

Kvantizacija zvučnog i video signala

Predmet: Multimedijalni signali i sistemi

Predavač: dr Nataša Savić

Asistent: Nikola Milutinović

Kvantizacija sinusnog signala

%% Parametri signala

```
A = 1;           % Amplituda sinusnog signala
f = 100;        % Frekvencija sinusnog signala [Hz]
Fs = 5000;     % Frekvencija uzorkovanja [Hz]
T = 0.01;      % Trajanje signala [s]
t = 0:1/Fs:T-1/Fs; % Vremenski vektor
```

% Originalni sinusni signal

```
x = A * sin(2*pi*f*t);
```

%% Parametri kvantizacije

```
nbits = 3;      % Broj bita kvantizatora
L = 2^nbits;    % Broj kvantizacionih nivoa
```

% Pretpostavimo uniformnu kvantizaciju u opsegu [-A, A]

```
xmin = -A;
xmax = A;
```

% Veličina koraka kvantizacije (quantization step)

```
delta = (xmax - xmin) / L;
```

% Nivoi kvantizacije (rekonstrukcioni nivoi – sredine intervala)

```
q_levels = xmin + delta/2 : delta : xmax - delta/2;
```

%% Kvantizacija (uniformni mid-tread)

% 1) Ograničavanje signala na zadati opseg

```
x_clipped = min(max(x, xmin), xmax);
```

% 2) Računanje indeksa kvantizacionih nivoa

% Pomeri se opseg da kreće od 0, podeli sa delta i zaokruži na najbliži nivo

```
q_index = floor( (x_clipped - xmin) / delta ); % indeksi 0,1,...,L-1
q_index(q_index == L) = L-1; % popravka za gornju granicu
```

% 3) Rekonstrukcija kvantizovanog signala preko nivoa

```
xq = q_levels(q_index + 1); % +1 jer MATLAB indeksira od 1
```

%% Kvantizaciona greška

```
e = x - xq;
MSE = mean(e.^2);
```

%% Ispis rezultata

```
fprintf('Broj bita: %d\n', nbits);
fprintf('Broj nivoa: %d\n', L);
fprintf('Korak kvantizacije delta = %f\n', delta);
fprintf('Srednja kvadratna greška (MSE) = %e\n', MSE);
```

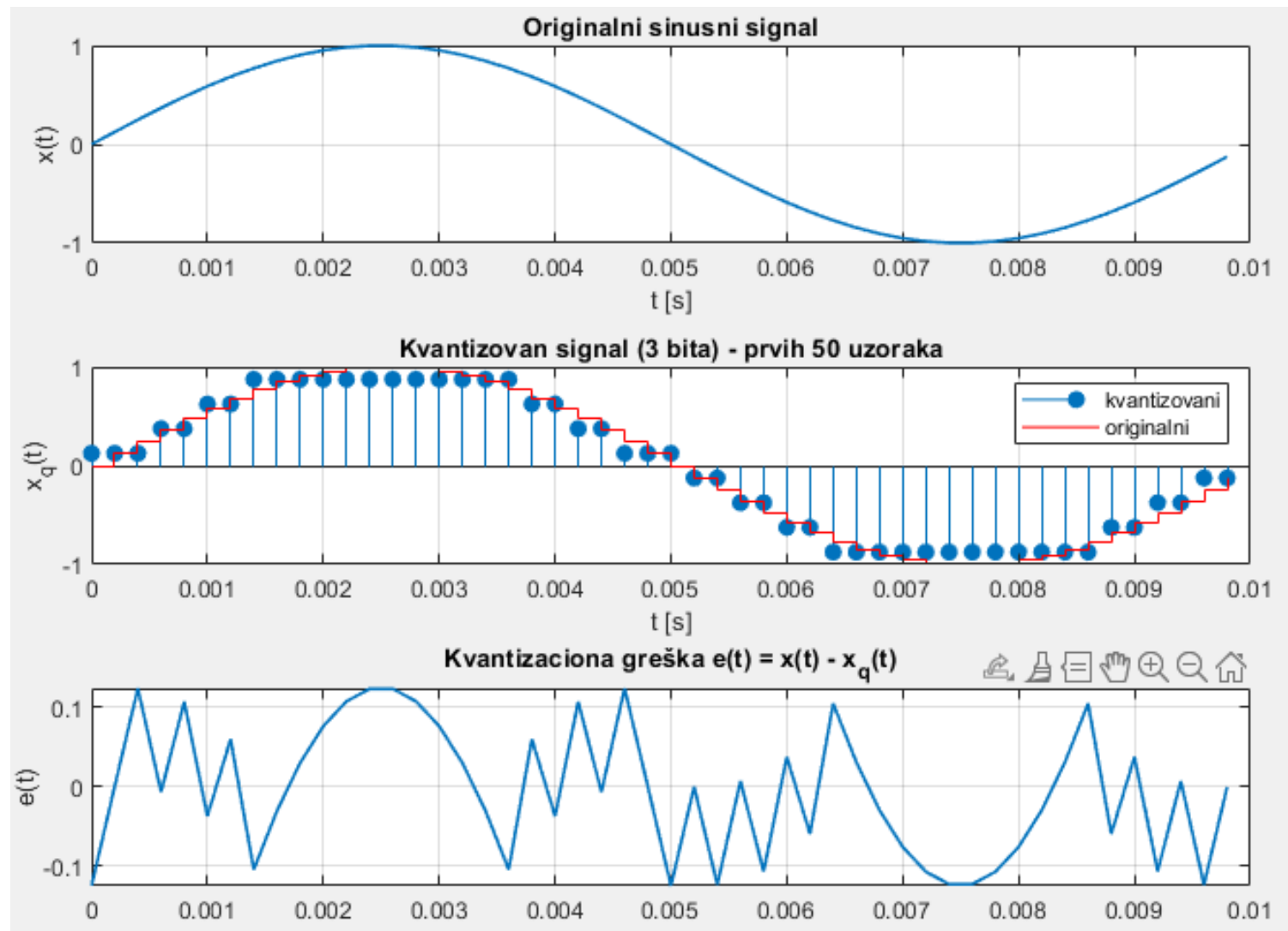
%% Crtanje rezultata

```
figure;
subplot(3,1,1);
plot(t, x, 'LineWidth', 1.2);
grid on;
title('Originalni sinusni signal');
xlabel('t [s]');
ylabel('x(t)');

subplot(3,1,2);
stem(t(1:50), xq(1:50), 'filled'); % prikazujemo prvih 50 uzoraka radi
preglednosti
hold on;
stairs(t(1:50), x(1:50), 'r');
grid on;
title(sprintf('Kvantizovan signal (%d bita) - prvih 50 uzoraka', nbits));
xlabel('t [s]');
ylabel('x_q(t)');
legend('kvantizovani', 'originalni');

subplot(3,1,3);
plot(t, e, 'LineWidth', 1.2);
grid on;
title('Kvantizaciona greška e(t) = x(t) - x_q(t)');
xlabel('t [s]');
ylabel('e(t)');
```

Rezultati



Kvantizacija audio signala

```
%% 1) Generisanje "audio" signala
```

```
% Parametri "audio" signala
```

```
Fs = 44100; % frekvencija uzorkovanja [Hz] (standard za audio)
```

```
T = 3; % trajanje signala [s]
```

```
t = (0:1/Fs:T-1/Fs)';
```

```
% Napravi signal kao kombinaciju nekoliko tonova (kao jednostavna muzika)
```

```
f1 = 440; % ton A4 (koncertni ton)
```

```
f2 = 660; % E5
```

```
f3 = 880; % A5
```

```
xa = 0.6*sin(2*pi*f1*t) + ... % osnovni ton
```

```
0.3*sin(2*pi*f2*t) + ... % viši harmonik
```

```
0.1*sin(2*pi*f3*t); % još viši harmonik
```

```
% Opcioni dodatak: malo šuma
```

```
% noise_level = 0.01;
```

```
% xa = xa + noise_level*randn(size(xa));
```

```
% Normalizacija na opseg [-1, 1] da liči na tipičan audio signal
```

```
max_val = max(abs(xa));
```

```
if max_val > 0
```

```
    xa = xa / max_val;
```

```
end
```

```
%% 2) Parametri kvantizacije
```

```
nbits = 8; % broj bita kvantizacije (probaj 4, 8, 12, 16...)
```

```
L = 2^nbits; % broj kvantizacionih nivoa
```

```
xmin = -1; % donja granica opsega
```

```
xmax = 1; % gornja granica opsega
```

```
% Korak kvantizacije
```

```
delta = (xmax - xmin) / L;
```

```
% Nivoi kvantizacije (sredine intervala)
```

```
q_levels = xmin + delta/2 : delta : xmax - delta/2; % 1 x L vektor
```

```
%% 3) Kvantizacija (uniformni mid-tread kvantizator)
```

```
% Ograničavanje signala na [xmin, xmax]
```

```
x_clipped = min(max(xa, xmin), xmax);
```

```
% Izračunavanje indeksa kvantizacionih nivoa
```

```
q_index = floor((x_clipped - xmin) / delta); % -> 0 .. L-1
```

```
q_index(q_index == L) = L - 1; % zaštita za gornju granicu
```

```
% Rekonstrukcija kvantizovanog signala preko nivoa kvantizacije
```

```
xq = q_levels(q_index + 1).'; % +1 zbog 1-based indeksiranja u MATLAB-u
```

```
%% 4) Kvantizaciona greška i SNR
```

```
e = xa - xq;
```

```
MSE = mean(e.^2);
```

```
P_signal = mean(xa.^2);
```

```
SNR = 10*log10(P_signal / MSE);
```

```
fprintf('--- Rezultati kvantizacije ---\n');
```

```
fprintf('Broj bita: %d\n', nbits);
```

```
fprintf('Broj nivoa: %d\n', L);
```

```
fprintf('Korak kvantizacije delta = %g\n', delta);
```

```
fprintf('MSE = %e\n', MSE);
```

```
fprintf('SNR = %.2f dB\n', SNR);
```

Kvantizacija audio signala

%% 5) Čuvanje originalnog i kvantizovanog signala kao .wav

```
audiowrite(sprintf('audio_original_%dHz.wav', Fs), xa, Fs);
audiowrite(sprintf('audio_kvant_%dbit_%dHz.wav', nbits, Fs), xq, Fs);
```

%% 6) Poređenje originalnog i kvantizovanog signala (talasni oblik)

```
Nplot = min(length(xa), 5000); % prikaži prvih 5000 uzoraka radi preglednosti
t_plot = (0:Nplot-1)/Fs;
```

```
figure;
subplot(3,1,1);
plot(t_plot, xa(1:Nplot));
grid on;
title('Originalni "audio" signal (deo)');
xlabel('t [s]');
ylabel('x(t)');
```

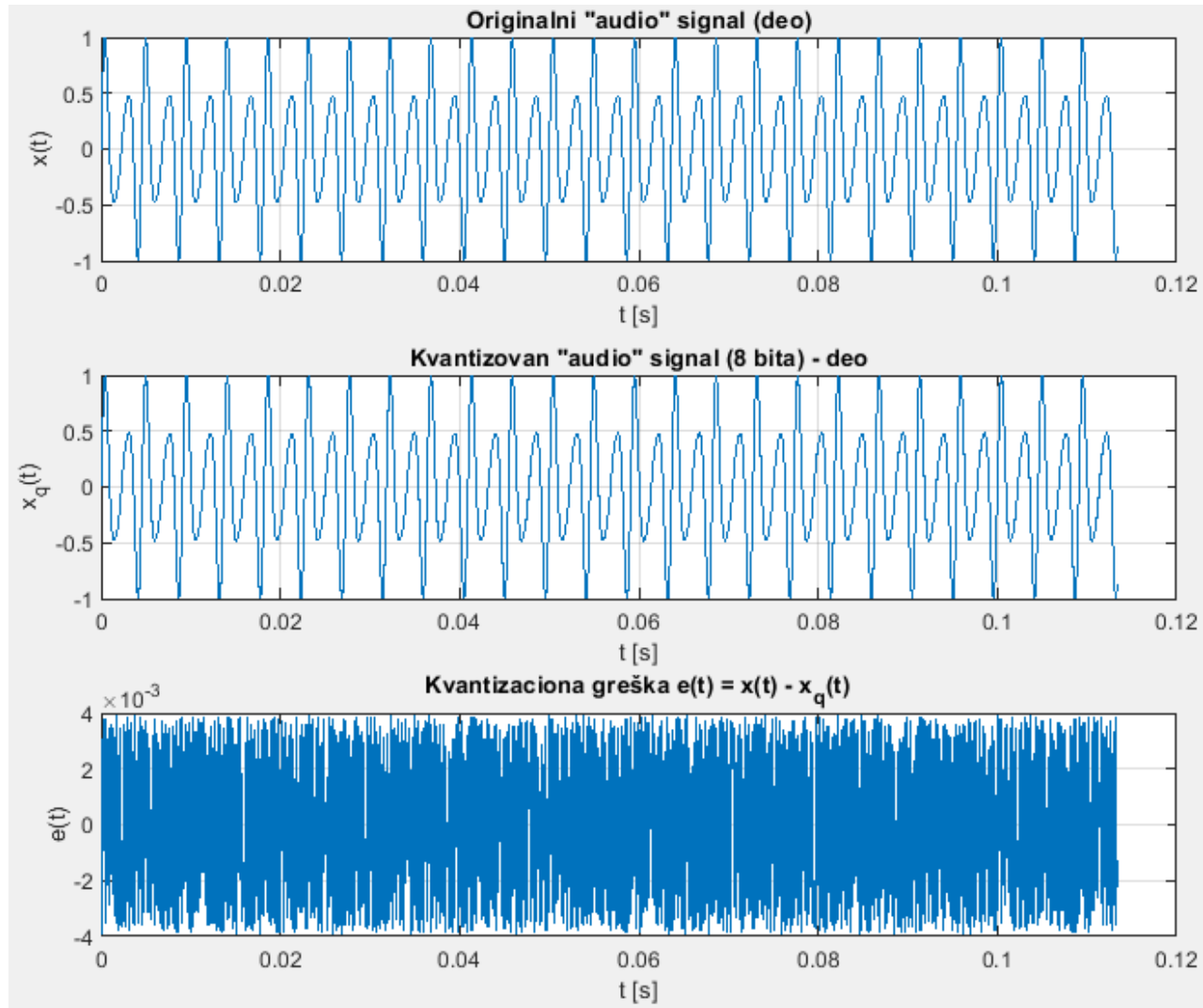
```
subplot(3,1,2);
stairs(t_plot, xq(1:Nplot));
grid on;
title(sprintf('Kvantizovan "audio" signal (%d bita) - deo', nbits));
xlabel('t [s]');
ylabel('x_q(t)');
```

```
subplot(3,1,3);
plot(t_plot, e(1:Nplot));
grid on;
title('Kvantizaciona greška e(t) = x(t) - x_q(t)');
xlabel('t [s]');
ylabel('e(t)');
```

%% 7) Reprodukција zvuka (opciono)

```
% Otkomenariši ako želiš da čuješ:
% sound(xa, Fs); % original
% pause(T + 1);
% sound(xq, Fs); % kvantizovan
```

Rezultati



Hvala na pažnji!

PITANJA?

