

# Traitement de signaux pour la détection de séismes et leur multilatération

Dalibard Louis

17 mai 2026

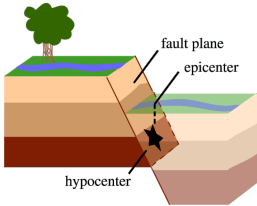
## Table des matières

<b>1 Séismes</b>	<b>2</b>
1.1 Introduction . . . . .	2
1.2 Ondes P et S . . . . .	2
1.3 Différence de vitesse de propagation . . . . .	2
<b>2 Théorie</b>	<b>3</b>
2.1 Principe . . . . .	3
2.2 DSP . . . . .	3
2.2.1 Acquisition des données . . . . .	3
2.2.2 Extraction des temps d'arrivée . . . . .	5
2.2.3 Convolution . . . . .	5
2.2.4 Propriétés algébriques de la convolution . . . . .	7
2.2.5 Moyennage . . . . .	11
2.2.6 Transformée de Fourier discrète . . . . .	11
2.2.7 Radix-2 decimation-in-time (DIT) - Factorisation Cooley-Tukey . . . . .	12
2.2.8 IFFT . . . . .	13
2.2.9 Passage du domaine temporel au domaine fréquentiel . . . . .	14
2.2.10 Corrélacion croisée . . . . .	14
2.3 Modélisation de la propagation des ondes sismiques . . . . .	15
2.3.1 Calcul du temps de propagation selon iasp91 . . . . .	15
2.3.2 Tabulation et interpolation . . . . .	16
2.4 Multilatération . . . . .	16
2.4.1 Fonction d'erreur . . . . .	17
2.4.2 Méthode de Nelder-Mead . . . . .	17
2.5 Magnitude sismique . . . . .	18
<b>3 Toutes les pièces mises ensembles</b>	<b>19</b>
3.1 Fonctionnement . . . . .	19
3.2 Résultats . . . . .	23
<b>4 Précision et erreur</b>	<b>25</b>
4.1 Cas réduit avec propagation rectiligne dans un milieu transparent, homogène, isotrope, linéaire à longues distances et 4 détecteurs . . . . .	25
4.2 Méthode de Monte-Carlo pour les cas plus complexes . . . . .	28
<b>5 Bonus</b>	<b>30</b>
<b>6 Bibliographie</b>	<b>31</b>

# 1 Séismes

## 1.1 Introduction

Un séisme, ou tremblement de terre, est une secousse du sol résultant de la libération brusque d'énergie accumulée par les contraintes s'exerçant sur les roches. Cette libération d'énergie se produit par rupture le long d'une faille, généralement préexistante. Plus rares sont les séismes dus à l'activité volcanique ou d'origine artificielle (explosions, par exemple). Le lieu où se produit la rupture des roches en profondeur est appelé le foyer ; la projection du foyer à la surface est l'épicentre du séisme. Le mouvement des roches près du foyer engendre des vibrations élastiques qui se propagent sous la forme de paquets d'ondes sismiques autour et à travers le globe terrestre.



## 1.2 Ondes P et S

Les ondes sismiques, ou ondes élastiques, sont des mouvements vibratoires qui se propagent à travers un milieu matériel et peuvent le modifier irréversiblement si leur amplitude est suffisante. Elles sont engendrées par un événement initial, généralement un séisme. L'impulsion de départ déplace des atomes, qui en poussent d'autres à leur tour avant de reprendre leur place, ces déplacements oscillatoires se propageant ensuite de proche en proche. Un séisme émet des ondes dans toutes les directions de l'espace.

La physique des milieux élastiques est régie par l'équation de Navier, qui implique l'existence de deux grands types d'ondes, mises en évidence expérimentalement : les ondes de volume qui traversent l'intérieur de la Terre et les ondes de surface qui se propagent dans une couche d'épaisseur limitée en suivant la surface terrestre. Les ondes de volume se subdivisent par ailleurs en ondes longitudinales (ondes P) et transversales (ondes S).

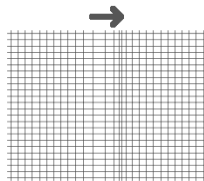


FIGURE 1 : Ondes P

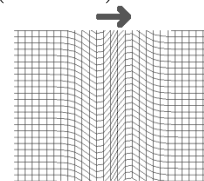


FIGURE 2 : Ondes S

## 1.3 Différence de vitesse de propagation

Les ondes P ou ondes primaires, appelées aussi ondes de compression (en anglais Pressure waves) ou ondes longitudinales. Le déplacement du sol qui accompagne leur passage se fait par des dilatations et des compressions successives. Ces déplacements du sol sont parallèles à la direction de propagation de l'onde. Elles se propagent dans tous les milieux et sont les plus rapides (autour de 6 km/s près de la surface), parcourant le chemin le plus court, même par noyau terrestre, et sont donc les premières enregistrées sur les sismogrammes. Elles sont responsables du grondement sourd que l'on peut entendre au début d'un tremblement de terre.

Les ondes S ou ondes secondaires, appelées aussi ondes de cisaillement (Shear waves) ou ondes transversales. À leur passage, les mouvements du sol s'effectuent perpendiculairement au sens de propagation de l'onde. Ces ondes ne se propagent pas dans les milieux liquides, elles sont en particulier arrêtées par le noyau externe de la Terre. Leur vitesse est d'environ 4 km/s. Elles apparaissent en deuxième sur les sismogrammes.

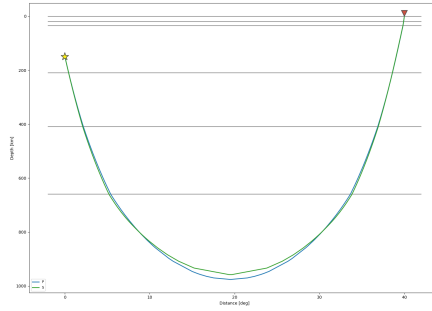


FIGURE 3 : 6 km/s (ondes P) vs 4 km/s (ondes S)

On peut donc exploiter les temps d'arrivée de ces différentes ondes pour localiser un séisme.

## 2 Théorie

### 2.1 Principe

Différentes étapes :

1. Acquisition de données en temps réel (SeedLink)
2. Reconnaissance d'un séisme et mesure automatique des temps
3. Calcul de la position et de la magnitude
  - (a) Modélisation de la propagation des ondes sismiques
  - (b) Méthode numérique d'optimisation de fonction à plusieurs variables pour la multilatération
  - (c) Calcul de la magnitude

### 2.2 DSP

#### 2.2.1 Acquisition des données

On utilise le protocole Seedlink pour récupérer les données des sismographes en temps réel.

Name	Host
AusPass	auspass.edu.au
BGR	eida.bgr.de
CISMID	www.cismid.uni.edu.pe
ENS	ephesite.ens.fr
GEOFON, GFZ	geofon.gfz-potsdam.de
GEONET	link.geonet.org.nz
Geoscience Australia	seis-pub.ga.gov.au
GSRAS (?)	89.22.182.133
Helsinki	finseis.seismo.helsinki.fi
Haiti	ayiti.unice.fr
ICGC	ws.icgc.cat
IDA Project	rtserve.ida.ucsd.edu
IFZ	data.ifz.ru
IPGP	rtserver.ipgp.fr

IRIS DMC	rtserve.iris.washington.edu
IRIS Jamaseis	jamaseis.iris.edu
ISNET - UNINA	185.15.171.86
LMU	erde.geophysik.uni-muenchen.de
NIGGG	195.96.231.100
NRCAN	earthquakescanada.nrcan.gc.ca
OBSEBRE	obsebre.es
OGS	nam.ogs.it
Oklahoma University	rtserve.ou.edu
ORFEUS	eida.orfeus-eu.org
PLSN (IGF Poland)	hudson.igf.edu.pl
Red Sismica de Puerto Rico	161.35.236.45
Red Sismica Baru	helis.redsismicabaru.com
RESIF	rtserve.resif.fr
SANET	147.213.113.73
RSIS	rsisl.on.br
SCSN-USC (South Carolina Seismic Network)	eeyore.seis.sc.edu:6382
Seisme IRD	rtserve.ird.nc
Staneo	vibrato.staneo.fr
SNAC NOA	snac.gein.noa.gr
TexNet	rtserve.beg.utexas.edu
Thai Meteorological Department	119.46.126.38
UFRN (Universidade Federal do Rio Grande do Norte)	sislink.geofisica.ufrn.br
Unical Universita Della Calabria	www.sismocal.org
UNITS Università degli studi di Trieste	rtweb.units.it
UNIV-AG Université des Antilles	seedsrv0.ovmp.martinique.univ-ag.fr
Universidade de Évora	clv-cge.uevora.pt
Universidad de Colima	148.213.24.15
UPR	worm.uprm.edu
USGS	cwbpublish.cr.usgs.gov
USP-IAG	seisrequest.iag.usp.br

## 2.2.2 Extraction des temps d'arrivée

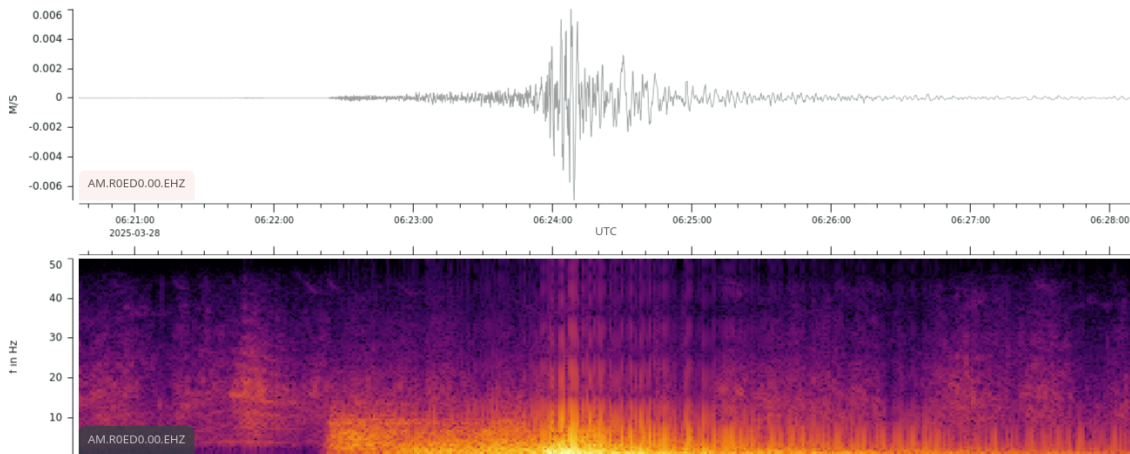


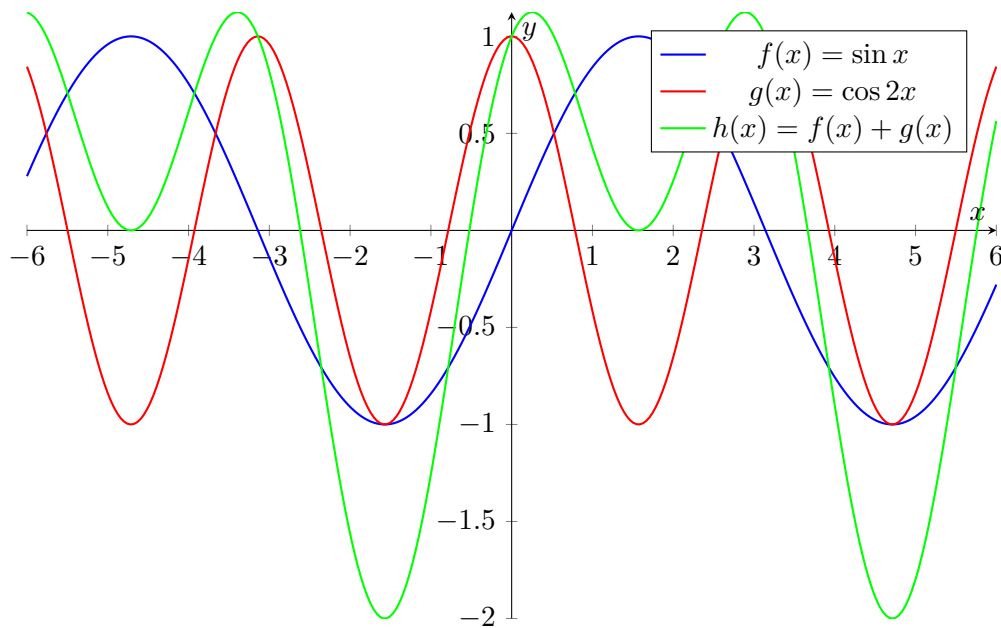
FIGURE 4 : Exemple d'un enregistrement de sismographe

On a un signal comme ça, on peut facilement voir l'onde P et l'onde S. Mais comment faire pour extraire ces temps d'arrivée automatiquement ? Notamment quand on aura des centaines de mesures venant de partout dans le monde ? Pour cela on va faire un détour par des maths.

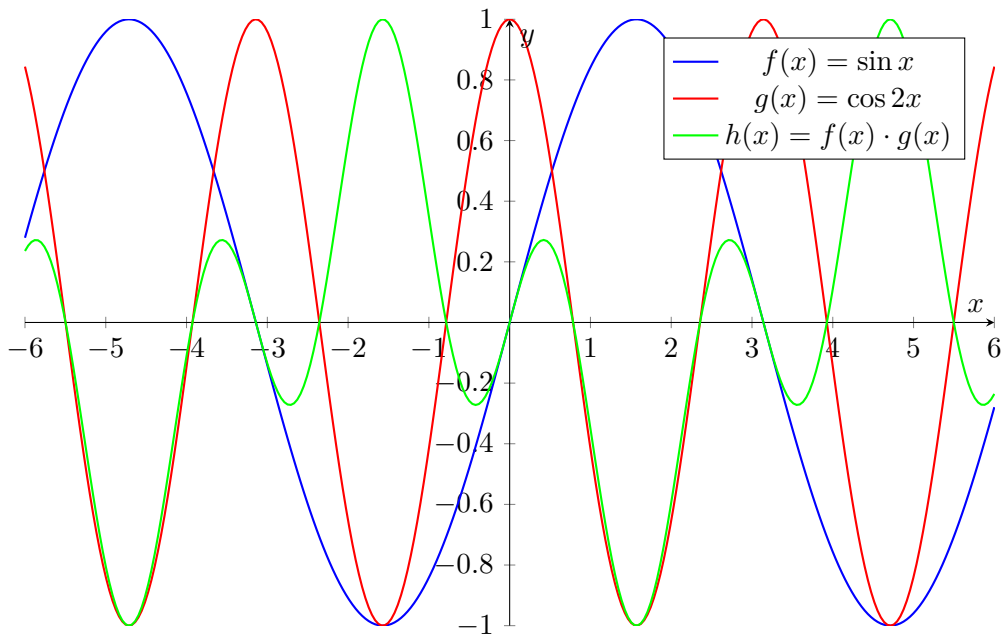
## 2.2.3 Convolution

Si on a deux fonctions, quelles opérations peut-on faire pour avoir une nouvelle fonction ?

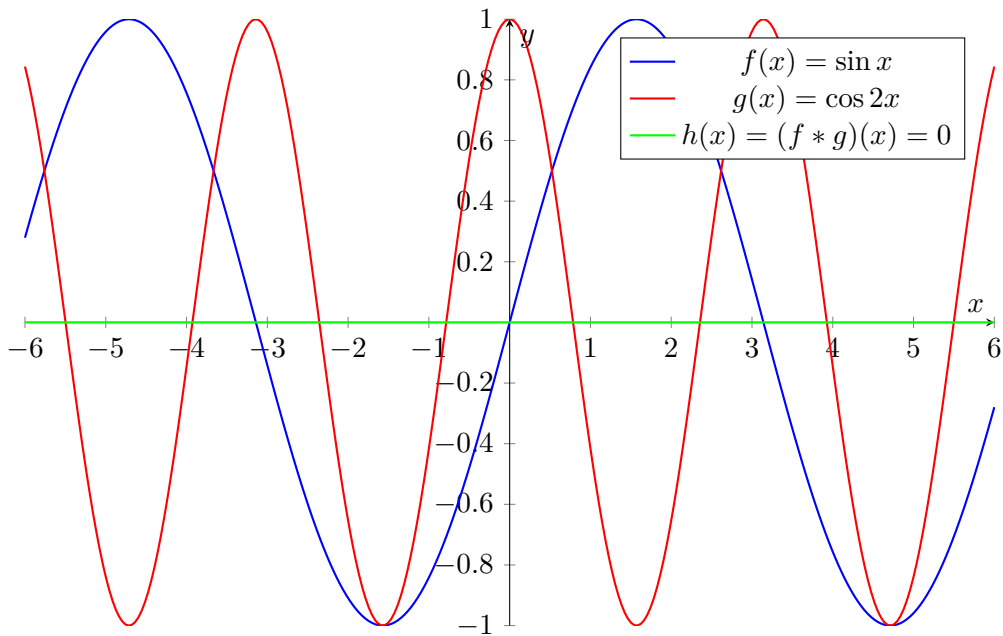
La plus simple semble être les additionner.



Une autre façon serait de les multiplier.



Mais il existe aussi une autre opération, la convolution.



Qui a de nombreuses applications dans le traitement d'images, la théorie de la probabilité, dans la solution d'équations différentielles (température), et comme nous le verrons (aussi la multiplication de polynômes).

On définit formellement la convolution comme un produit sur l'espace des fonctions intégrables.

Soit  $f$  et  $g$  deux fonctions intégrables sur  $\mathbb{R}$ .

Pour tout  $t \in \mathbb{R}$ ,

$$(f * g)(t) = \int_{-\infty}^{+\infty} f(x) \cdot g(t - x) dx$$

Dans le cas discret on utilisera une somme.

Soit  $f$  et  $g$  deux fonctions de  $\mathbb{Z}$  dans  $\mathbb{C}$ .

Pour tout  $n \in \mathbb{Z}$ ,

$$(f * g)[n] = \sum_{m=-\infty}^{+\infty} f[m] \cdot g[n - m]$$

et dans le cas de fonction périodiques, il suffit de faire l'intégrale sur une période. Pour tout  $t \in \mathbb{R}$ ,

$$(f * g)(t) = \int_0^T f(x) \cdot g(t - x) dx$$

(on peut également généraliser cela à des fonctions discrètes périodiques)

#### 2.2.4 Propriétés algébriques de la convolution

— Commutatif

On remarquera que si

$$(f * g)(t) = \int_{-\infty}^{+\infty} f(x) \cdot g(t - x) dx$$

Et on fait le changement de variable  $u = t - x$

On a

$$(f * g)(t) = \int_{+\infty}^{-\infty} f(t - u) \cdot g(u) - du = \int_{-\infty}^{+\infty} f(t - u) \cdot g(u) du = (g * f)(t)$$

— Associatif

$$(f * g) * h = f * (g * h)$$

C'est une conséquence du théorème de Fubini.

— Distributif

$$f * (g + h) = f * g + f * h$$

Par linéarité de l'intégrale.

L'espace des fonctions intégrables muni de  $*$  forme un demi-groupe commutatif (car pas d'élément neutre).

Pourquoi une telle construction ?

Déjà à quoi ça ressemble ?

Si on voulait compter le nombre de combinaisons de dés qui donnent un certain total, on les disposerait en liste, on mettrait une d'elle à l'envers et on regarderait le nombre de dés qui s'alignent à chaque fois.

Somme donnant 2 : 

1	2	3	4	5	6
6	5	4	3	2	1

Somme donnant 3 : 

1	2	3	4	5	6
6	5	4	3	2	1

Somme donnant 4 : 

1	2	3	4	5	6
6	5	4	3	2	1

Somme donnant 5 : 

1	2	3	4	5	6
6	5	4	3	2	1

Somme donnant 6 : 

1	2	3	4	5	6
6	5	4	3	2	1

Somme donnant 7 : 

1	2	3	4	5	6
6	5	4	3	2	1

Somme donnant 8 : 

1	2	3	4	5	6	
	6	5	4	3	2	1

Somme donnant 9 : 

1	2	3	4	5	6		
		6	5	4	3	2	1

Somme donnant 10 : 

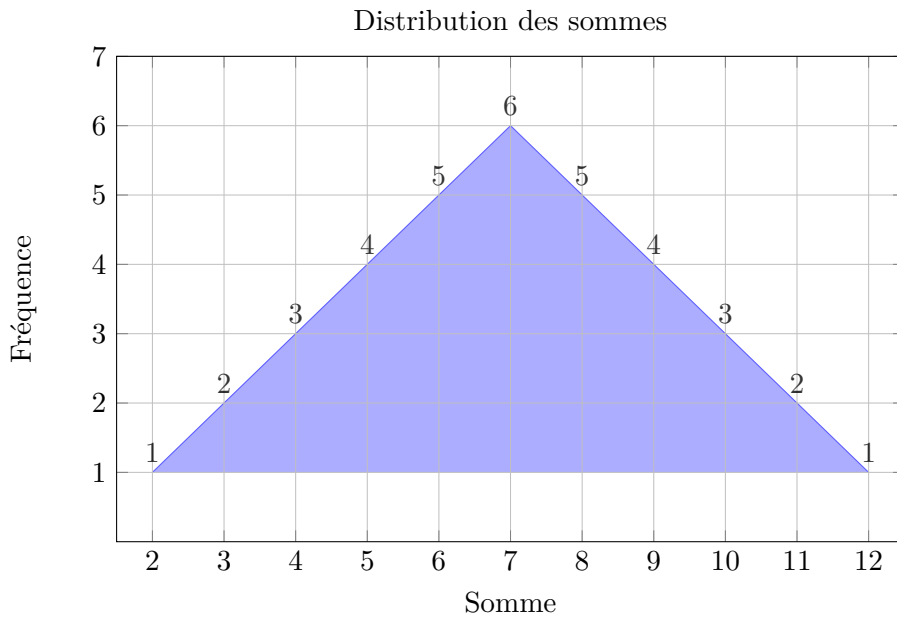
1	2	3	4	5	6			
			6	5	4	3	2	1

Somme donnant 11 : 

1	2	3	4	5	6				
				6	5	4	3	2	1

Somme donnant 12 : 

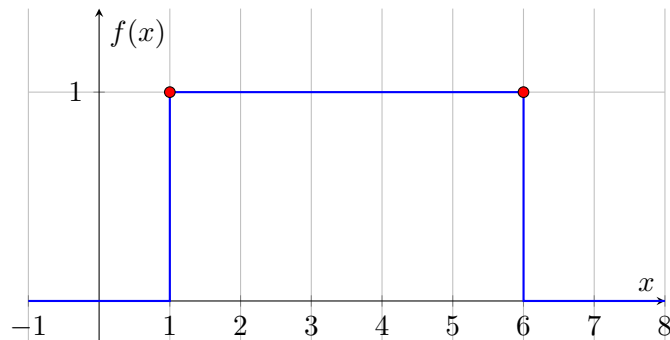
1	2	3	4	5	6					
					6	5	4	3	2	1



On obtient les fréquences d'apparition des différentes combinaisons. Nous pouvons également remarquer que si à la place de rajouter 1 à chaque fois, on pondérerait les probabilités et faisait le produit des probabilités qui se trouvent face à face, on aurait également le résultat pour le cas non équiprobable.

On vient de faire la convolution entre cette fonction et elle même.

Fonction nulle puis constante égale à 1 puis nulle

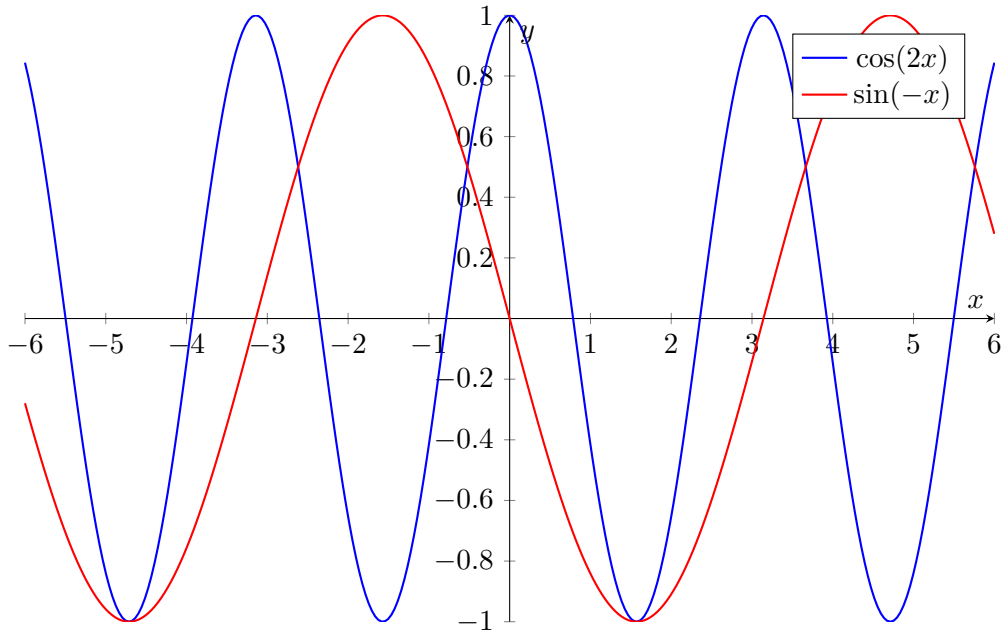
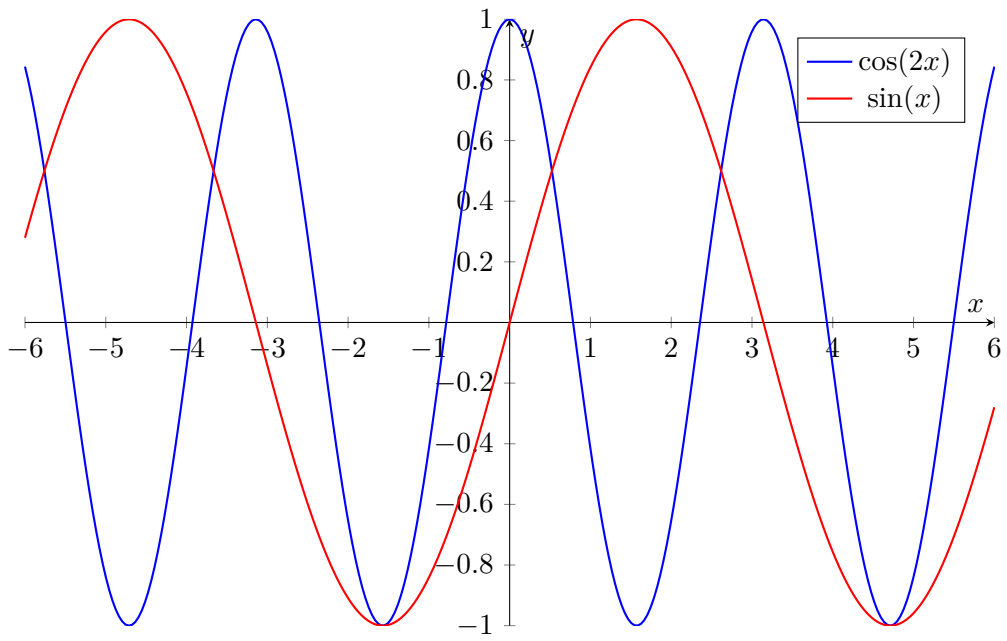


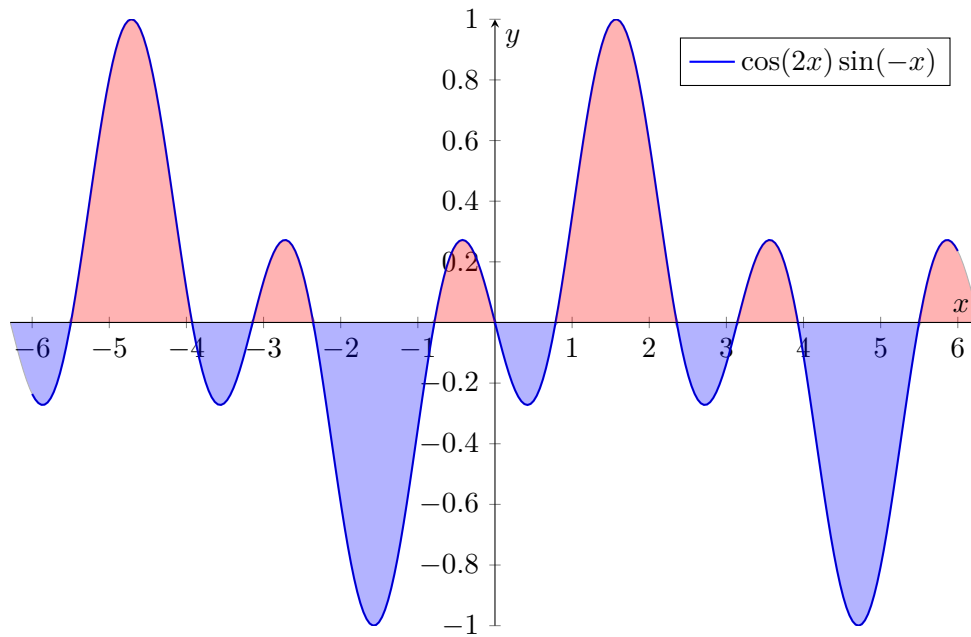
```
>>> np.convolve([0,1,1,1,1,1,1,0],[0,1,1,1,1,1,1,0])
array([0, 0, 1, 2, 3, 4, 5, 6, 5, 4, 3, 2, 1, 0, 0])
```

FIGURE 5 : numpy confirme ce résultat

On remarque que si on le fait dans le cas discret, on a des valeurs en plus sur les bords que l'on peut ignorer.

Faire la convolution revient donc à retourner une des listes des coefficients et le résultat est une fonction qui à un décalage  $t$  entre les deux listes associe une somme des produits deux à deux des coefficients qui se retrouvent en face l'un de l'autre par ce décalage.



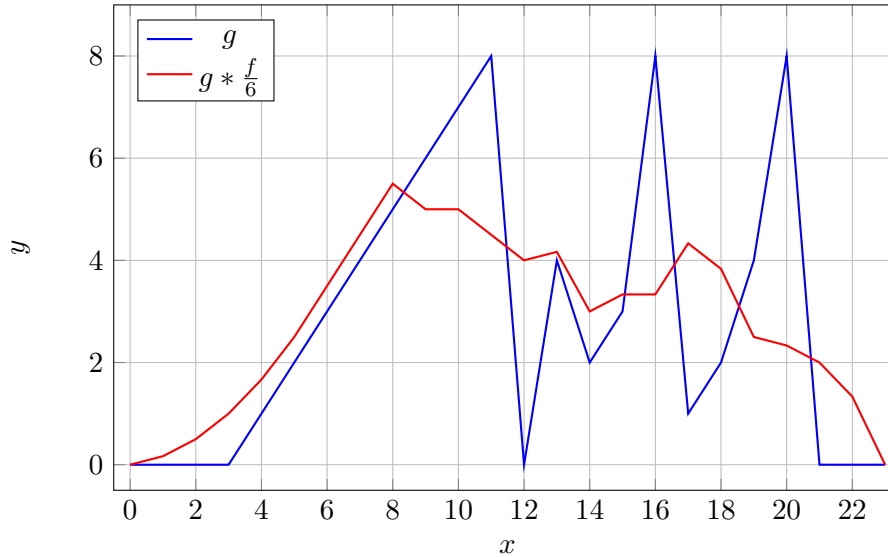


On voit que c'est nul dans ce cas.

### 2.2.5 Moyennage

Si on applique la fonction  $\frac{f}{6}$  à une fonction  $g$  quelconque on observe une sorte de moyennage qui se fait entre 6 valeurs proches.

Moyennage par convolution



Ce processus est fait en deux dimensions sur des images avec différentes fonctions pour avoir des effets comme la détection de bord ou le floutage.

### 2.2.6 Transformée de Fourier discrète

Si on doit faire la convolution de deux listes  $a$  et  $b$  telles que  $a = [a_0, a_1, \dots, a_{n-1}]$  et  $b = [b_0, b_1, \dots, b_{k-1}]$ ,

$$\text{On a } (a * b)[p] = \sum_{\substack{i \in \mathbb{Z} \\ 0 \leq p-i \leq k-1 \\ 0 \leq i \leq n-1}} a_i b_{p-i}$$

On reconnaît la formule du produit des polynômes  $P(X) = \sum_{i=0}^{n-1} a_i X^i$  et  $Q(X) = \sum_{j=0}^{k-1} a_j X^j$

$(a * b)[p]$  est le coefficient du terme de degré  $p$  dans le produit

$$PQ(X) = \sum_{j=0}^p \left( \sum_{\substack{i \in \mathbb{Z} \\ 0 \leq p-i \leq k-1 \\ 0 \leq i \leq n-1}} a_i b_{j-i} \right) X^j$$

Bon, de premier abord, ça n'a pas l'air de beaucoup nous aider, le produit de polynômes est  $\mathcal{O}(n^2)$ .

Mais pour un polynôme, du fait de leur rigidité, on peut juste évaluer leur produit en un nombre fini de points pour retrouver les coefficients.

Et donc l'idée si on avait un nombre discret de coefficients serait de les traiter comme les coefficients d'un polynôme, on les évalue un certain nombre de fois et on multiplie ensemble les évaluations, puis on retrouve les coefficients en résolvant le système de Vandermonde correspondant.

Le problème c'est que cela semble prendre plus de temps, un pivot de Gauss est en  $\mathcal{O}(n^3)$

Mais si on choisit d'évaluer nos valeurs en les racines de l'unité, on a un système plus simple à résoudre.

On va pouvoir faire les opérations en  $\mathcal{O}(n \cdot \log(n))$

Les coefficients que l'on obtient quand on évalue en les  $\omega_n^k = e^{-\frac{2ki\pi}{n}}$  valeurs sont appelés la transformation de Fourier discrète TFD (un équivalent discret de la transformation de Fourier continue). Il existe aussi d'autres algorithmes.

### 2.2.7 Radix-2 decimation-in-time (DIT) - Factorisation Cooley-Tukey

Evaluer notre polynôme  $P(x) = \sum_{i=0}^{n-1} a_i x^i$  en les  $\omega_n^k$  revient à faire la multiplication matricielle suivante :

$$R = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_{n-1} \end{bmatrix} = \begin{bmatrix} P(\omega_n^0) \\ P(\omega_n^1) \\ \vdots \\ P(\omega_n^{n-1}) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_n^1 & \omega_n^2 & \cdots & \omega_n^{n-1} \\ 1 & \omega_n^2 & \omega_n^4 & \cdots & \omega_n^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_n^{n-1} & \omega_n^{2(n-1)} & \cdots & \omega_n^{(n-1)(n-1)} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix}$$

On reconnaît la matrice de Vandermonde transposée associée à l'évaluation en les  $\omega_n^k$

On se restreint au cas où  $n = 2^p$  ( $p \in \mathbb{N}$ ) pour simplifier l'explication même si en théorie ça ne change pas grand chose vu qu'il y a des algorithmes généraux possibles et on peut rajouter des 0 à la fin pour avoir une liste de taille  $2^p$ .

On note la matrice de Vandermonde transposée, qui permet de calculer le DFT pour tous nos coefficients,

$$F_{2^p} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_n^1 & \omega_n^2 & \cdots & \omega_n^{n-1} \\ 1 & \omega_n^2 & \omega_n^4 & \cdots & \omega_n^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_n^{n-1} & \omega_n^{2(n-1)} & \cdots & \omega_n^{(n-1)(n-1)} \end{bmatrix}$$

On peut permuter les lignes, à la fin, ce qu'on a c'est que, du fait des propriétés des racines de l'unité,

si on note,  $D_{2^{p-1}} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \omega_{2^{p-1}} & 0 & \cdots & 0 \\ 0 & 0 & \omega_{2^{p-1}}^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \omega_{2^{p-1}}^{2^{p-1}-1} \end{bmatrix}$

est :

$$R = F_{2^p} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \begin{pmatrix} I_{2^{p-1}} & D_{2^{p-1}} \\ I_{2^{p-1}} & -D_{2^{p-1}} \end{pmatrix} \begin{pmatrix} F_{2^{p-1}} & \mathbf{0} \\ \mathbf{0} & F_{2^{p-1}} \end{pmatrix}$$

Blocs diagonaux, de l'ordre de  $\mathcal{O}(n)$  opérations

Coefficients d'indice pair

Coefficients d'indice impair

Récurrence

On évalue la complexité de l'algorithme.

On note  $u_p$  sa complexité en fonction de  $p$  et  $C_n$  sa complexité en fonction de  $n$ .

Complexité du produit sur la diagonale

$$u_{p+1} = A \cdot 2^{p+1} + 2u_p$$

↑ Traitement des coefficients par récurrence

On factorise par la solution homogène,  $\frac{u_{p+1}}{2^{p+1}} = A + \frac{u_p}{2^p}$

$$\frac{u_p}{2^p} = u_0 + A \cdot p$$

$$u_p = u_0 \cdot 2^p + A \cdot p \cdot 2^p$$

$$\text{Or } p = \log_2 n$$

$$\text{Donc } C_n = u_{\log_2 n} = u_0 \cdot n + A \cdot \log_2 n \cdot n = \mathcal{O}(n \log n)$$

### 2.2.8 IFFT

On peut montrer que :  $F_{2^p}^{-1} = \frac{1}{2^p} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_n^{-1} & \omega_n^{-2} & \cdots & \omega_n^{-n-1} \\ 1 & \omega_n^{-2} & \omega_n^{-4} & \cdots & \omega_n^{-2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_n^{-n-1} & \omega_n^{-2(n-1)} & \cdots & \omega_n^{-(n-1)(n-1)} \end{bmatrix}$

$$A = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = F_{2^p}^{-1} R = F_{2^p}^{-1} \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_{n-1} \end{bmatrix}$$

Le conjugué passe au produit et à la somme, donc aussi pour les matrices (prendre le conjugué d'une matrice c'est prendre le conjugué des termes de la matrice).

$$A = \overline{\overline{A}} = \overline{\overline{F_{2^p}^{-1} R}} = \overline{F_{2^p}^{-1} \overline{R}} = \frac{1}{2^p} \overline{F_{2^p} R}$$

On peut donc utiliser la même technique, en prenant le conjugué avant d'appliquer un FFT et en le prenant après puis en renormalisant.

### 2.2.9 Passage du domaine temporel au domaine fréquentiel

Sans passer trop de temps dessus, le fait de multiplier par ces coefficients spécifiques, revient à décomposer en ondes sinusoidales de différentes fréquences et phases notre signal. Il y a beaucoup d'applications dans d'autres domaines. Mais ce qui est remarquable c'est que pour calculer plus rapidement notre résultat. On est passé par une autre représentation.

Maintenant qu'on peut calculer des convolutions rapidement, revenons à notre problème.

#### 2.2.10 Corrélation croisée

Si on définit  $\tilde{f}(t) = f(-t)$

La corrélation croisée de  $f$  et  $g$  est  $(g * \tilde{f})(t) = \int_{-\infty}^{\infty} \overline{f(x-t)} g(x) dx$

Ou sur une seule période pour une fonction périodique.

On va donc faire glisser les fonctions dans le même sens et multiplier les valeurs de la fonction évaluée en ses points ensemble deux à deux puis faire la somme.

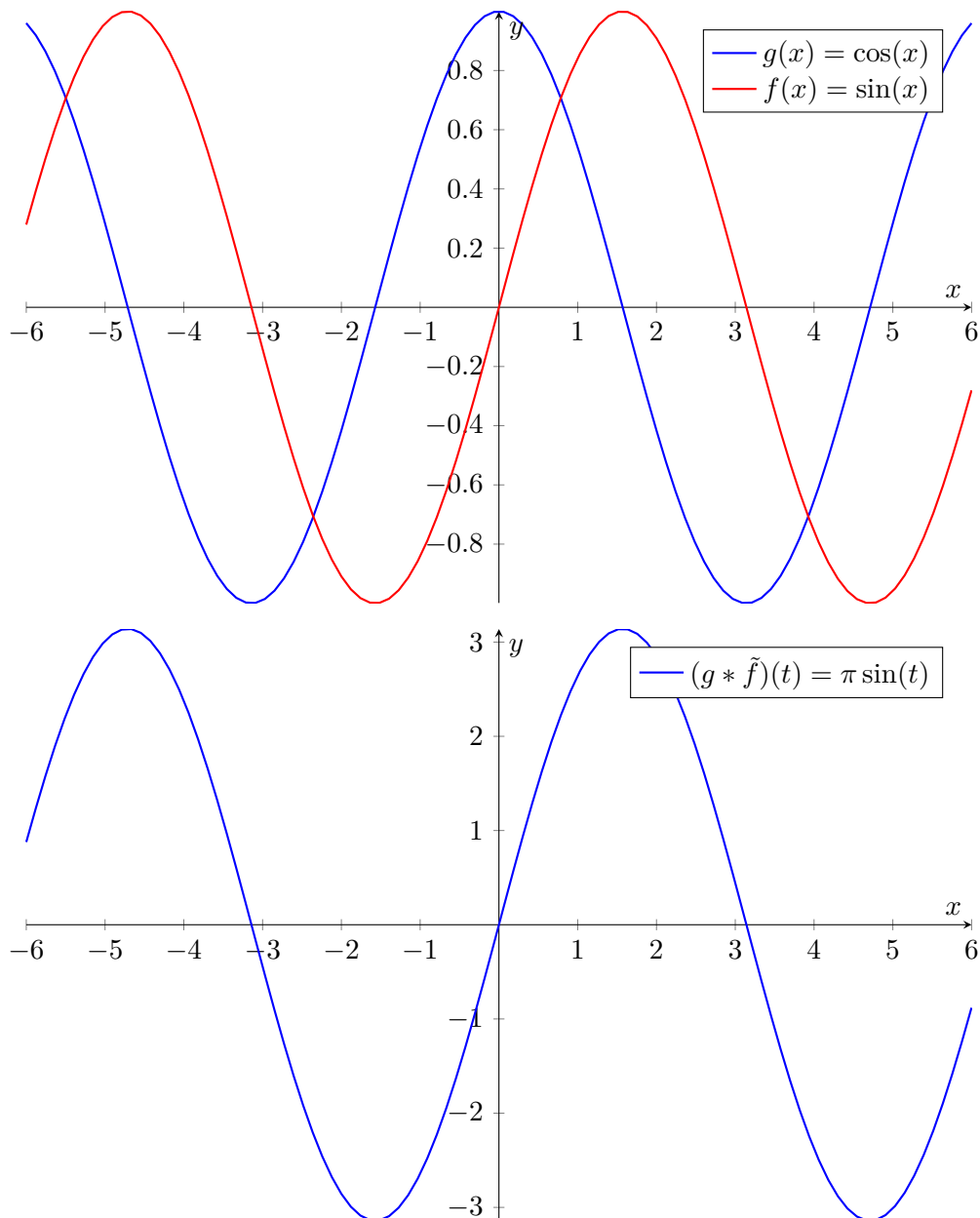
Quand les fonctions sont similaires et bien alignées, alors, on multiplie les grandes valeurs ensemble et les petites valeurs ensemble et donc on a résultat grand, alors que quand elles ne le sont pas, on multiplie les grandes valeurs avec les petites valeurs et on aura un résultat plus petit.

On prend le conjugué pour que si c'est des fonctions complexes, que quand elles s'alignent elles donnent une plus grande valeur au lieu d'une plus petite à cause de la valeur imaginaire.

Et donc on ne regardera que la partie réelle et à quel point elle est grande.

Donc si on a une fonction  $f$  et une fonction  $g$  où  $g$  est dans  $f$  mais avec un certain retard  $\tau$  alors  $(g * \tilde{f})$  sera maximale en  $\tau$ .

Il s'agit donc d'un produit scalaire entre les deux familles de nombres dans le cas discret pour chaque décalage. Le produit scalaire des deux fonctions étant donc maximal quand elles se "ressemblent" le plus.



Ainsi, si on a un enregistrement de référence pour nos différentes ondes  $P$  et  $S$  alors on peut utiliser la corrélation croisée entre eux et le signal en temps réel pour détecter ces ondes et mesurer avec exactitude le temps d'arrivée de ces ondes.

## 2.3 Modélisation de la propagation des ondes sismiques

### 2.3.1 Calcul du temps de propagation selon iasp91

On va utiliser un modèle de vitesse de propagation des différentes ondes sismiques selon la profondeur et tracer des rayons pour calculer le temps de propagation. On remarque que la courbure qu'on obtient est due à la réfraction. L'onde va emprunter le chemin le plus court, même si cela veut dire aller plus en profondeur d'abord.

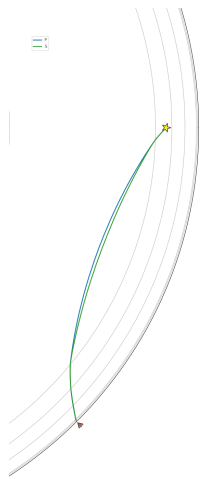


FIGURE 6 : TauPy

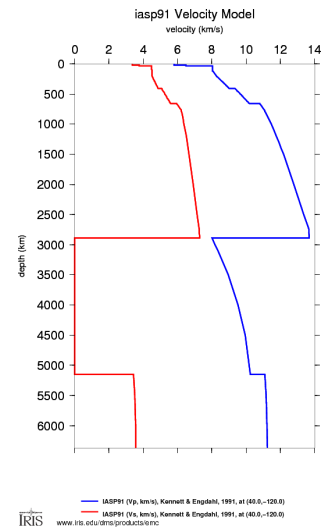


FIGURE 7 : iasp91

### 2.3.2 Tabulation et interpolation

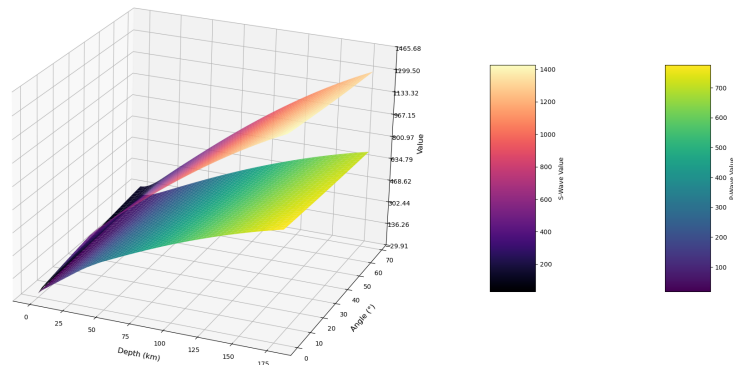


FIGURE 8 : Visualization des deux tables précalculées

## 2.4 Multilatération

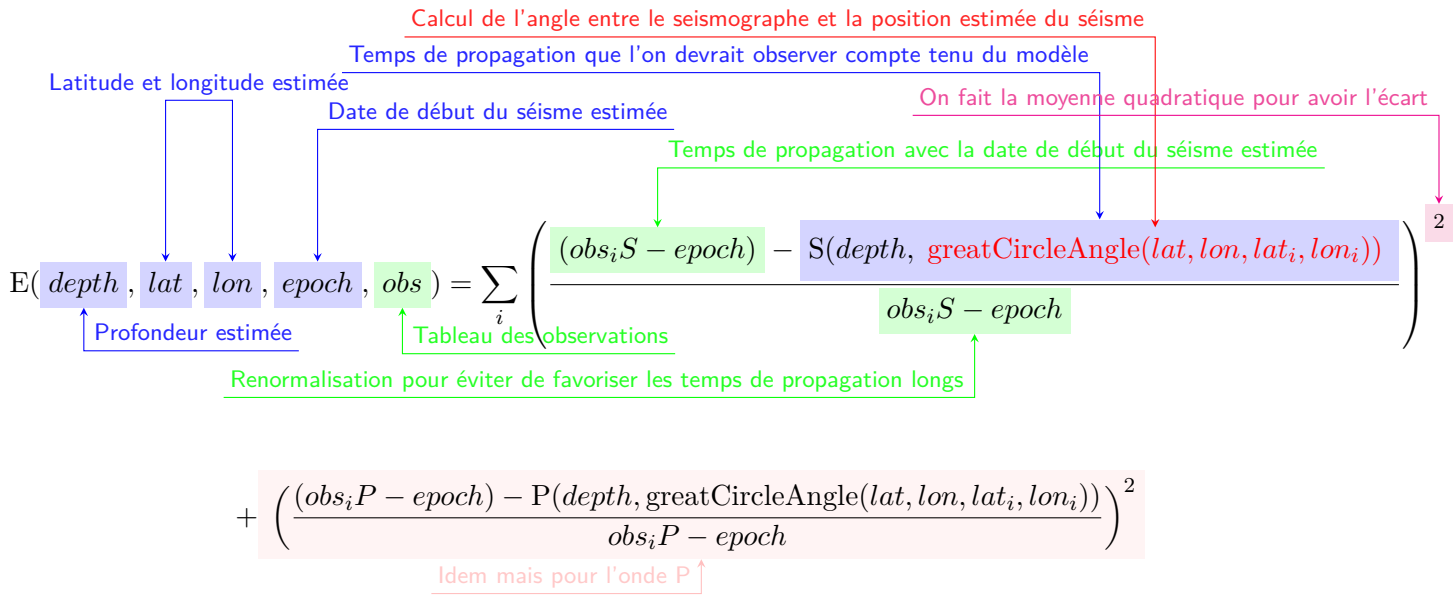
Une fois qu'on a une table des différentes vitesses de propagation pour les différentes profondeurs et les ondes P et S ainsi que nos mesures, on peut résoudre l'équation avec nos mesures.

Sauf qu'on a plusieurs mesures donc on veut une moyenne pour nos mesures pour avoir le meilleur résultat et nos équations ne sont pas linéaires comme vous pouvez le constater sur la diapositive précédente.

Ainsi, on va utiliser une méthode pour trouver le minimum sur une fonction d'erreur.

### 2.4.1 Fonction d'erreur

Voici notre fonction d'erreur :



```

116 func ErrorFunction(observations *[]Observation, guess Guess) float64 {
117     cumuErrorSquared := 0.0
118     for _, observation := range *observations {
119         diffAngle := GreatCircleAngle(guess.Lat, guess.Lon, observation.Lat, observation.Lon)
120         pDelay := observation.PWaveArrival - guess.Epoch
121         sDelay := observation.SWaveArrival - guess.Epoch
122         cumuErrorSquared += Square((sDelay - InterpolateWave(guess.Depth, diffAngle)) / sDelay) + Square((pDelay - InterpolateWave(guess.Depth, diffAngle)) / pDelay)
123     }
124     return cumuErrorSquared // no unit
125 }

```

FIGURE 9 : Implémentation de la fonction d'erreur

### 2.4.2 Méthode de Nelder-Mead

La méthode d'optimisation que j'utilise est celle de Nelder-Mead. Elle permet de trouver un minimum local dans une fonction continue avec une recherche directe heuristique.

Elle utilise ce qu'on appelle des simplexes, c'est la généralisation des triangles à plus de dimensions, en dimension 1 c'est une ligne, en dimension 3 ça correspond à un tétraèdre, à chaque fois en dimension  $n$ , il y a  $n + 1$  côtés (il y a 4 dimensions ici donc il faudra  $n + 1 = 5$  cotés au simplexe).

L'approche la plus simple consiste à remplacer le plus mauvais point par un point reflétant le centroïde des  $n$  points restants. Si ce point est meilleur que le meilleur point actuel, nous pouvons essayer de l'étirer de manière exponentielle le long de cette ligne. En revanche, si ce nouveau point n'est pas beaucoup mieux que la valeur précédente, nous traversons une vallée et nous rétrécissons le simplexe vers un meilleur point.

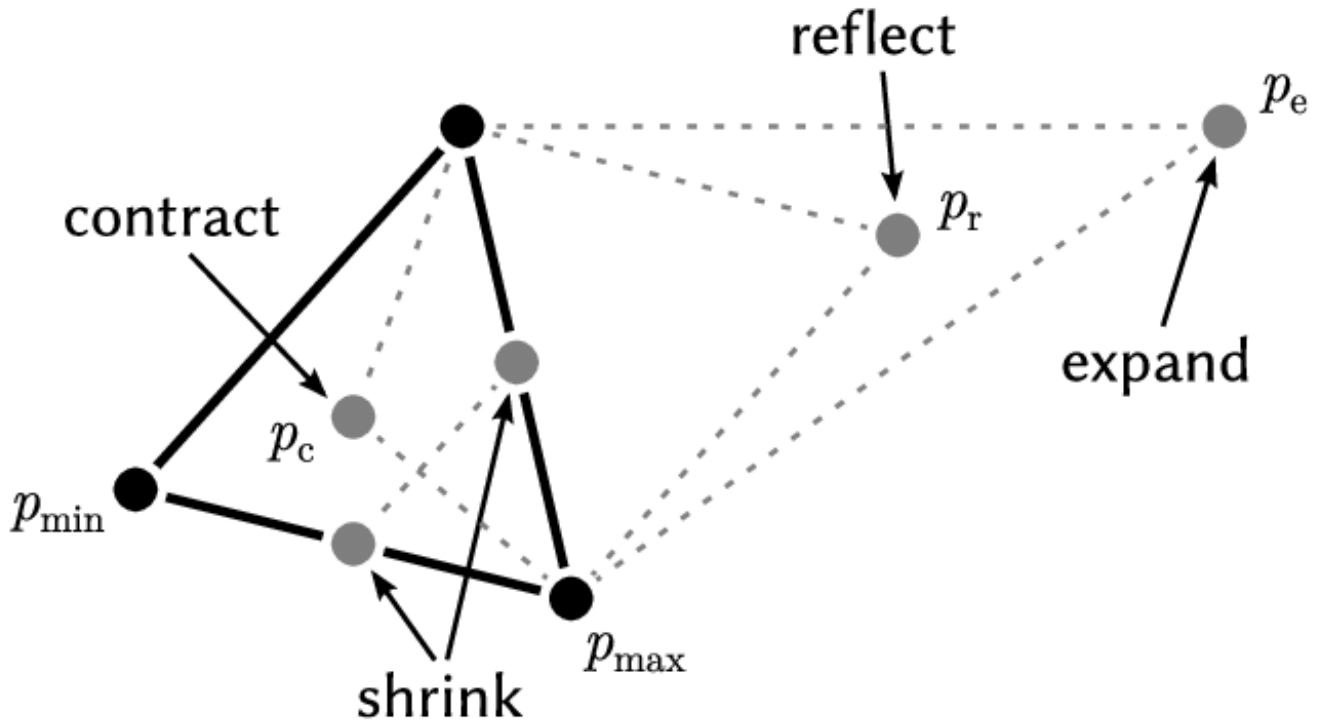


FIGURE 10 : Une itération de Nelder-Mead sur un espace de dimension 2

## 2.5 Magnitude sismique

On utilise l'échelle logarithmique de Richter pour quantifier la magnitude d'un séisme pour laquelle on a la formule suivante :

$$M_L = \log_{10} \left[ \frac{A}{A_0(\delta)} \right]$$

où  $A$  correspond à l'amplitude maximale mesurée (en m) par le sismographe et  $A_0(\delta)$  un coefficient de correction qui dépend de la distance ( $\delta$ ) à l'épicentre et dont le calcul diffère selon les modèles employés (généralement on utilise une table de corrélation empirique).

Pour les grands séismes on préfère utiliser la magnitude de moment qui rend mieux compte de l'énergie du séisme plutôt que l'échelle de Richter (magnitude locale), mais elle est plus compliquée à calculer et nécessite une étude approfondie du séisme (largeur et longueur de la faille, rigidité du milieu, déplacement sur la faille) et n'est donc pas adaptée à un système de prévention en temps réel.

Il existe plusieurs formules et tables empiriques pour calculer la magnitude locale d'un séisme.

On utilisera la formule empirique de Tsuboi (Université de Tokyo) :

$$M_L = \log_{10} A + 1.73 \log_{10} \Delta - 0.83$$

où  $A$  est l'amplitude maximale en micromètres et  $\Delta$  est la distance en kilomètres.

Les différentes valeurs obtenues permettent de quantifier les dégâts provoqués par les séismes :

Magnitude	Description	MMI Typique	Effets Moyens du Séisme
1.0 - 1.9	Micro	I	Micro-séismes, non ressentis. Enregistrés par les sismographes.
2.0 - 2.9	Mineur	I	Légèrement ressenti par certaines personnes. Aucun dommage aux bâtiments.
3.0 - 3.9	Léger	II à III	Souvent ressenti, mais cause rarement des dégâts. Secousses perceptibles d'objets à l'intérieur.
4.0 - 4.9	Faible	IV à V	Secousses intérieures notables et bruits de cliquetis. Légèrement ressenti à l'extérieur. Dégâts minimes possibles.
5.0 - 5.9	Modéré	VI à VII	Peut endommager les bâtiments mal construits ; ressenti par tous. Peu ou pas de dégâts aux bâtiments solides.
6.0 - 6.9	Fort	VII à IX	Dégâts modérés aux structures solides ; dégâts sévères aux structures faibles. Ressenti sur de grandes régions.
7.0 - 7.9	Majeur	VIII ou plus	Dégâts majeurs et effondrements possibles. Dommages concentrés dans un rayon de 250 km.
8.0 - 8.9	Très fort	VIII+	Destructions majeures à totales. Dommages sur des zones très vastes. Ressenti à très grande distance de l'épicentre.
9.0 - 9.9	Extrême	XII	Destruction quasi-totale, dégâts graves ou effondrement de tous les bâtiments. Modification du relief.

### 3 Toutes les pièces mises ensembles

#### 3.1 Fonctionnement

Ainsi, le programme fonctionne comme ceci : Il se connecte à tous les sismographes, commence à streamer leurs mesures. Il fait tourner en continu une corrélation croisée entre les signaux et plusieurs signaux de référence pour les ondes S et P. Si suffisamment de sismographes dans une zone géographique proche détectent des ondes P et S alors une détection d'événement sismique est enregistrée. Ainsi, le programme met dans une liste les sismographes qui ont détecté cet événement (donc leur position), et les temps d'arrivée des ondes P et S. À mesure que plus de détecteurs sismiques détectent cet événement, ils sont rajoutés à la liste. À chaque fois que la liste est mise à jour, il calcule avec la méthode de Nelder-Mead la position (longitude,

latitude, profondeur) et temps de séisme qui minimise l'erreur et renvoie celle-ci. Il calcule ensuite à partir de cette position et l'amplitude des mesures la magnitude du séisme est calculée. Toutes ces informations sont renvoyées en temps réel à l'utilisateur et peuvent être utilisées pour alimenter un système d'alerte des populations.

Il renvoie une fois toutes les détections finies un compte rendu final avec ces paramètres ainsi qu'une carte qui montre la position du séisme, la fonction d'erreur superposée sur celle-ci selon la position à temps de début de séisme fixé, ainsi que les détecteurs qui ont permis leur détection.

Par ailleurs, il y a une phase de calibration qui prend en compte les spécificités de chaque détecteur (sensitivité, bruit de fond, signal continu present à tout moment).

```
95953 "AKCZ": {
95954   "Name": "Akaroa Harbour",
95955   "Lat": -43.8721,
95956   "Lon": 172.90983,
95957   "SampleRate": 100,
95958   "DC": 189,
95959   "NoiseFloor": 70,
95960   "Calibrated": true,
95961   "Sensitivity": 167772160
95962 },
```

FIGURE 11 : Les données de stations telles qu'elles sont stockées pour éviter de les recalculer



```
Thu, 14 May 2026 23:16:15 CEST INF Detected wave arrival
connection=rtserve.iris.washington.edu:18000 pWaveArrival=29609
peakAmplitudeUm=0.1251697540283203 sWaveArrival=35281 station=MBB26
```

FIGURE 14 : Arrivée d'une onde

```
Fri, 15 May 2026 00:07:37 CEST WRN Missing station metadata connection=ephesite.ens.fr:18000
Fri, 15 May 2026 00:07:37 CEST WRN Missing station metadata connection=sislink.geofisica.ufrn.br:18000
Fri, 15 May 2026 00:07:37 CEST WRN Missing station metadata connection=rtweb.units.it:18000
Fri, 15 May 2026 00:07:39 CEST INF Registered new station station=TTR url=rtserve.resif.fr:18000
Fri, 15 May 2026 00:07:56 CEST INF Starting cold start calibration
Fri, 15 May 2026 00:07:57 CEST INF Serializing station data to cache
Fri, 15 May 2026 00:07:57 CEST INF Successfully serialized station data to cache
Fri, 15 May 2026 00:07:57 CEST INF Calibration complete
```

FIGURE 15 : Phase de calibration terminée

```
Fri, 15 May 2026 01:17:40 CEST INF Earthquake prediction date=2026-05-15T01:11:28.007 depth_km=-2.707677171266685
detections=5 latitude=16.11988516202063 longitude=-61.697028305587324 magnitude=3.1032857380058068 type="Partial Earthquake Guess"
```

FIGURE 16 : Détection d'un séisme de magnitude 3

Il y a même une interface web pour voir tout ça en temps réel (notamment les traces des sismographes).

### Tracking 475 stations

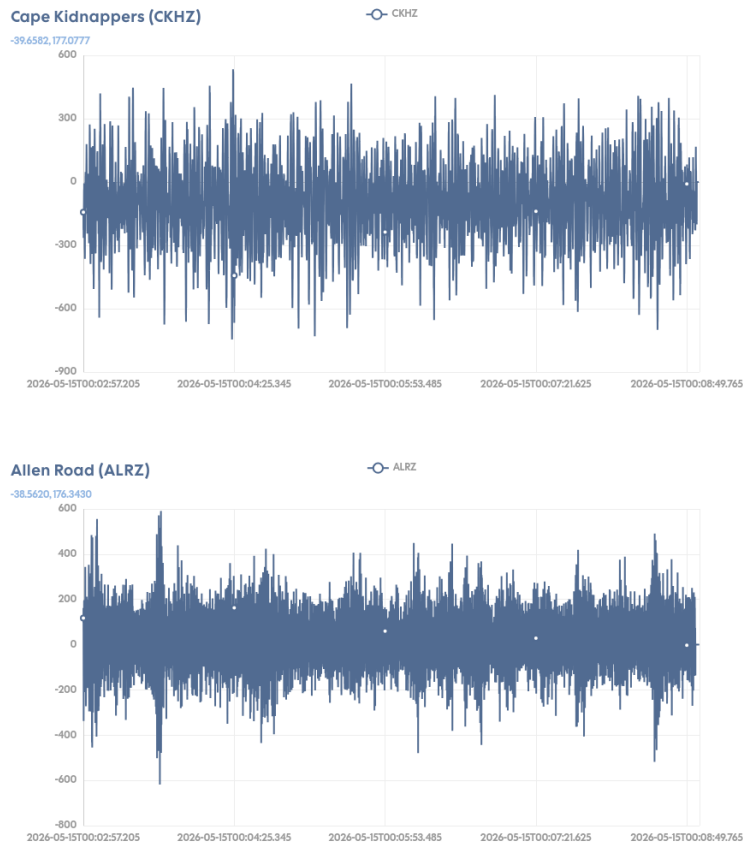


FIGURE 17 : Interface web

### 3.2 Résultats

Voici par exemple un séisme plutôt faible (M6.2) détecté par mon système.

On constate que les positions prédites sont en accord avec celles données par l'USGS (United States Geological Survey) ainsi que les magnitudes.

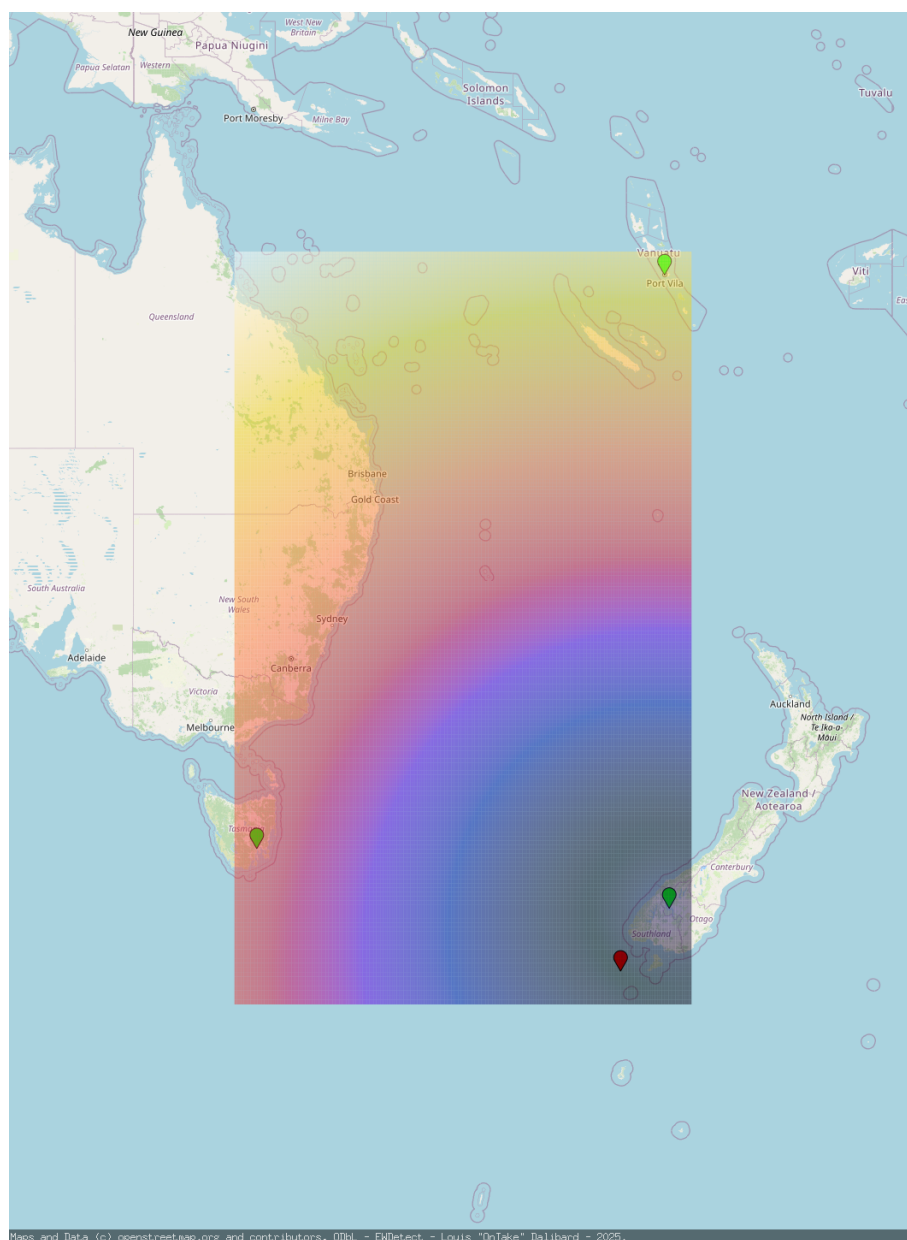


FIGURE 18 : rs2025fwmrzv - Mon estimation - Il donne également une magnitude de M5.9

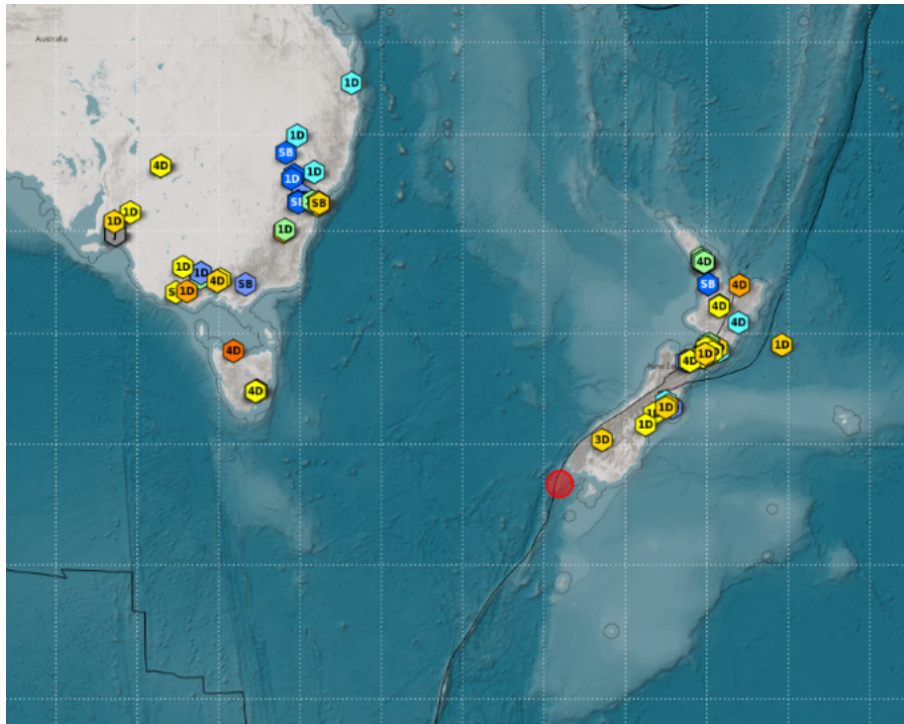


FIGURE 19 : rs2025fwmrzv - Estimation de l'USGS - M6.2 selon eux

## 4 Précision et erreur

### 4.1 Cas réduit avec propagation rectiligne dans un milieu transparent, homogène, isotrope, linéaire à longues distances et 4 détecteurs

Dans certains cas, comme le GPS (cadre dans lequel on va se placer pour l'analogie), les détecteurs (pour le GPS ils émettent mais c'est symétrique) sont suffisamment loin par rapport à l'erreur pour pouvoir linéariser les équations régissant la détermination de la position (il faudrait encadrer la position de l'appareil avec des surfaces de sphères qui deviennent donc des plans à cette distance). Ainsi dans le cas de 4 détecteurs (pour avoir exactement une solution, quatre inconnues  $(x, y, z, t)$  donc quatre équations,  $t$  étant la date de l'envoi du ping par tous les satellites), on a une dépendance linéaire entre les temps d'arrivée et la position. Ainsi si on fixe la position du récepteur à l'origine et qu'on désigne la position des quatre satellites par les vecteurs  $\vec{a}, \vec{b}, \vec{c}$  et  $\vec{d}$ .

Soit  $\vec{M} = (x, y, z)^\top$  notre mesure et  $t$  le temps d'émission du signal par tous les satellites (de façon synchrone).

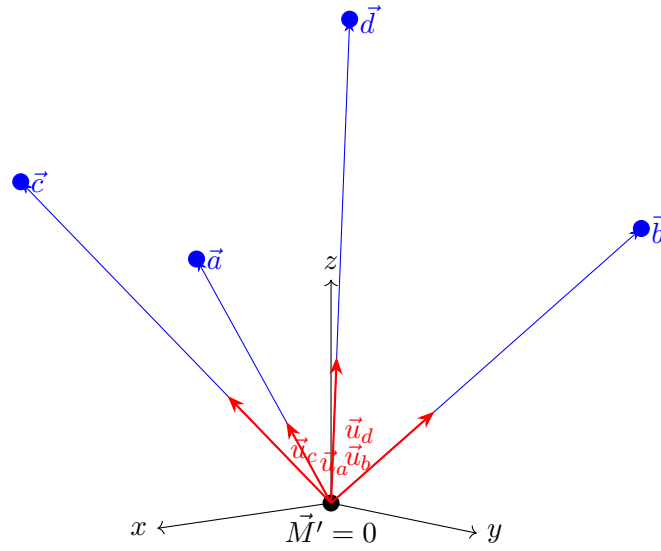


FIGURE 20 : Géométrie GPS : satellites et vecteurs unitaires  $\vec{u}_i$  associés

$$\|\vec{M} - \vec{a}\|^2 = v^2(\tau_a - t)^2 \quad (1)$$

$$\|\vec{M} - \vec{b}\|^2 = v^2(\tau_b - t)^2 \quad (2)$$

$$\|\vec{M} - \vec{c}\|^2 = v^2(\tau_c - t)^2 \quad (3)$$

$$\|\vec{M} - \vec{d}\|^2 = v^2(\tau_d - t)^2 \quad (4)$$

On écrit  $\vec{M} = \vec{M}' + \delta\vec{M}$ ,  $t = t' + \delta t$ ,  $\tau_a = \tau'_a + \delta\tau_a$ ,  $\tau_b = \tau'_b + \delta\tau_b$ ,  $\tau_c = \tau'_c + \delta\tau_c$ ,  $\tau_d = \tau'_d + \delta\tau_d$ .

$\vec{M}'$  est la vraie position, donc  $\vec{M}' = 0$  vu notre choix de coordonnées, et  $t' = 0$  également ;  $\tau'_a, \tau'_b, \tau'_c$  et  $\tau'_d$  sont les temps de propagation entre les satellites et l'origine.

On a donc en développant à l'ordre 1 et en identifiant :

$$-2\langle\delta\vec{M},\vec{a}\rangle=2v^2\tau'_a(\delta\tau_a-\delta t) \quad (5)$$

$$-2\langle\delta\vec{M},\vec{b}\rangle=2v^2\tau'_b(\delta\tau_b-\delta t) \quad (6)$$

$$-2\langle\delta\vec{M},\vec{c}\rangle=2v^2\tau'_c(\delta\tau_c-\delta t) \quad (7)$$

$$-2\langle\delta\vec{M},\vec{d}\rangle=2v^2\tau'_d(\delta\tau_d-\delta t) \quad (8)$$

Ce qui donne en réarrangeant :

$$\langle\delta\vec{M},\vec{a}\rangle+v^2\tau'_a(\delta\tau_a-\delta t)=0 \quad (9)$$

$$\langle\delta\vec{M},\vec{b}\rangle+v^2\tau'_b(\delta\tau_b-\delta t)=0 \quad (10)$$

$$\langle\delta\vec{M},\vec{c}\rangle+v^2\tau'_c(\delta\tau_c-\delta t)=0 \quad (11)$$

$$\langle\delta\vec{M},\vec{d}\rangle+v^2\tau'_d(\delta\tau_d-\delta t)=0 \quad (12)$$

On divise chaque ligne par la distance entre le récepteur au repos et le satellite correspondant ( $v\tau'_i$ ). On note  $\vec{u}_i$  le vecteur unitaire correspondant.

$$\langle\delta\vec{M},\vec{u}_a\rangle+v(\delta\tau_a-\delta t)=0 \quad (13)$$

$$\langle\delta\vec{M},\vec{u}_b\rangle+v(\delta\tau_b-\delta t)=0 \quad (14)$$

$$\langle\delta\vec{M},\vec{u}_c\rangle+v(\delta\tau_c-\delta t)=0 \quad (15)$$

$$\langle\delta\vec{M},\vec{u}_d\rangle+v(\delta\tau_d-\delta t)=0 \quad (16)$$

Si on pose :

$$A=\frac{1}{v}\begin{pmatrix} u_{ax} & u_{ay} & u_{az} & -v \\ u_{bx} & u_{by} & u_{bz} & -v \\ u_{cx} & u_{cy} & u_{cz} & -v \\ u_{dx} & u_{dy} & u_{dz} & -v \end{pmatrix}, \quad X=\begin{pmatrix} \delta x \\ \delta y \\ \delta z \\ \delta t \end{pmatrix}, \quad Y=\begin{pmatrix} \delta\tau_a \\ \delta\tau_b \\ \delta\tau_c \\ \delta\tau_d \end{pmatrix}$$

On a  $AX=Y$ .

Ainsi si on choisit  $Y$  comme variable aléatoire où chaque coordonnée est indépendante, centrée et de  $\mathbb{E}[(\delta\tau_i)^2]=\sigma_i^2$ , alors  $X=A^{-1}Y$  est une variable aléatoire. Calculons l'espérance de sa norme au carré (on peut également calculer l'espérance du carré d'une coordonnée en projetant et en mettant au carré).

$$\|X\|^2=X^\top X=Y^\top A^{-1\top} A^{-1}Y=Y^\top(AA^\top)^{-1}Y$$

$(AA^\top)^{-1}$  étant symétrique positive et inversible (matrice de Gram associée à  $A^\top$ , i.e. aux vecteurs  $\vec{u}_i$  complétés d'un  $-1$ ), on peut donc la diagonaliser dans une BON par le théorème spectral.

$$Q=(\vec{v}_1 \mid \vec{v}_2 \mid \vec{v}_3 \mid \vec{v}_4)$$

où  $\vec{v}_1, \vec{v}_2, \vec{v}_3$  et  $\vec{v}_4$  sont orthonormés.

$$\|X\|^2 = X^\top X = Y^\top A^{-1\top} A^{-1} Y \quad (17)$$

$$= Y^\top Q^\top \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4) Q Y \quad (18)$$

$$= \sum_{i=1}^4 \lambda_i \langle \vec{v}_i, Y \rangle^2 \quad (19)$$

On passe à l'espérance :

$$\mathbb{E}(\|X\|^2) = \sum_{i=1}^4 \lambda_i \mathbb{E}(\langle \vec{v}_i, Y \rangle^2) \quad (20)$$

$$= \sum_{i=1}^4 \lambda_i \sum_{j=1}^4 (v_i)_j^2 \sigma_j^2 \quad (21)$$

La dernière égalité étant donnée par indépendance.

Si on suppose de plus que l'incertitude en temps ( $\sigma_j$ ) est la même pour tous les détecteurs, égale à  $\sigma_\tau$ , alors cette égalité devient :

$$\mathbb{E}(\|X\|^2) = \text{tr}((AA^\top)^{-1}) \sigma_\tau^2$$

On remarque donc que l'erreur explose quand les valeurs propres de  $AA^\top$  sont très petites, i.e. quand les vecteurs  $\vec{u}_i$  sont proches (dilution de précision géométrique).

Si on regarde la forme quadratique induite par  $Q(Y) = Y^\top (AA^\top)^{-1} Y$ , l'ensemble  $Q(Y) = C$  pour une constante  $C > 0$  définit un ellipsoïde. Sur cette surface, plus  $\|Y\|^2$  est petit, plus le résultat est sensible à l'erreur en temps de mesure selon la direction (dans l'espace des temps de mesure) de  $Y$ .

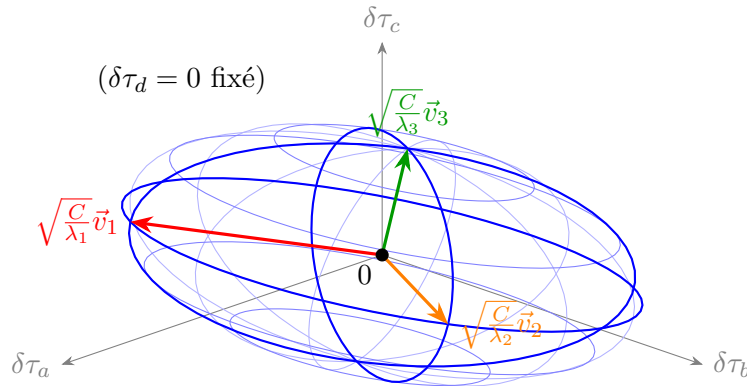


FIGURE 21 : Ellipsoïde  $Q(Y) = Y^\top (AA^\top)^{-1} Y = C$  projeté dans l'espace  $(\delta\tau_a, \delta\tau_b, \delta\tau_c)$ . Les grands axes correspondent aux petites valeurs propres de  $AA^\top$  : une faible norme de  $Y$  dans ces directions suffit à atteindre la surface, signifiant une grande sensibilité à l'erreur.

## 4.2 Méthode de Monte-Carlo pour les cas plus complexes

Pour les cas plus complexes où la position est déterminée par recherche de minimum ou encore que les distances sont suffisamment faibles pour ne pas pouvoir linéariser les équations, les calculs deviennent rapidement compliqués. Ainsi on peut opter pour une méthode de Monte-Carlo. On estime l'espérance de la norme carrée du vecteur position selon nos paramètres en faisant pleins de simulations en faisant varier aléatoirement les valeurs des mesures selon l'erreur qu'on veut.

J'ai écrit un programme pour le faire dans le cas de mesures faites à la main avec la propagation d'un onde sonore (clappement de main) avec 4 microphones synchronisés et placés dans une petite salle (hypothèse de linéarisation non vérifiée).

D'une part je calcule la position par la même méthode qu'avec les séismes et d'autre part je calcule l'erreur par Monte-Carlo selon la disposition des microphones avec la même erreur de mesure pour chacun (distribution uniforme par ailleurs).

On voit bien l'effet de la dilution de précision géométrique.

Voici une configuration écrasée, il y a une grande dilution de précision géométrique dans le plan engendré par les détecteurs.



FIGURE 22 : Rouge : détecteurs, Vert : position mesurée, Bleu : référence (configuration écrasée)

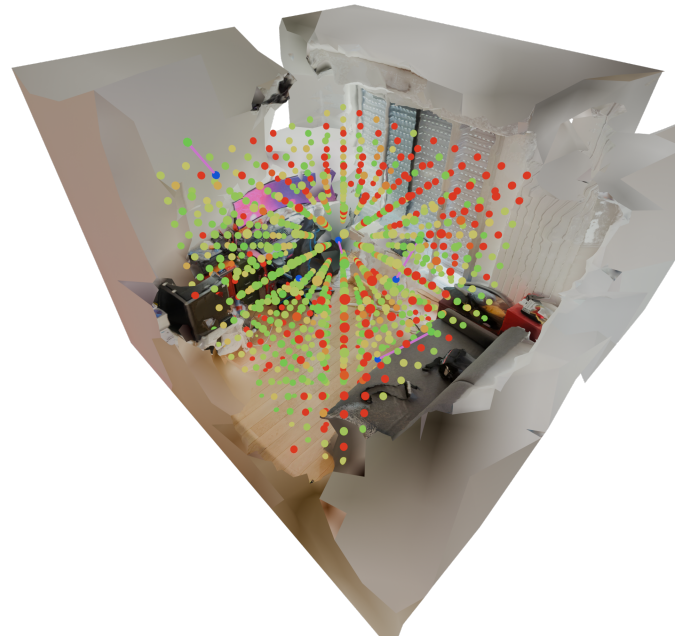


FIGURE 23 : Erreur calculée par Monte-Carlo selon la position dans l'espace à erreur de mesure égale par détecteur (configuration écrasée)

Lorsque la configuration donne des vecteurs moins colinéaires on a bien une meilleure précision.

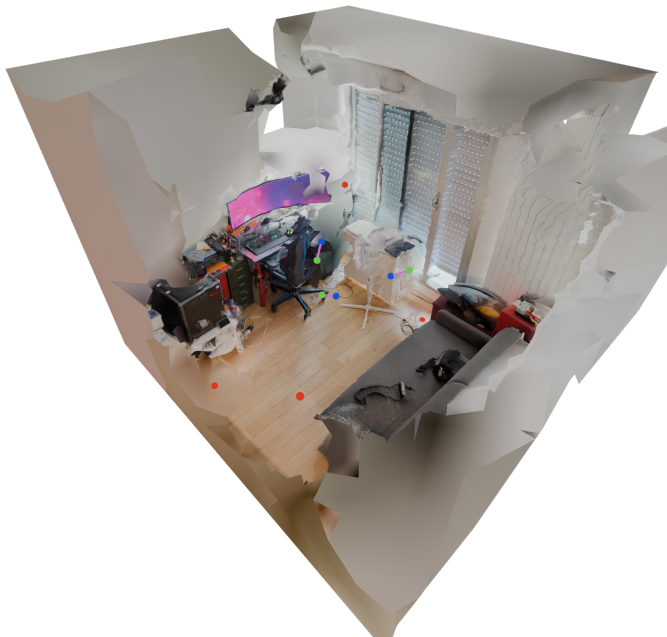


FIGURE 24 : Rouge : détecteurs, Vert : position mesurée, Bleu : référence (configuration favorable)

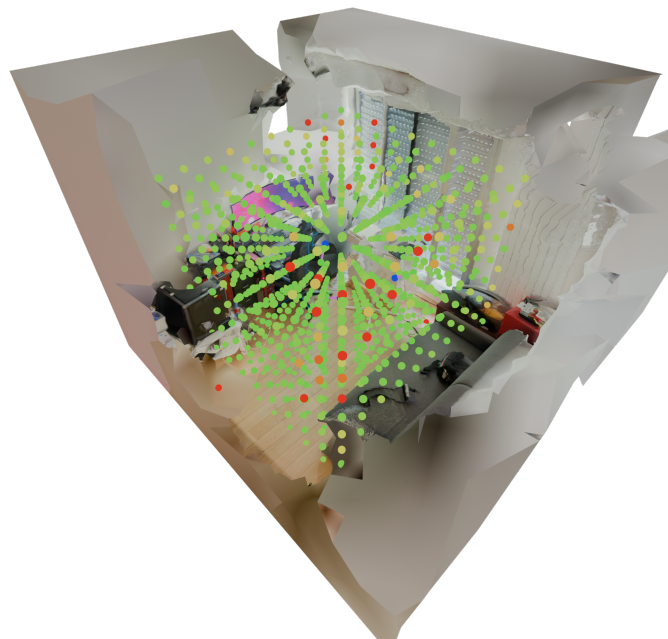


FIGURE 25 : Erreur calculée par Monte-Carlo selon la position dans l'espace à erreur de mesure égale par détecteur (configuration favorable)

## 5 Bonus

De plus ces expérimentations de traitements de signaux m'ont amené à construire un oscilloscope logiciel avec auto-correlation (pour détecter les périodes) et corrélation avec une onde sinusoïdale de la période détectée pour toujours avoir la même phase au centre (<https://github.com/make-42/xyosc>).

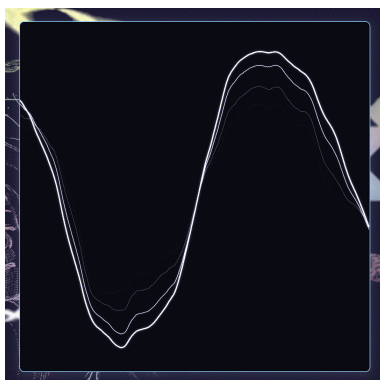


FIGURE 26 : Mode canal unique avec auto-correlation

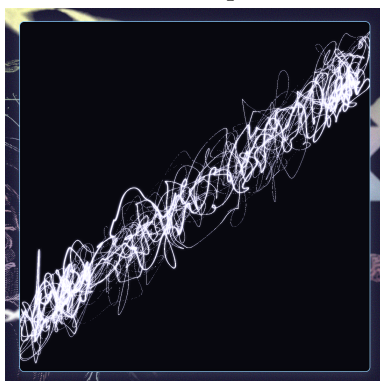


FIGURE 27 : Mode XY

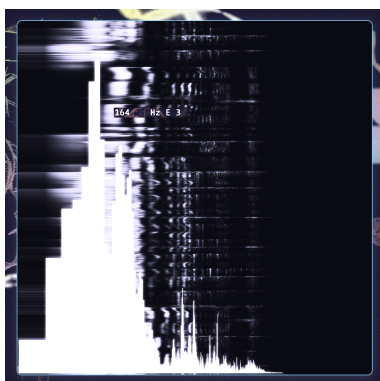


FIGURE 28 : Mode spectrogramme avec détection de note

## 6 Bibliographie

### Références

- [1] Charles Van Loan, *The FFT Via Matrix Factorizations*, Lecture notes, 2010.  
<https://www.cs.cornell.edu/~bindel/class/cs5220-s10/slides/FFT.pdf>
- [2] R.J. Mitchell, *Earthquake Sources*, Lecture notes, Western Washington University.  
[https://www.geol.wvu.edu/rjmitch/L4\\_EQsources.pdf](https://www.geol.wvu.edu/rjmitch/L4_EQsources.pdf)
- [3] University of Hawaii, *Compare, Contrast, and Connect : Seismic Waves and Determining Earth's Structure*.  
<https://manoa.hawaii.edu/exploringourfluidearth/physical/ocean-floor/layers-earth/compare-contrast-connect-seismic-waves-and-determining-earth-s-structure>
- [4] Peter Bormann, *Global 1-D Earth models (IASP91 tables)*, GFZ German Research Centre for Geosciences.  
[https://gfzpublic.gfz-potsdam.de/rest/items/item\\_4031/component/file\\_4032/content](https://gfzpublic.gfz-potsdam.de/rest/items/item_4031/component/file_4032/content)
- [5] Yacine Boussoufa, *Earthquake Data Centers*, GitHub repository.  
<https://github.com/YacineBoussoufa/EarthquakeDataCenters>
- [6] School of Informatics, University of Edinburgh, *Computational Foundations of Cognitive Science : Lecture 15 (Convolutions and Kernels)*.  
[https://www.inf.ed.ac.uk/teaching/courses/cfcs1/lectures/cfcs\\_115.pdf](https://www.inf.ed.ac.uk/teaching/courses/cfcs1/lectures/cfcs_115.pdf)
- [7] Penn State Eberly College of Science, *STAT 510 : Lesson 8.2 - Cross Correlation Functions and Lagged Regressions*, Online course material.  
<https://online.stat.psu.edu/stat510/lesson/8/8.2>
- [8] Jason Cantarella, *Nelder-Mead Method*, Lecture notes.  
<https://jasoncantarella.com/downloads/NelderMeadProof.pdf>
- [9] Tatsuhiko Hara, *Determination of Broadband Moment Magnitude*, IISEE/BRI.  
[https://iisee.kenken.go.jp/lna/download.php?f=2011082925678c01.pdf&n=T0-100-2007\\_Mwp-2-new.pdf&cid=T0-100-2007](https://iisee.kenken.go.jp/lna/download.php?f=2011082925678c01.pdf&n=T0-100-2007_Mwp-2-new.pdf&cid=T0-100-2007)
- [10] Moritz Beyreuther, Robert Barsch, Lion Krischer, Tobias Megies, Yannik Behr and Joachim Wassermann,  
*ObsPy : A Python Toolbox for Seismology*,  
Seismological Research Letters, vol. 81, no. 3, pp. 530–533, 2010.  
doi:10.1785/gssrl.81.3.530
- [11] Elias M. Stein & Rami Shakarchi. *Fourier Analysis : An Introduction*.  
2003. (ISBN 978-0-691-11384-5)
- [12] 3Blue1Brown, *But what is a convolution ?*, YouTube video, 2022.  
<https://www.youtube.com/watch?v=KuXjwB4LzSA>
- [13] Michael Pound, *The Fast Fourier Transform Algorithm*, YouTube video, 2023.  
[https://www.youtube.com/watch?v=toj\\_IoCQE-4](https://www.youtube.com/watch?v=toj_IoCQE-4)