TP3 – Représentation de signaux non stationnaires. Signal analytique, temps-fréquence, ondelettes. Exemple des ondes gravitationnelles

Indication : une archive contenant des signaux et codes utiles au TP est à récupérer sur la page web suivante : `http://www.syncpoint.fr/#Teaching`

On prendra comme signal modèle un chirp (ou sinus glissant) suivant la forme :

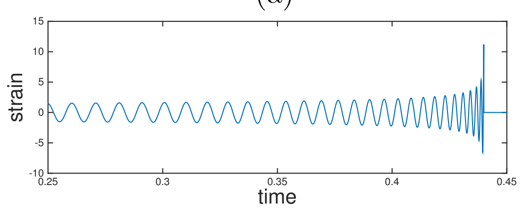
$$s(t) = A\,(t_0 - t)^{-\alpha}\ \cos\left[2\pi C\,(t_0 - t)^{\beta} + \varphi_0\right]\ H(t_0 - t),$$

où H est la fonction de Heaviside ($H(\tau) = 1$ si $\tau > 0$, 0 sinon), et A, C, α, β et t_0 sont des paramètres ajustables.

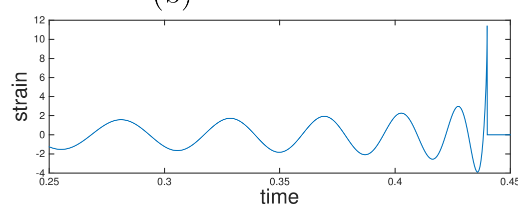
On fixera $\alpha = 1/4$ et $\beta = 5/8$ qui sont les valeurs attendues pour les ondes gravitationnelles en chirps associés à la coalescence d'un système binaire. L'événement GW150914, détecté en septembre 2015 par les instruments de LIGO, nous servira de données réelles. L'article et les données associées à cet évènement sont disponibles dans l'archive dont le lien est donné au début du sujet. La fréquence instantanée de ce chirp est proportionnelle à $(t_0 - t)^{-3/8}$ et le signal diverge en t_0 à la fois en amplitude et en fréquence instantanée. La phase φ_0 est inconnue dans le modèle.

1. **Création des signaux.**
   Créer numériquement ce chirp et régler les paramètres A, C et t_0 pour être dans l'une des deux situations suivantes : signal (a) avec beaucoup d'oscillations et signal (b) avec peu (qui est le cas plus réaliste tandis que les tests numériques seront plus faciles dans un premier temps avec (a)).

(a)

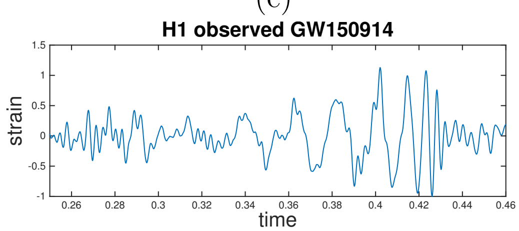


(b)

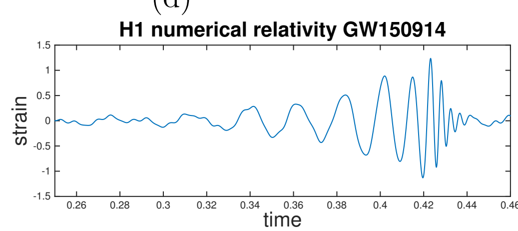


   Ensuite, importer avec la fonction `dlmread` le signal réel (c) observé par l'instrument LIGO d'Hanford (fichier `observed_signal.dat` dans l'archive) et la forme d'onde théorique (d) venant de la relativité (fichier `theoretical_signal.dat` dans l'archive).

(c)



(d)



On pourra regarder toutes les questions suivantes pour ces quatre signaux l'un après l'autre : simulé (a), simulé (b), observé H1 (c) ou forme d'onde H1 (d) selon les équations de la relativité.

2. **Signal analytique.**
   On rappelle que le signal analytique d'un signal réel $s(t)$ est défini comme :

$$z_s(t) = s(t) + i\ \hat{s}(t),$$

   où $\hat{s}(t)$ est la transformée de Hilbert de s. La transformée de Hilbert peut être interprétée comme une opération permettant de déphaser de $\pi/2$ les fréquences négatives et de $-\pi/2$ les fréquences positives. Ainsi, le signal analytique z_s est l'équivalent complexe du signal original s sans les fréquences négatives.

   Le signal original peut être reconstruit en prenant la partie réel du signal analytique :

$$\begin{aligned} x(t) &= \Re\left[z_x(t)\right] \\ &= |z_x(t)|\ \cos\left[\arg z_x(t)\right]. \end{aligned}$$

   On peut donc définir l'amplitude instantanée $A_s(t)$ et la fréquence instantanée $f_s(t)$ comme :

$$\begin{aligned} A_s(t) &= |z_x(t)| \\ f_s(t) &= \frac{1}{2\pi}\frac{\mathrm{d}}{\mathrm{d}t}\left[\arg z_x(t)\right] \end{aligned}$$

   Construire le signal analytique associé à (a-d) à l'aide de la fonction `hilbert`. Calculer son amplitude avec la fonction `abs` et sa phase avec les fonctions `angle` et `unwrap` (qui permet d'éviter des sauts de 2π dans la phase calculée).

   Montrer que pour (a), ce signal donne une bonne estimation numérique de l'amplitude et de la fréquence instantanée.
   Faire de même pour le cas (b) ; que voyez vous ?
   Mêmes questions en ajoutant du bruit (par exemple gaussien), même avec un SNR modéré.

3. **Représentation temps-fréquence**.
   Pour calculer facilement en Matlab les différentes représentations, vous pouvez utiliser la toolbox temps-fréquence TFTB (`http://tftb.nongnu.org`), dont une copie a été fournie dans l'archive. Ajouter simplement le dossier contenant les fichiers `*.m` de la toolbox au *path* de Matlab avec la fonction `addpath`.

(a) Calculer et tracer le spectrogramme de ces signaux.

La fonction qui permet ce calcul est `tfrsp`. Pour l'affichage : éliminer les fréquences négatives, et s'inspirer du code suivant pour afficher le spectrogramme et remettre l'axe des fréquences dans le bon sens :

```
imagesc([tmin tmax],[fmin fmax],spectro);
set(gca,'YDir','normal');
```

(b) Utiliser la fonction `hanning` pour créer une fenêtre d'analyse et passer la comme argument à `tfrsp`. Faire varier la taille de cette fenêtre d'analyse et commenter sur son impact.

(c) Regarder ce que la représentation de Wigner-Ville permettrait d'obtenir.

La fonction qui la calcule est `tfrwv`. Ne pas oublier qu'il y a des interférences quand on calcule la transformée de Wigner-Ville : on en enlève certaines (celles entre les fréquences positives et négatives) en calculant la transformée de Wigner-Ville du signal analytique plutôt que celle du signal original.

(d) Pour améliorer la localisation des chirps, une méthode intéressante est celle de la réallocation dans le plan temps-fréquence (non discutée en cours).

La fonction qui permet ce calcul est `tfrrsp` ; utilisez la et constatez qu'on améliore cette localisation (en perdant cependant sur la structure du bruit qui se localise alors aussi).

(e) On peut aussi étudier la transformée en ondelette, qui permet d'avoir une résolution variant dans le plan temps-fréquence.

On se limitera ici au choix d'une ondelette de Morlet (exponentielle complexe oscillante modulée par une gaussienne), qui permettra une comparaison facile avec les représentations précédentes.

Calculer et afficher le scalogramme (carré de la transformée en ondelette) de chaque signal à l'aide de la fonction `tfrscalo`.

(f) Discuter pour dire, selon vous, quelle représentation est la plus adaptée pour analyser des signaux tels que les chirps regardés ici et estimer leurs propriétés (comme leurs paramètres).