Odmjeravanje i spektralna analiza audio signala

1 Osnovni signali u MATLAB-u

Osnovni signali koji se često koriste u digitalnoj obradi signala su jedinični impuls, eksponencijalne funkcije, sinusne funkcije i njihova generalizacija na kompleksne eksponencijalne funkcije. Budući da je digitalni signal sekvenca (niz) brojeva, a osnovna struktura podataka u MATLAB-u je matrica dimenzija $M \times N$, onda je predstavljanje signala u MATLAB-u očigledno – pomoću matrica dimenzija $N \times 1$ (vrsta-vektor) ili $1 \times M$ (kolona-vektor) čiji su elementi odmjerci signala sa kojim radimo.

Kada se radi sa signalima u MATLAB-u važno je voditi računa o dva pitanja. Prvo je da su svi signali u MATLAB-u konačnog trajanja što se razlikuje od analitičkog rješavanja problema kada je signal beskonačnog trajanja moguće predstaviti matematičkim izrazom. Drugo pitanje se odnosi na korespondenciju vrijednosti indeksa vektora u kojem se nalazi signal i indeksa vremena. U MATLAB-u vrijednosti indeksa nizova počinju od 1. Dakle, indeksi elemenata vektora sa N elemenata su $1, 2, \dots, N$. Sa druge strane, signali se često posmatraju u vremenskom intervalu koji počinje u nuli tako da indeks vremena uzima vrijednosti od 0 do $N-1$. Moguće je čak i da posmatranje signala počne u bilo kojem proizvoljnom vremenskom trenutku kojem može odgovarati čak i negativan vremenski indeks, npr. $-N$. Nažalost, informaciju o vremenskom intervalu, tj. vremenskim indeksima u kojima su definisani odmjerci signala ne možemo pridružiti vektoru u kojem se nalazi signal već se ova informacija mora čuvati odvojeno od signala i koristiti kada je to potrebno. Obično ovo pitanje nije od značaja sve dok se ne dođe do grafičke reprezentacije signala. U ovom slučaju potrebno je apscisu označiti na adekvatan način.

Od posebnog značaja su prostoperiodični, sinusni, signali. Diskretna sinusoida je u potpunosti određena pomoću tri parametra, amplitude (A), (digitalne) frekvencije (ω) i početne faze (ϕ)

$$x(n) = A\cos(\omega n + \phi)\,. \tag{1}$$

Često diskretni signali nastaju odmjeravanjem kontinualnih signala, kao što je npr. sinusni signal. U opštem slučaju kontinualna sinusoida je data sledećom jednačinom:

$$x_a(t) = A\cos(2\pi F t + \phi), \tag{2}$$

gdje je A amplituda, F frekvencija u Hercima, a ϕ početna faza signala. Ako se diskretni signal dobija odmjeravanjem signala $x_a(t)$, brzinom odmjeravanja $F_s = \frac{1}{T}$, dobijamo

$$x(n) = x_a(nT) = x_a(t)|_{t=nT} = A\cos\left(2\pi\frac{F}{F_s}n + \phi\right). \tag{3}$$

Poređenjem ove jednačine sa jednačinom diskretne sinusoide vidimo da je frekvencija dobijene diskretne sinusoide (digitalna frekvencija) jednaka

$$\omega = 2\pi\frac{F}{F_s} \tag{4}$$

Prilikom generisanja signala dobro je nastojati da se iskoriste MATLAB-ove mogućnosti za rad sa vektorima kako bi se izbjegle spore for petlje. Na primjer, da bi se generisao vektor sa odmjercima sinusnog signala moguće je upotrebiti sledeći kod:

```
Fs = 11025;
nsamp = 100;
for n = 0:nsamp-1
   x(n+1) = sin(2*pi*440*n/Fs);
end
```

Međutim, mnogo efikasnije je to uraditi na sledeći način:

```
Fs = 11025;
nsamp = 100;
n = 0:nsamp-1;
x = sin(2*pi*440*n/Fs);
```

Iako je MATLAB jezik sa slabom tipizacijom podataka i dinamičkom alokacijom memorije, bolje performanse se dobijaju ako se memorija za velike nizove (matrice) alocira unaprijed, npr. kao niz odgovarajućih dimenzija čiji su svi elementi nule.

2 Spektralna analiza audio signala

Furijeova transformacija diskretnog signala $x(n)$ je

$$X\left(e^{j\omega}\right)=\sum_{n=-\infty}^{\infty} x(n)\, e^{-j\omega n}. \tag{5}$$

Važno je uočiti da je Furijeova transformacija diskretnog signala kontinualna funkcija veličine označene sa ω koja se obično naziva digitalna frekvencija. Veza između digitalne i analogne frekvencije (F, izražene u Hercima) data je sa (4).

Pošto je naš cilj da pomoću računara izračunamo spektar signala onda je neophodno diskretizovati frekvencijsku promjenljivu ω. Pošto je Furijeova transformacija diskretnog signala periodična sa periodom 2π, uobičajeno je da se diskretizacija vrši u N tačaka ravnomjerno raspoređenih na osnovnom periodu Furijeove transformacije, tj. u frekvencijama

$$\omega_k=\frac{2\pi k}{N},\ k=0,\ldots,N-1. \tag{6}$$

Može se pokazati da ukoliko je broj tačaka N u kojima se odmjerava Furijeova transformacija veći ili jednak od trajanja signala onda odmjerci Furijeove transformacije odgovaraju vrijednostima diskretne Furijeove transformacije (DFT)

$$X(k)=\sum_{n=0}^{N-1} x(n)\, e^{-j\frac{2\pi kn}{N}}. \tag{7}$$

DFT je obično implementirana pomoću FFT (Fast Fourier Transform) algoritma. Važno je napomenuti da se pomoću FFT tačno izračunava DFT signala. Postoji mnogo implementacija FFT algoritma i, u opštem slučaju, one rade za proizvoljnu dužinu signala, ali najbrže izračunavanje se postiže kada je dužina signala stepen broja 2. Srećom, ovo se uvijek može postići dopunjavanjem uzorka signala nulama do trajanja koje je stepen broja 2. Dodatna korist od dopunjavanja nulama je i finija interpolacija u frekvencijskom domenu. Za izračunavanje FFT u MATLAB-u se koriste sledeće funkcije:

- `fft(x, N)`, brza Furijeova transformacija niza `x` u `N` tačaka,
- `ifft(X, N)`, inverzna Furijeova transformacija niza `X` u `N` tačaka.

Ukoliko se ne zada broj tačaka u kojima se računa FFT (parametar N), onda se podrazumijeva da je N jednako dužini signala. Ukoliko je N veće od dužine signala, signal se interno dopunjava nulama do dužine N. Ukoliko je

N manje od dužine signala, FFT se izračunava samo za prvih N odmjeraka signala.

Sljedećim primjerom je ilustrovano izračunavanje i crtanje spektra sinusnog signala:

```
w = pi/16;
L = 64;
n = 0:L-1;
x = cos(w * n);
figure, stem(n, x);

N = 64;
X = fft(x, N);
w = 2*pi*[0:N-1]/N;
% DFT se racuna u diskretnim frekvencijama
figure, stem(w, abs(X));
title('DFT')
xlabel('\omega'); ylabel('|X(\omega)|')
% spektar diskretnog signala je kontinualna funkcija
% crtamo aproksimaciju dobijenu pomocu DFT
figure, plot(w, abs(X));
title('Amplitudni spektar')
xlabel('\omega'); ylabel('|X(\omega)|')

% cesto se digitalna frekvencija normalizuje sa pi
w = 2*[0:N-1]/N;
figure, stem(w, abs(X));
figure, plot(w, abs(X));
title('Amplitudni spektar')
xlabel('\omega/\pi'); ylabel('|X(\omega)|')
```

Važno je primijetiti da je u gornjem isječku koda za grafičko prikazivanje spektra signala korištena naredba plot. Razlog za ovo je što je Furijeova transformacija diskretnog signala kontinualna funkcija frekvencije ω. Često se amplitudni spektar crta na logaritamskoj skali i u tom slučaju se izražava u decibelima (dB).

$$X_{dB}(\omega) = 20 \log|X\left(e^{j\omega}\right)| \text{ [dB]}. \tag{8}$$

Amplitudni spektar izražen u decibelima se može nacrtati pomoću sljedećeg koda

```
figure, plot(w, 20*log10(abs(X)));
title('Amplitudni spektar')
xlabel('\omega/\pi'); ylabel('|X(\omega) [dB]|')
```

U slučaju nestacionarnih signala (kao što su npr. govor i muzika) spektar signala se mijenja sa vremenom. Za analizu takvih signala koristi se kratkotrajna Furijeova transformacija (Short Time Fourier Transform, STFT). Kratkotrajna Furijeova transformacija predstavlja sekvencu DFT primijenjenih na kratke segmente signala. Slična analiza zvučnog signala se vrši i u čovjekovom uhu. Grafička reprezentacija kratkotrajne Furijeove transformacije u funkciji vremena i frekvencije naziva se spektrogram. Spektrogram se intenzivno koristi u analizi i sintezi govora i muzike. U MATLAB-u (Signal Processing Toolboxu) spektrogram je implementiran funkcijom spectrogram čija je sintaksa:

```
[S,F,T,P] = spectrogram(x, window, overlap, Nfft, Fs)
```

Parametri su:

- `x` – vektor odmjeraka signala koji se analizira,
- `window` – vektor koji sadrži odmjerke prozorske funkcije koja se koristi prilikom izračunavanja spektrograma,
- `overlap` – broj odmjeraka za koliko se sukcesivni blokovi signala preklapaju.
- `Nfft` – broj tačaka u kojima se računa FFT bloka signala. Ukoliko je veći od trajanja signala vrši se dopunjavanje nulama,
- `Fs` – frekvencija odmjeravanja u hercima. Ova vrijednost se ne koristi pri izračunavanju spektrograma već samo za skaliranje frekvencijske ose.

Izlazne promjenljive su:

- `S` – vrijednosti spektrograma,
- `F` – frekvencije u kojima su izračunate vrijednosti spektrograma,
- `T` – trenuci (pozicije prozora) u kojima su izračunate vrijednosti spektrograma. Ove vrijednosti odgovaraju položaju centra prozora,
- `P` – estimacija spektralne gustine snage signala.

Ukoliko se funkcija pozove bez izlaznih argumenata onda funkcija crta 3D grafik estimacije spektralne gustine snage. U ovom slučaju, poslednji argument je lokacija frekvencijske ose. Na primjer,

```
spectrogram(x, hamming(128), 120, 128, 1000, 'yaxis')
```

daje isti rezultat kao:

```
[y,f,t,p] = spectrogram(x, hamming(128), 120, 128, 1000, ... 'yaxis');
surf(t,f,10*log10(abs(p)),'EdgeColor','none');
axis xy; axis tight; colormap(jet); view(0,90);
xlabel('Time');
ylabel('Frequency (Hz)')
```

Ako je za frekvencijsku osu izabrana y-osa, kao u gornjem primjeru, onda su u kolonama matrice y odmjerci DFT za odgovarajući položaj prozora.

3 Aditivna sinteza zvuka

Helmholtz je 1859. godine pokazao da je moguće sintetizovati zvukove pomoću elektromagnetno pobuđenih muzičkih viljušaka čiju je glasnoću kontrolisao pomoću pokrivača. Najefektnija demonstracija ove ideje je bila sintetizovanje zvukova vokala. Pošto je talasni oblik zvuka koji proizvodi zvučna viljuška približno sinusoida to znači da se sintetizovani signal iz Helmholtzovih eksperimenata može prikazati kao superpozicija sinusoida različitih frekvencija i amplituda. Ovo je osnova aditivne sinteze zvuka. Frekvencije sinusoidalnih komponenata mogu, ali ne moraju, biti harmonijski vezane, tj. biti cjelobrojni umnošci osnovne frekvencije. U slučaju da su frekvencije komponenata harmonijski vezane, sinusoidalne komponente se zovu harmonici. Za datu osnovnu frekvenciju f_1 i amplitude harmonika p_n, sintetizovani signal je dat sa:

$$x\left(t\right) = \sum_{n=1}^{N} p_n \cos\left(2\pi f_1 t + \theta_n\right). \tag{9}$$

Početne faze harmonika θ_n ne igraju nikakvu ulogu u sintezi audio signala pa se mogu zanemariti.

Jednačina (9) predstavlja sumu konačnog broja članova Furijeovog reda sa osnovnom frekvencijom f_1. Zbog toga se vrijednosti amplituda harmonika mogu dobiti Furijeovom analizom signala koji želimo da sintetizujemo.

Vrijednosti osnovne frekvencije i amplituda harmonika određuju visinu i boja sintetizovanog tona. Kako bi se dobilo realistično zvučanje tona potrebno je modelovati i njegovu anvelopu $e(t)$. Konačni ton se dobija množenjem polaznog tona i anvelope

$$y(t) = x(t)\, e(t)\,. \tag{10}$$

4 Zadaci

4.1 Generisanje osnovnih signala i teorema odmjeravanja

1. Generisati vektor t sa vrijednostima iz intervala $[0, 3]$ sa korakom 0.05.

2. Generisati signal $x_1(t) = \cos(2\pi t)$, pri čemu je t vektor definisan u prethodnoj tački. Nacrtati ovaj signal i adekvatno označiti ose. Koliki je period (frekvencija) ovog signala? Na istom grafiku (u istom prozoru) nacrtati ovaj signal korištenjem naredbe `stem` kako biste vidjeli pojedine odmjerke. Kolika je frekvencija odmjeravanja korištena?

3. Generisati i nacrtati prethodni signal, ali koristeći frekvenciju odmjeravanja $F_s = 10$ Hz.

4. Generisati i nacrtati prethodni signal, ali koristeći frekvenciju odmjeravanja $F_s = 2$ Hz.

5. Grafički prikaz signala odmjerenog visokom frekvencijom odmjeravanja možemo smatrati prikazom kontinualnog signala. U ovom slučaju to može biti grafik iz tačke 2. U novim prozorima nacrtati grafike na kojima su na grafik iz tačke 2. dodati odmjerci iz tačaka 3, odnosno, 4.

6. Generisati i nacrtati signal $x_2(t) = \cos(2\pi 19t)$ koristeći frekvenciju odmjeravanja $F_s = 10$ Hz. Uporediti ovaj signal sa signalom iz tačke 3. i komentarisati rezultat.

7. Generisati 2 s sinusnog signala frekvencije 440 Hz i amplitude 0.5. Frekvencija odmjeravanja je 11025 Hz. Reprodukovati ovaj signal pomoću `sound`.

8. Ponoviti prethodnu tačku za signale frekvencije 220 Hz i 880 Hz.

4.2 Spektri osnovnih signala

1. Generisati sinusoidu frekvencije 440 Hz, sa amplitudom 1 i trajanjem 1 s. Frekvencija odmjeravanja je 8 kHz. Izračunati i nacrtati njen spektar. Na apscisi treba da bude označena frekvencija u Hz.

2. Generisati složenoperiodični signal čija je osnovna frekvencija 440 Hz, trajanje 1 s, a amplitude harmonika su, redom, 1, 0.8, 0.1 i 0.04. Frekvencija odmjeravanja je 8 kHz. Izračunati i nacrtati spektar ovog signala. Na apscisi treba da bude označena frekvencija u Hz.

3. Generisati pravougaoni signal frekvencije 440 Hz. Izračunati i nacrtati spektar signala.

4. Generisati trougaoni signal frekvencije 440 Hz. Izračunati i nacrtati spektar signala.

5. Učitati signal `http://dsp.etfbl.net/multimedia/media/piano_A4.wav`. Reprodukovati signal, nacrtati talasni oblik i spektar signala. Uporediti dobijene rezultate sa prethodnim signalima.

4.3 Aditivna sinteza zvuka

1. Na osnovu amplitudnog spektra tona u fajlu `http://dsp.etfbl.net/multimedia/media/piano_A4.wav` sintetizovati ton koji ima istu visinu i boju i traje 1 s. Kako biste izbjegli odsijecanje signala prilikom reprodukcije normalizujte vrijednosti odmjeraka na interval $[-1, +1]$. Reprodukovati sintetizovani signal.

2. Pomnožiti sintetizovani ton eksponencijalnom anvelopom čija je vremenska konstanta jednaka 0.35 s. Reprodukovati dobijeni signal.

3. Napišite program u MATLAB-u koji će generisati C-dur ljestvicu. Vrijednosti frekvencija pojedinih tonova možete preuzeti iz članka u Vikipediji (`http://en.wikipedia.org/wiki/Piano_key_frequencies`)

4.4 Spektrogram

1. Generisati 4 tona:

$$\begin{aligned} x_1(t) &= \cos(2\pi 100t) \\ x_2(t) &= \cos(2\pi 200t) \\ x_3(t) &= \cos(2\pi 500t) \\ x_4(t) &= \cos(2\pi 1000t)\,. \end{aligned}$$

Koristiti frekvenciju odmjeravanja od 8000 Hz. Nacrtati spektrogram signala koji se dobije konkatenacijom ova 4 tona. Koristiti Hemingov (`hamming`) prozor dužine 256, preklapanje 128 odmjeraka, a FFT računati u 1024 tačke uz dopunjavanje nulama.

2. Dat je frekvencijski modulisan (chirp) signal $x(t) = \cos(2\pi\mu t^2)$. Generisati odmjerke ovog signala na intervalu $t = 0$ do $t = 1$ s. Uzeti da je $\mu = 11025$, a frekvencija odmjeravanja iznosi 22050 Hz. Reprodukovati signal. Nacrtati spektrogram ovog signala. Koristiti Hemingov (`hamming`) prozor dužine 256, preklapanje 255 odmjeraka, a FFT računati u 1024 tačke uz dopunjavanje nulama.

3. Učitati signal `http://dsp.etfbl.net/multimedia/media/piano_A4.wav`. Nacrtati spektrogram ovog signala. Koristiti Hemingov prozor dužine 512 i preklapanje 256 odmjeraka. FFT računati u 1024 tačke.

4. Učitati govorni signal u fajlu `http://dsp.etfbl.net/multimedia/media/iaeao.wav`. Nacrtati spektrogram ovog signala korištenjem Hemingovog prozora dužine 512 i preklapanje 480 odmjeraka. FFT računati u 1024 tačke. Pošto se govorni signal za vokale u prvoj aproksimaciji može modelirati izlazom iz filtra pobuđenog nizom impulsa, njegov talasni oblik je kvaziperiodičan i spektrogram dobijen sa ovim parametrima bi trebalo da se sastoji od mnogo linija na jednakim razmacima zbog kvaziperiodičnosti signala sa periodom (pitch period) od približno 10ms.

5. Nacrtati spektrogram signala iz prethodne tačke korištenjem Hemingovog prozora dužine 64 i preklapanje 32 odmjerka. FFT računati u 1024 tačke. U ovom slučaju bi spektralni oblik filtra vokalnog trakta trebalo da bude vidljiv u nekoliko (manje od 5) velikih rezonantnih pikova spektrograma.

6. Učitati govorni signal u fajlu `http://dsp.etfbl.net/multimedia/media/sat.wav`. Pošto je govorni signal nestacionaran njegov spektar je funkcija vremena i najčešće se predstavlja korištenjem spektrograma. Nacrtati spektrogram datog signala korišćenjem Hemingovog prozora dužine 512 odmjeraka sa preklapanjem sukcesivnih prozora od 481 odmjerak. Uočiti kakakteristične strukture koje postoje u govornom signalu.