

```

%% 周期信号的傅里叶级数分析与合成
%
% 2019_07_02
%可以改进的地方:
% 显示多个周期, 或者说t可以是任何范围。 2019_07_03 搞定
%
% t1 = -3:1/100:5;
%
% t2 = mod(t1+T/2, T) - T/2; 把t1映射
到 [-T/2, T/2] 区间。

% 方便更改参数T1, T
% 合成波形显示更友好
% 显示误差大小和波形, 当均方误差小于1%时停止迭代, 同时输出均方误差值, 谐波
数和y。
%改成 函数
%改成GUI

%% 三角波合成
% step1: 时域三角波
tri_wav = @(t, T1, T) ( (1/T1 * (t+T1)) .* ((t>-T1) & (t<=0)) ) + ...
(( -1/T1*(t-T1)) .* ((t> 0) & (t<=T1)) );

%三角波函数

T1 =1 , T=3;

t = -5:1/100:5;
t1 = mod(t+T/2, T)-T/2;

x = tri_wav(t1, T1, T);

figure
plot(t, x, 'linewidth', 2)
axis([-5.5 5.5 -0.2 1.2])

%axis ([-1.5 1.5 -0.5 1.5])
%axis([-3 , 5 - 0.5 1.5])
axis([-3 , 5 , -0.5, 1.5] )

%step2: fourier级数 合成
a0 = T1/T;

Sa =@(x) sin(x)./x ;

a_k = @(k, T1 , T) T1/T*Sa(k*2*pi/T*T1/2).^2; %三角波傅里叶系数公式

M = 10 %用M个谐波叠加合成

ak = a_k(1:M , T1, T);

```

```

abs_ak = abs(ak);
theta_ak = angle(ak);

%开始合成!
y = a0*ones(size(t));

hold on
for k = 1 : M
    y = y + 2*abs_ak(k)*cos(k*2*pi/T*t +theta_ak(k));

    plot(t,y),title(int2str(k))
    pause

end

```