

TP 4 – Série de Fourier, transformée de Fourier - Partie I

S.Gibet

Année 2021-2022

Objectifs : Comprendre les bases des séries de Fourier et la transformée de Fourier discrète (TFD). Application en Python sur des exemples de synthèse.

Ce TP comprend une partie à rédiger sur papier ou document pdf (Exercice 1, questions 1 à 9). L'exercice 1 complet sur les séries de Fourier correspond à un exercice de cours qui peut être fait à la maison, en suivant le cours. Les autres exercices sur la transformée de Fourier discrète (question 2) seront à rendre en fin de semaine. Il est fortement conseillé cependant de poser des questions en TP sur ce deuxième exercice.

Dans les rendus de TP demandés, vous donnerez l'ensemble des programmes en Python en prenant soin d'écrire pour chaque programme et chaque fonction les cartouches et les commentaires. L'objectif pour vous est de réaliser pour chaque TP des sortes de tutoriels permettant de vous relire et de bien comprendre les concepts abordés, l'utilisation des paramètres (à quoi ils correspondent) et quand c'est nécessaire l'interprétation des résultats.

Il est conseillé de réaliser les fonctions dans un fichier à part (*serie_fourier.py*) et de tester ces fonctions dans un autre fichier (*serie_fourier_tests.py*). Vous pouvez également séparer les questions/modules en utilisant le séparateur de bloc de code : `#%%`

1 Série de Fourier – Compréhension du cours

Cet exercice a déjà été traité en cours. Il permet de revoir la théorie sur les séries de Fourier et de comprendre par la pratique le programme qui en découle. L'idée est de remplacer le signal carré du cours par une somme infinie de sinus et cosinus. En pratique, cette somme ne sera pas infinie dans le programme, elle sera tronquée à un nombre fini appelé *ordre*. On considère ainsi le signal périodique carré vu en cours, nommé $s(t)$, et défini par :

$$\begin{cases} s(t) = -1 & \text{si } -\pi < t < 0 \\ s(t) = 1 & \text{si } 0 < t < \pi \end{cases} \quad (1)$$

1. Quelle est la période T_0 de ce signal ? Que valent la pulsation ω_0 et la fréquence f_0 ?
2. On échantillonne ce signal. Comment doit-on choisir la fréquence d'échantillonnage ? Pourquoi ? Tracez cette fonction sur une période (de 0 à 2π).
3. Réécrire s à partir de la décomposition de s en série de Fourier (somme infinie de sinus et cosinus). Quelles conditions doit respecter s ?
4. Comment s'écrit cette décomposition précédente si l'on tronque la somme à un certain ordre ?

5. Donnez les équations générales qui permettent de calculer les coefficients de Fourier a_k et b_k (formules du cours).
6. Calculez le coefficient a_0 . Que représente-t-il ?
7. Calculez les coefficients a_k pour tout k .
8. Calculez les coefficients b_k , en considérant les 2 cas : k pair et k impair.
9. Écrivez sur papier le programme en python qui permet de calculer un terme de la série de Fourier (à l'ordre k).
10. Ecrivez le programme en Python *serie_fourier* en sommant les termes jusqu'à l'ordre donné par le paramètre *ordre*.
11. Testez la fonction avec plusieurs paramètres (nombre de points N et nombre de termes en sinus, soit *ordre*) ; visualisez à chaque fois le résultat grâce à la librairie *matplotlib*.
12. A partir de quel ordre retrouve-t-on le signal carré ?
13. Écrivez un programme qui permet d'afficher de manière superposée les décompositions avec $\text{ordre} = 1, 2, \dots, N$
14. Observez-vous un artefact dans le signal synthétisé ? Dépend-il de l'ordre de la série de Fourier ?