

DOING PHYSICS WITH MATLAB

BEATS

Ian Cooper

matlabvisualphysics@gmail.com

MATLAB SCRIPTS

<https://github.com/D-Arora/Doing-Physics-With-Matlab/tree/master/mpScripts>

<https://drive.google.com/drive/u/3/folders/1j09aAhfrVYpiMavaJrgSvUMc89ksF9Jb>

beats.m

The Matlab script is for a simulation of beats. A **GUI** (Graphical User Interface) is used to input the second frequency f_2 while the first frequency f_1 is fixed at 1000 Hz. A graphical output is displayed showing the beat pattern and a **sound** can be played so that you can hear the beats (figure 1).

beats.Calculations.m

Graphical output of the beat pattern for two superposed sinusoidal signals (figure 2).

wav_SoundRecordings.m

How a sound can be generated in Matlab and the sound signal saved as a **wav** file.

Notes on BEATS

BEATS

Beats are heard when two sounds with slightly different frequencies f_1 and f_2 are sounded together. It is due to the interference between the two waves. Suppose that at a certain position the two waves are given by

$$(1) \quad y_1(t) = A \sin(2\pi f_1 t) \quad \text{and} \quad y_2(t) = A \sin(2\pi f_2 t)$$

The resultant wave can be expressed as

$$(2) \quad y(t) = y_1(t) + y_2(t) = 2A \sin\left(2\pi\left(\frac{f_1 - f_2}{2}\right)t\right) \cos\left(2\pi\left(\frac{f_1 + f_2}{2}\right)t\right)$$

The cosine term varies with the average frequency of $(f_1 + f_2)/2$. The factor in front of the cosine term gives the *envelope* (or amplitude factor) which varies slowly with a frequency of $|f_1 - f_2|/2$. The ear responds to the intensity of the wave which varies as the square of the amplitude factor and goes through two maxima or minima per cycle, giving a *beat frequency* of

$$(3) \quad f_{beat} = |f_1 - f_2|$$

Sample Results `beats.m`

The resultant wave for frequencies $f_1 = 1000$ Hz and $f_2 = 1100$ Hz are displayed in figure 1. A beat frequency of 100 Hz can be heard when the SOUND button is pressed.

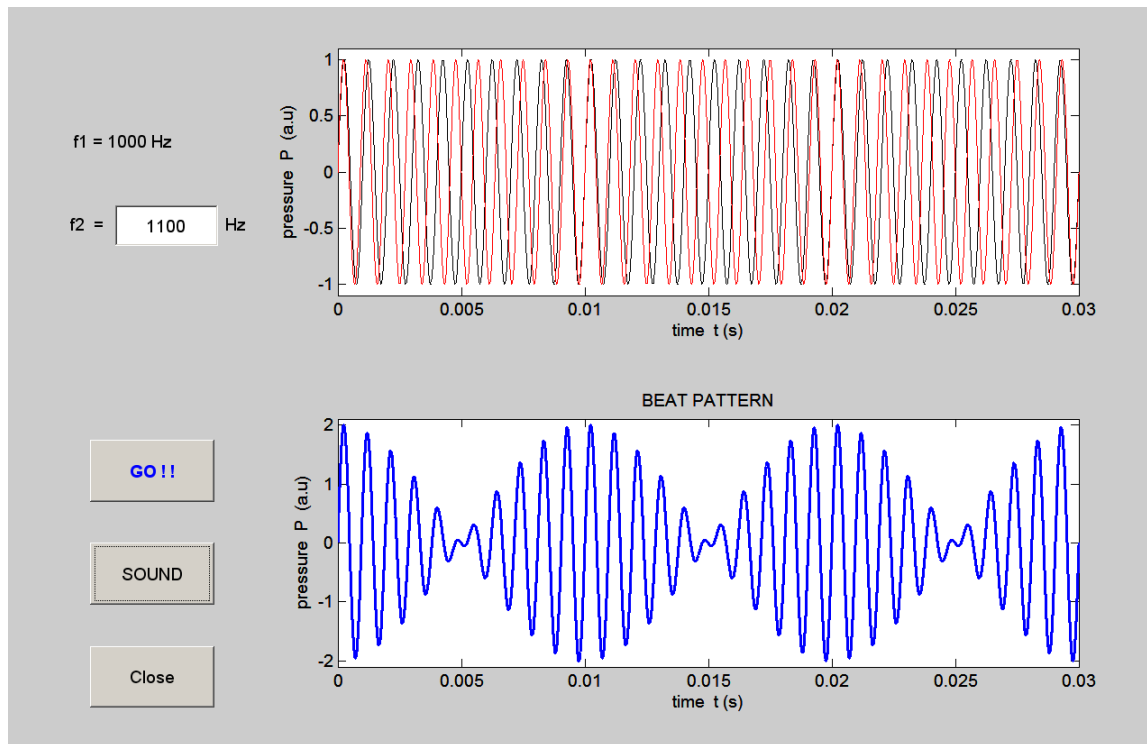


Fig. 1. A screen dump of for the GUI interface and the plot showing the interference of the two waves that produce the beat pattern. The beat frequency is 100 Hz and the beat period is 0.01 s. `beats.m`

Investigations and Questions

Inspect and *run* the m-script **beats.m** so that you are familiar with what the program and the code does. For a range of input parameters, view the plots and listen to the sounds. How does a plot relate to the sound?

- 1 Start with the two frequencies set at 1000 Hz. Increase the input frequency above 1000 Hz. Decrease the input frequency to values less than 1000 Hz. Observe the changes in the plots and the sounds.
- 2 Set the input frequency to $f_2 = 1100$ Hz. Use the Data Cursor to measure the period of the rapid fluctuations and the period of the envelope. From the period measurements, calculate the frequencies of the rapidly varying fluctuations, the envelope and the beats. How well do your results agree with the theoretical results?

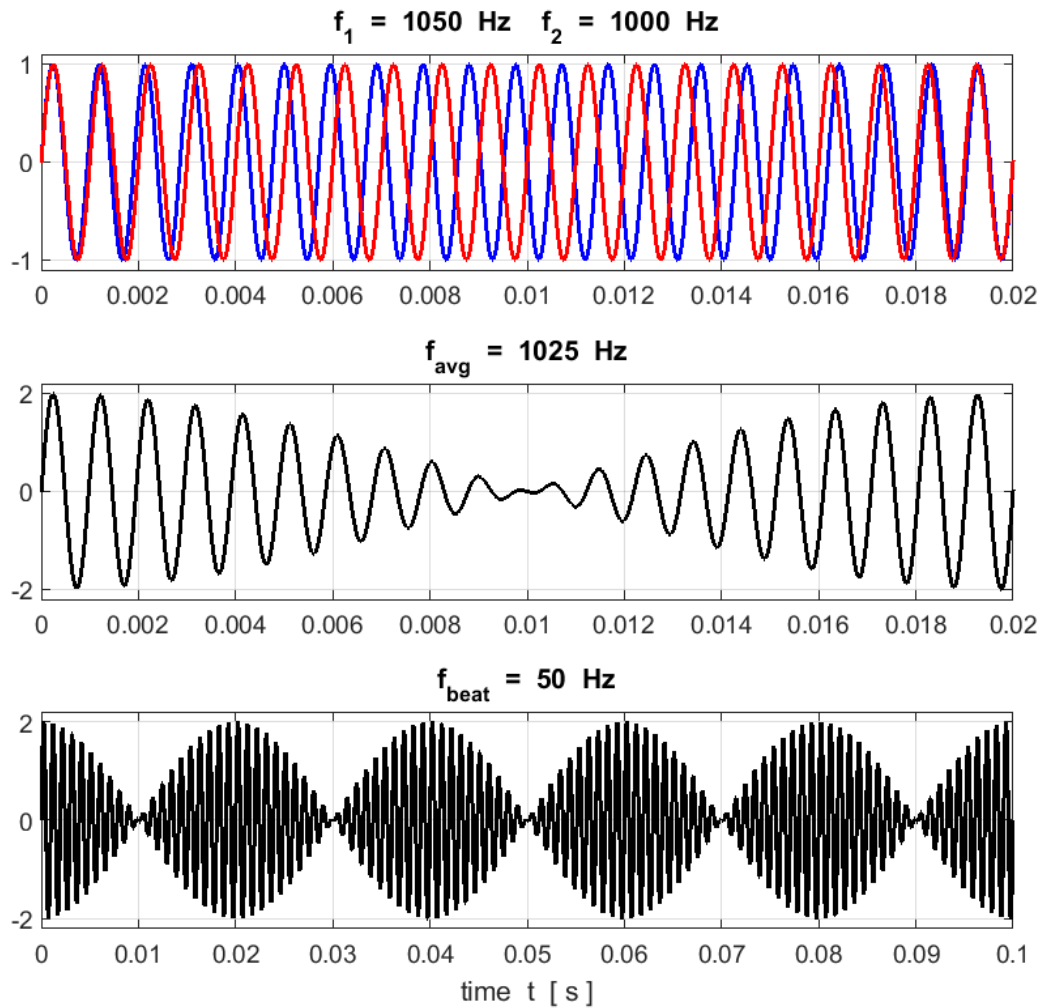


Fig. 2. Graphical output of the beat pattern for two superposed sinusoidal signals. [beats_Calculations.m](#)

SOUNDS IN MATLAB

The mscript **wav_SoundRecording.m** can be used to generate a sound and save as a wav file.

```
% wav_SoundRecording.m
% Generate and save sound files for two
frequency inputs

% Ian Cooper
% email: matlabvisualphysics@gmail.com
% 170511
% Ignore Warning about clipping

clear; close all; clc;

% Frequency inputs
    f1 = 3000;
    f2 = 3003;

% Calculate Waveform
    fs = 22050;           % sample frequency (Hz)
    d = 4.0;             % duration (s)
    n = fs * d;         % number of samples
    t = (1:n) / fs;     % sound data preparation
    % s = sin(2 * pi * f1 * s); % pure tone
    s = sin(2*pi*f1*t) + sin(2*pi*f2*t);
    s = s./max(s);

sound(s, fs);           % Generate sound
pause(d + 0.5);        % waiting for sound end

% Save wav file to disk
    filename = 'wav_S3000-3003.wav';
    audiowrite(filename,s,fs);
```