

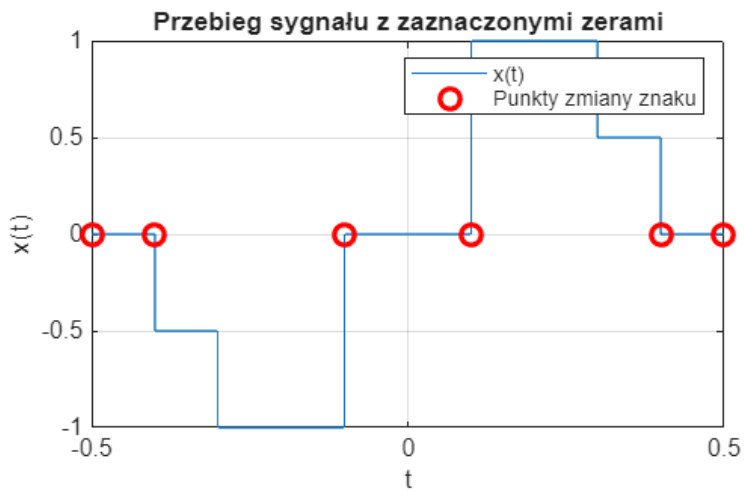
SPRAWOZDANIE 4 (cz. II)

Zajęcia nr 5 - Analiza harmoniczna (cz. II) - Michał Midor, gr. 5 - 29.10.2025, godz: 16:45

Jest to druga część sprawozdania, tym razem ponownie usuwam zbędny komentarz do zadań. Poniżej znajduje się kod podobny do tego w części I tego sprawozdania, natomiast zmieniłem definicję sygnału na modyfikowaną sinusoidę.

1)

```
clear all; close all;
syms t t1 t2 x
T0 = 1.0; % Okres
t1 = -0.5;
t2 = t1+T0; % t2 = 0.5
f0 = 1/T0; % Czystotliwosc
w0 = 2*pi*f0; % Pulsacja
BND = [t1,t2];
x = piecewise( ...
    (t >= -0.5) & (t < -0.4), 0, ...
    (t >= -0.4) & (t < -0.3), -0.5, ...
    (t >= -0.3) & (t < -0.1), -1.0, ...
    (t >= -0.1) & (t < 0.1), 0, ...
    (t >= 0.1) & (t < 0.3), 1.0, ...
    (t >= 0.3) & (t < 0.4), 0.5, ...
    (t >= 0.4) & (t <= 0.5), 0 ...
);
t_zero_in_boundaries = [-0.5, -0.4, -0.1, 0.1, 0.4, 0.5];
x_zero_value = [0, 0, 0, 0, 0, 0];
figure;
fplot(x, BND);
grid on;
ylabel('x(t)')
xlabel('t')
title('Przebieg sygnału z zaznaczonymi zerami');
hold on;
plot(t_zero_in_boundaries, x_zero_value, 'ro', 'MarkerSize', 8, 'LineWidth', 2);
legend('x(t)', 'Punkty zmiany znaku', 'Location', 'NorthEast');
hold off;
```

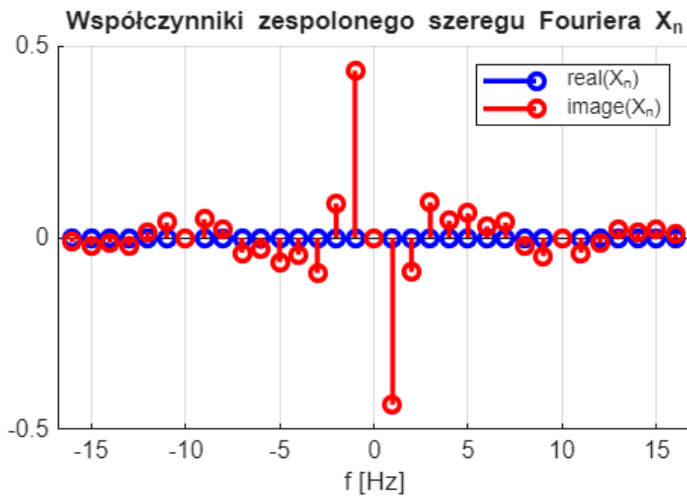


2)

```
syms n
NT = 16;
X=[];
ind = -NT : NT;
for n = ind
    Xn = (1/T0)*int(x*exp(-1i*w0*n*t),t,BND); % bezpieczniej użyć -1i zamiast -i
    X(n + NT + 1) = Xn;
end
X_numeric = double(X)
```

```
X_numeric = 1x33 complex
    0.0000 - 0.01111i    0.0000 - 0.02121i    0.0000 - 0.01271i    0.0000 - 0.02131i ...
```

```
figure; hold on;
stem(ind*f0,real(X_numeric),'b','LineWidth',2);
xlabel('f [Hz]')
stem(ind*f0,imag(X_numeric),'r','LineWidth',2);
grid on
legend('real(X_n)','image(X_n)','Location','NorthEast')
title('Współczynniki zespolonego szeregu Fouriera X_n')
hold off
```



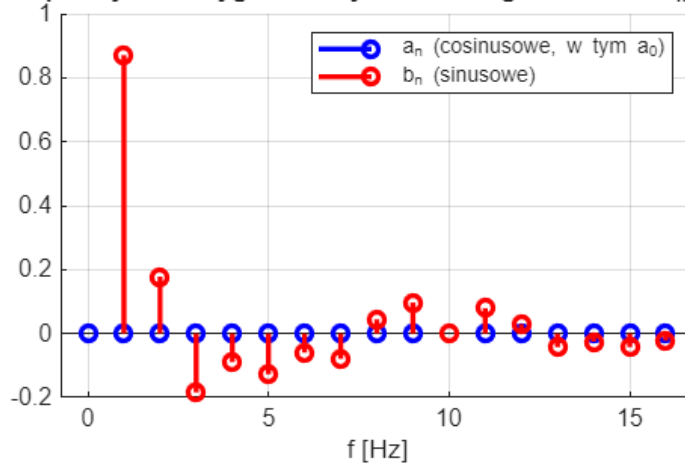
3)

```
n_ind = 1:NT;
a0 = (1/T0)*int(x, t, BND);
a = zeros(1, NT + 1);
b = zeros(1, NT + 1);
a(1) = a0;
for k = n_ind
    n = k;

    an = (2/T0) * int(x * cos(n * w0 * t), t, BND);
    a(k + 1) = an;

    bn = (2/T0) * int(x * sin(n * w0 * t), t, BND);
    b(k + 1) = bn;
end
a_numeric = double(a);
b_numeric = double(b);
figure;
hold on;
f_ind = (0:NT)*f0;
stem(f_ind, a_numeric, 'b', 'LineWidth', 2);
stem(f_ind(2:end), b_numeric(2:end), 'r', 'LineWidth', 2);
xlabel('f [Hz]');
title('Współczynniki trygonometryczne szeregu Fouriera a_n i b_n');
legend('a_n (cosinusowe, w tym a_0)', 'b_n (sinusowe)', 'Location', 'NorthEast');
grid on;
hold off;
```

Współczynniki trygonometryczne szeregu Fouriera a_n i b_n



4)

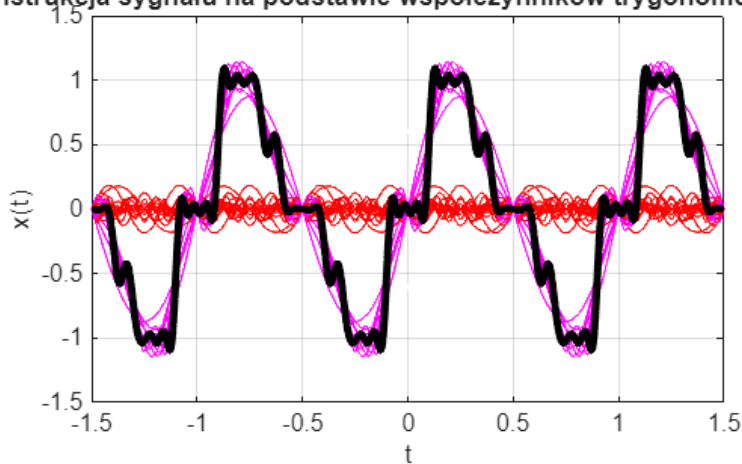
```

step = (BND(2) - BND(1))/1000;
tt = [BND(1)-T0 : step: BND(2) + T0];
xx = zeros(1,length(tt));
xx = xx + a(1); % składowa stała

figure
plot(tt,xx,'m'); grid on, hold on;
plot([0,0],[-0.6,0.6],'w.')
xlabel('t'); ylabel('x(t)');
pause(0.5)

for n = 1 : NT
    xx_n = (a_numeric(n+1)*cos(w0*n*tt) + b_numeric(n+1)*sin(w0*n*tt));
    xx = xx + xx_n;
    plot(tt,xx_n,'r'); plot(tt,xx,'m');
    title(sprintf('n = %d',n+1)); pause(0.5)
end
plot(tt,xx,'k','LineWidth',3);
title('Rekonstrukcja sygnału na podstawie współczynników trygonometrycznych');
    
```

instrukcja sygnału na podstawie współczynników trygonometrycznych



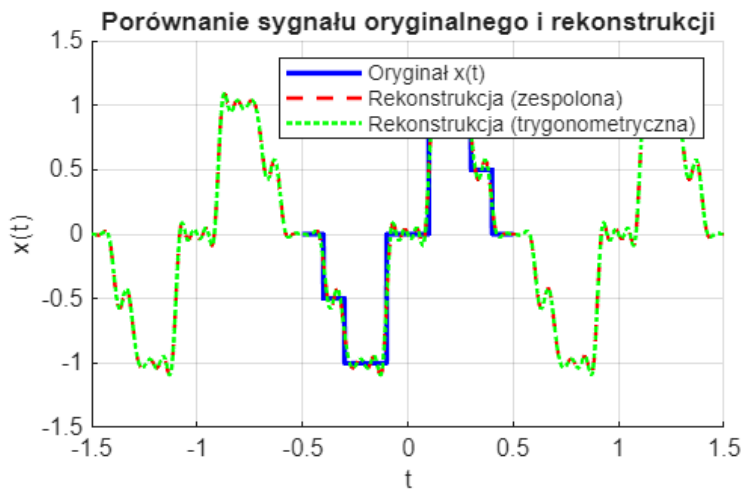
5)

```
xx_zespolony = zeros(1, length(tt));
for n_idx = 1:length(ind)
    n = ind(n_idx);
    Xn = X_numeric(n_idx);

    harmoniczna_n = Xn * exp(1i*n*w0*tt);
    xx_zespolony = xx_zespolony + harmoniczna_n;
end

xx_zespolony_real = real(xx_zespolony);
x_original_plot = double(subs(x, t, tt));

figure;
hold on;
plot(tt, x_original_plot, 'b', 'LineWidth', 2); % Oryginał
plot(tt, xx_zespolony_real, 'r--', 'LineWidth', 1.5); % Nowy (zespolony)
plot(tt, xx, 'g:', 'LineWidth', 1.5); % Stary (trygonometryczny)
grid on;
xlabel('t');
ylabel('x(t)');
title('Porównanie sygnału oryginalnego i rekonstrukcji');
legend('Oryginał x(t)', 'Rekonstrukcja (zespolona)', 'Rekonstrukcja (trygonometryczna)');
hold off;
```



```
% W tym przypadku sygnał jest określony jedynie w jednym okresie
idx_central_period = (tt >= BND(1)) & (tt <= BND(2));
x_original_central = x_original_plot(idx_central_period);
xx_zespolony_central = xx_zespolony_real(idx_central_period);
blad_sygnal_central = x_original_central - xx_zespolony_central;
mse_error = mean(blad_sygnal_central.^2);
fprintf('Błąd średniokwadratowy (MSE) dla NT=%d (w centralnym okresie) wynosi:
%e\n', NT, mse_error);
```

Błąd średniokwadratowy (MSE) dla NT=16 (w centralnym okresie) wynosi: 9.402490e-03

6)

```
syms A w0 t;
T0_sin = 2*pi/w0;

signal_squared = (A*sin(w0*t)).^2;
mean_signal_squared = (1/T0_sin)*int(signal_squared, t, 0, T0_sin);
rms_signal_symbolic = sqrt(mean_signal_squared)
```

rms_signal_symbolic =

$$\sqrt{\frac{A^2}{2}}$$

```
skd = double(subs(rms_signal_symbolic, A, 1.0)) % Wynik jest poprawny.
```

```
skd =
0.7071
```

Wartość skuteczna sygnału x(t).

```
x_squared = x.^2;
mean_x_squared = (1/T0) * int(x_squared, t, BND(1), BND(2));
rms_x_symbolic = sqrt(mean_x_squared);
```

```
x_rms = double(rms_x_symbolic);
fprintf('Wartość skuteczna sygnału x(t) wynosi: %f\n', x_rms);
```

Wartość skuteczna sygnału $x(t)$ wynosi: 0.670820

Wartość skuteczna sygnału zrekonstruowanego (z ćw. 4).

```
% Moc składowej stałej (n=0)
P_0 = a_numeric(1)^2;

% Moce harmonicznych (n=1 do NT)
P_n = sum( (a_numeric(2:end).^2 + b_numeric(2:end).^2) / 2 );

P_c = P_n + P_0;

rms_xx = sqrt(P_c);
fprintf('Wartość skuteczna sygnału zrekonstruowanego (z %d harm.) wynosi: %f\n',
NT, rms_xx);
```

Wartość skuteczna sygnału zrekonstruowanego (z 16 harm.) wynosi: 0.663780

7)

```
P_podst = a_numeric(2)^2 + b_numeric(2)^2;
W_skut_podst = sqrt(P_podst)
```

```
W_skut_podst =
0.8709
```

```
N_values = [5, 10, 15];
THD_results = [];

for NT = N_values
    P_harmonic = sum(a_numeric(3:NT+1).^2 + b_numeric(3:NT+1).^2);
    W_skut_harmonic = sqrt(P_harmonic);

    THD = W_skut_harmonic/W_skut_podst;
    THD_results(end+1) = THD;

    fprintf('Dla NT = %d, THD = %f (czyli %.2f%%)\n', NT, THD, THD*100);
end
```

```
Dla NT = 5, THD = 0.343929 (czyli 34.39%)
Dla NT = 10, THD = 0.382226 (czyli 38.22%)
Dla NT = 15, THD = 0.401410 (czyli 40.14%)
```

Widać, że zmodyfikowana sinusoida zawiera znacznie większy udział harmonicznych w energii sygnału, niż czysta funkcja s