

Μάθημα 3: Η έννοια του φάσματος

Θωμάς Καμαλάκης

30 Αυγούστου 2023

1 Η σειρά Fourier

Είμαι σίγουρος ότι αρκετοί από εσάς έχετε δει από κοντά κάποια στιγμή τις περίφημες κονσόλες των DJ στις καφετέριες, όπως αυτή της εικόνας 1. Σε πολλές από αυτές υπάρχει η δυνατότητα να ενισχυθούν τα μπάσα ή τα πρίμα - στην συγκεκριμένη εικόνα είναι οι μπλέ διακόπτες στο πάνω μέρος της κονσόλας. Συνήθως κάθε διακόπτης αντιστοιχεί σε μία συχνότητα (π.χ. 2KHz κτλ). Η κονσόλα επομένως μπορεί να ενισχύσει ή να εξασθενίσει ορισμένες συχνότητες ώστε το ακουστικό αποτέλεσμα να είναι το επιθυμητό. Πως όμως γίνεται αυτό;

Καταρχήν θα πρέπει να καταλάβουμε πως ακριβώς αντιστοιχούμε ένα ακουστικό και εν γένει ένα οποιοδήποτε σήμα σε συχνότητες. Ξεκινάμε από σήματα που έχουν πεπερασμένη διάρκεια, δηλαδή εκτός ενός διαστήματος $[t_a, t_b]$ είναι ίσα με μηδέν,

$$x(t) = 0 \text{ εάν } t > t_b \text{ ή } t < t_a \quad (1)$$

Τότε αν υπολογίσουμε τους συντελεστές

$$X_m = \frac{1}{T} \int_{t_a}^{t_b} x(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt \quad (2)$$

όπου το $T = t_b - t_a$ είναι η διάρκεια του σήματος. Τότε θα έχουμε:

$$x(t) = \sum_{m=-\infty}^{+\infty} X_m \exp\left(j \frac{2\pi m t}{T}\right) \quad (3)$$

Τα X_m όπως ορίζονται στην (2) ονομάζονται *συντελεστές Fourier* του $x(t)$. Το άθροισμα (3) ονομάζεται *σειρά Fourier* του $x(t)$. Πως όμως η σειρά Fourier μας διευκολύνει να αντιστοιχήσουμε το σήμα μας σε συχνότητες;

Η (2) αναπαριστά το σήμα ως ένα άθροισμα όρων της μορφής $X_m F_m(t)$ όπου $F_m(t)$ είναι οι συναρτήσεις:

$$F_m(t) = \exp\left(j \frac{2\pi m t}{T}\right) \quad (4)$$

Αν θυμηθούμε την ιδιότητα του Euler $e^{j\phi} = \cos \phi + j \sin \phi$ τότε μπορούμε να γράψουμε τις $F_m(t)$ ως εξής:

$$F_m(t) = \cos(2\pi f_m t) + j \sin(2\pi f_m t) \quad (5)$$

όπου οι συχνότητες f_m δίνονται από την σχέση

$$f_m = \frac{m}{T} \quad (6)$$

```
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 def harmonic(t, m, T):
13     return np.exp(1j * 2 * np.pi * m / T * t)
14
15 mm = [1, 2, 3]
```



Εικόνα 1: Κονσόλα DJ

```

16 T = 2
17 N = 1000
18 t = np.linspace(0, T, N)
19 plt.close('all')
20 for m in mm:
21     Fm = harmonic(t, m, T)
22     plt.figure()
23     plt.plot(t, np.real(Fm), label = 'Real')
24     plt.plot(t, np.imag(Fm), '--r', label = 'Imag')
25     plt.title('$F_{%d}(t)$' % m)
26     plt.legend()
27     plt.xlabel('$t$')
28     plt.ylabel('$F_m(t)$')

```

Listing 1: harmonic.py

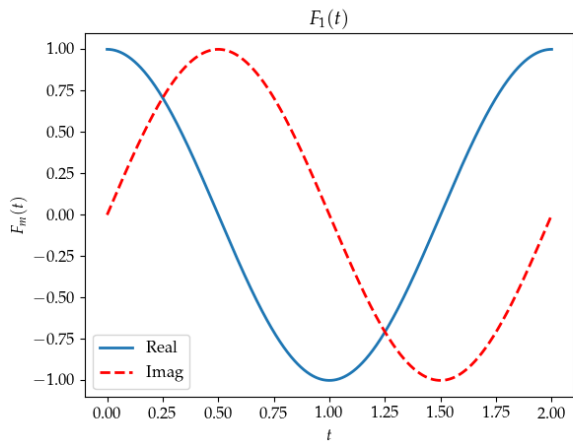
Οι συναρτήσεις $F_m(t)$ ονομάζονται *αρμονικά* σήματα. Ας κάνουμε μία γραφική παράσταση των $F_m(t)$ για διάφορα m . Στο [listing 1](#) υπολογίζουμε τα τρία πρώτα αρμονικά σήματα για $m = 1, 2, 3$. Θα δείτε ότι στο [listing](#) έχουμε αλλάξει τις παραμέτρους των γραφικών παραστάσεων με την `rcParams.update` έτσι ώστε να μπορούμε να χρησιμοποιήσουμε $\LaTeX$ στις γραφικές παραστάσεις μας. Το $\LaTeX$ συχνά δίνει πολύ καλύτερο αποτέλεσμα στην εμφάνιση μαθηματικών συμβόλων. Αυτές οι σημειώσεις είναι γραμμένες σε $\LaTeX$!

Στη συνέχεια ορίζουμε μία συνάρτηση `harmonic` η οποία υπολογίζει τα αρμονικά σήματα $F_m(t)$ και κάνουμε τις γραφικές παραστάσεις όπως φαίνονται στην εικόνα 2. Δεδομένου ότι οι F_m όπως δίνονται από την (5) έχουν και πραγματικό και φανταστικό μέρος, παριστάνουμε και τα δύο. Παρατηρούμε ότι όσο αυξάνεται το m , τα F_m ταλαντώνονται όλο και πιο γρήγορα. Για $m = 1$ έχουμε μία πλήρη περίοδο στο διάστημα $[0, 2]$, για $m = 2$ έχουμε δύο κ.ο.κ. σε συμφωνία και με την (6) που μας δείχνει ότι η συχνότητα του αρμονικού σήματος είναι ανάλογο του m .

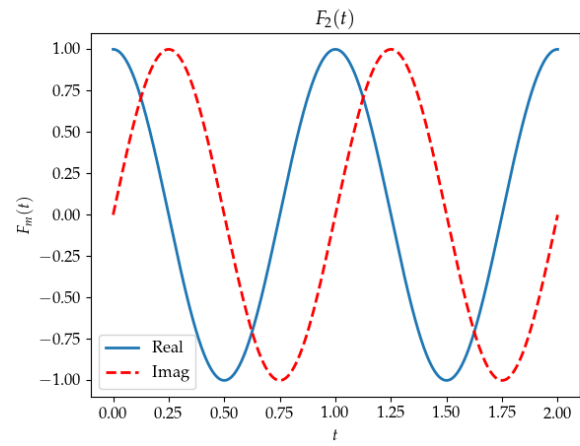
Μπορούμε να αντικαταστήσουμε την (5) στην (3) για να γράψουμε:

$$x(t) = \sum_{m=-\infty}^{+\infty} X_m F_m(t) \quad (7)$$

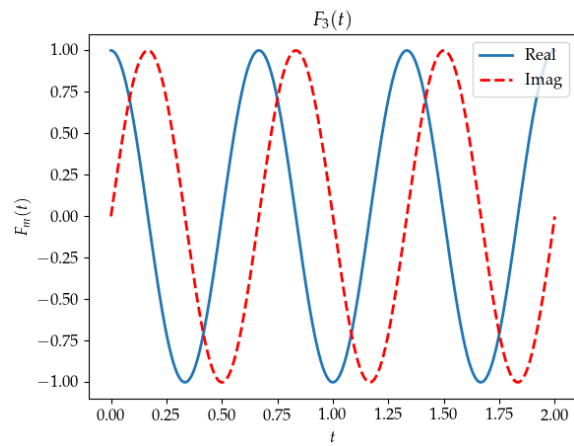
Σύμφωνα με την (7), το σήμα $x(t)$ γράφεται ως άθροισμα των αρμονικών σημάτων $F_m(t)$ όπως αυτά που έχουμε παραστήσει στην εικόνα 2 το κάθε ένα πολλαπλασιασμένο με τον συντελεστή X_m . Αν ο συντελεστής X_m είναι μεγάλος τότε η αντίστοιχη συνεισφορά του όρου $X_m F_m(t)$ θα είναι και αυτή μεγάλη. Επομένως η τιμή του συντελεστή X_m καθορίζει το κατά πόσο ένα αρμονικό σήμα $F_m(t)$ παίζει ρόλο στη σειρά [Fourier](#). Για παράδειγμα όταν καταγράφουμε το σήμα ενός μπάσου τραγουδιστή και υπολογίσουμε τους συντελεστές [Fourier](#) του, περιμένουμε αυτοί που αντιστοιχούν σε χαμηλές συχνότητες f_m να είναι έχουν μεγαλύτερο μέγεθος από αυτούς που είναι σε υψηλές.



(a) $m = 1$



(b) $m = 2$



(c) $m = 3$

Εικόνα 2: Τα αρμονικά σήματα για $m = 1, 2, 3$.

2 Η σειρά Fourier του τετραγωνικού παλμού

Ας προσπαθήσουμε τώρα να υπολογίσουμε την σειρά Fourier του τετραγωνικού παλμού:

$$p(t) = \begin{cases} 1 & , -\frac{T_1}{2} \leq t \leq \frac{T_1}{2} \\ 0 & , \text{διαφορετικά} \end{cases} \quad (8)$$

Θεωρούμε μία διάρκεια $[-T/2, T/2]$ όπου $T_1 < T$ και χρησιμοποιούμε την (2) θέτοντας $t_a = -T/2$ και $t_b = T/2$ οπότε οι συντελεστές Fourier P_m του σήματος θα υπολογίζονται ως εξής:

$$P_m = \frac{1}{T} \int_{-T/2}^{T/2} p(t) \exp(-j2\pi f_m t) dt = \frac{1}{T} \int_{-T_1/2}^{T_1/2} \exp(-j2\pi f_m t) dt \quad (9)$$

Το παραπάνω ολοκλήρωμα υπολογίζεται αν θυμηθούμε ότι

$$\int \exp(\alpha t) dt = \frac{\exp(\alpha t)}{\alpha} + C \quad (10)$$

όπου C μία οποιαδήποτε σταθερά. Στην περίπτωση μας, $\alpha = -j2\pi f_m$ οπότε:

$$\int_{-T_1/2}^{T_1/2} \exp(-j2\pi f_m t) dt = \frac{\exp(j\pi f_m T_1) - \exp(-j\pi f_m T_1)}{j2\pi f_m} \quad (11)$$

Μπορούμε να απλοποιήσουμε λίγο το αποτέλεσμα αυτό εάν θυμηθούμε ότι

$$e^{j\phi} = \cos \phi + j \sin \phi \quad (12)$$

οπότε και

$$e^{-j\phi} = \cos \phi - j \sin \phi \quad (13)$$

Αφαιρώντας κατά μέλη προκύπτει ότι

$$2j \sin \phi = e^{j\phi} - e^{-j\phi} \quad (14)$$

Οπότε από την (11) έχουμε:

$$P_m = \frac{1}{T} \frac{\sin(\pi f_m T_1)}{\pi f_m} = \frac{T_1}{T} \text{sinc}(f_m T_1) \quad (15)$$

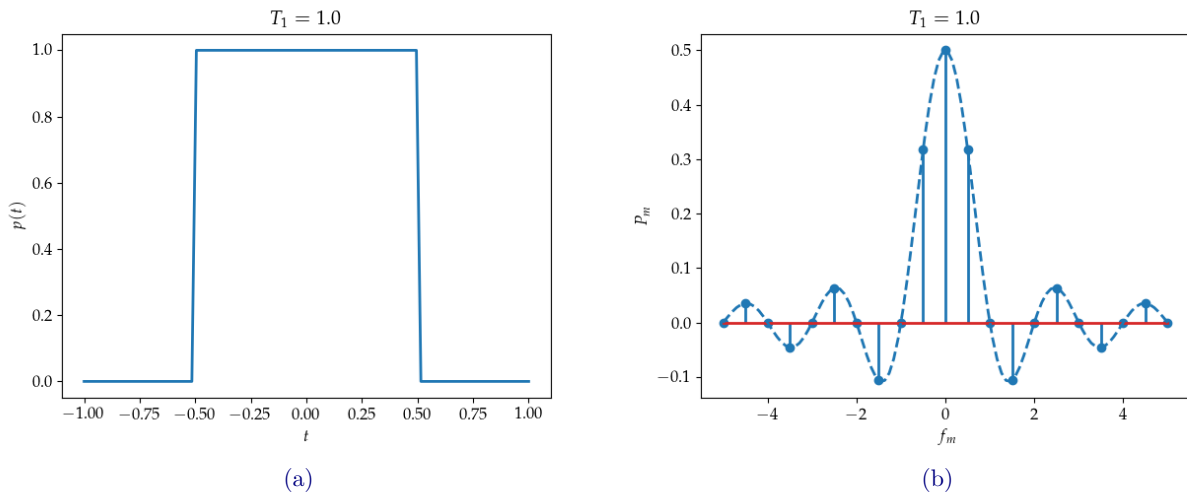
όπου έχουμε ορίσει την συνάρτηση sinc ως εξής:

$$\text{sinc}(x) = \begin{cases} \frac{\sin(\pi x)}{\pi x}, & x \neq 0 \\ 1, & x = 0 \end{cases} \quad (16)$$

Η (15) μας δίνει τους συντελεστές Fourier του τετραγωνικού παλμού.

Ας χρησιμοποιήσουμε τώρα την Python για να κάνουμε μία γραφική παράσταση των συντελεστών P_m . Χρησιμοποιούμε το listing 2 όπου θεωρούμε δύο διαφορετικές περιπτώσεις. Μία όπου $T_1 = 1$ και μία όπου $T_1 = 0.5$. Τα αποτελέσματα του listing φαίνονται στην Εικόνες 3 και 4 αντίστοιχα.

```
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 # 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 coefficients
20 def Pm(m, T1, T):
21     fm = m / T
```



Εικόνα 3: Οι συντελεστές Fourier του τετραγωνικού παλμού για $T_1 = 1$.

```

22     return T1 / T * np.sinc( fm * T1 )
23
24 T = 2
25 t = np.linspace(-T/2, T/2, 100)
26 T1s = [1, 0.5]
27 m = np.arange(-10, 11)
28 f = np.linspace(-10/T, 10/T, 100)
29 plt.close('all')
30 for T1 in T1s:
31     p = square_pulse(t, T1)
32     plt.figure()
33     plt.plot(t, p)
34     plt.title('$T_1=%6.1f$' %T1)
35     plt.xlabel('$t$')
36     plt.ylabel('$p(t)$')
37     plt.figure()
38
39     Pms = Pm(m, T1, T)
40     fm = m/T
41     plt.figure()
42     plt.stem(fm, Pms)
43     plt.xlabel('$f_m$')
44     plt.ylabel('$P_m$')
45     plt.plot(f, T1/T * np.sinc(f*T1), '--')
46     plt.title('$T_1=%6.1f$' %T1)

```

Listing 2: Pm.py

Οι τιμές των συντελεστών Fourier για κάθε συχνότητα f_m όπως αυτές που απεικονίζονται στις Εικόνες 3b και 4b λέμε ότι περιγράφουν το φάσμα του παλμού. Συγκρίνοντας τα δύο φάσματα, μία πρώτη παρατήρηση είναι ότι ο στενός παλμός στην εικόνα 4a έχει πιο ευρύ φάσμα από ότι ο ευρύς παλμός στην εικόνα 3a. Αυτό είναι ένα γενικό συμπέρασμα που ισχύει για κάθε είδος σήματος: ένας στενός παλμός όπως αυτός της εικόνας 4a έχει πιο γρήγορες μεταβάσεις καθώς παραμένει στην υψηλή τιμή για λιγότερη χρονική διάρκεια (το T_1 είναι πιο μικρό). Ως εκ τούτου, το φάσμα του θα πρέπει να εκτείνεται σε υψηλότερες συχνότητες.

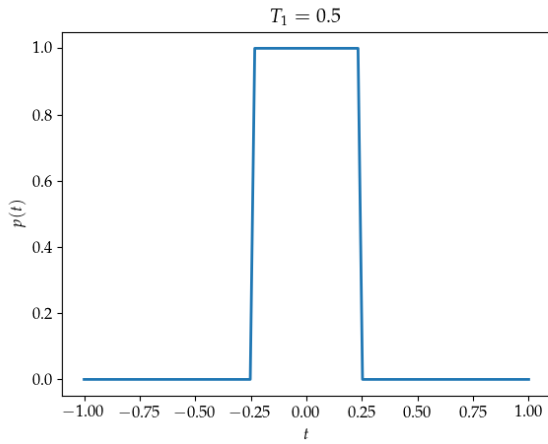
3 Ψαλιδίζοντας την σειρά Fourier

Η σειρά Fourier όπως δίνεται από την (3) έχει άπειρους όρους. Ξεκινάει από το $m = -\infty$ και φτάνει ως το $m = +\infty$. Όπως είδαμε όμως στην περίπτωση του τετραγωνικού παλμού δεν είναι όλοι οι όροι εξίσου σημαντικοί καθώς όσο απομακρυνόμαστε από το $m = 0$ στις εικόνες 3b και 4b οι τιμές των συντελεστών τείνουν να μειώνονται. Πόσους συντελεστές Fourier επομένως πρέπει πρακτικά να λάβουμε υπόψη μας; Η απάντηση είναι ότι εξαρτάται από το σήμα.

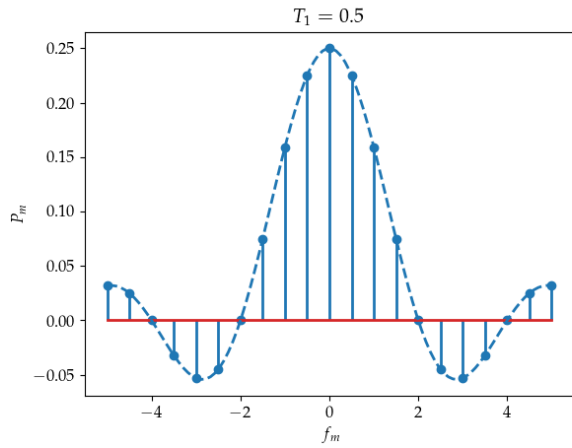
```

1 import numpy as np
2 import matplotlib.pyplot as plt
3

```



(a)



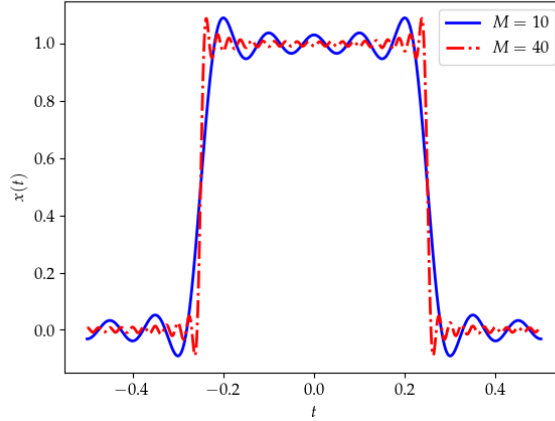
(b)

Εικόνα 4: Οι συντελεστές Fourier του τετραγωνικού παλμού για $T_1 = 0.5$.

```

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 # 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 coefficients
20 def Pm(m, T1, T):
21     fm = m / T
22     return T1 / T * np.sinc( fm * T1 )
23
24 # Harmonic functions
25 def harmonic(t, m, T):
26     return np.exp(1j * 2 * np.pi * m / T * t)
27
28 # Fourier sum
29 def fourier_sum(t, M, T, Xm):
30     x = np.zeros(t.size, dtype=complex)
31     i = 0
32     for m in range(-M, M+1 ):
33         x += Xm[i] * harmonic(t, m, T)
34         i += 1
35
36     return x
37
38 MM = [10, 40]
39 pltypes = ['-b', '-.r']
40
41 T1 = 0.5
42 T = 1
43 t = np.linspace(-T/2, T/2, 1000)
44 plt.close('all')
45 plt.figure()
46
47 for i, M in enumerate(MM):
48     m = np.arange(-M, M+1, 1)
49     Xm = Pm(m, T1, T)
50     x = fourier_sum(t, M, T, Xm)
51     plt.plot(t, np.real(x), pltypes[i], label='$M=%d$' %M )
52
53 plt.legend()

```



Εικόνα 5: Ψαλιδίζοντας την σειρά Fourier.

```
54 plt.xlabel('$t$')
55 plt.ylabel('$x(t)$')
```

Listing 3: fouriersum.py

Μπορούμε να χρησιμοποιήσουμε την Python για να δούμε τι συμβαίνει όταν ψαλιδίζουμε την σειρά Fourier κρατώντας τους όρους από $m = -M$ έως $m = M$. Χρησιμοποιούμε το listing 3 όπου υλοποιούμε μία νέα συνάρτηση, την `fourier_sum` για να υπολογίζουμε το άθροισμα (3) από $m = -M$ έως $m = M$. Στην εικόνα 5 δείχνουμε το αποτέλεσμα για $M = 10$ και $M = 40$.

Για $M = 10$, λαμβάνουμε μία πρώτη προσέγγιση για τον τετραγωνικό παλμό. Ωστόσο παρατηρούμε μερικές ταλαντώσεις τόσο στις υψηλές τιμές (κοντά στο $t = 0$) όσο και στις χαμηλές τιμές ($|t| > T_1/2$). Επίσης οι μεταβάσεις από τις υψηλές στις χαμηλές τιμές (κοντά στο $t = \pm T_1/2$) είναι σχετικά αργές. Αυτό οφείλεται στο ότι έχουμε περιορίσει τις συχνότητες f_m που λαμβάνουμε υπόψη στο ανάπτυγμα και ως εκ τούτου οι μεταβάσεις δεν μπορούν να είναι πολύ γρήγορες. Αν θεωρήσουμε περισσότερους όρους ($M = 40$) τότε οι μεταβάσεις γίνονται πιο γρήγορες καθώς τώρα λαμβάνουμε υπόψη υψηλότερες συχνότητες f_m .

Για να μελετήσουμε πόσο καλά προσεγγίζει η ψαλιδισμένη σειρά Fourier το αρχικό σήμα ορίζουμε ένα μέτρο του σφάλματος μεταξύ της ψαλιδισμένης σειράς

$$x_M(t) = \sum_{m=-M}^M X_m \exp\left(j \frac{2\pi m t}{T}\right) \quad (17)$$

και του αρχικού σήματος $x(t)$. Ας ορίσουμε το ϵ ως εξής:

$$\epsilon = \int_{t_a}^{t_b} |x(t) - x_M(t)|^2 dt \quad (18)$$

Είναι φανερό ότι αν $x(t) = x_M(t)$ τότε το ϵ είναι ίσο με μηδέν. Επίσης καταλαβαίνουμε ότι όσο μικρότερο είναι το ϵ τόσο καλύτερη είναι η προσέγγιση του $x(t)$ από το $x_M(t)$. Το ολοκλήρωμα διατρέχει όλες τις τιμές $t_a \leq t \leq t_b$. Αν η διαφορά μεταξύ $x(t)$ και $x_M(t)$ σε κάποια περιοχή τιμών είναι μεγάλη το $|x(t) - x_M(t)|^2$ θα είναι επίσης μεγάλο κάτι που θα αυξάνει την τιμή του ϵ . Επομένως το ϵ μετράει το σφάλμα της προσέγγισης του $x(t)$ από το $x_M(t)$. Χρησιμοποιούμε το listing 4 για να υπολογίσουμε πόσο βελτιώνεται το σφάλμα ϵ καθώς αυξάνουμε τον αριθμό των όρων που λαμβάνουμε στο άθροισμα (17).

```
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 # 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 coefficients
20 def Pm(m, T1, T):
21     fm = m / T
22     return T1 / T * np.sinc( fm * T1 )
23
24 # Harmonic functions
25 def harmonic(t, m, T):
26     return np.exp(1j * 2 * np.pi * m / T * t)
27
28 # Fourier sum
29 def fourier_sum(t, M, T, Xm):
30     x = np.zeros(t.size, dtype=complex)
31     i = 0
32     for m in range(-M, M+1 ):
33         x += Xm[i] * harmonic(t, m, T)
34         i += 1
35
36     return x
37
38 # error function
39 def error(t, x, x0):
40     return np.trapz( np.abs(x-x0) ** 2, t)
41
42 MM = np.arange(10, 100, 1)
43 errors = np.zeros(MM.size)
44
45 T1 = 0.5
46 T = 1
47 t = np.linspace(-T/2, T/2, 1000)
48 plt.close('all')
49 plt.figure()
50 p = square_pulse(t, T1)
51
52 for i, M in enumerate(MM):
53     m = np.arange(-M, M+1, 1)
54     Xm = Pm(m, T1, T)
55     x = fourier_sum(t, M, T, Xm)
56     errors[i] = error(t, x, p)
57
58
59 plt.plot(MM, errors )
60 plt.ylabel('$\epsilon$')
61 plt.xlabel('$M$')

```

Listing 4: fourierconv.py

Στην εικόνα 6 βλέπουμε την μεταβολή του ϵ με το M όπως προκύπτει από το listing 4. Όπως είναι αναμενόμενο το σφάλμα μειώνεται όσο αυξάνει ο αριθμός των όρων που λαμβάνουμε στο άθροισμα. Από την εικόνα προκύπτει ότι $\epsilon \rightarrow 0$ όταν $M \rightarrow \infty$, όπως άλλωστε είναι αναμενόμενο.

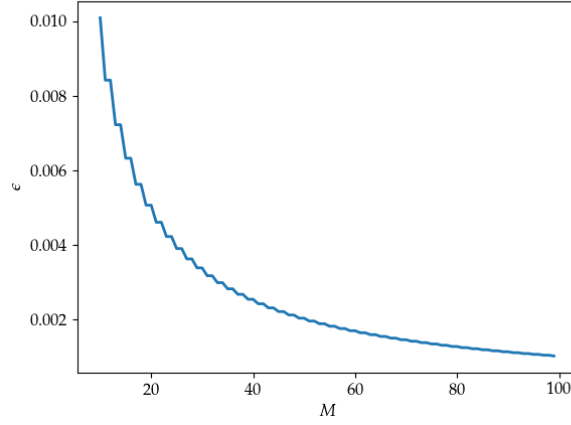
4 Ιδιότητες των συντελεστών Fourier

Υπάρχουν μερικές ιδιότητες των συντελεστών Fourier που είναι ιδιαίτερα χρήσιμες. Καταρχήν σύμφωνα με την (2), οι συντελεστές Fourier προκύπτουν από την ολοκλήρωση του σήματος $x(t)$ επί το μιγαδικό σήμα $\exp(-j2\pi f_m t)$. Οπότε εν γένει είναι μιγαδικοί αριθμοί. Στην περίπτωση όπου το σήμα $x(t)$ είναι πραγματικό τότε μπορούμε να δείξουμε ότι

$$X_{-m} = \frac{1}{T} \int_{t_a}^{t_b} x(t) \exp\left(j \frac{2\pi m t}{T}\right) dt \quad (19)$$

Ας υπολογίσουμε τώρα το μιγαδικό συζυγές του X_m .

$$X_m^* = \frac{1}{T} \left(\int_{t_a}^{t_b} x^*(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt \right)^* = \frac{1}{T} \int_{t_a}^{t_b} x^*(t) \exp\left(j \frac{2\pi m t}{T}\right) dt \quad (20)$$



Εικόνα 6: Το $\epsilon = \epsilon(M)$ για την περίπτωση του τετραγωνικού παλμού.

Υποθέσαμε ότι το σήμα $x(t)$ είναι πραγματικό οπότε $x(t) = x^*(t)$. Οπότε συνδυάζοντας την (19) και την (20) έχουμε

$$X_m^* = X_{-m} \quad (21)$$

δηλαδή ο X_m είναι το μιγαδικό συζυγές του X_{-m} .

Μπορούμε να δείξουμε και μία ιδιότητα που ισχύει όταν το $x(t)$ είναι συμμετρικό ως προς το $t = 0$, δηλαδή $x(t) = x(-t)$ και όταν τα άκρα είναι συμμετρικά, δηλαδή $t_a = -t_b$. Τότε θα έχουμε:

$$X_m = \frac{1}{T} \int_{-t_b}^{t_b} x(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt = \frac{1}{T} \int_0^{t_b} x(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt + \frac{1}{T} \int_{-t_b}^0 x(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt \quad (22)$$

Για το δεύτερο ολοκλήρωμα θέτουμε $t' = -t$ οπότε

$$\frac{1}{T} \int_{-t_b}^0 x(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt = -\frac{1}{T} \int_{t_b}^0 x(t') \exp\left(j \frac{2\pi m t'}{T}\right) dt' = \frac{1}{T} \int_0^{t_b} x(t') \exp\left(j \frac{2\pi m t'}{T}\right) dt' \quad (23)$$

Για τους συντελεστές Fourier μπορούμε επομένως να γράψουμε:

$$X_m = \frac{1}{T} \int_0^{t_b} x(t) \exp\left(-j \frac{2\pi m t}{T}\right) dt + \frac{1}{T} \int_0^{t_b} x(t) \exp\left(j \frac{2\pi m t}{T}\right) dt = \frac{1}{T} \int_0^{t_b} x(t) \left[\exp\left(-j \frac{2\pi m t}{T}\right) + \exp\left(j \frac{2\pi m t}{T}\right) \right] dt \quad (24)$$

Προσθέτοντας κατά μέλη τις (12) και (13) προκύπτει ότι:

$$2 \cos \phi = e^{j\phi} + e^{-j\phi} \quad (25)$$

Βάσει και τις παραπάνω εξίσωσης,

$$X_m = \frac{2}{T} \int_0^{t_b} x(t) \cos\left(\frac{2\pi m t}{T}\right) dt \quad (26)$$

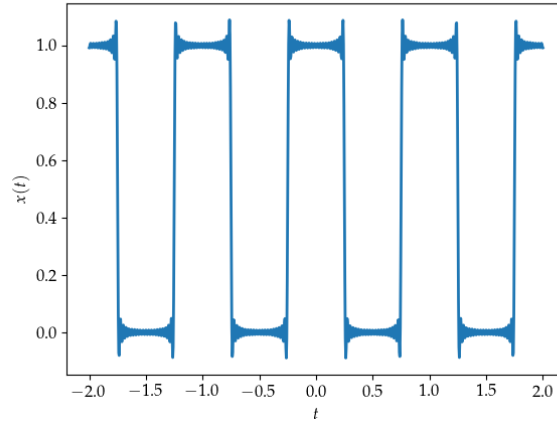
Το συνημίτονο έχει την ιδιότητα $\cos(-\phi) = \cos(\phi)$ και επομένως προκύπτει ότι

$$X_m = X_{-m} \quad (27)$$

Για ένα σήμα που είναι πραγματικό και συμμετρικό γύρω από το $t = 0$ θα ισχύουν ταυτόχρονα οι (22) και (27), δηλαδή:

$$X_{-m} = X_m = X_m^* \quad (28)$$

από όπου προκύπτει ότι τα X_m είναι πραγματικά και συμμετρικά ως προς το m . Οι συνθήκες αυτές πληρούνται για το τετραγωνικό σήμα της (8) οπότε οι συντελεστές P_m που δείχνουμε στις εικόνες 3b και 4b είναι πραγματικοί και συμμετρικοί ως προς το m .



Εικόνα 7: Υπολογίζοντας την σειρά [Fourier](#) εκτός του πεδίου ορισμού.

5 Περιοδικά σήματα

```

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 # 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 coefficients
20 def Pm(m, T1, T):
21     fm = m / T
22     return T1 / T * np.sinc( fm * T1 )
23
24 # Harmonic functions
25 def harmonic(t, m, T):
26     return np.exp(1j * 2 * np.pi * m / T * t)
27
28 # Fourier sum
29 def fourier_sum(t, M, T, Xm):
30     x = np.zeros(t.size, dtype=complex)
31     i = 0
32     for m in range(-M, M+1 ):
33         x += Xm[i] * harmonic(t, m, T)
34         i += 1
35
36     return x
37
38
39 T1 = 0.5
40 T = 1
41 M = 40
42 t = np.linspace(-2*T, 2*T, 1000)
43 plt.close('all')
44 plt.figure()
45 p = square_pulse(t, T1)
46 m = np.arange(-M, M+1, 1)
47 Xm = Pm(m, T1, T)
48 xM = fourier_sum(t, M, T, Xm)
49

```

```

50 plt.close('all')
51 plt.plot(t, xM, '-')
52 plt.ylabel('$x(t)$')
53 plt.xlabel('$t$')

```

Listing 5: fouriersum.py

Μέχρι τώρα, χρησιμοποιήσαμε την σειρά [Fourier \(3\)](#) για να αναπαραστήσουμε ένα σήμα μέσα στο διάστημα $[t_a, t_b]$. Τι θα συμβεί αν χρησιμοποιήσουμε τη σειρά [Fourier](#) για μία τιμή του t που δεν ανήκει στο $[t_a, t_b]$; Ας πάρουμε μία ιδέα πρώτα από την Python. Χρησιμοποιούμε το [listing 5](#) για να υπολογίσουμε την σειρά [Fourier](#) εκτός του διαστήματος $[-T/2, T/2]$ και λαμβάνουμε το αποτέλεσμα στην εικόνα [7](#). Παρατηρούμε επομένως ότι εκτός του διαστήματος που λαμβάνουμε υπόψη μας για τον υπολογισμό των συντελεστών [Fourier](#), η σειρά [Fourier](#) μας δίνει μία περιοδική επανάληψη του αρχικού σήματος. Δεν είναι δύσκολο να καταλάβουμε γιατί: τα αρμονικά σήματα που δίνονται από την [\(5\)](#) είναι περιοδικά:

$$F_m(t+T) = \exp\left(j\frac{2\pi m(t+T)}{T}\right) = \exp\left(j\frac{2\pi mt}{T} + j2\pi m\right) = \exp\left(j\frac{2\pi mt}{T}\right) = F_m(t) \quad (29)$$

οπότε σύμφωνα και με την [\(7\)](#) θα έχουμε:

$$x(t+T) = \sum_{m=-\infty}^{\infty} X_m F_m(t+T) = \sum_{m=-\infty}^{\infty} X_m F_m(t) = x(t) \quad (30)$$

Υπολογίζοντας δηλαδή την σειρά [Fourier](#) εκτός του αρχικού διαστήματος λαμβάνουμε μία περιοδική επανάληψη του σήματος. Εναλλακτικά μπορούμε να πούμε ότι εάν μας δώσουν ένα περιοδικό σήμα $x(t)$ με περίοδο T μπορούμε να χρησιμοποιήσουμε την σειρά [Fourier](#) για να το αναπαραστήσουμε υπολογίζοντας τους συντελεστές [Fourier](#) X_m μέσα σε μία βασική περίοδο, π.χ. $t_a = -T/2$ και $t_b = T/2$. Αυτό είναι ένα αρκετά χρήσιμο συμπέρασμα που θα το χρησιμοποιήσουμε και στα παρακάτω.

6 Ώρα για σπαζοκεφαλίες

1. Θεωρείστε το σήμα του φέροντος $x(t) = \cos(2\pi f_0 t)$ που είναι περιοδικό με περίοδο $T = 1/f_0$. Να υπολογίσετε τη σειρά [Fourier](#) θεωρώντας ότι $t_a = 0, t_b = T$. Ποιοί συντελεστές [Fourier](#) είναι μη μηδενικοί; Πως ερμηνεύεται αυτό;
2. Στην εικόνα [6](#) φαίνεται ότι το σφάλμα μειώνεται όσο αυξάνει το M αλλά διακρίνουμε κάποια σκαλοπατάκια. Που κατά την γνώμη σας οφείλεται αυτό;
3. Να φτιάξετε ένα Python script που να υπολογίζει αριθμητικά τους συντελεστές [Fourier](#) χρησιμοποιώντας την `trapz` της `numpy`. Να επαληθεύσετε τον κώδικα υπολογίζοντας τους συντελεστές [Fourier](#) του τετραγωνικού σήματος και να συγκρίνετε το αποτέλεσμα με τη θεωρητική πρόβλεψη της [\(15\)](#).
4. Δείξτε ότι το σήμα:

$$x(t) = \sum_{m=-\infty}^{\infty} \delta(t - mT_s) \quad (31)$$

είναι περιοδικό και υπολογίστε τους συντελεστές [Fourier](#) του.