

Μάθημα 4: Ο μετασχηματισμός Fourier

Θωμάς Καμαλάκης

30 Αυγούστου 2023

1 Εισαγωγή

Στο προηγούμενο μάθημα είδαμε ότι η σειρά Fourier μπορεί να χρησιμοποιηθεί για την αναπαράσταση σημάτων στο φασματικό πεδίο αρκεί τα σήματα αυτά να είναι πεπερασμένα σε διάρκεια (δηλαδή να έχουμε $x(t) = 0$ για $t < t_a$ και $t > t_b$) ή να είναι περιοδικά. Όταν ένα σήμα δεν ανήκει σε αυτές τις κατηγορίες, χρησιμοποιούμε τον μετασχηματισμό Fourier για να το περιγράψουμε στο πεδίο των συχνοτήτων. Ο μετασχηματισμός Fourier $\mathcal{F}$, ορίζεται ως:

$$X(f) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt \quad (1)$$

Μπορούμε επίσης να υπολογίσουμε το σήμα $x(t)$ στο πεδίο του χρόνου χρησιμοποιώντας τον αντίστροφο μετασχηματισμό Fourier $\mathcal{F}^{-1}$,

$$x(t) = \mathcal{F}^{-1}\{X(f)\} = \int_{-\infty}^{+\infty} X(f)e^{j2\pi ft} df \quad (2)$$

Υπό μία έννοια, ο μετασχηματισμός Fourier είναι κάτι αντίστοιχο με την σειρά Fourier. Το ολοκλήρωμα της (2) είναι στην ουσία ένα άθροισμα όρων $X(f_m)\exp(j2\pi f_m t)\Delta f$ παρόμοιο με το άθροισμα στην σειρά Fourier.

1.1 Ο μετασχηματισμός Fourier του τετραγωνικού σήματος

Ας υπολογίσουμε τον μετασχηματισμό Fourier του τετραγωνικού σήματος

$$p(t) = \begin{cases} 1 & , -\frac{T_1}{2} \leq t \leq \frac{T_1}{2} \\ 0 & , \text{διαφορετικά} \end{cases} \quad (3)$$

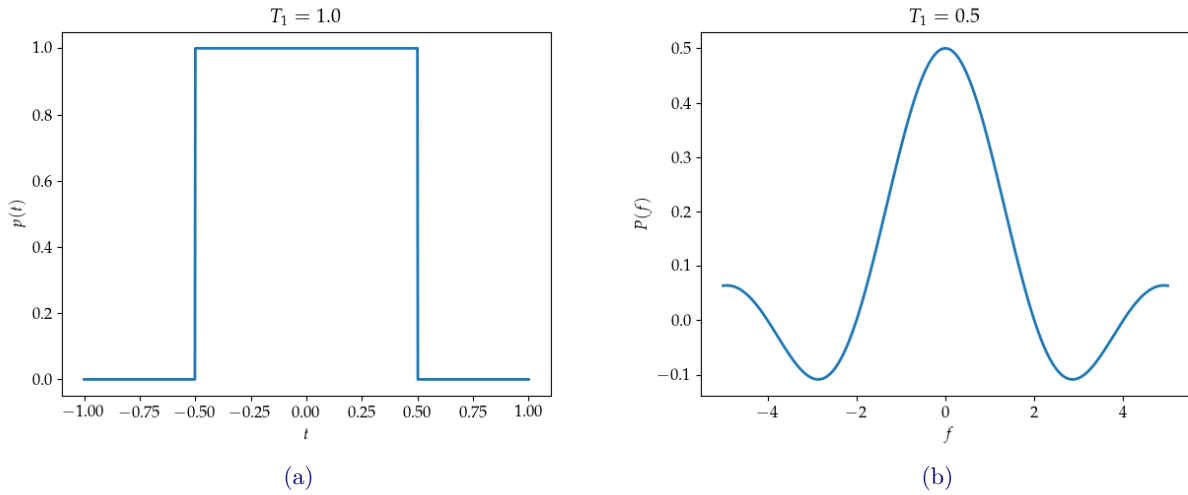
Χρησιμοποιούμε την (1) και έχουμε

$$P(f) = \int_{-\infty}^{+\infty} p(t)e^{-j2\pi ft} dt = \int_{-T_1/2}^{+T_1/2} e^{-j2\pi ft} dt = \frac{e^{-j\pi f T_1} - e^{j\pi f T_1}}{-j2\pi f} = T_1 \frac{\sin(\pi f T_1)}{\pi f T_1} = T_1 \text{sinc}(f T_1) \quad (4)$$

Στην παραπάνω σχέση χρησιμοποιήσαμε τον ορισμό της συνάρτησης $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$ και παρατηρούμε ότι ο μετασχηματισμός Fourier $P(f)$ μοιάζει ως συνάρτηση με τους συντελεστές Fourier P_m που είχαμε υπολογίσει στο προηγούμενο μάθημα.

Χρησιμοποιούμε το listing 1 για να παραστήσουμε γραφικά τον μετασχηματισμό Fourier $P(f)$ του $p(t)$. Το αποτέλεσμα φαίνεται στις εικόνες 1 και 2 για $T_1 = 1$ και $T_1 = 0.5$ αντίστοιχα. Προκύπτει ένα παρόμοιο συμπέρασμα όπως και για τους συντελεστές Fourier: όσο πιο στενός είναι ο παλμός τόσο πιο γρήγορες μεταβάσεις έχει και επομένως τόσο πιο απλωμένο είναι το φάσμα του.

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 plt.rcParams.update({
5     "text.usetex": True,
6     "font.family": "serif",
7     "font.serif": ["Palatino"],
8     "font.size" : 12,
9     "lines.linewidth" : 2,
10 })
11
```



Εικόνα 1: Μετασχηματισμός Fourier του τετραγωνικού παλμού για $T_1 = 1$.

```

12 # Square pulse
13 def square_pulse(t, T1):
14     samples = np.zeros(t.size)
15     i = np.where( np.abs(t) <= 0.5 * T1)
16     samples[ i ] = 1
17     return samples
18
19 # Fourier transform
20 def pulse_fourier(f, T1):
21     return T1 * np.sinc( f * T1 )
22
23 T = 2
24 t = np.linspace(-T/2, T/2, 1000)
25 f = np.linspace(-10/T, 10/T, 1000)
26 T1s = [1, 0.5]
27 plt.close('all')
28 for T1 in T1s:
29     p = square_pulse(t, T1)
30     plt.figure()
31     plt.plot(t, p)
32     plt.title('$T_1=%6.1f$' %T1)
33     plt.xlabel('$t$')
34     plt.ylabel('$p(t)$')
35     plt.figure()
36
37     P = pulse_fourier(f, T1)
38     plt.figure()
39     plt.plot(f, P)
40     plt.xlabel('$f$')
41     plt.ylabel('$P(f)$')
42     plt.title('$T_1=%6.1f$' %T1)

```

Listing 1: fourierpulse.py

Ας δούμε λίγο πιο προσεκτικά τη σχέση μεταξύ διάρκειας παλμού και φασματικού εύρους. Η διάρκεια του παλμού $p(t)$ είναι ίση με T_1 . Ως φασματικό εύρος μπορούμε να ορίσουμε την συχνοτική απόσταση μεταξύ των δύο σημείων γύρω από το μέγιστο όπου το $P(f)$ μηδενίζεται. Οι μηδενισμοί του $P(f)$ καθορίζονται από το sinc στην (4),

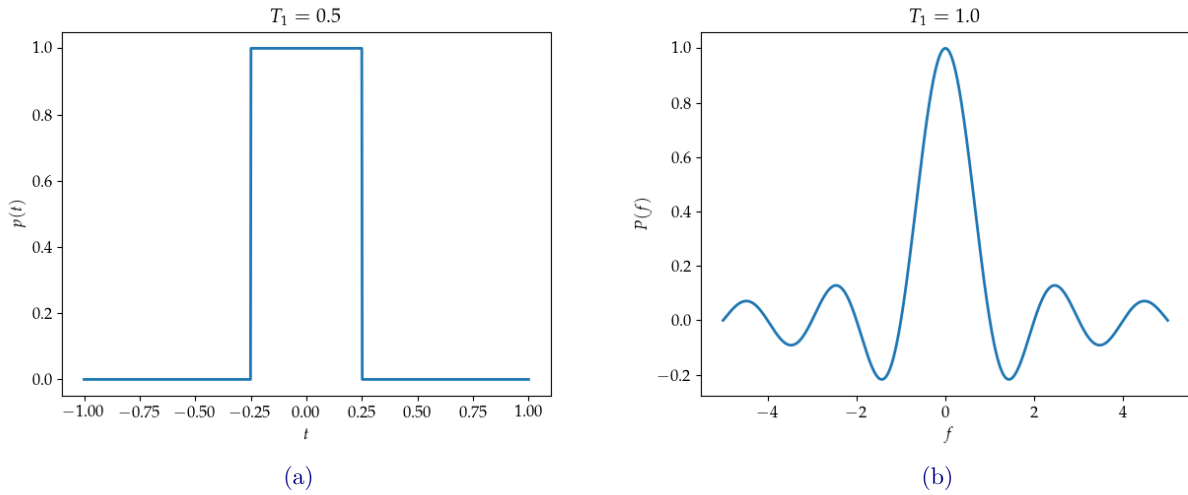
$$\text{sinc}(fT_1) = 0 \quad (5)$$

Το μέγιστο του sinc αντιστοιχεί στο $f = 0$ και είναι ίσο με 1. Οι μηδενισμοί αντιστοιχούν στις συχνότητες,

$$fT_1 = m \quad (6)$$

όπου m ένας οποιοσδήποτε ακέραιος που όμως δεν είναι μηδέν (για $m = 0$ έχουμε $f = 0$ που αντιστοιχεί στο μέγιστο του sinc). Οι δύο μηδενισμοί που είναι πλησιέστεροι στο μέγιστο του sinc θα είναι το $f_l = -1/T_1$ (αριστερά του $f = 0$) και το $f_r = 1/T_1$ (δεξιά του $f = 0$). Το συχνοτικό εύρος B θα είναι ίσο με

$$B = \frac{2}{T_1} \quad (7)$$



Εικόνα 2: Μετασχηματισμός **Fourier** του τετραγωνικού παλμού για $T_1 = 1$.

Το B συχνά αναφέρεται ως *εύρος ζώνης*. Το εύρος ζώνης μπορεί να οριστεί και με διαφορετικό τρόπο όπως θα δούμε και παρακάτω. Προς το παρόν ας δούμε ότι σύμφωνα με την (7), το εύρος ζώνης είναι αντιστρόφως ανάλογο με την διάρκεια του παλμού T_1 : Ένας παλμός με την μισή διάρκεια θα έχει διπλάσιο εύρος ζώνης.

Γενικά στις επικοινωνίες, το εύρος ζώνης είναι μία σημαντική παράμετρος. Στις ασύρματες επικοινωνίες αν εξαιρέσει κανείς περιοχές συχνοτήτων που είναι ελεύθερες π.χ. οι ζώνες του **WiFi** στα 2.4 και τα 5GHz, η χρήση του φάσματος αδειοδοτείται και το εύρος ζώνης είναι αυστηρά καθορισμένο. Ακόμα και στις ελεύθερες ζώνες όμως το εύρος ζώνης καθορίζει την απόδοση του συστήματος: αν διαθέτουμε μεγάλο εύρος ζώνης, χρησιμοποιούμε στενούς παλμούς και μεταδίδουμε τα σήματα μας πιο γρήγορα. Θα ξανααυξητήσουμε τα θέματα αυτά σε επόμενα μαθήματα.

1.2 Μετασχηματισμός **Fourier** ενός φέροντος σήματος

Έχει ενδιαφέρον να υπολογίσουμε τον μετασχηματισμό **Fourier** του φέροντος σήματος:

$$x(t) = A \cos(2\pi f_0 t + \phi) \quad (8)$$

Αν εφαρμόσουμε τη σχέση (1), θα πάρουμε ένα ολοκλήρωμα της μορφής

$$X(f) = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi f t} dt = A \int_{-\infty}^{+\infty} \cos(2\pi f_0 t + \phi) e^{-j2\pi f t} dt \quad (9)$$

Χρησιμοποιώντας την ιδιότητα $2 \cos \theta = e^{j\theta} + e^{-j\theta}$ προκύπτει ότι

$$X(f) = \frac{Ae^{j\phi}}{2} \int_{-\infty}^{+\infty} e^{-j2\pi(f-f_0)t} dt + \frac{Ae^{-j\phi}}{2} \int_{-\infty}^{+\infty} e^{-j2\pi(f+f_0)t} dt \quad (10)$$

Χρησιμοποιώντας την (1) έχουμε:

$$X(f) = \frac{Ae^{j\phi}}{2} \mathcal{F} \{e^{j2\pi f_0 t}\} + \frac{Ae^{-j\phi}}{2} \mathcal{F} \{e^{-j2\pi f_0 t}\} \quad (11)$$

Στην παραπάνω σχέση εμφανίζονται οι μετασχηματισμοί **Fourier** $e^{\pm j2\pi f_0 t}$. Για να τα υπολογίσουμε ας θεωρήσουμε τον αντίστροφο μετασχηματισμό του $\delta(f - f_0)$. Σύμφωνα με την (2) θα έχουμε

$$\mathcal{F}^{-1} \{\delta(f - f_0)\} = \int_{-\infty}^{+\infty} \delta(f - f_0) e^{j2\pi f t} df = e^{j2\pi f_0 t} \quad (12)$$

Η παραπάνω σχέση δείχνει ότι ο αντίστροφος μετασχηματισμός **Fourier** του $\delta(f - f_0)$ είναι το $e^{j2\pi f_0 t}$, επομένως ο μετασχηματισμός **Fourier** του $e^{j2\pi f_0 t}$ θα είναι το $\delta(f - f_0)$, δηλαδή:

$$\mathcal{F} \{e^{j2\pi f_0 t}\} = \delta(f - f_0) \quad (13)$$

Με τον ίδιο τρόπο υπολογίζουμε τον αντίστροφο μετασχηματισμό **Fourier** του $\delta(f + f_0)$,

$$\mathcal{F}^{-1}\{\delta(f + f_0)\} = \int_{-\infty}^{+\infty} \delta(f + f_0) e^{j2\pi f t} df = e^{-j2\pi f_0 t} \quad (14)$$

οπότε ο μετασχηματισμός **Fourier** του $e^{-j2\pi f_0 t}$ θα είναι το $\delta(f + f_0)$,

$$\mathcal{F}\{e^{-j2\pi f_0 t}\} = \delta(f + f_0) \quad (15)$$

Τελικά θα έχουμε

$$X(f) = \frac{Ae^{j\phi}}{2} \delta(f - f_0) + \frac{Ae^{-j\phi}}{2} \delta(f + f_0) \quad (16)$$

Σύμφωνα με την (16), ο μετασχηματισμός **Fourier** του φέροντος σήματος αποτελείται από δύο συναρτήσεις $\delta(f \pm f_0)$. Προκύπτει δηλαδή ότι το φάσμα του $X(f)$ είναι παντού μηδέν εκτός από τις $f = \pm f_0$.

1.3 Μετασχηματισμός **Fourier** του εκθετικού σήματος

Θεωρούμε το σήμα:

$$x(t) = \begin{cases} e^{-\alpha t} & t \geq 0 \\ 0 & , \text{διαφορετικά} \end{cases} \quad (17)$$

όπου το α είναι μία θετική σταθερά, $\alpha > 0$. Χρησιμοποιώντας την (1) θα έχουμε

$$X(f) = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi f t} dt = \int_0^{+\infty} e^{-(\alpha + j2\pi f)t} dt = \left[-\frac{e^{-(\alpha + j2\pi f)t}}{\alpha + j2\pi f} \right]_0^{+\infty} \quad (18)$$

Δεδομένου ότι $\alpha > 0$ θα έχουμε $e^{-(\alpha + j2\pi f)t} \rightarrow 0$ όταν $t \rightarrow +\infty$. Οπότε:

$$X(f) = \frac{1}{\alpha + j2\pi f} \quad (19)$$

2 Αριθμητικός υπολογισμός του μετασχηματισμού **Fourier**

2.1 Ο μετασχηματισμός **DFT**

Χρειαζόμαστε καταρχήν έναν αποτελεσματικό τρόπο για να υπολογίζουμε τον μετασχηματισμό **Fourier** για τα σήματα μας. Θα μπορούσαμε να χρησιμοποιήσουμε αριθμητική ολοκλήρωση τόσο για την (1) όσο και για την (2). Ωστόσο δεδομένου ότι θα πρέπει να υπολογίσουμε το αποτέλεσμα για πολλές συχνότητες f ή πολλές χρονικές στιγμές t καταλαβαίνουμε ότι ο χρόνος υπολογισμού ενδέχεται να είναι πολύ μεγάλος.

Ας περιοριστούμε σε συγκεκριμένες N συχνότητες f_k μέσα σε ένα διάστημα $[-F_{\max}, F_{\max}]$ που δίνονται από την σχέση $f_k = -F_{\max} + k\Delta f$ με $\Delta f = 2F_{\max}/N$. Επίσης θεωρούμε ότι το σήμα $x(t)$ έχει αμελητέες τιμές εκτός ενός μεγάλου διαστήματος $[-T_{\max}, T_{\max}]$. Μέσα στο διάστημα αυτό θεωρούμε N χρονικές στιγμές $t_n = -T_{\max} + n\Delta t$ όπου $\Delta t = 2T_{\max}/N$. Επιλέγουμε την παρακάτω σχέση να συνδέει τα Δf και Δt :

$$\Delta t \Delta f = \frac{1}{N} \quad (20)$$

Αν ισχύει η παραπάνω σχέση θα έχουμε

$$F_{\max} T_{\max} = (N/2)^2 \Delta f \Delta t = N/4 \quad (21a)$$

$$F_{\max} \Delta t = (N/2) \Delta f \Delta t = 1/2 \quad (21b)$$

$$T_{\max} \Delta f = (N/2) \Delta t \Delta f = 1/2 \quad (21c)$$

Στην περίπτωση αυτή θα έχουμε

$$f_k t_n = (-F_{\max} + k\Delta f)(-T_{\max} + n\Delta t) = F_{\max} T_{\max} - F_{\max} n\Delta t - T_{\max} k\Delta f + kn\Delta t \Delta f = N/4 - k/2 - n/2 + kn/N \quad (22)$$

Μπορούμε να προσεγγίσουμε το ολοκλήρωμα στην (1) με το εξής άθροισμα,

$$X(f_k) \approx \Delta t \sum_{n=0}^{N-1} x(t_n) \exp(-j2\pi f_k t_n) = \Delta t \sum_{n=0}^{N-1} x(t_n) \exp(-j\pi N/2 + j\pi k + j\pi n + j2\pi kn/N) \quad (23)$$

Αν γράψουμε λίγο διαφορετικά την παραπάνω σχέση θα έχουμε:

$$X(f_k)e^{-j\pi k} = \Delta t e^{-j\pi N/2} \sum_{n=0}^{N-1} x_n e^{j\pi n} \exp(-j2\pi kn/N) \quad (24)$$

όπου

$$x_n = x(t_n) \quad (25)$$

Το x_n είναι το σήμα που προκύπτει από την δειγματοληψία του $x(t)$ στις χρονικές στιγμές $t = t_n$. Στην ουσία είναι ένα διακριτό σήμα: αλλάζει τιμή μόνο τις χρονικές στιγμές $t = t_n$ και μπορούμε να βάλουμε τις τιμές του σε ένα μονοδιάστατο πίνακα. Στα μαθηματικά το x_n ονομάζεται και ακολουθία καθώς τελικά οι τιμές του εξαρτώνται από έναν φυσικό αριθμό n . Στην παραπάνω σχέση το άθροισμα που εμφανίζεται περιγράφει έναν διακριτό μετασχηματισμό **Fourier (Discrete Fourier Transform - DFT)**. Γενικότερα εάν έχουμε δείγματα y_n τότε ο **DFT** του y_n ορίζεται ως:

$$Y_k = \text{DFT}\{y_n\} = \sum_{n=0}^{N-1} y_n \exp(-j2\pi kn/N) \quad (26)$$

Παρατηρούμε επομένως ότι αν έχουμε ένα γρήγορο τρόπο να υπολογίζουμε τον **DFT** μπορούμε να τον χρησιμοποιήσουμε για να προσεγγίσουμε και τον μετασχηματισμό **Fourier (1)**.

Στη συνέχεια θα δούμε πως υπολογίζεται ο **DFT** στην **Python**. Μπορούμε να αντιστρέψουμε τον **DFT** και να υπολογίσουμε τα y_n από τα Y_k . Αποδεικνύεται ότι ο αντίστροφος μετασχηματισμός **DFT (inverse DFT - IDFT)** δίνεται από την σχέση:

$$y_n = \text{IDFT}\{Y_k\} = \frac{1}{N} \sum_{k=0}^{N-1} Y_k \exp(j2\pi kn/N) \quad (27)$$

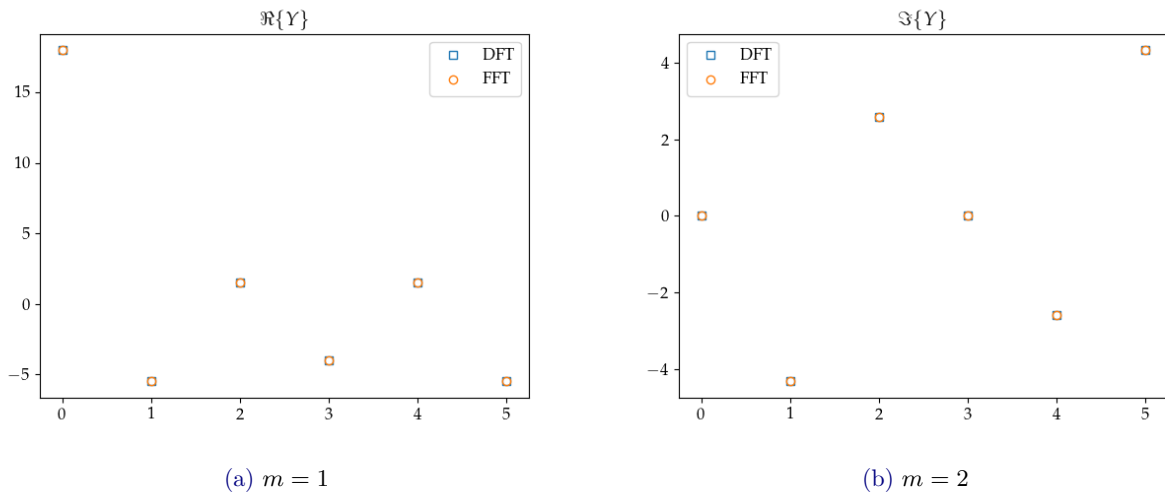
2.2 Ο αλγόριθμος FFT

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 plt.rcParams.update({
5     "text.usetex": True,
6     "font.family": "serif",
7     "font.serif": ["Palatino"],
8     "font.size" : 12,
9     "lines.linewidth" : 2,
10 })
11
12 y = np.array([1, 3, 5, 6, 1, 2])
13 Y = np.fft.fft(y)
14 N = y.size
15 Y2 = np.zeros(N, dtype = complex)
16
17 for n in range(N):
18     for k in range(N):
19         Y2[n] += y[k] * np.exp( -1j * 2 * np.pi * n * k / N )
20
21 plt.close('all')
22 plt.figure()
23 plt.plot( np.real(Y2), 's', fillstyle='none', label='DFT')
24 plt.plot( np.real(Y), 'o', fillstyle='none', label='FFT' )
25 plt.legend()
26 plt.title('$\text{Re}\{Y\}$')
27
28 plt.figure()
29 plt.plot( np.imag(Y2), 's', fillstyle='none', label='DFT')
30 plt.plot( np.imag(Y), 'o', fillstyle='none', label='FFT' )
31 plt.legend()
32 plt.title('$\text{Im}\{Y\}$')

```

Listing 2: fftexample.py



Εικόνα 3: Ο αλγόριθμος FFT

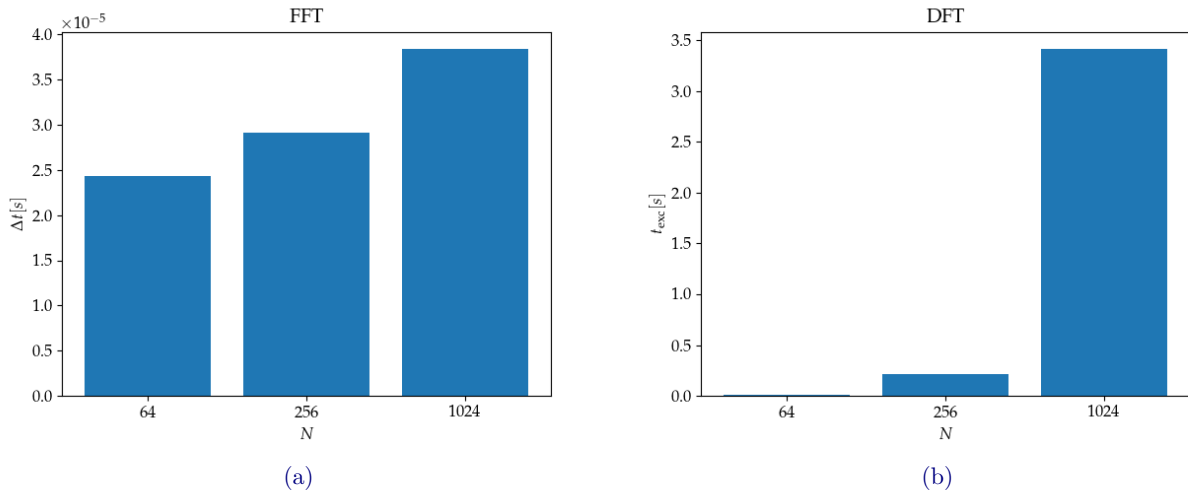
Στη βιβλιοθήκη `numpy` ο `DFT` υπολογίζεται με την βοήθεια του γρήγορου μετασχηματισμού Fourier (fast Fourier transform - FFT). Ο `FFT` είναι ένας πολύ αποτελεσματικός τρόπος να υπολογίζεται ο `DFT`. Χρησιμοποιούμε το `listing` στην 2 όπου συγκρίνουμε τον παραδοσιακό τρόπο υπολογισμού του αθροίσματος (26) με ένα διπλό `for` και το αποτέλεσμα που παράγει η συνάρτηση `fft` της `numpy` στην περίπτωση της ακολουθίας y_n με στοιχεία $[1, 3, 5, 6, 1, 2]$. Όπως δείχνει και η εικόνα 3, τα αποτελέσματα ταυτίζονται.

Ας συγκρίνουμε όμως τους χρόνους εκτέλεσης των δύο προσεγγίσεων, δηλαδή της χρήσης της δικής μας υλοποίησης με το διπλό `for loop` και την συνάρτηση `fft` της `numpy`. Για τον σκοπό αυτό χρησιμοποιούμε το `listing` 3 στο οποίο αξιοποιούμε την βιβλιοθήκη `time` για να μετρήσουμε τον χρόνο που διαρκεί ο κάθε υπολογισμός. Τα αποτελέσματα φαίνονται στην εικόνα 4. Προκύπτουν δύο συμπεράσματα: καταρχήν ο χρόνος εκτέλεσης του διπλού `for loop` είναι πέντε τάξεις μεγέθους μεγαλύτερος από ότι ο αντίστοιχος της συνάρτησης `fft`. Επίσης όσο αυξάνουμε το πλήθος των δειγμάτων της ακολουθίας N , ο χρόνος εκτέλεσης αυξάνει πολύ γρηγορότερα στην δική μας περίπτωση από ότι στην `fft`. Θα πρέπει να θυμόμαστε ότι η βιβλιοθήκη `numpy` χρησιμοποιεί την γλώσσα `C` για τους αριθμητικούς υπολογισμούς και ως `compiled` γλώσσα είναι πολύ πιο γρήγορη. Ρίξτε μία ματιά στην υλοποίηση της `numpy` στο [Github](#) έχει αρκετό ενδιαφέρον. Επίσης ο αλγόριθμος `FFT` έχει εγγενώς μικρότερη πολυπλοκότητα από το διπλό μας `for`: όσο αυξάνει το N ο χρόνος υπολογισμού στην περίπτωση του `FFT` αυξάνεται σύμφωνα με το $N \log_2 N$ ενώ στην περίπτωση του διπλού `for` σύμφωνα με το N^2 . Σε αυτούς τους λόγους οφείλεται η μεγάλη διαφορά στους χρόνους εκτέλεσης.

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 from time import time
4
5 plt.rcParams.update({
6     "text.usetex": True,
7     "font.family": "serif",
8     "font.serif": ["Palatino"],
9     "font.size": 12,
10    "lines.linewidth": 2,
11 })
12
13 def dft(y):
14     N = y.size
15     Y = np.zeros(N, dtype = complex)
16     for n in range(N):
17         for k in range(N):
18             Y[n] += y[k] * np.exp( -1j * 2 * np.pi * n * k / N )
19     return Y
20
21 NN = [64, 256, 1024]
22 time_fft = np.zeros(len(NN))
23 time_dft = np.zeros(len(NN))
24
25 for i, N in enumerate(NN):
26     y = np.linspace(0, 1, N)
27     t1 = time()
28     Y = dft(y)

```



Εικόνα 4: Χρόνος υπολογισμού του DFT

```

29     t2 = time()
30     time_dft[i] = t2-t1
31
32     t1 = time()
33     Y2 = np.fft.fft(y)
34     t2 = time()
35     time_fft[i] = t2-t1
36     print('N = %d exec. time for FFT: %6.5f secs time for DFT: %6.5f secs' %(N, time_fft[i],
37                                     time_dft[i]))
38
39 labels = [str(n) for n in NN]
40 plt.close('all')
41 plt.figure()
42 plt.bar(labels, time_fft)
43 plt.xlabel('$N$')
44 plt.ylabel(r'$t_{\mathrm{exec}}$ [s]')
45 plt.title('FFT')
46
47 plt.figure()
48 plt.bar(labels, time_dft)
49 plt.xlabel('$N$')
50 plt.ylabel(r'$t_{\mathrm{exec}}$ [s]')
51 plt.title('DFT')

```

Listing 3: fftperformance.py

2.3 Υπολογισμός του φάσματος με τον FFT

Όπως αναφέραμε και παραπάνω, στόχος μας είναι να υπολογίσουμε τον μετασχηματισμό Fourier που δίνεται από την (1) μέσω του DFT. Ας ξαναγράψουμε την (24) εδώ για ευκολία θεωρώντας ότι το πλήθος των σημείων N είναι πολλαπλάσιο του 4 οπότε υπάρχει ακέραιος q τέτοιος ώστε $N = 4q$ οπότε και $e^{-j\pi N/2} = e^{-j2\pi q} = 1$.

$$X_k e^{-j\pi k} = \Delta t \text{DFT} \{x_n e^{j\pi n}\} \quad (28)$$

όπου με X_k έχουμε συμβολίσει τα $X(f_k)$ και έχουμε χρησιμοποιήσει τον ορισμό του DFT στην (26). Ας ορίσουμε τώρα και τον μετασχηματισμό SHIFT ως εξής:

$$\text{SHIFT} \{z_n\} = \begin{cases} z_{k-N/2} & \text{εάν } n \geq N/2, \\ z_{k+N/2} & \text{εάν } 0 \leq n < N/2 \end{cases} \quad (29)$$

Αν προσέξουμε την (29), θα δούμε ότι φέρνει τα στοιχεία του z_n που είναι από την μέση και μετά ($k \leq N/2$) στην αρχή του τελικού πίνακα και τα στοιχεία του z_n που είναι στην αρχή ($0 \leq n < N/2$) μετατοπίζονται $/2$ δεξιά. Στην numpy αυτό υλοποιείται από την συνάρτηση `fftshift`.

Υπάρχουν μερικές χρήσιμες ιδιότητες που διέπουν την SHIFT. Είναι φανερό ότι όταν εφαρμόσουμε δύο φορές της SHIFT τότε θα πάρουμε το αρχικό πίνακα, δηλαδή:

$$\text{SHIFT} \{\text{SHIFT} \{z_n\}\} = z_n \quad (30)$$

Επίσης όταν έχουμε γινόμενο δύο πινάκων το **SHIFT** του γινομένου θα είναι το γινόμενο των επιμέρους **SHIFT**,

$$\text{SHIFT} \{z_n w_n\} = \text{SHIFT} \{z_n\} \text{SHIFT} \{w_n\} \quad (31)$$

Στην περίπτωση όπου $z_n = e^{j\pi n}$ τότε οι τιμές του πίνακα z_n θα είναι $[1, -1, 1, -1, \dots]$ δηλαδή επαναλαμβάνονται ως προς το n με περίοδο 2. Είναι επομένως προφανές ότι αν $N = 4q$ οπότε και $N/2 = 2q$ το **SHIFT** δεν επιδρά καθόλου στο σήμα δηλαδή $\text{SHIFT} \{z_n\} = z_n$.

Χρησιμοποιώντας τις ιδιότητες που είδαμε παραπάνω μπορούμε να γράψουμε:

$$X_k e^{-j\pi k} = \Delta t \text{DFT} \{x_n e^{j\pi n}\} = \Delta t \text{SHIFT} \{\text{DFT} \{x_n\}\} \quad (32)$$

Η παραπάνω σχέση μπορεί να γραφεί και ως:

$$\text{SHIFT} \{X_k e^{-j\pi k}\} = \text{SHIFT} \{X_k\} e^{-j\pi k} = \Delta t \text{DFT} \{x_n\} \quad (33)$$

οπότε:

$$\text{SHIFT} \{X_k\} = \Delta t \text{DFT} \{\text{SHIFT} \{x_n\}\} \quad (34)$$

που τελικά δίνει

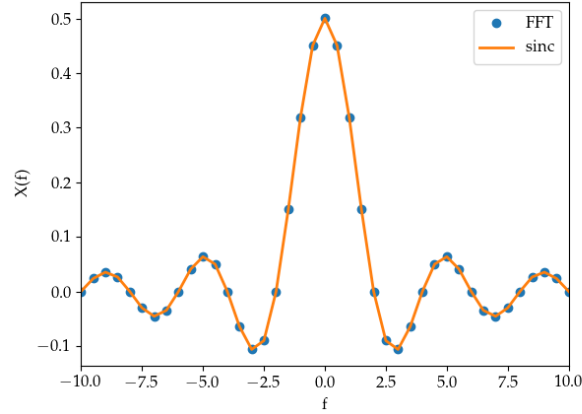
$$X_k = \Delta t \text{SHIFT} \{\text{DFT} \{\text{SHIFT} \{x_n\}\}\} \quad (35)$$

Η (35) μας δείχνει έναν τρόπο να υπολογίζουμε τις τιμές του μετασχηματισμού **Fourier** $X(f)$ σε συγκεκριμένα σημεία $X(f_k) = X_k$ χρησιμοποιώντας τα δείγματα του σήματος $x(t_n)$ χρησιμοποιώντας μία αλυσίδα μετασχηματισμών **SHIFT - DFT - SHIFT**.

Ας δούμε ένα παράδειγμα για να καταλάβουμε καλύτερα πως υπολογίζουμε τον μετασχηματισμό **Fourier** με την βοήθεια του **FFT**. Στο **listing 4** υπολογίζουμε τον μετασχηματισμό **Fourier** ενός τετραγωνικού παλμού με την χρήση του **FFT** και το συγκρίνουμε το θεωρητικό αποτέλεσμα στην (4). Στην εικόνα 5 δείχνουμε τα φάσματα που υπολογίζονται από το **listing**. Είναι φανερό ότι έχουμε μία πολύ καλή συμφωνία του **FFT** με το θεωρητικό αποτέλεσμα.

```

1
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5
6 plt.rcParams.update({
7     "text.usetex": True,
8     "font.family": "serif",
9     "font.serif": ["Palatino"],
10    "font.size" : 12,
11    "lines.linewidth" : 2,
12 })
13
14 # Square pulse
15 def square_pulse(t, T1):
16     samples = np.zeros(t.size)
17     i = np.where( np.abs(t) <= 0.5 * T1)
18     samples[ i ] = 1
19     return samples
20
21 # Calculate spectrum using FFT
22 def calc_spectrum(x, t):
23     Dt = t[1] - t[0]
24     N = t.size
25     Df = 1 / Dt / N
26     f = np.arange( -N / 2, N/2 ) * Df
27     X = Dt * np.fft.fftshift(
28         np.fft.fft(
29             np.fft.fftshift(x) ) )
30     return X, f
31
32 Tmin = -1
33 Tmax = 1
34 T1 = 0.5
35 N = 1024
36 t = np.linspace(Tmin, Tmax, N)
37
38 x = square_pulse(t, T1)
39 X, f = calc_spectrum(x, t)
40
41 plt.close('all')
```



Εικόνα 5: Σύγκριση του φάσματος που λαμβάνουμε με τον FFT και του θεωρητικού αποτελέσματος για την περίπτωση του τετραγωνικού σήματος.

```

42 plt.figure()
43 plt.plot(f, np.real(X), 'o', label = 'FFT' )
44 X_an = T1 * np.sinc(f * T1)
45 plt.plot(f, np.real(X), label = '$\mathrm{sinc}$')
46 plt.xlabel('f')
47 plt.ylabel('X(f)')
48 plt.legend()
49
50 plt.xlim([-10, 10])

```

Listing 4: fftcheck.py

2.4 Υπολογισμός του αντίστροφου μετασχηματισμού Fourier

Ξεκινώντας από την (35) μπορούμε να βρούμε έναν τρόπο να προσεγγίζουμε τον αντίστροφο μετασχηματισμό Fourier. Χρησιμοποιώντας την (30) θα έχουμε:

$$\text{SHIFT} \{X_k\} = \Delta t \text{DFT} \{\text{SHIFT} \{x_n\}\} \quad (36)$$

Εφόσον ο IDFT είναι ο αντίστροφος του DFT θα έχουμε:

$$\text{IDFT} \{\text{SHIFT} \{X_k\}\} = \Delta t \text{SHIFT} \{x_n\} \quad (37)$$

Τελικά χρησιμοποιώντας και την (20), λαμβάνουμε:

$$x_n = \frac{1}{N\Delta f} \text{SHIFT} \{\text{IDFT} \{\text{SHIFT} \{X_k\}\}\} \quad (38)$$

Στην Python ο αντίστροφος μετασχηματισμός IDFT μπορεί να υπολογιστεί μέσω του γρήγορου αντίστροφου μετασχηματισμού Fourier (inverse FFT - IFFT). Η (38) μας δείχνει πως να υπολογίσουμε τον αντίστροφο μετασχηματισμό Fourier ενός σήματος χρησιμοποιώντας τον IFFT.

3 Ιδιότητες του μετασχηματισμού Fourier

3.1 Ταυτότητα Parseval

Μία χρήσιμη ιδιότητα του μετασχηματισμού Fourier είναι η ταυτότητα του Parseval κατά την οποία η ενέργεια του σήματος μπορεί να υπολογιστεί τόσο στο πεδίο του χρόνου όσο και στο πεδίο των συχνοτήτων:

$$\mathcal{E}_t = \int_{-\infty}^{+\infty} |x(t)|^2 dt = \int_{-\infty}^{+\infty} |X(f)|^2 df = \mathcal{E}_f \quad (39)$$

```

1 import numpy as np
2
3 # Square pulse
4 def square_pulse(t, T1):
5     samples = np.zeros(t.size)
6     i = np.where( np.abs(t) <= 0.5 * T1)
7     samples[ i ] = 1
8     return samples
9
10 # Calculate spectrum using FFT
11 def calc_spectrum(x, t):
12     Dt = t[1] - t[0]
13     N = t.size
14     Df = 1 / Dt / N
15     f = np.arange( -N / 2, N/2 ) * Df
16     X = Dt * np.fft.fftshift(
17         np.fft.fft(
18             np.fft.fftshift(x) ) )
19     return X, f
20
21
22 # Energy
23 def energy(x, t):
24     return np.trapz(np.abs(x) ** 2, t)
25
26 Tmin = -1
27 Tmax = 1
28 T1 = 0.5
29 N = 1024
30 t = np.linspace(Tmin, Tmax, N)
31
32 x = square_pulse(t, T1)
33 X, f = calc_spectrum(x, t)
34
35 Et = energy(x, t)
36 Ef = energy(X, f)
37
38 print('Energy in time domain:', Et)
39 print('Energy in frequency domain:', Ef)

```

Listing 5: parseval.py

Μπορούμε να χρησιμοποιήσουμε την Python για να επαληθεύσουμε την ιδιότητα του Parseval όπως φαίνεται στο listing 5 όπου υπολογίζουμε αριθμητικά τόσο την ενέργεια $\mathcal{E}_t$ στο πεδίο του χρόνου όσο και την ενέργεια $\mathcal{E}_f$ στο πεδίο των συχνοτήτων. Στο πεδίο του χρόνου θα έχουμε:

$$\mathcal{E}_t = \int_{-\infty}^{+\infty} |x(t)|^2 dt = \int_{-T_1/2}^{T_1/2} dt = T_1 \quad (40)$$

Αν δούμε τα αποτελέσματα που παράγονται από το listing 5 παρατηρούμε ότι οι τιμές που υπολογίζονται με τη χρήση της trapz είναι πολύ κοντά στην θεωρητική τιμή που στην συγκεκριμένη περίπτωση είναι $T_1 = 0.5$.

3.2 Δυσισμός

Μία άλλη ενδιαφέρουσα ιδιότητα του μετασχηματισμού Fourier είναι ο δυσισμός. Δηλαδή εάν το $x(t)$ έχει μετασχηματισμό Fourier το $X(f)$ τότε εάν ορίσουμε ένα νέο σήμα που λαμβάνεται από το $X(f)$ αντικαθιστώντας το f με το t , $\tilde{x}(t) = X(t)$ τότε το $\tilde{x}(t)$ θα έχει μετασχηματισμό Fourier το $\tilde{X}(f) = x(-f)$. Είναι λίγο παράξενο αλλά προκύπτει από τον ορισμό του μετασχηματισμού Fourier,

$$\tilde{X}(f) = \mathcal{F}\{\tilde{x}(t)\} = \int_{-\infty}^{+\infty} \tilde{x}(t) e^{-j2\pi ft} dt = \int_{-\infty}^{+\infty} X(t) e^{j2\pi(-f)t} dt \quad (41)$$

Για να μην μπερδεύμαστε μπορούμε να αλλάξουμε την μεταβλητή ολοκλήρωσης στο τελευταίο ολοκλήρωμα,

$$\tilde{X}(f) = \int_{-\infty}^{+\infty} X(r) e^{j2\pi(-f)r} dr \quad (42)$$

Σύμφωνα με τον αντίστροφο μετασχηματισμό Fourier στην (2) θα πρέπει να ισχύει

$$\int_{-\infty}^{+\infty} X(r) e^{j2\pi r q} dr = x(q) \quad (43)$$

και θέτοντας $q = -f$ θα έχουμε

$$\tilde{X}(f) = x(-f) \quad (44)$$

Ας δούμε ένα παράδειγμα. Θέλουμε να υπολογίσουμε το φάσμα του σήματος

$$x(t) = \frac{1}{T_0} \text{sinc}\left(\frac{t}{T_0}\right) \quad (45)$$

Είδαμε στα προηγούμενα ότι ο μετασχηματισμός **Fourier** $P(f)$ του τετραγωνικού σήματος $p(t)$ στην (3) δίνεται από την (4). Αν προσέξουμε καλά την (4) παρατηρούμε ότι μπορούμε να γράψουμε

$$x(t) = P(t) = T_1 \text{sinc}(tT_1) \quad (46)$$

αρκεί να θέσουμε $T_1 = 1/T_0$. Οπότε σύμφωνα με την ιδιότητα του διτισμού θα πρέπει ο μετασχηματισμός **Fourier** $X(f)$ του $x(t)$ να προκύπτει από το $p(-f)$ δηλαδή

$$X(f) = p(-f) = \begin{cases} 1 & , -\frac{T_1}{2} \leq -f \leq \frac{T_1}{2} \\ 0 & , \text{διαφορετικά} \end{cases} = \begin{cases} 1 & , -\frac{1}{2T_0} \leq f \leq \frac{1}{2T_0} \\ 0 & , \text{διαφορετικά} \end{cases} \quad (47)$$

Επομένως το φάσμα του σήματος (46) είναι τετραγωνικό με κέντρο το $f = 0$ και φασματικό εύρος $\frac{1}{2T_0}$.

3.3 Συγκερασμός

Ο συγκερασμός $z(t)$ δύο σημάτων $x(t)$ και $y(t)$ ορίζεται ως:

$$z(t) = x(t) \circledast y(t) = \int_{-\infty}^{+\infty} x(\tau)y(t-\tau)d\tau \quad (48)$$

Το φάσμα $Z(f)$ του $z(t)$ μπορούμε να δείξουμε ότι προκύπτει από το γινόμενο των φασμάτων $X(f)$ και $Y(f)$, δηλαδή:

$$Z(f) = X(f)Y(f) \quad (49)$$

Για να δείξουμε αυτή την ιδιότητα, ξεκινάμε με τον υπολογισμό του $Z(f)$ βάσει της (1),

$$Z(f) = \int_{-\infty}^{\infty} z(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} \left\{ \int_{-\infty}^{+\infty} x(\tau)y(t-\tau)d\tau \right\} e^{-j2\pi ft}dt \quad (50)$$

Η τελευταία σχέση γράφεται:

$$Z(f) = \int_{-\infty}^{\infty} \int_{-\infty}^{+\infty} x(\tau)e^{-j2\pi f\tau}y(t-\tau)e^{-j2\pi f(t-\tau)}d\tau dt \quad (51)$$

Θέτουμε $\tau' = t - \tau$ οπότε θα έχουμε:

$$Z(f) = \int_{-\infty}^{\infty} \int_{-\infty}^{+\infty} x(\tau)e^{-j2\pi f\tau}y(\tau')e^{-j2\pi f\tau'}d\tau d\tau' = \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f\tau}d\tau \int_{-\infty}^{+\infty} y(\tau')e^{-j2\pi f\tau'}d\tau' \quad (52)$$

Δεδομένου ότι τα $X(f)$ και $Y(f)$ είναι οι μετασχηματισμοί **Fourier** των $x(t)$ και $y(t)$ θα έχουμε

$$X(f) = \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f\tau}d\tau \quad (53)$$

$$Y(f) = \int_{-\infty}^{+\infty} y(\tau')e^{-j2\pi f\tau'}d\tau' \quad (54)$$

οπότε προκύπτει η (49).

3.4 Άλλες ιδιότητες

Αναφέρουμε εδώ μερικές ακόμα χρήσιμες ιδιότητες του μετασχηματισμού **Fourier**:

- *Γραμμικότητα.* Αν $z(t) = c_1x(t) + c_2y(t)$ όπου c_1 και c_2 σταθερές, τότε

$$Z(f) = c_1X(f) + c_2Y(f) \quad (55)$$

- *Κλιμάκωση*. Αν $y(t) = x(\alpha t)$ τότε

$$Y(f) = \frac{1}{|\alpha|} X\left(\frac{f}{\alpha}\right) \quad (56)$$

- *Διαφόριση*. Αν το $x(t)$ είναι η παράγωγος του $y(t)$, $x(t) = \frac{dy(t)}{dt}$ τότε

$$X(f) = j2\pi f Y(f) \quad (57)$$

- *Χρονική ολίσθηση*. Αν $y(t) = x(t - t_0)$ τότε

$$Y(f) = X(f) e^{-j2\pi t_0 f} \quad (58)$$

- *Συχνотική ολίσθηση*. Αν $y(t) = x(t) e^{j2\pi f_0 t}$ τότε

$$Y(f) = X(f - f_0) \quad (59)$$

- *Συζυγή σήματα*. Ο μετασχηματισμός **Fourier** του συζυγούς σήματος $x^*(t)$ είναι το

$$\mathcal{F}\{x^*(t)\} = X^*(-f) \quad (60)$$

- *Πραγματικά σήματα*. Αν το $x(t)$ είναι πραγματικό σήμα, τότε ισχύει:

$$X(f) = X^*(-f) \quad (61)$$

4 Φασματική πυκνότητα ισχύος

Η φασματική πυκνότητα ισχύος (**power spectral density - PSD**) είναι ένα μέτρο που μας δείχνει πια τμήματα του φάσματος ενός σήματος είναι ισχυρά και συμμετέχουν περισσότερο στην διαμόρφωση της ισχύος του σήματος. Αν θεωρήσουμε ένα σήμα $x(t)$ τότε η ισχύς του $\mathcal{P}(T)$ στο διάστημα $[-T/2, T/2]$ δίνεται από την σχέση:

$$\mathcal{P}(T) = \frac{1}{T} \int_{-T/2}^{T/2} |x(t)|^2 dt \quad (62)$$

Αν ορίσουμε τώρα μία “παραθυρωμένη” έκδοση του $x(t)$, το $x_T(t)$ το οποίο ισούται με το $x(t)$ εντός του $[-T/2, T/2]$ και με μηδέν εκτός αυτού,

$$x_T(t) = \begin{cases} x(t) & , -\frac{T}{2} \leq t \leq \frac{T}{2} \\ 0 & , \text{διαφορετικά} \end{cases} \quad (63)$$

Η ισχύς του σήματος μπορεί να γραφτεί και ως εξής:

$$\mathcal{P}(T) = \frac{1}{T} \int_{-\infty}^{\infty} |x_T(t)|^2 dt \quad (64)$$

Αν ορίσουμε το $X_T(f)$ να είναι ο μετασχηματισμός **Fourier** του $x_T(t)$ τότε βάσει της ιδιότητας του **Parseval**,

$$\mathcal{P}(T) = \frac{1}{T} \int_{-\infty}^{\infty} |X_T(f)|^2 df \quad (65)$$

Στο όριο όπου $T \rightarrow \infty$ το $\mathcal{P}(T)$ γίνεται η μέση ισχύς για όλο τον άξονα του χρόνου,

$$\mathcal{P} = \lim_{T \rightarrow \infty} \int_{-\infty}^{\infty} \frac{1}{T} |X_T(f)|^2 df \quad (66)$$

Αν ορίσουμε επομένως την **PSD** ως:

$$S_x(f) = \lim_{T \rightarrow \infty} \frac{1}{T} |X_T(f)|^2 df \quad (67)$$

τότε η μέση ισχύς είναι το ολοκλήρωμα της **PSD** σε όλο τον άξονα των συχνοτήτων,

$$\mathcal{P} = \int_{-\infty}^{\infty} S_x(f) df \quad (68)$$

Από την (68) προκύπτει ότι το $S_x(f)$ καθορίζει ποιες συχνότητες παίζουν σημαντικό ρόλο στον υπολογισμό του $\mathcal{P}$. Αν σε μία περιοχή συχνοτήτων το $S_x(f)$ είναι μεγάλο τότε έπεται ότι η εν λόγω περιοχή θα έχει σημαντική συνεισφορά στο ολοκλήρωμα και επομένως σημαντικό μέρος της μέσης ισχύος θα καθορίζεται από αυτήν.

Δεδομένου ότι $|X_T(f)|^2 = X_T(f)X_T^*(f)$ και ο αντίστροφος μετασχηματισμός Fourier του $X_T^*(f)$ είναι $x_T^*(-t)$ οπότε στο πεδίο του χρόνου το $S_x(f)$ αντιστοιχεί σε ένα σήμα

$$\mathcal{F}^{-1}\{S_x(f)\} = \lim_{T \rightarrow \infty} \frac{1}{T} x_T(t) \otimes x_T^*(-t) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-\infty}^{+\infty} x_T(\tau) x_T^*(\tau - t) d\tau \quad (69)$$

Στην (69) εμφανίζεται η *συνάρτηση αυτοσυσχέτισης* $R_x(t)$ του σήματος $x(t)$ η οποία ορίζεται ως εξής:

$$R_x(t) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-\infty}^{+\infty} x_T(\tau) x_T^*(\tau - t) d\tau \quad (70)$$

από την οποία προκύπτει ότι:

$$S_x(f) = \mathcal{F}\{R_x(t)\} \quad (71)$$

5 Ώρα για σπαζοκεφαλίες

1. Με χαρτί και μολύβι δείξτε τις ιδιότητες του μετασχηματισμού Fourier της παραγράφου 3.4.
2. Χρησιμοποιώντας την Python υπολογίστε τον μετασχηματισμό Fourier του σήματος $x(t)$ που δίνεται από την

$$x(t) = \exp\left(-\frac{t^2}{2\tau^2}\right) \quad (72)$$

και συγκρίνεται το αριθμητικό αυτό αποτέλεσμα με το θεωρητικό:

$$X(f) = \tau\sqrt{2\pi}\exp(-2\tau^2\pi^2 f^2) \quad (73)$$