Laboratory 2A:

Fourier Series 3D-Plot

In this lab you will experiment to synthesize periodic signals using the definition of the Fourier Series with real coefficients. You will write a Matlab script which generates helpful plots for the visualisation of the single harmonics and their sum.

Step-1

(a) Open the template script *Fourier_series_3D_graph_step_1.m*
Look for the positions indicated with the label "WORK HERE!!!", and fill out the required commands.

(b) The Fourier coefficients a_k, b_k, should be defined so, that you obtain an odd periodic function x(t). Which of the coefficients (a_k or b_k) need to be different from zero?
Define these coefficients to be inversely proportional to k.

(c) Add the required plot command, and check which form has the generated periodic function (expecting a sawtooth). Vary the number of harmonics and observe the effect. Decrease the pause intervals (in seconds) if you wish.

(d) Clean up your script, by removing the unneeded comments and proceed to the next step.

Step-2

Change now the definition of the b_k coefficients, in order to have only non-zero values for odd values of k. Hints:

- you will need an IF statement (use help to check the syntax);
- you can use the modulo operator to check the parity of k (check the syntax with help).

Step-3

(a) We want now to save and visualize the single harmonics being summed to x(t). Define for this purpose a matrix M(t) , with dimensions M versus length(t).
Like x_t, the matrix m_t should be defined and initialised before the usage within the loop.

(b) Add within the loop an statement allowing to calculate each harmonic and store it in the corresponding row of M(t).
Use this row of M(t) to update the x(t) value (such that you avoid to calculate twice the harmonic component).

Step-4

(a) Insert the following lines before the loop to open the figure in a fixed position (left upper corner) and with an appropriated dimension for our purpose (long wise).

```
% Open figure with specific
% Position Property: [left bottom width height]
scrsz = get(groot,'ScreenSize');
figure('Position',[1 0.6*scrsz(4) 0.8*scrsz(3) 0.3*scrsz(4)])
```

(b) Split the figure you generate within the loop into two subwindows using the *subplot()* command. Let the previous plot from x(t) in the right side (*subplot(122)*).

(c) On the left side let us add a 3D plot using the command *plot3()*, allowing to visualise the single rows of M(t). Check the syntax of plot3 with the help command.
For each line you want to plot in this 3D space, you need to provide corresponding X,Y,Z coordinates.
You can use the time vector t for X-coordinates and the single rows of M(t) for the Z-coordinates.
For the Y-coordinates you need a vector with the same length of t, but having constant values equal to the index k.

(d) Do not forget to add a *hold on* command, such that the single harmonics already plot, stay visible. You can also add grid lines (grid on), and check if a rotation can improve the visualisation (use the "Rotate 3D" button in figure menu, and once you find the desired values, take these in your script with the command *view()*. For example `view(65,50)` .

(e) Add appropriated labels and titles. The expected final figure is shown below.