

Digitalna obrada signala

Laboratorijska vježba 3

5. decembar 2022.

Furijeova transformacija diskretnog signala. DFT i FFT.

Teorijska osnova vježbe

U sklopu ovih laboratorijskih vježbi analiziraće se diskretna Furijeova transformacija (DFT) kao i inverzna diskretna Furijeova transformacija (IDFT). DFT predstavlja jedan od najkorišćenijih alata u digitalnoj obradi signala.

Diskretna i inverzna diskretna Furijeova transformacija

Diskretna Furijeova transformacija signala $x(n)$, $n = 0, 1, \dots, N_0 - 1$ se definiše na sljedeći način:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}nk}, \quad k = 0, 1, \dots, N-1,$$

dok se originalni signal može dobiti iz DFT koeficijenata korišćenjem inverzne DFT:

$$x(n) = \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}nk}, \quad n = 0, 1, \dots, N-1.$$

Implicitno se podrazumijeva da je signal dobijen inverznom DFT **periodično** produžen. Parametar N predstavlja broj tačaka u kojima se DFT računa, i on mora zadovoljavati uslov $N \geq N_0$, gdje je sa N_0 označena dužina signala $x(n)$. DFT je moguće računati samo za signale konačnog trajanja, budući da diskretizacija Furijeove transformacije diskretnog signala dovodi do periodičnog produžavanja signala u vremenskom domenu. Za vježbu, napisati MATLAB/Octave program kojim se implementiraju gornje relacije.

Numerička složenost računanja DFT-a po definiciji je takva da je broj množenja reda $(N-1)^2$, dok je broj sabiranja reda N^2 . Drugim riječima, numerička složenost je $O(N^2)$. U praksi, DFT se računa primjenom algoritama za brzo izračunavanje, poznatih kao *Fast Fourier Transform (FFT)* algoritmi. U slučaju kada je $N = 2^l$, gdje je l cio broj, numerička složenost je $O(N \log_2 N)$. Za vježbu, samostalno napisati programe kojima se implementiraju FFT algoritmi koji su rađeni na časovima predavanja i računskih vježbi.

U MATLAB/Octave okruženju, FFT se računa korišćenjem funkcije `x=fft(x,N)`, dok se inverzna DFT računa pomoću `x=ifft(X,N)`.

Primjer

Odredimo DFT signala

$$x(n) = \sum_{l=0}^{199} l\delta(n-l).$$

Kod kojim se rješava zadatak je prikazan ispod. Prve dvije linije u kodu služe za definisanje posmatranog signala. U trećoj liniji definiše se $N = N_x$, odnosno, DFT će se računati u najmanjem mogućem broju tačaka. U četvrtoj liniji računamo DFT, dok u petoj liniji definišemo diskretni indeks frekvencije. Amplitudski spektar prikazujemo u linijom 9.

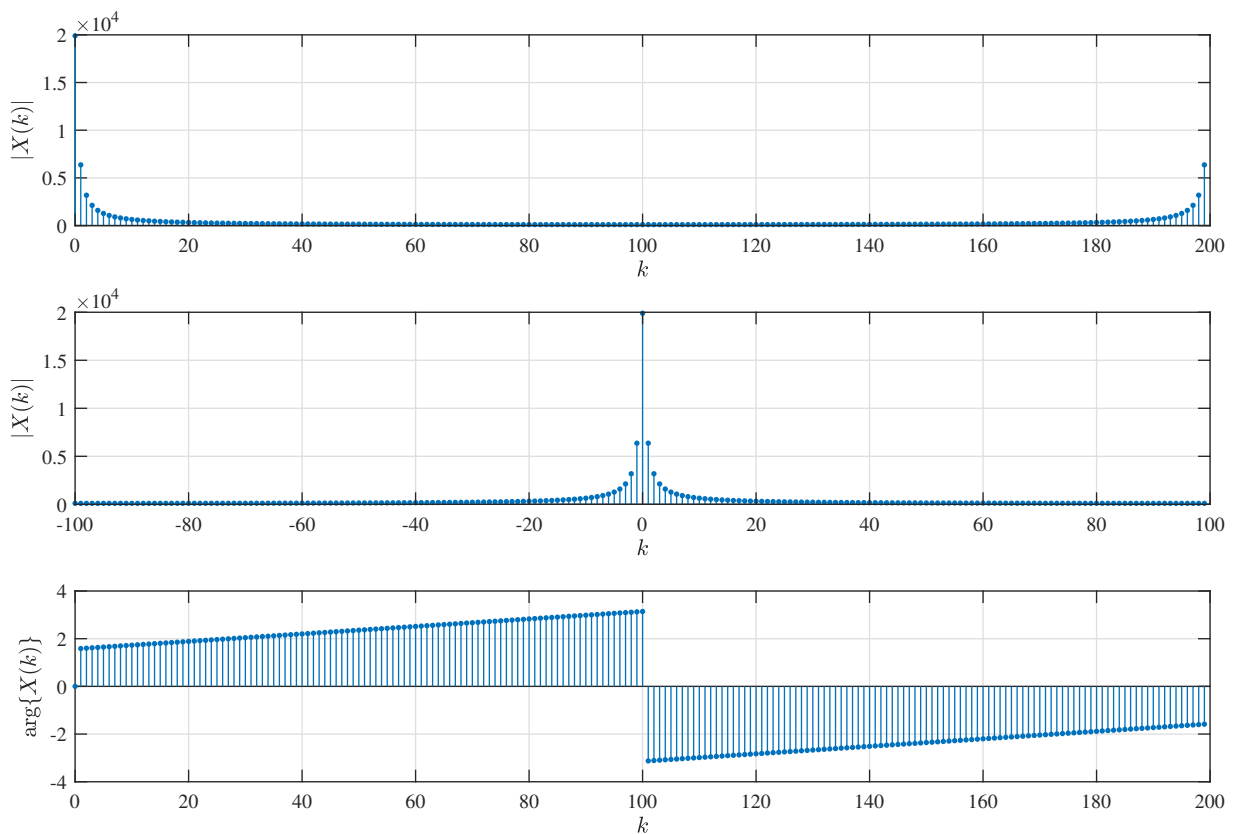
Linija 16 sadrži prikaz amplitudskog spektra, ali tako da su DFT odbirci sa frekvencijskim idneksima od $N/2$ do $N-1$ prebačeni ispred $k=0$. Navedno se postiže funkcijom `fftshift`. Odgovarajući frekvencijski indeksi su definisani u liniji 15. Isprobajte šta daje kao rezultat `fftshift(1:10)` i `fftshift(1:11)`. U liniji 22 prikazana je faza DFT-a posmatranog signala (fazni spektar). Uočiti da pri računanju faze takođe ima smisla primijeniti `fftshift`. U liniji 27 izračunata je IDFT, pa je rezultat u liniji 28 upoređen sa originanim signalom (izračunata je srednja kvadratna greška

$$MSE = \sum_{n=0}^{N-1} |x(n) - x_i(n)|^2,$$

gdje je $x_i(n)$ signal dobijen inverznom DFT od $X(k)$). Kao što očekujemo, dobija se da je MSE jednaka nuli. Rezultat pokretanja opisanog koda je prikazan na slici 1 (uz određena dodatna formatiranja).

```
1 n=0:199;
2 x=n;
3 N=length(x)
4 X=fft(x);
5 k=0:N-1;
6
7 figure(1)
8 subplot(3,1,1)
9 stem(k,abs(X),'.')
10 xlabel('k')
11 ylabel('|X(k)|')
12 grid
13
14 subplot(3,1,2)
15 k1=-N/2:N/2-1;
16 stem(k1,fftshift(abs(X)),'.')
17 xlabel('k')
18 ylabel('|X(k)|')
19 grid
20
21 subplot(3,1,3)
22 stem(k,angle(X),'.')
23 xlabel('k')
24 ylabel('arg{X(k)}')
25 grid
26
27 x_i=ifft(X);
28 greska=sum(abs(x-x_i).^2)
```

Funkcija `fft` može se koristiti sa dva argumenta. Drugi argument je broj tačaka u kojima se računa DFT. Ukoliko se ne zada, uzima se N koje je jednako dužini posmatranog signala. Sa većim N se postiže bolja rezolucija, što je naročito bitno u slučaju kada su analizirani signali kratkog trajanja.



Slika 1: DFT signala: prvi red – amplitudski spektar bez korišćenja `fftshift` funkcije, drugi red – amplitudski spektar uz korišćenje `fftshift` funkcije, treći red – fazni spektar.

Primjer

Prikazati DFT signala $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)$ za najmanje moguće N , kao i za $N = 32$ i za $N = 256$. Zadatak je riješen narednim kodom, dok je odovarajući rezultat prikazan na slici 2. U slučajevima kada se DFT računa u velikom broju tačaka, za prikaz spektra se najčešće koristi funkcija `plot` (prikazati za vježbu posljednji rezultat pomoću ove funkcije).

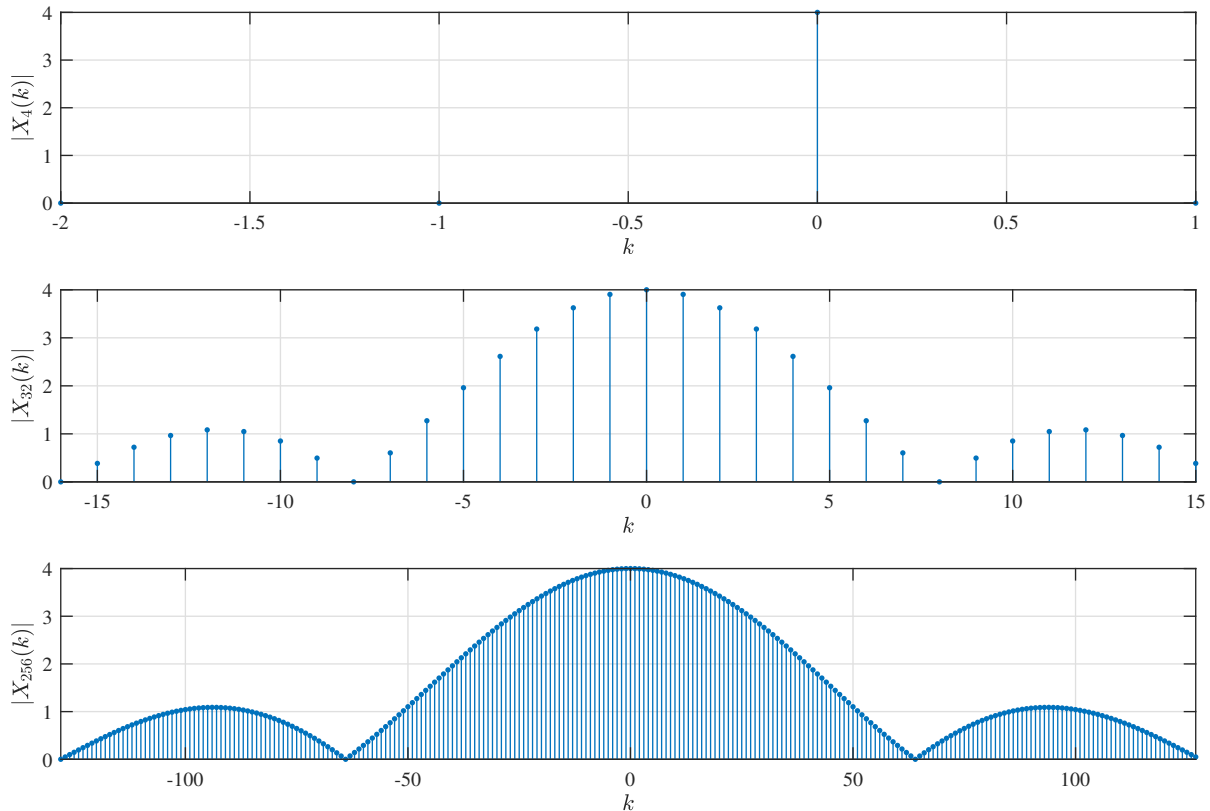
```

1 x=ones(1,4);
2
3 figure(1)
4 subplot(3,1,1)
5 N=4;
6 X_4=fft(x); % ili X_4=fft(x,N);
7 k=-N/2:N/2-1;
8 stem(k,fftshift(abs(X_4)),'.')
9 xlabel('k')
10 ylabel('|X_4(k)|')
11 axis tight
12
13 subplot(3,1,2)
14 N=32;
15 X_32=fft(x,N);
16 k=-N/2:N/2-1;
17 stem(k,fftshift(abs(X_32)),'.')
18 xlabel('k')
19 ylabel('|X_{32}(k)|')
```

```

20 axis tight
21
22 subplot(3,1,3)
23 N=256;
24 X_256=fft(x,N);
25 k=-N/2:N/2-1;
26 stem(k,fftshift(abs(X_256)),'.')
27 xlabel('k')
28 ylabel('|X_{256}(k)|')
29 axis tight

```



Slika 2: DFT računata za različite vrijednosti parametra N (broja tačaka).

DFT signala dobijenih odabiranjem kontinualnih signala

DFT se često koristi za Furijeovu analizu kontinualnih signala, koji se, za potrebu takve analize, odabiraju. Za analizu takvih signala, pogodno je prikazivati analogne frekvencije Ω . DFT predstavlja odabranu FT diskretnog signala, odnosno:

$$X(k) = X(e^{j\omega})|_{\omega=\frac{2\pi k}{N}}, \quad k = 0, 1, \dots, N-1.$$

Ako se na DFT prethodno primijeni `fftshift`, tada se uzima $k = -N/2, \dots, N/2 - 1$, gdje, bez gubljenja opštosti, pretpostavljamo da je N parno. Ako N nije parno, na odgovarajućim mjestima se uzima $(N-1)/2$. Kako je $\omega = \Omega\Delta t$, to analogne frekvencije možemo izraziti preko k na sljedeći način:

$$\Omega = \frac{2\pi k}{N\Delta t}, \quad k = -N/2, \dots, N/2 - 1. \quad (1)$$

Primjer

Odredimo Furijeovu transformaciju analognog signala:

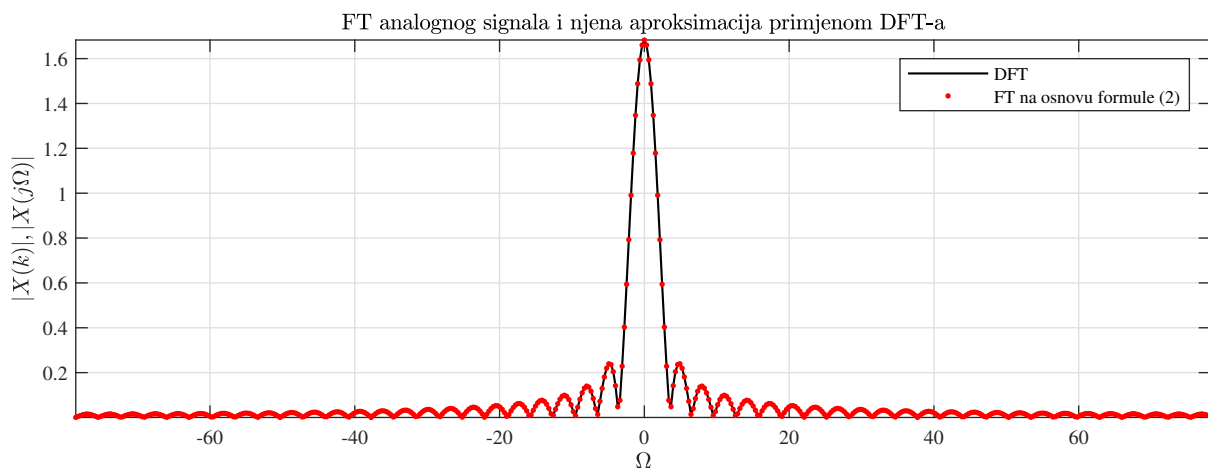
$$x(t) = \cos(t)[u(t+1) - u(t-1)].$$

Furijeova transformacija je:

$$X(j\Omega) = \int_{-\infty}^{\infty} x(t)e^{-j\Omega t} dt = \int_{-1}^1 \cos(t)e^{-j\Omega t} dt = j \left(\frac{\sin(\Omega-1)}{\Omega-1} + \frac{\sin(\Omega+1)}{\Omega+1} \right). \quad (2)$$

Odaberimo signal sa „dovoljno” malim korakom $\Delta t = 0.04$. Izračunajmo DFT signala, i prikazimo ga pomoću funkcije `plot`, sa Ω računatim kao u (1), što je dato u linijama koda 6–10. Zatim na istom grafiku prikazimo i (2), linije koda 12–13. Vidimo da se spektri gotovo u potpunosti poklapaju. Izvjesna greška postoji zbog korišćenja relativno velikog koraka odabiranja (koristimo relativno veliki korak u svrhu adekvatnog prikaza rezultata). Kod je dat ispod, dok je rezultat, uz određene vizuelne modifikacije, prikazan na slici 3.

```
1 dt=0.04;
2 t=-1:dt:1-dt;
3 x=dt*cos(t);
4
5 figure(1)
6 N=1024;
7 X=fft(x,N);
8 k=-N/2:N/2-1;
9 Omega=2*pi*k/(N*dt);
10 plot(Omega,fftshift(abs(X)),'k');
11 hold on
12 Xa=j*(sin(Omega-1)./(Omega-1))+j*(sin(Omega+1)./(Omega+1));
13 plot(Omega,abs(Xa),'r. ');
14 axis tight
```



Slika 3: Analogna Furijeova transformacija i njena aproksimacija pomoću DFT-a odgovarajućeg odabranog signala.

Uočimo da je vjerodostojan spektar diskretnog signala moguće dobiti **samo za** $x(n) = \Delta t x(n\Delta t)$, odnosno, prilikom diskretizacije, neophodno je množenje sa korakom odabiranja Δt .

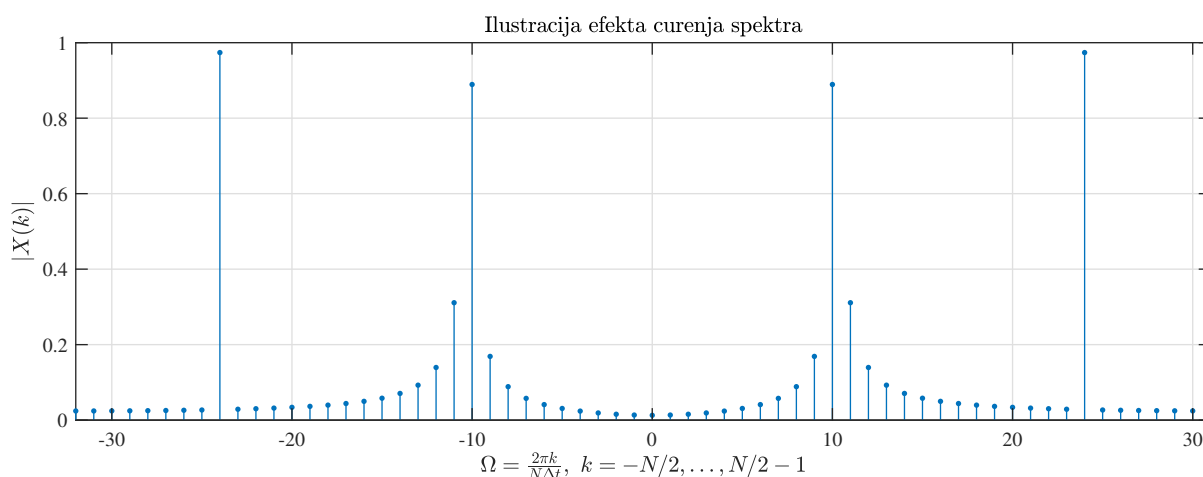
Efektat „curenja” spektra

U primjerima koje smo posmatrali, dobijeni rezultati se u potpunosti slažu sa očekivanjem. U nastavku će biti ukazano na neke probleme koji se mogu pojaviti korišćenjem diskretne Furijeove transformacije za dobijanje odbiraka Furijeove transformacije signala. Najčešće se dešava da nije uzet cio broj perioda odabiranog signala. Posledica navedenog je pojava tzv. „curenja” spektra. Kako to izgleda u konkretnom slučaju vidjećemo na sledećem primjeru.

Primjer

Posmatraćemo dvokomponentni signal, kod kojeg je odabiranjem „uzeto” 10.25 perioda prve komponente i 24 periode druge komponente. Rezultat je prikazan na slici 4.

```
1 dt=pi/32; Tmax=2*pi-dt; t=0:dt:Tmax;
2 x=dt*(sin(10.25*t)+cos(24*t));
3 N=length(x);
4 X=fftshift(abs(fft(x)));
5 k=-N/2:N/2-1;
6 Omega=2*pi*k/(N*dt);
7 stem(Omega,X/pi)
```



Slika 4: Ilustracija curenja spektra. Čak je i amplituda komponente kod koje se efekat ne pojavljuje oštećena (smanjena) usljed uticaja druge komponente kod koje se pojavilo curenje spektra.

Napomena: prisjetiti se zadatka sa odabiranjem koji je rađen na računskim vježbama, u kojem je objašnjeno zašto je neophodno dijeljenje sa π rezultujuće Furijeove transformacije!

Sa dobijenog grafika primjećujemo da je druga komponenta signala (sa $\Omega = 24$) prikazana u skladu sa očekivanjima dok se kod prve komponente javlja „curenje” spektra, odnosno dobijamo spektralne komponente koje u kontinualnom signalu ne postoje. Dodatno, uočimo da je komponenta na $\Omega = 24$ dodatno oštećena spektralnom komponentom koja potiče od komponente sa frekvencijom $\Omega = 10.25$. Ovu pojavu možemo eliminisati tako što ćemo podesiti dužinu analiziranog signala (odnosno korak odabiranja) tako da se perioda posmatranog signala u njemu sadrži cio broj puta, ili uzeti signal dovoljne dužine tako da se smanji „curenje”. Uočiti šta se dešava ako se signal u prethodnom primjeru definiše kao:

```
1 x=dt*(cos(24*t));
```

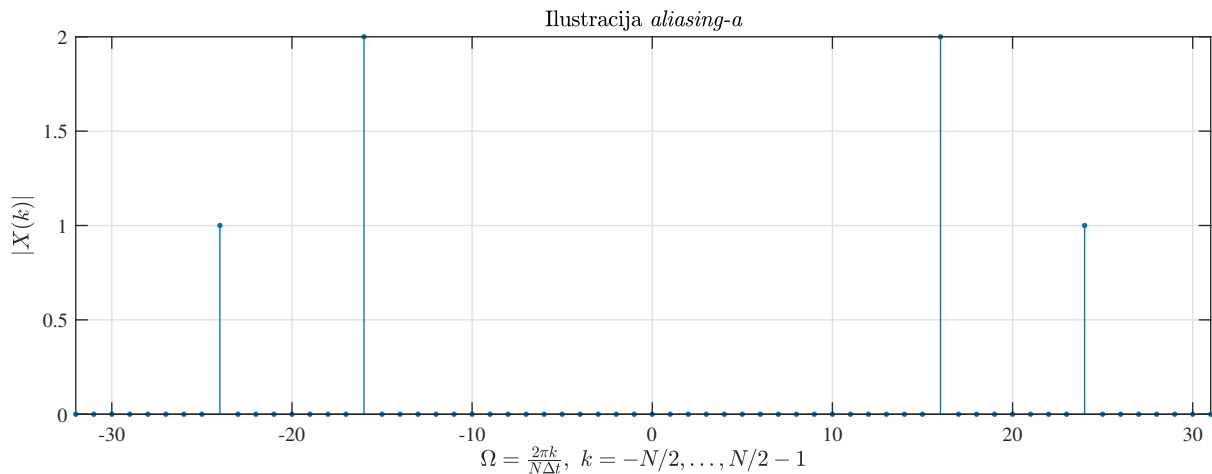
Pri radu sa odbircima kontinualnog signala mora se voditi računa o teoremi o odabiranju. Ukoliko ona nije zadovoljena, dolazi do *aliasing*-a. Primjer u kojem dolazi do *aliasing*-a je dat u kodu ispod:

```

1 dt=pi/32;Tmax=2*pi-dt;t=0:dt:Tmax;
2 x=dt*(sin(24*t)+2*cos(48*t));
3 N=length(x);
4 X=fftshift(abs(fft(x)));
5 k=-N/2:N/2-1;
6 Omega=2*pi*k/(N*dt);
7 stem(Omega,X/pi)

```

U prethodnom primjeru druga komponenta signala **ne zadovoljava** uslove teoreme o odabiranju, pa se u spektru pojavljuje komponenta na $\Omega = 16$, koje u originalnom signalu nema, kao što je prikazano na slici 5.



Slika 5: Ilustracija *aliasing* efekta i pojavljivanja spektralne komponente na pogrešnoj frekvenciji.

Računanje konvolucije primjenom DFT-a

Sada ćemo pokušati da konvoluciju dva diskretna signala izračunamo pomoću diskretne Furijeove transformacije. Iz teorije znamo da je dobijeni rezultat **cirkularna konvolucija**, pa ako želimo dobiti pravu konvoluciju dvije sekvence, treba ih dopuniti nulama (minimalna dužina produženih sekvenci treba biti dužina rezultata).

```

1 x=[1 2 -1 3];
2 y=[1 0 3];
3 minduz=length(x)+length(y)-1;
4 xp=[x zeros(1,minduz-length(x))];
5 yp=[y zeros(1,minduz-length(y))];
6 X=fft(xp);Y=fft(yp);
7 Z=X.*Y;
8 z=ifft(Z)
9 zt=conv(x,y)

```

Vrijednosti z i z_t možemo uporediti sljedećom linijom koda:

```

1 abs(z-zt)

```

Algoritam računanja DFT u MATLAB/Octave okruženju zavisi od broja elemenata polazne sekvence i najbolji rezultati po pitanju brzine se dobijaju kada je dužina polazne sekvence stepen broja 2. Tako prethodni primjer možemo modifikovati sa:

```

1 minduz=2^nextpow2(length(x)+length(y)-1);

```

čime se naznačava da za dužinu sekvenci i rezultata treba uzeti prvi veći stepen broja dva.

Drugi način za realizaciju primjera sa računanjem konvolucije je da ne koristimo produžene signale x_p i y_p već da `fft` naredbi damo do znanja da treba izvršiti produžavanje signala, odnosno:

```
1 X=fft(x,minduz);Y=fft(y,minduz);  
2 Z=X.*Y;  
3 z=real(ifft(Z))
```

Zadaci

1. Otkucati sljedeće komande i protumačiti rezultate:

```
1 fft([1,1,1,1]); fftshift([1,2,3,4]); n=0:31;  
2 x=cos(pi*n*0);  
3 X=fftshift(abs(fft(x))); stem(X)
```

Zatim liniju 2 u gornjem kodu zamijeniti sa:

- (a) `x=sin(pi*n);`
- (b) `x=sin(pi*n*15/16);`
- (c) `x=sin(pi*n*8/16);`
- (d) `a=8.25; x=sin(pi*n*a/16);`

Zadatak pod d) se razlikuje od ostalih. Smanjivati vrijednost parametra $a = 8.25$ na 8.15, 8.10, 8.01 i 8, i promjene pratiti u istom grafičkom prozoru. Kako se naziva pojava koja nastaje u posmatranom slučaju i čega je ona posljedica?

U svim slučajevima izračunati energije signala. Računati na dva načina – na osnovu odbiraka u vremenskom, i u frekvencijskom domenu. Uporediti vrijednosti, i obrazložiti razloge za sličnosti, odnosno razlike.

2. Dat je diskretni signal $x=[1, 1.5, 1, 0.5]$. Odrediti i nacrtati njegovu DFT za $N = 4, 8, 16, 32$ i 64 (signal se dopunjava nulama). Za $N = 16$ nacrtati tri periode DFT.
3. Koristeći se DFT sa $N = 16$ odrediti konvoluciju $y(n)$ signala: $x=[1, 2, 3, 2, 1]$ i $h=[1, 1, 1]$.
4. Ponoviti prethodni zadatak uz $N = 5$. Prokomentarisati rezultat.
5. Dat je analogni signal $x(t) = 2\sin(\frac{5}{8}\pi t) + \cos(\frac{7}{8}\pi t)$. Signal diskretizovati sa korakom $\Delta t = 0.25$, $\Delta t = 0.5$, $\Delta t = 1$ i $\Delta t = 2$, pri čemu treba uzeti 256 odbiraka signala. Nacrtati DFT dobijenih diskretnih signala (na apscisnoj osi postaviti analognu frekvenciju Ω). Ponoviti proceduru sa 257 odbiraka uz $\Delta t = 0.5$. Obrazložiti razlike.
6. $x(n)$ je periodičan, sa periodom $N_0 = 4$, realan signal. Poznati su koeficijenti njegovog DFT-a $X_4(0) = 3$, $X_4(1) = j$, $X_4(2) = -1$, za $N = 4$. Odrediti odbirke DFT $X_{12}(k)$ za $N = 12$, za svako k .