

Procesarea Semnalelor

Laboratorul 4

Eșantionare - Partea a II-a

1 Eșantionare sub-Nyquist

Reamintim că un semnal limitat în bandă este un semnal ale cărui componente de frecvență sunt 0 sau, mai general, ne semnificative, în afara unui interval $[-B(\text{Hz}), B(\text{Hz})]$. Semnalele centrate în $f_c \neq 0$ pot fi eșantionate corect și la frecvențe mai mici decât frecvența Nyquist prin centrarea unei replici în 0. Intervalul $[0, f_c]$ este astfel utilizat eficient. Procedura poartă numele de eșantionare trece-bandă (*bandpass sampling*). Ca orice eșantionare, și aceasta produce replici, iar pentru a evita alierea, este necesar ca acestea să nu se suprapună. Condițiile pentru o eșantionare eficientă sub-Nyquist, pentru un număr m de replici, sunt așadar

$$\frac{2f_c - B}{m} \geq f_s \geq \frac{2f_c + B}{m + 1} \quad (1)$$

$$f_s > 2B \quad (2)$$

2 Ghid Python

Debugging

Dacă lucrați în PyCharm, puteți vizualiza valorile variabilelor în modul Debugging. Adăugați un breakpoint la linia la care vreți să opriți programul și apăsați Debug în loc de Run (Shift+F9 în loc de Shift+10). Veți putea vedea detalii despre variabilele din program calculate până la acea linie în fereastra Debug. Folosiți varianta aceasta pentru a verifica dacă ați făcut corect eșantionarea (axa de timp conține momentele corecte) pentru Exercițiul 1.

Reprezentare grafică

Dacă doriți ca funcția `stem` să nu afișeze și liniile verticale, iar culoarea folosită să nu fie cea implicită, puteți folosi secvența de mai jos. Parametrul pentru

culoare, transmis ca *string* între apostrofuri (*red* în exemplul de mai jos) poate fi transmis și prin codul acesteia în format [hexazecimal](#).

```
markerline,_,_ = axs[1].stem(x2, y2, linefmt='none', basefmt = 'none')
markerline.set_markerfacecolor('red')
markerline.set_markeredgecolor('red')
```

3 Exerciții

1. Prespuneți că într-o incintă temperatura este controlată astfel încât să se obțină o periodicitate noapte-zi asemănătoare unei sinusoide; maximul de temperatură (1°C) se atinge la ora 12:00, minimul (-1°C) la ora 24:00, nu există variații între zile. Începeți să măsurați temperatura din incintă într-o zi de miercuri, la ora 12:00.
 - (a) Dacă măsurați temperatura la fiecare 2 ore, care va fi frecvența de eșantionare (exprimată în Hz)?
 - (b) Care este perioada semnalului de temperatură și care este frecvența acestuia? Este eșantionarea din subpunctul precedent corectă în sensul teoremei Nyquist? Care este intervalul de timp minim la care trebuie măsurată temperatura astfel încât să îndeplinească criteriul de eșantionare Nyquist?
 - (c) Afișați grafic cu o curbă sinusoidă de temperatură pentru o săptămână și eșantioanele (semnalate grafic prin puncte) obținute cu frecvența de eșantionare de la punctul (a). Verificați conținutul variabilei de timp urmând indicațiile din Secțiunea 2.
 - (d) Dacă măsurați temperatura o dată la 5 ore, în ce zi și la ce oră veți obține din nou valoarea din primul eșantion (cel de miercuri, ora 12:00)?
 - (e) Dacă la (c) ați folosit funcția `numpy.linspace` pentru a genera axa de timp, cum va trebui modificată aceasta (sau ce altă opțiune de a genera axa de timp aveți?) pentru o nouă perioadă de eșantionare, de 5 ore? Indicație: care este relația între frecvența sinusoidei și frecvența de eșantionare în acest caz?
 - (f) Generați eșantioanele pentru noua perioadă de eșantionare (5 ore). Afișați-le pe un grafic nou folosind `plt.plot()`. Ce observați?
 - (g) Creați un nou grafic în care afișați eșantioanele obținute la subpunctul precedent folosind, de data aceasta, `plt.stem()`. Adăugați pe acest grafic, cu alte culori, sinusoida și eșantioanele de la subpunctul (c).
2. Afișați grafic o sinusoidă de frecvență și fază aleasă de voi, ale cărei valori minime și maxime sunt 3, respectiv 7.

3. Calculați frecvența optimă de eșantionare sub-Nyquist pentru un semnal de bandă $B = 10\text{Hz}$ centrat în jurul lui $f_c = 90\text{Hz}$ pentru următoarele 3 valori posibile ale numărului de replici: a) $m = 1$, b) $m = 2$, c) $m = 4$.