GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL of ELECTRICAL and COMPUTER ENGINEERING

ECE 2026 Fall 2013

Lab #5: Synthesis of Sinusoidal Signals—A Music Illusion

Date: 30 September-3 October, 2012

Each Lab assignment in ECE2026 consists of three parts: Pre-Lab, In-lab Tasks, and Take-home Questions. It requires you to come into lab prepared. Be sure to read the entire lab carefully before arriving.

Pre-Lab: You should read the Pre-Lab section of the lab and go over all exercises in this section before going to your assigned lab session. Although you do not need to turn in results from the Pre-Lab, doing the exercises therein will help make your in-lab experience more rewarding and go more smoothly with less frustration and panicking.

In-lab Tasks and Verification: There are a number of designated tasks for each student to accomplish during the lab session. Students are encouraged to read, prepare for and even attempt at these tasks beforehand. These tasks must be completed **during your assigned lab time** and the steps marked *Instructor Verification* must also be signed off **during the lab time**. One of the laboratory instructors must verify the appropriate steps by signing on the **Instructor Verification** line. When you have completed a step that requires verification, simply put a plastic cup on top of your PC and demonstrate the step to one of the TAs or the professor. (You can also use the plastic cups to indicate if you have a more general question, i.e. you can use it to get our attention even if you don't have an Instructor Verification ready.)

Take-home Questions: At the end of each lab sheet below all the verification steps, several questions are to be answered by the student, who can choose to complete the answers while in the lab or after leaving the lab.

The lab sheet with all verification signatures and answers to the questions needs to be turned in to the Lab-grading TA at the beginning of the next lab session.

Lab Report: This is the only lab in the semester that requires you to write a lab report to describe your approach to sound synthesis (Section 4). ***This lab report will be worth 150 points.***

Section 4 defines the task that will lead to the lab report. More information on the lab report format can be found on t-square; go to the lab portal page. Please *label* the axes of your plots and include a title and Figure number for every plot. In order to reduce *orphan plots*, include each plot as figure *embedded* within your report. This can be done easily with MATLAB's `notebook` capability. For more information on how to include figures and plots from MATLAB in your report file, consult your TA for details.

The formal report is due the week of October 21 at the start of your lab. This will give you plenty of time to write a good formal report. You can send your synthesis result to your Lab Report grader electronically. Find out how to use `wavwrite` (recall Lab2) to save a wave file from MATLAB in Section 4.1.

1. Introduction

This lab includes a project on sound synthesis with sinusoids. The project requires an extensive programming effort and should be documented with a complete **formal** lab report[1]. A good report should include the following items: a cover sheet, **commented** MATLAB code, explanations of your approach, conclusions and any additional tweaks that you implemented for the synthesis. Since the project must be evaluated by listening to the quality of the synthesized sounds, the criteria for judging a good result are given at the end of this lab description.

The sound synthesis will be done with sinusoidal waveforms of the form

$$x(t) = \sum_k A_k \cos(\omega_k t + \varphi_k) \qquad (1)$$

[1]Refer to the ECE-2026 t-square page for more details on the required format.

where the amplitudes will be manipulated to produce a musical illusion. The challenge of the lab is to adjust the amplitudes to maximize the subjective illusive effect for listening.

2. Pre-Lab

In this lab, the periodic waveforms and music signals will be created with the intention of playing them out through a speaker or a headphone. Therefore, it is necessary to take into account the fact that a conversion is needed from the digital samples, which are numbers stored in the computer memory to the actual voltage waveform that will be amplified and heard through the speakers/headphones/earphones.

2.1 Piano Keyboard

Section 4 of this lab will consist of synthesizing notes in one octave of a musical scale[2]. Since these signals require sinusoidal tones to represent piano notes, a quick introduction to the layout of the piano keyboard is needed. (See also Lab 4. If you remember Lab 4, you can skip this sub-section 2.1.) On a piano, the keyboard is divided into octaves—the notes in one octave being twice the frequency of the notes in the next lower octave. The white keys in each octave are named A through G. In order to define the frequencies of all the keys, one key must be designated as the reference. Usually, the reference note is the A above middle-C, called A-440 (or A4) because its frequency is 440 Hz. (In this lab, we are using the number 40 to represent middle-C. The numbering scheme is somewhat arbitrary; for instance, the Musical Instrument Digital Interface (MIDI) standard represents middle-C with the number 60). Each octave contains 12 notes (5 black keys and 7 white) and the ratio between the frequencies of the notes is constant between successive notes. As a result, this ratio

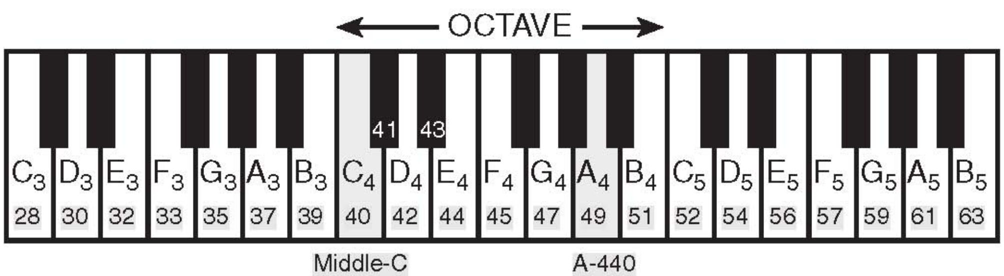

Figure 1: Layout of a piano keyboard. Key numbers are shaded. The notation C4 means the *C*-key in the fourth octave.

must be $2^{1/12}$. Since middle-C is 9 keys below A-440, its frequency is approximately 261 Hz. Consult the text for even more details.

Musical notation shows which notes are to be played and their relative timing (half, quarter, or eighth). Figure 2 shows how the keys on the piano correspond to notes drawn in musical notation. The white keys are labeled as A, B, C, D, E, F, and G; but the black keys are denoted with "sharps" or "flats." A sharp such as $A^{\#}$ is one key number larger than A; a flat is one key lower, e.g., A_4^b (A-flat) is key number 48.

The music system we are now most familiar with in the Western world, as introduced above, is based on the so-called "equal temperament." The Wikipedia has some brief but interesting historical accounts:

> **Equal temperament** is a musical temperament, or a system of tuning in which every pair of adjacent notes has an identical frequency ratio. In equal temperament tunings, an interval — usually the octave — is divided into a series of equal steps (equal frequency ratios between successive notes). For classical music, the most common tuning system is **twelve-tone equal temperament** (also known as **12 equal temperament**), inconsistently abbreviated as **12-TET**, **12TET**, **12tET**, **12tet**, **12-ET**, **12ET**, or **12et**, which divides the octave into 12 parts, which are equal on a logarithmic scale. It is usually tuned relative to a standard pitch of 440 Hz, called A 440. …………………………..

[2]If you have little or no experience reading music, don't be intimidated. Only a little music knowledge is needed to carry out this lab. On the other hand, the experience of working in an application area where you must quickly acquire new knowledge is a valuable one. Many real-world engineering problems have this flavor, especially in signal processing which has such a broad applicability in diverse areas such as geophysics, medicine, radar, speech, etc.

Thus, you can use the ratio $2^{1/12}$ to calculate the frequency of notes anywhere on the piano keyboard. For example, the E-flat above middle-C (black key number 43) is 6 keys below A-440, so its frequency should be $f_{43} = 440 \times 2^{-6/12} = 400\sqrt{2} \approx 311$ Hertz.

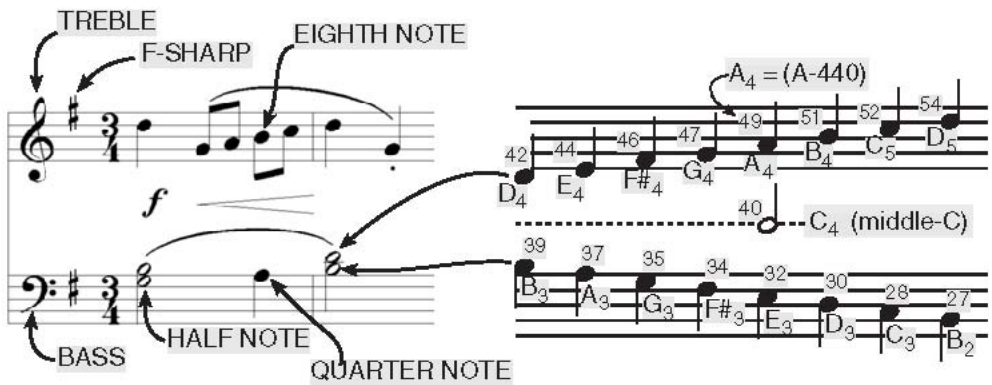

Figure 2: Musical notation is a time-frequency diagram where vertical position indicates which note is to be played. Notice that the shape of the note defines it as a half, quarter or eighth note, which in turn defines the duration of the sound.

2.2 Gaussian Forms

The Gaussian form is used in many different fields, but most often in probability where it is called the Gaussian distribution[3]. In this lab, we will use the Gaussian to control the amplitude weighting of sinusoids. The mathematical form of the Gaussian is

$$g(v) = \alpha\, e^{-(v-\mu)^2/2\sigma^2}$$

where v is the independent variable. The plot of a Gaussian is the well-known "bell curve" where μ determines the peak location and σ controls the width of the bell-shaped peak. The plots in Fig. 3 show the general shape for three values of σ. Notice that the *width of the bell* when measured at the $e^{-1/2} = 0.607$ level is equal to 2σ, which confirms the role of σ as the width-control parameter. In our application we will use $\alpha = 1$, but for a probability distribution α is chosen so that the total area under the Gaussian is equal to one, which leads to the value of α being $\alpha = 1/(\sigma\, 2\pi)$.

Write a few lines of MATLAB code to make a plot of $10e^{-(v-1)^2/2(3)^2}$ over the range $-10 \le v \le 10$.

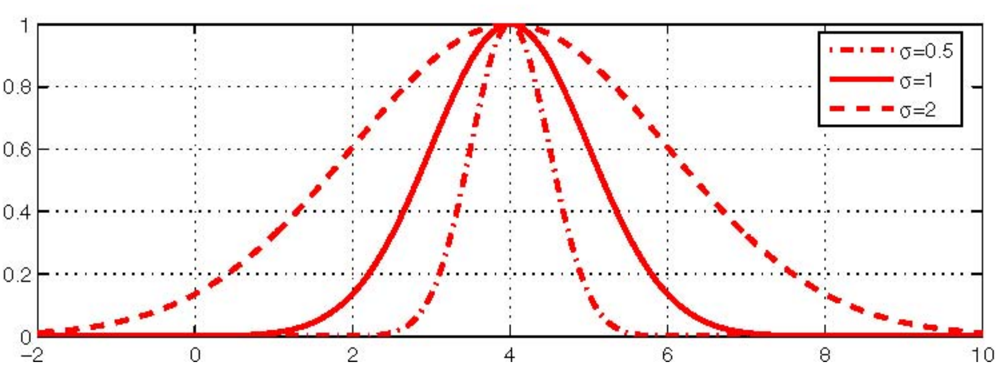

Figure 3: Plot of a Gaussian with parameters $\alpha = 1$, $\mu = 4$, and $\sigma = ½, 1, 2$. When $\sigma = 2$ the 0.607 level extends from $v = 2$ to $v = 6$ giving a width of 4, i.e., the width is defined as 2σ.

[3]Here's a link for more info: **http://mathworld.wolfram.com/NormalDistribution.html**.

3. Exercise

In this exercise you will generate a Gaussian function over a frequency range and use its values as weights to define the amplitude of sinusoids at chosen harmonic frequencies to play a musical note.

3.1 Gaussian Weighting

When used with music synthesis, the Gaussian form is applied as a function of the logarithm of frequency, i.e. $\log_2(f)$. (Our auditory perception responds to frequency approximately in a log scale.) Thus we need to define the following *frequency-weighting* function:

$$W(f) = g(\log_2(f)) = e^{-(\log_2(f)-\log_2(f_c))^2/2\sigma^2}$$

where f_c is the *center frequency* of the Gaussian at which the sinusoid with peak amplitude is located. The parameter σ, which is usually called the *standard deviation* in probability distributions, determines the width (2σ) of the Gaussian curve along the log frequency axis.

(a) Write a short M-file that will synthesize a Gaussian given three inputs: center frequency, half-width (σ), and a vector of frequencies where the Gaussian is to be evaluated.

(b) The MATLAB code `ff=2.^(5:1/12:10)` will generate frequencies that are uniformly spaced on a logarithmic axis.

(c) In order to demonstrate that your M-file works, make a plot of a Gaussian that is centered at 440 Hz and has a half-width equal to one octave. Make the plot over a frequency range of five octaves, i.e., $55 \le f \le 1760$ Hz. In addition, make the horizontal axis logarithmic, i.e., $\log_2(f)$. *Note:* Since the independent variable is $\log_2(f)$, σ controls the width along the $\log_2(f)$ axis. Thus $\sigma = 1$ corresponds to $\log_2(f/f_c) = 1$ which is a factor of two change in frequency and the width (2σ) spans over 2 octaves.

(d) Make two more plots of the Gaussian, but make these plots versus f. The first plot should use MATLAB's **`plot`** command. Explain why the Gaussian appears distorted. Then use MATLAB's **`semilogx`** function to make the same plot. Explain why the Gaussian once again has its expected bell shape.

Instructor Verification (separate page)

3.2 Synthesize Octaves with Gaussian Weighting

Create a note that is the sum of five keys spaced by octaves and weighted by the Gaussian in the previous part. In a previous lab you completed the **`key2note.m`** function which synthesizes the correct sinusoidal signal for a particular key number, so you can use that function to do this task.

(a) Synthesize a signal that is 2.5s long, sampled at f_s = 8000 Hz. Create five sinusoids (notes) and add them together so that they will be played simultaneously. The five notes should be centered at middle-C (also called C4) with the other four notes split on both sides of C4, i.e., C5, C6, on the right (higher octaves) and C2 and C3 on the left (lower octaves). The amplitudes of the five notes should be different; each note's amplitude should be obtained according to a Gaussian function at the note's frequency. The Gaussian function is centered at f = 440 Hz, and has a width (i.e. 2σ), of two octaves (see description in 3.1c).

(b) Show a spectrogram for your synthesized sound, and identify all five octave components. Use a long window length to get good frequency resolution. Check that they are at the correct frequencies, and also justify that they have different amplitudes.

Instructor Verification (separate page)

4. Lab: A Musical Illusion

The objective in this lab project is to reproduce a musical illusion called *Shepard's Scale* in which the tones played seem to be continually rising forever, yet seem to stay within one octave. If you are familiar with M. C. Escher's drawing of a perpetual staircase[4] called "Ascending and Descending," then this audio illusion is analogous. The illusion is referred to as "circularity in pitch judgment" in an article on the web site of the Acoustical Society of America (`http://asa.aip.org/demo27.html`). You can also find a demonstration audio file at the same ASA web site.

The basis of Shepard's scale is to create notes as the sum of many individual sinusoids, all separated by octaves. By cleverly controlling the amplitudes of each sinusoid, the illusion is created by playing the amplitude-weighted notes in a scale (or within an octave) over and over. The use of several octaves allows the weighting to make the illusion. For example, to produce the note A-440 you would add sinusoids at the frequencies of 55Hz, 110Hz, 220 Hz, 440Hz, 880 Hz, 1760 Hz, 3520 Hz, and so on, with the amplitude of the sinusoids being largest at 440 Hz, a little smaller at 220 and 880 Hz, even smaller at 110 and 1760 Hz, and continually smaller for frequencies farther away from 440 Hz. In other words, the *amplitudes are frequency dependent*.

If we followed the guide of the ASA web site, we would use a cosine amplitude weighting. However, we will use a Gaussian instead, because we can get nearly the same amplitude dependence, and we will have the σ parameter available to control the width of the Gaussian. For Shepard's scale, the Gaussian should be centered somewhere between 260 and 500 Hz, and σ should be fixed to get a fixed Gaussian shape from which the amplitude weights for all notes are obtained. Since the notes on a piano are spaced on a log scale, the Gaussian used for amplitude weighting must also be a function of $\log_2(f)$, instead of f.

The following steps should help you produce a working illusion:

(a) First of all, forget about the amplitude weighting. Synthesize a C-major scale, starting at middle-C, in which each note in the scale is accompanied by eight other notes separated by octaves, extending four octaves below and four octaves above. For example, to synthesize A-440, you would create the sum of nine sinusoids at the frequencies 27.5, 55, 110, 220, 440, 880, 1760, 3520 and 7040 Hz. Use a sampling frequency of 22050 Hz, so that you will have f_s greater than twice the highest frequency.

(b) Playing once through a scale requires that you play seven different notes, each note being the sum of nine octave-spaced keys. However, we will need to play the scale over and over, so modify your code to play the C-Major scale five times (35 total notes). Adjust the length of the notes and introduce silence between the notes so that you can easily identify each one.

(c) Now introduce amplitude weighting using the Gaussian form centered somewhere between 260 and 500 Hz. The trick here is that the weighting must be a function of $\log_2(f)$, not f. Make a plot of the weighting function that you are going to use. Make the plot versus $\log_2(f)$ for the case when $\sigma = 2$.

(d) You should now be able to produce the illusion by using the Gaussian weight function to generate the amplitudes of the nine octave-related sinusoids that are added together for each note. Experiment with the σ parameter to get a good sounding illusion.

(e) Make a spectrogram of playing the scale three times and explain the illusion from the features that you can identify in the spectrogram. A spectrogram made by **`spectrogram`** displays a linear amplitude scale, and that should make it easier to explain the illusion.

(f) *Variations:* Play every note within the octave; all twelve of them and repeat. Does the illusion sound better in this case?

(g) *Listening:* Prepare your MATLAB code and include it in your report. Follow Section 4.1 below and submit your audio file to your grading TA via email so as to demonstrate the illusion.

4.1 Save Sound Data in `.wav` File Format (Review previous labs)

[4] The idea of the perpetual staircase is attributed to Roger Penrose and his father Lionel (1958).

You can save a data array in MATLAB into a file in various data formats, one of which is the Microsoft **`wav`** file format. This can be accomplished by using the MATLAB command "**`wavwrite`**." (You may want to review Lab 1 again.)

`WAVWRITE` writes Microsoft WAVE (".wav") sound file. A typical calling sequence is

```
WAVWRITE(Y,WAVEFILE)
```

where the character string **`WAVEFILE, e.g., 'mysong.wav',`** contains the name of the file that you'd like to store the data **`Y`**. In the expression above, a sampling rate of 8000 samples per second is assumed. You can specify other sampling rates if necessary with **`WAVWRITE(Y,FS,WAVEFILE)`** where **`FS`** is the sampling rate. For more details, do

```
help wavwrite
```

Once the file is written, it can be distributed electronically. You can also play the file with your media player.

4.2 Read .wav File into MATLAB Data Array

In Lab 01, you have learned to read a data file in **`wav`** format into MATALB with the following calling sequence

```
Y = WAVREAD(FILE);
```

which transfers the data stored in the file named **`file`** into a MATLAB data array **`Y`**. As in **`wavwrite`**, the filename is a character string. Again, consult **`help wavread`** for more details about the use of the function.

Lab #5
ECE-2026 Fall-2013
LAB SHEET

For each verification, be prepared to explain your answer and respond to other related questions that the lab TA's or professors might ask. Turn this page in together with the report.

Name: ___________________________ Date of Lab:_____________________

Part 3.1 Complete and demonstrate the function that will synthesize a Gaussian form versus $\log_2(f)$.

Verified: __________________________ Date/Time:______________________

Part 3.2 Complete and demonstrate the synthesis of five Gaussian weighted sinusoids. Show a spectrogram and identify the frequencies and amplitudes of the five spectral components to the TA.

Verified: __________________________ Date/Time:______________________

Sound Evaluation Criteria

Does the file play the correct notes?

All Notes______ Most______ Missing octaves______ Wrong Amplitude Weighting______

Overall Impression:

Good: Correct illusion; proper use of Gaussian envelope; all notes synthesized and in sync.

***OK*:** Basic sinusoidal synthesis, but the illusion is not obvious

Poor: Synthesis does not work; incorrect illusion.