver **toutes les valeurs possibles** de x . On aura alors $x_{\text{exp}} = \bar{x}$ la valeur centrale de cette plage et Δ sa demi-largeur.

Elle traduit qu'il y a **100% de probabilité** de trouver la valeur mesurée dans l'intervalle $[\bar{x} - \Delta, \bar{x} + \Delta]$.

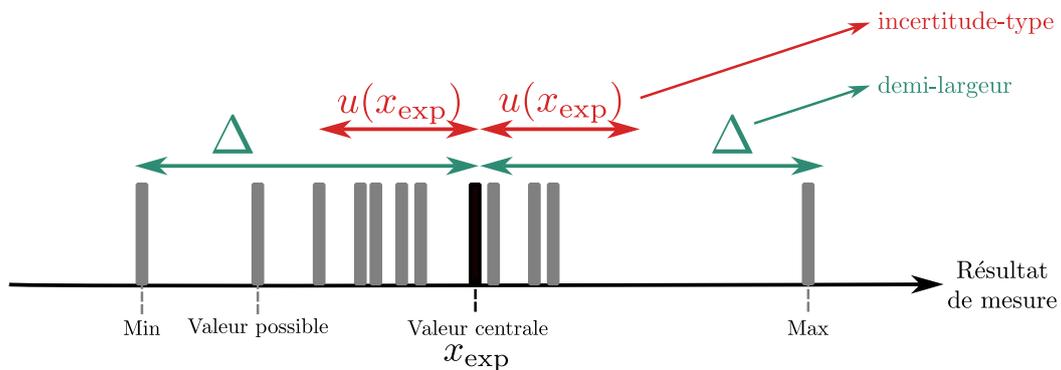


FIGURE N2.1 – Représentation d'un résultat de mesure expérimentale.

Définition N2.2 : Incertitude-type et relative

La variabilité d'une mesure x_{exp} sur une grandeur x est appelée **incertitude-type**, et se note $u(x_{\text{exp}})$.

Elle traduit qu'il y a **68% de probabilité** de trouver la valeur mesurée dans l'intervalle $[x_{\text{exp}} - u(x_{\text{exp}}), x_{\text{exp}} + u(x_{\text{exp}})]$.

Il est utile de définir l'incertitude-type **relative**, en pourcentages :

$$u_r(x_{\text{exp}}) = \frac{u(x_{\text{exp}})}{x_{\text{exp}}}$$

Attention N2.1 : Valeurs possibles d'une mesure

Une incompréhension classique serait de considérer que l'incertitude-type regroupe toutes les valeurs d'une mesure, mais *ça n'est pas le cas* : elle traduit une **erreur moyenne attendue**.

Les valeurs possibles d'une mesure sont plus larges que l'incertitude-type !

Important N2.1 : Présentation d'un résultat de TP

Le résultat numérique d'une grandeur x s'écrit :

$$x = (x_{\text{exp}} \pm u(x_{\text{exp}})) 10^n \text{ unité}$$

x_{exp} est la valeur **mesurée**, la meilleure estimation possible de la grandeur mesurée.

Par convention,

- ◇ Une incertitude-type comporte 2 chiffres significatifs;
- ◇ Le dernier chiffre significatif de la mesure correspond à celui de l'incertitude :

$$d = (12,35 \pm 0,27) \text{ m} \quad \text{et pas} \quad d = (12,35\cancel{2} \pm 0,27) \text{ m}$$

Application N2.1 : Présenter un résultat numérique

Pour chaque mesure suivante : corriger la présentation, indiquer le nombre de chiffres significatifs, calculer **et commenter** leurs incertitudes relatives u_r .

$$\lambda = (589,0 \pm 11,0) \text{ nm} \quad t = (0,473 \pm 0,122) \text{ s} \quad V = 14 \pm 1,5 \times 10^{-3} \text{ mL}$$

- 1) $\lambda = (589 \pm 11) \text{ nm}$; 3 CS ; A.N. : $u_r(\lambda) = 1,9\%$ Satisfaisant en général
- 2) $t = (0,47 \pm 0,12) \text{ s}$; 2 CS ; A.N. : $u_r(t) = 26\%$ Insatisfaisant
- 3) $V = (14,0000 \pm 0,0015) \text{ mL}$; 6 CS ; A.N. : $u_r(t) = 0,011\%$ Soit c'est industriel donc TB, soit sous-estimé.

I/C Comparaison de deux mesures

Par essence, les sciences physiques s'attardent à rendre compte d'observations de manière quantitative. Ainsi, il est fréquent de devoir comparer deux valeurs mesurées entre elles, ou avec une valeur de référence. Pour cela, il faut un critère quantitatif pour indiquer si ces deux mesures sont considérées comme compatibles ou incompatibles.

♥ Définition N2.3 : Écart normalisé

L'**écart normalisé** E_N ¹ entre deux mesures de valeurs m_1 et m_2 et d'incertitudes $u(m_1)$ et $u(m_2)$ est défini par :

$$E_N = \frac{|m_1 - m_2|}{\sqrt{u(m_1)^2 + u(m_2)^2}}$$

Par *convention*, on considère que **deux résultats sont compatibles** si $E_N \lesssim 2$.

Interprétation N2.1 : Écart normalisé

Pour justifier cette convention, on peut revenir à la définition de l'incertitude-type. Celle-ci quantifie les fluctuations potentielles de la valeur mesurée annoncée. Lorsque deux mesures sont cohérentes, on ne s'attend pas à ce qu'elles coïncident exactement, mais qu'elles ne s'écartent pas l'une de l'autre de plus que de quelques incertitudes-type.

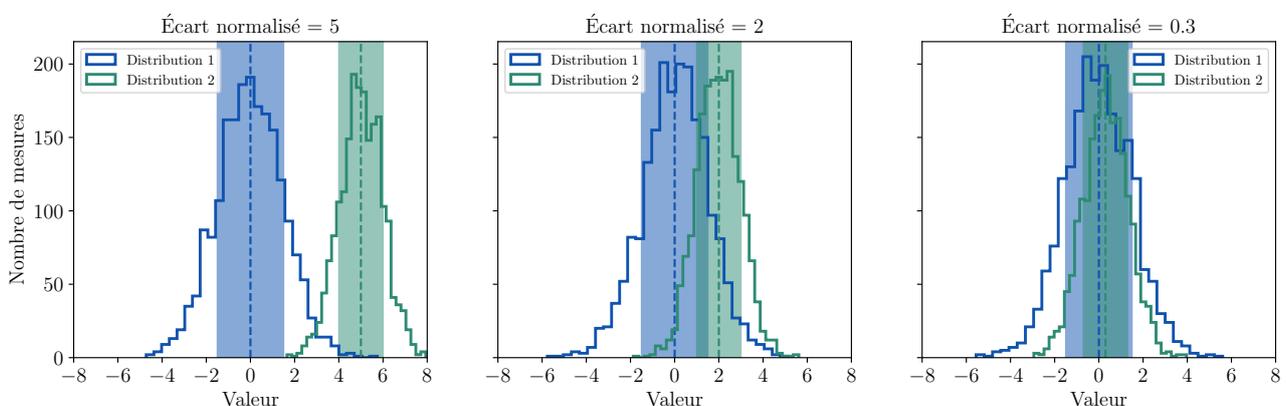


FIGURE N2.2 – Tracé de deux distributions de résultats de mesures.

1. Il est parfois appelé *z-score*, et noté z .

Corollaire N2.1 : Comparaison avec une valeur de référence

Dans le cas d'une valeur donnée sans incertitude, ou avec une incertitude très faible comparée à la nôtre, $u(m_{\text{exp}}) \gg u(m_{\text{ref}})$ implique donc

$$E_N = \frac{|m_{\text{exp}} - m_{\text{ref}}|}{u(m_{\text{exp}})}$$

♥ Définition N2.4 : Écart absolu et relatif

Si aucune des incertitudes n'est connue ou déterminée, on peut utiliser :

Écart absolu

$$\varepsilon = |x_{\text{exp}} - m_{\text{ref}}| \quad \text{unité} \quad \text{celle de } m$$

Écart relatif

$$\varepsilon_r = \left| \frac{m_{\text{exp}} - m_{\text{ref}}}{m_{\text{ref}}} \right| \quad \text{unité} \quad \text{aucune : \%}$$

II Méthodes analytiques d'estimation d'incertitudes**II/A Estimation pour une mesure invariable : type B**

Pour une mesure soit unique soit invariable, il existe tout de même une incertitude. On l'évalue à l'aide de connaissances préalables sur le dispositif expérimental : mesures antérieures, spécifications du fabricant, expérience ou connaissances du comportement ou des propriétés des instruments utilisés.

Important N2.2 : Incertitude de type B

Pour une mesure sans variabilité observée, on trouve l'intervalle d'existence $[\bar{x} - \Delta, \bar{x} + \Delta]$. Alors on a :

Valeur mesurée

C'est la valeur centrale de l'intervalle.

$$x_{\text{exp}} = \bar{x}$$

Incertitude

C'est la demi-largeur divisée par $\sqrt{3}$.

$$u(x_{\text{exp}}) = \frac{\Delta}{\sqrt{3}}$$

Outils N2.1 : Détermination de Δ

On a trois cas de figures courant pour la mesure de Δ :

- 1) Δ est estimée directement avec honnêteté ;
- 2) Δ est *reliée* à **une graduation** de l'appareil ;
- 3) Δ est fournie sur la notice.

Application N2.2 : Incertitudes de type B

Évaluer les incertitudes et donnez les résultats des expériences suivantes :

- 1) Mesure de la position d d'un écran sur un banc optique. L'image semble nette pour des positions allant de 29,7 à 30,5 cm.

$$d_{\text{exp}} = \bar{d} = \frac{d_{\text{max}} + d_{\text{min}}}{2} ; \Delta = \frac{|d_{\text{max}} - d_{\text{min}}|}{2} \Rightarrow u(d) = \frac{\Delta}{\sqrt{3}} \quad \text{soit} \quad \underline{d = (30,10 \pm 0,23) \text{ cm}}$$

- 2) Mesure d'une longueur $\ell = 13$ cm avec une règle graduée au mm.

$$\Delta = 1 \text{ mm} \quad \text{soit} \quad u(\ell) = \frac{\Delta}{\sqrt{3}} \Rightarrow \underline{\ell = (13,000 \pm 0,058) \text{ cm}}$$

- 3) Utilisation d'un multimètre pour mesurer une tension U . Il affiche 2,5462 V et la notice indique *Accuracy=0.3% rdg + 2 digits*, autrement dit de 0,3% de la valeur lue (*rdg = reading*) auquel on ajoute 2 fois la valeur du dernier chiffre.

$$u(U) = 0,003U + 2 \times 0,002 \Rightarrow \underline{U = (2,5462 \pm 0,0080) \text{ V}}$$

II/B Estimation pour une série de mesures : type A

Lorsque l'on réalise plusieurs fois et de manière indépendante une mesure, il est possible d'évaluer **statistiquement** l'incertitude. Ainsi, pour n mesures notées x_i avec $1 \leq i \leq n$:

Important N2.3 : Incertitude de type A

Valeur mesurée

C'est la moyenne des mesures.

$$x_{\text{exp}} = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

Incertitude sur x_i

C'est l'écart-type.

$$u(x_i) = \sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

Incertitude sur \bar{x}

C'est l'écart-type divisé par \sqrt{n} .

$$u(\bar{x}) = \frac{\sigma}{\sqrt{n}}$$

Interprétation N2.2 : Incertitude de type A

L'écart-type σ est une mesure de la dispersion des mesures x_i : plus elles sont étalées, plus σ est grand. Cependant, chaque nouvelle mesure apporte une information supplémentaire sur la valeur mesurée, et augmente ainsi la précision : la moyenne \bar{x} est d'autant plus précise que le nombre de mesures n est grand. C'est pourquoi **l'incertitude sur la moyenne réduit** avec le nombre de mesures.

On remarquera que le procédé n'est pas linéaire : pour avoir une incertitude 10 fois plus faible, il faut 100 fois plus de mesures !

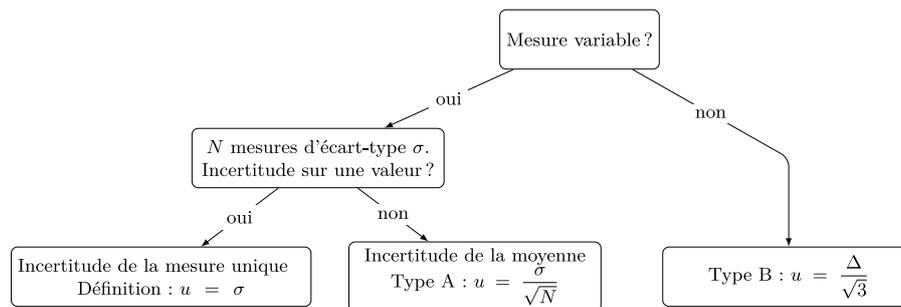


FIGURE N2.3 – Schématisation du choix de la méthode d'estimation de l'incertitude.

Attention N2.2 : Écarts-types automatiques

Attention, selon les logiciels ou calculatrices, c'est n au lieu $n - 1$! Notamment pour les Casio, σx est la version non désirée, et sx est la version avec $n - 1$ comme attendu.

Avec Python, on utilise `np.std(vals, ddof=1)` : `std` pour *standard deviation* (écart-type), et `ddof` pour *in delta degrees of freedom* (delta degré de liberté) : `np.std` remplace $n - 1$ par $n - ddof$.

Application N2.3 : Incertitude de type A

On a obtenu par dosage rédox le degré d'alcool d'un vin à 8 reprises. Les mesures ont donné :

Degré (°)	11,9	12,5	13,1	12,4	12,9	12,6	12,8	12,6

À l'aide de votre calculatrice, d'un tableur ou d'un script Python, exprimer le résultat de ces mesures ainsi que son incertitude.

```

1 import numpy as np
2
3 vals = np.array([11.9, 12.5, 13.1, 12.4, 12.9, 12.6, 12.8, 12.6])
4
5 mean = np.mean(vals)
6 ecarttype = np.std(vals, ddof=1)
7 incertype = ecarttype / np.sqrt(len(vals))
8
9 print(f"D = ({mean:.2f} ± {incertype:.2f})°")
  
```

On a $D = \bar{D} \pm u(\bar{D})$ avec $u(\bar{D}) = \frac{\sigma}{\sqrt{n}}$ soit $D = (12,60 \pm 0,13)^\circ$

II/C Incertitudes composées

Dans de très nombreuses situations, on est amené-e à **calculer** la valeur d'une grandeur à partir de valeurs mesurées, et donc possédant des incertitudes. Comment exprimer alors l'**incertitude sur ce calcul** en fonction de celles sur les données utilisées ?

II/C) 1 Calcul à partir d'une seule valeur mesurée

Important N2.4 : Incertitudes composées à 1 variable

L'incertitude sur le calcul dépend de la manière dont la fonction varie selon la grandeur mesurée : si on souhaite calculer $y_{\text{calc}} = 3 \times x_{\text{exp}}$, alors l'incertitude sur y_{calc} est 3 fois plus importante que celle sur x_{exp} ! D'une manière générale, le lien se fait grâce à la **tangente à la fonction**, c'est-à-dire sa dérivée au point de calcul.

Soit $y_{\text{calc}} = f(x_{\text{exp}})$ une valeur calculée à partir d'une mesure x_{exp} . L'incertitude-type sur y_{calc} est donnée par :

$$u(y_{\text{calc}}) = f'(x_{\text{exp}}) \cdot u(x_{\text{exp}}) = \left. \frac{df}{dx} \right|_{x_{\text{exp}}} \cdot u(x_{\text{exp}})$$

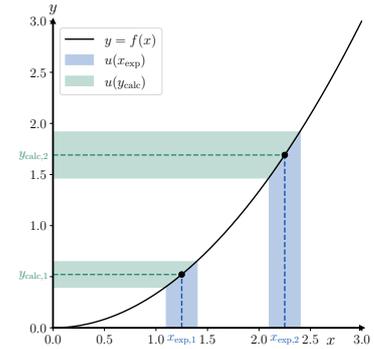


FIGURE N2.4

♥ Application N2.4 : Incertitude composée à une variable

- On cherche à déterminer la période T d'un pendule simple de longueur L . On a pour cela mesuré 8 périodes, avec $T_8 = (24,2 \pm 1,0)$ s. Quelle est la valeur de la période T et son incertitude ?
- Soit $\overline{OF'} = (10,35 \pm 0,20)$ cm une focale mesurée d'une lentille. On cherche à déterminer sa vergence $V = 1/\overline{OF'}$ avec son incertitude.

1) Soit $y_{\text{calc}} = f(x)$ avec $y_{\text{calc}} = T$; $x_{\text{exp}} = T_8$; $f(x) = \frac{x}{8}$
 ainsi $f'(x) = \frac{1}{8} \Rightarrow u(y_{\text{calc}}) = \frac{u(x_{\text{exp}})}{8} \Leftrightarrow u(T) = \frac{u(T_8)}{8}$ A.N. : $T = (3,03 \pm 0,13)$ s

2) Soit $y_{\text{calc}} = f(x)$ avec $y_{\text{calc}} = V$; $x_{\text{exp}} = \overline{OF'}$; $f(x) = \frac{1}{x}$
 ainsi $|f'(x)| = \frac{1}{x^2} \Rightarrow u(y_{\text{calc}}) = \frac{u(x_{\text{exp}})}{x_{\text{exp}}^2} \Leftrightarrow u(V) = \frac{u(\overline{OF'})}{\overline{OF'}^2}$ A.N. : $V = (9,7 \pm 0,2) \text{ m}^{-1}$

II/C) 2 Calcul avec deux valeurs mesurées

Important N2.5 : Incertitudes composées à 2 variables

Somme ou différence

Pour $y = \alpha x_1 \pm \beta x_2$, on obtient

$$u(y) = \sqrt{(\alpha u(x_1))^2 + (\beta u(x_2))^2}$$

Produit de puissances

Pour $y = a x_1^{\alpha_1} x_2^{\alpha_2}$, on obtient

$$\frac{u(y)}{y} = \sqrt{\left(\alpha_1 \frac{u(x_1)}{x_1}\right)^2 + \left(\alpha_2 \frac{u(x_2)}{x_2}\right)^2}$$

♥ Application N2.5 : Incertitude composée à 2 variables

- On mesure la distance $d = x_2 - x_1$ entre deux points repérés avec la même incertitude $u(x_1) = u(x_2)$. Quelle est l'incertitude sur d ?
- On détermine la célérité c du son dans l'air grâce à la relation $\lambda = c/f$, avec $f = (500 \pm 10)$ Hz et $\lambda =$

(68,0 ± 2,5) cm. Donner sa valeur.

1) On trouve élémentairement : $u(d) = \sqrt{u(x_1)^2 + u(x_2)^2} = u(x_1)\sqrt{2}$

2) $c = \lambda f \Rightarrow u(c) = c \times \sqrt{\left(\frac{u(\lambda)}{\lambda}\right)^2 + \left(\frac{u(f)}{f}\right)^2} \Rightarrow c = (340 \pm 14) \text{ m}\cdot\text{s}^{-1}$

II/C) 3 Incertitudes-types composées quelconques

Seuls les cas à 1 et 2 variables sont à connaître. Dans les cas généraux, il existe une formule hors-programme mais potentiellement utile en TIPE :

Propriété N2.1 : Incertitude composée générale (HP)

Pour $y = f(x_1, x_2, \dots, x_n)$, on a

$$u(y) = \sqrt{\sum_{i=1}^n \left(\frac{\partial f}{\partial x_i} \Big|_{x_{\text{exp},i}} \cdot u(x_i) \right)^2}$$

Pour tous les autres cas, nous allons revenir à la définition des incertitudes puis, à l'aide d'une simulation informatique comportant une part d'aléatoire, calculer l'incertitude-type.

III Méthode numérique : simulations MONTE-CARLO

III/A Principe

Définition N2.5 : Simulations MONTE-CARLO

Un algorithme utilisant la variabilité d'une mesure pour simuler un calcul d'incertitude fait parti des algorithmes de type MONTE-CARLO.

On dispose généralement de plusieurs jeux de données pour lesquels on a des incertitudes de mesure, et on veut calculer z qui dépend de ces données mais d'une manière complexe. On peut alors réaliser une simulation.

En effet, connaissant l'intervalle d'existence des mesures, on peut prendre aléatoirement d'autres valeurs possibles pour les mesures, et faire toute une série de calculs avec des valeurs « simulées », légèrement modifiées mais toujours dans un intervalle raisonnable. On pourra alors finalement prendre la moyenne des valeurs calculées et leur écart-type pour avoir la propagation des incertitudes !

Interprétation N2.3 : Cœur de la simulation

Finalement, le cœur de la simulation revient (presque) à réaliser une estimation d'incertitude de type A sur les valeurs calculées !

Attention N2.3 : Distribution uniforme ou gaussienne ?

Selon que l'on dispose d'une incertitude-type ou d'une plage de valeurs, on n'effectue pas le même tirage :

◇ Si l'on dispose d'une incertitude-type $u(x)$ obtenue par type A (ou explicitement donnée), on considère que la grandeur x suit une **distribution gaussienne**, aussi appelée **loi normale** (courbe en cloche), centrée en x_{exp} et d'écart-type $u(x)$:

$$x \sim \mathcal{N}(x_{\text{exp}}, u(x)) = \frac{1}{u(x)\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x - x_{\text{exp}}}{u(x)} \right)^2}$$

On tirera donc aléatoirement des valeurs selon cette distribution. En Python, on utilisera alors `np.random.normal(x_exp, u_x)` ;

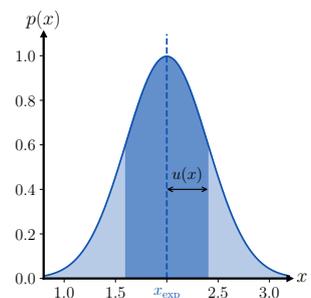


FIGURE N2.5

- ◇ Si l'on dispose d'une incertitude-type $u(x)$ obtenue par type B, on considère que la grandeur x suit une **distribution uniforme** centrée en x_{exp} et de demi-largeur $\Delta = u(x)\sqrt{3}$:

$$x \sim \mathcal{U}(x_{\text{exp}} - \Delta, x_{\text{exp}} + \Delta) = \begin{cases} \frac{1}{2\Delta} & \text{pour } x \in [x_{\text{exp}} - \Delta, x_{\text{exp}} + \Delta] \\ 0 & \text{sinon} \end{cases}$$

On tirera donc aléatoirement des valeurs dans cet intervalle. En Python, on utilisera alors `np.random.uniform(x_exp - Delta_x, x_exp + Delta_x)`.

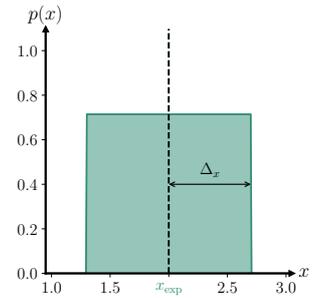


FIGURE N2.6

III/B MONTE-CARLO sur un scalaire

Prenons un exemple. On peut mesurer la focale d'une lentille convergente par la méthode de BESSEL :

$$f' = \frac{D^2 - d^2}{4D}$$

avec d la plage de positions de la lentille qui garde une image nette sur l'écran, et D la distance objet-écran, **mesurées avec un intervalle de valeurs possibles** (type B). Il est possible de faire le calcul analytique ici, mais il peut être plus rapide de réaliser une propagation des incertitudes des valeurs d et D sur la valeur calculée de f' . Pour cela :

Important N2.6 : MONTE-CARLO scalaire

- ◇ On associe les demi-largeurs Δ_d et Δ_D aux variables `Delta_d` et `Delta_D`.
- ◇ On crée une liste `f_liste` vide qui accueillera les valeurs de f' simulées ;
- ◇ On fixe $N \gtrsim 10^4$ le nombre de simulations ;
- ◇ Pour i allant de 0 à $N - 1$:
 - ▷ On tire aléatoirement et uniformément une valeur `d_simu` dans l'intervalle $[d - \Delta_d, d + \Delta_d]$;
 - ▷ On tire aléatoirement et uniformément une valeur `D_simu` dans l'intervalle $[D - \Delta_D, D + \Delta_D]$;
 - ▷ On calcule f'_{simu} avec ces données **simulées** ;
 - ▷ On ajoute cette valeur simulée à la liste des valeurs probables mesurées `f_liste`.
- ◇ On a alors N valeurs de f'_{simu} . La valeur la plus probable sera la moyenne, et son incertitude-type est l'écart-type de la liste des f'_{simu} ^a.

Ainsi, en Python :

```
1 import numpy as np
2
3 d = 12 # cm
4 Delta_d = 0.1 # demi-largeur de d
5 D = 50 # cm
6 Delta_D = 0.5 # demi-largeur de D
7
8 N = 10000 # nombre de régressions à effectuer
9
10 f_liste = [] # création de la liste vide pour stocker les valeurs
11 for i in range(N):
12     # on prend des valeurs de d_simu parmi toutes les valeurs possibles
13     # à l'aide d'un tirage aléatoire à l'intérieur de l'intervalle
14     # [d ± Delta_d]
15     d_simu = np.random.uniform(d - Delta_d, d + Delta_d)
16     # Pareil pour D
17     D_simu = np.random.uniform(D - Delta_D, D + Delta_D)
```

^a. Semblable à une incertitude de type A, seulement on ne **divise pas par** \sqrt{n} puisque le nombre de simulation est arbitraire : ce ne sont pas des nouvelles mesures !

```

18
19     # On calcule les f' simulés avec ces valeurs
20     f_simu = (D_simu**2 - d_simu**2) / (4 * D_simu)
21
22     # On les sauvegarde dans a liste correspondante
23     f_liste.append(f_simu)
24
25     # La valeur moyenne est la moyenne de la liste
26     fmoy = np.mean(f_liste)
27     # L'incertitude est l'écart-type de la liste
28     uf = np.std(f_liste, ddof=1)
29     # Affichage, à modifier pour l'écriture scientifique
30     print(f"f' = {fmoy:.2f} ± {uf:.2f} cm")

```

III/C MONTE-CARLO sur un vecteur

Prenons l'exemple de la régression linéaire :

$$y = ax + b$$

On a une série de mesures de x et y dans les listes X et Y avec leurs **listes d'incertitudes associées** uX et uY (type A ou explicite). On obtient a et b avec `np.polyfit(X, Y, 1)`, mais ce calcul ne donne pas l'incertitude sur a et b . Les deux valeurs étant interdépendantes, on n'a pas d'expression analytique pour les déterminer : on va donc les simuler.

Chaque valeur des x_i est ici décrite par une distribution normale, puisqu'on a les incertitudes-types. On va alors calculer une série de a_{simu} et b_{simu} pour des valeurs x_{simu} et y_{simu} légèrement modifiées, à l'intérieur d'une distribution normale, puis réaliser un **grand nombre de régressions linéaires simulées**. Les moyennes de a et b sont alors les valeurs moyennes de chaque liste, et les incertitudes-types seront leurs écarts-types. Pour cela :

Important N2.7 : MONTE-CARLO vecteur

- ◇ On associe les incertitudes-types $u(x)$ et $u(y)$ aux **listes** uX et uY ;
- ◇ On fixe un nombre N très grand.
- ◇ On crée des listes vides `a_liste` et `b_liste` pour y stocker les futures valeurs des a_{simu} et des b_{simu} calculés.
- ◇ Pour chaque i compris entre 0 et $N - 1$:
 - ▷ On tire aléatoirement mais **normalement** un vecteur `X_simu` ;
 - ▷ On tire aléatoirement mais **normalement** un vecteur `Y_simu` ;
 - ▷ on calcule `a_simu` et `b_simu` avec ces valeurs simulées ;
 - ▷ on les stocke dans `a_liste` et `b_liste`.
- ◇ On a alors N valeurs de a_{simu} et de b_{simu} : les valeurs les plus probables sont les moyennes, et leurs incertitudes-types sont les écarts-types des listes de a_{simu} et de b_{simu} .

Attention N2.4 : Δ ou u pour MONTE-CARLO vecteur

Pour la régression linéaire, les Δ_x ou $u(x)$ doivent être des vecteurs (des listes) ! Il doit y avoir autant de valeurs de Δ_x (ou $u(x)$) que de valeurs de x !

Ainsi, en Python :

```

1  import numpy as np
2
3  X = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]) # unité
4  # VECTEUR d'incertitude de 0.1 sur chaque valeur
5  uX = 0.1 * np.ones(len(X)) # unité
6
7  Y = np.array(
8      [2.20, 2.00, 1.60, 1.55, 1.16, 1.00, 0.95, 0.60, 0.36, 0.36, 0.18]
9  ) # unité
10 # VECTEUR d'incertitude de 0.12 sur chaque valeur
11 uY = 0.12 * np.ones(len(Y)) # unité

```

```
12
13
14 N = 10000 # nombre de régressions à effectuer
15
16 a_liste, b_liste = [], [] # création des listes vides pour stocker les valeurs
17 for i in range(N):
18     # on prend des valeurs de X_simu parmi les valeurs possibles
19     # à l'aide d'un tirage aléatoire suivant une loi normale
20     X_simu = np.random.normal(X, uX)
21     # Pareil pour Y
22     Y_simu = np.random.normal(Y, uY)
23
24     # On fait une régression linéaire avec ces données « simulées »
25     a_simu, b_simu = np.polyfit(X_simu, Y_simu, 1)
26
27     # On les sauvegarde dans les listes correspondantes
28     a_liste.append(a_simu)
29     b_liste.append(b_simu)
30
31 # Les valeurs moyennes sont les moyennes des listes
32 a_moy, b_moy = np.mean(a_liste), np.mean(b_liste)
33 # Les incertitudes sont les écarts-types des listes
34 ua, ub = np.std(a_liste, ddof=1), np.std(b_liste, ddof=1)
35
36 # Affichage, à modifier pour l'écriture scientifique
37 print(f"Coef.directeur = {a_moy:.3e} ± {ua:.3e} unité")
38 print(f"Ordonnée à l'origine = {b_moy:.3e} ± {ub:.3e} unité")
```