For Lab Session: Thursday, February 7, 2013 in Detkin Lab.
Due: Wednesday, February 13, 2013 by 12:00pm.

Collaboration: Work in lab in teams of 2 (we will pair you up). Perform individual writeups. See course collaboration policy in the Administrative Handout.

Objective: Synthesis musical instruments based on an analysis of the sound produced by the actual instrument.

Prelab Requirements: Download and read this entire lab assignment especially the Motivation section and the Theory subsection of the Lab Writeup Guidelines section. Please do the prelab questions in the Theory Section. These will be checked at the beginning of lab. Bring your headphones.

Deliverable:

- Names of all lab group members
- Answers to all lab questions (including prelab question)
- Your modified version of generateNote.vi

Handin: All labs will be turned in electronically through the Penn Blackboard website. Go to the assignment submission link and follow the instructions. Please submit all your files as one zip file. Also, remember that your writeup should be a PDF file.

Exit Ticket: Play for your TA a melody on your synthesizer.
Motivation

Every sound can be created by combining the appropriately weighted pure tones together. Representing a signal in frequency space is a convenient way to analyze sounds and discover what pure tones are contained in it. In this lab we will analyze the recordings of four musical instruments and build a musical synthesizer that tries to recreate the sounds of those instruments.

Lab Procedure

You should dedicate about a third of the time to the first three section and the remaining two thirds to the section labeled Building a Synthesizer.

Exploring the Frequency Domain

You may remember from lab 1, that a pure tone is a sinusoidal signal that has a constant frequency. Using LabView, we can convert the signal from the time domain to the frequency domain and discover the frequency of the signal.

- From the lab website, download `lab4.zip` and unzip it to a local directory.
- Open LabView 10.0 and load `generateWave.vi` from the unzipped files of `lab4.zip`.
- Connect your headphones to the computer and run the VI.
- If no sound plays, right-click on the speaker icon on the taskbar, select playback devices, and switch which speaker is listed as the default. Then restart Labview and open the VI.

On the left side you will find the controls that operate the VI. You can control the frequency, amplitude and phase of the waveform as well as change the type of waveform being produced. On the right, there is a graph showing the waveform in the time domain, above, and in frequency domain, below. The red vertical line on the frequency graph, is a cursor you can slide, by grabbing it with the mouse, to quickly estimate points on the graph. When you first run the VI, you will see a sine wave at 220Hz, you will see a clear spike at 220Hz in the frequency domain graph.

FOR QUESTIONS: Spend some time exploring how the different controls affect the waveforms both in time and frequency.

Analyzing Sound Waves

In lab 1, you observed four sound files in the time domain and gave qualitative descriptions of what they were. Using LabView, we can now analyze these four samples in the frequency domain and can better describe the sound.
• Close `generateWave.vi` and open `measureWave.vi` which allows you to load a `.wav` file and look at it in both time and frequency.

• Included in `lab4.zip` is the four files from lab 1 in `.wav` format instead of `.mp3`. Load the first file `ese250-1.wav` and run the VI.

Along with the time and frequency graphs, you will find the `Harmonic Information` array. It analyzes the signal and reports the frequency, amplitude and phase of the first few most important pure tones composing the sound. You should see it reporting that `ese250-1.wav`’s primary pure tone is at 100Hz with an amplitude of almost 1 and a phase of almost 0. This should be consistent with the black spike on the frequency graph.

Since `ese250-1.wav` is itself a pure tone, the Harmonic Information panel will only report one tone. If the sound were composed of multiple tones, you could use the arrows to the left of the box with a 0 to explore the other tones found by the Harmonic Information panel.\(^1\)

FOR QUESTIONS:

• Load and run the VI for the remaining three wave files, `ese250-2.wav`, `ese250-3.wav` and `ese250-4.wav`.

• For each, step through the different tones reported by the Harmonic Information panel and visually confirm them in the frequency graph.

Building Sounds

From lecture you know that by adding together the correct pure tones, we can build arbitrary sound waves. In this section, we will use LabView to see how pure tones combine.

• Open `combineWave.vi` and run it.

On the top left you will find 30 amplitude sliders and below each, a corresponding phase slider. Each slider allows you to adjust the amplitude and phase of one pure tone. Each tone has a frequency that is a multiple integer of 80Hz. The Individual Waves graph will show any pure tone with an amplitude greater than zero. The result of adding together all the waves shown in the Individual Waves graph will be shown in the Combined Wave graph. Finally to the left you will find 6 controls, one that clears the amplitude and phase sliders, four that generate different types of waves and a noise generator. As you adjust the sliders, you can hear how the sound changes.

\(^1\)The Harmonic Information panel is not perfect. Sometimes it will report three or four tones all very close to the same frequency when in reality there is only one tone at that frequency. To pick out the true tone, compare the amplitude of the cluster and use the one with the greatest amplitude.
Click on Sine, Triangle, Sawtooth and Square buttons. Each of these will adjust the sliders to create the described wave.

Once you’ve seen all four waves, press Clear and spend no more than a minute or two moving the sliders to create your own waves.

FOR QUESTIONS: Create a wave composed of two pure tones by first clearing the wave (press the Clear button) and then changing the magnitude of two pure tones of your choosing. Consider both the shape of the waveform and the sound for the following questions.

Building a Synthesizer

In this last part of the lab we will build a simple musical instrument synthesizer. A musical synthesizer simulates musical instruments by generating a sound wave that approximates the sound of the real instrument. In order to do this you will first use measureWave.vi to analyze a note played by a musical instrument. Once you’ve figured out what pure tones compose the sound, you will modify generateNote.vi so that it generates the same pure tones.

The following steps will walk you through the process of analyzing a note played by a bass, you will have to repeat the process for another instrument, either the alto sax, French horn or clarinet.

- Open measureWave.vi.
- In Path to sound browse for bass.wav and then run the VI.
- The harmonic information panel will report the first five tones ranging from 85Hz to 255Hz. You can use the arrows to the left of the harmonic information panel to explore all five.
- Using the cursor on the Frequency graph, confirm that the frequencies of the tones discovered by the harmonic information panel match the spikes on the graph.

FOR QUESTIONS: Write down the frequency, amplitude and phase for all five tones of the bass recording as reported by the harmonic information panel. Rounding to the nearest half integer, also record what the lowest frequency needs to be multiplied by in order to produce each of the four higher frequencies of the other tones. These are the frequency multipliers of the tones.

Once we have this information, we can modify generateNote.vi to have it generate a wave similar to that of the bass.

- Open generateNote.vi
- Switch to the Block Diagram by pressing Ctrl+E

generateNote.vi takes a reference frequency and instrument type, and generates a note that corresponds to the given frequency, using the given instrument.
There are three main parts in `generateNote.vi`. The most important is the *Case Structure* in the middle. To the left of the Case Structure block is some initialization and to the right there is some cleanup. You will only work inside of the Case Structure. *It is very important that you don’t modify any of the blocks and connections outside the Case Structure*, if you do, the synthesizer may stop working.

When you first open the block diagram, the Case Structure will show the “Example” case. The example instrument produces a note composed of three pure tones generated by the three copies of the `generateSine.vi` subVI. The first tone has a frequency of $1 \cdot f_{reference}$, an amplitude of 180 and a phase of 3. Tone two’s frequency, amplitude and phase are $2 \cdot f_{reference}$, 30 and -27. The final tone is $2.5 \cdot f_{reference}$, 60 and 32. The three tones are combined by the *Combine Waves* block.

Before we begin creating the wave for a bass, here’s a quick review of some of LabView’s tips and tools that will help simplify this process.

- To see what the terminals of a subVI are, open the *Context Help* by pressing Ctrl+H. To see the terminals of a subVI in the Context Help, simply place your mouse over the subVI in question.
- Triple clicking on a wire will highlight all the branches of that wire. This can be helpful when tracing where a wire is going.
- To duplicate a block, select it, hold down Ctrl and drag it with the mouse.
- You can quickly create a constant connected to a terminal by right clicking on the terminal and choosing *Create → Constant*.
- Finally, you can click on *Edit → Clean Up Diagram* or press Ctrl+U to automatically cleanup the layout of blocks and wires.

Make sure you have the information you took down for the questions handy, you will use these to generate the bass note.

- The Case structure has 6 cases: Example, Bass, Alto Sax, French Horn, Clarinet and Custom. You can switch to a case directly by clicking on the down arrow to the right of the case name at the top, or step through the cases by clicking the left and right arrows on the sides of the case name. Switch to the “Bass” case, it should currently be empty.

Our analysis of the bass note revealed that it has 5 primary tones. This means we will need to generate five tones in the “Bass” case.

- Right click inside the case box to open the *Functions* panel.
- Select *Select a VI...*, search for and select `generateSine.vi` and click OK.
- Place the `generateSine.vi` subVI inside the Case Structure.
- Create four more copies of `generateSine.vi` and space them vertically.
There are three orange terminals coming into the Case Structure on the left. You can identify what signal each wire carries, by opening the Context Help (Ctrl+H) and placing your mouse over the wire to the left of the terminals.

- Connect the Sampling Frequency, Frequency and Duration terminals coming into the case structure to the corresponding terminals on each of the 5 generateSine.vi subVIs as shown in Figure 1. Use the Context Help to help you make the correct connections.

![Figure 1: Bass case initial wiring](image)

- For the topmost generateSine.vi subVI, create a constant for the frequency multiplier, amplitude and phase.
- Change the three constants to reflect the values recorded in Question ?? for the first (lowest frequency) tone. Since this is the first tone, its frequency multiplier will be 1.
- For each of the remaining four generateSine.vi subVIs, create constants for and fill in the frequency multiplier, amplitude and phase based on the values you took down.

The last step is to combine the five tones and connect the combination to the output terminal on the right side of the Case Structure.

- From the Functions pallet, select Express → Arithmetic & Comparison → Numeric → Compound Arithmetic.
- Place the Compound Arithmetic block inside the Case Structure.
• Currently it only has two inputs and one output, grabbing one of the blue dot handles, extend the block so that it has five inputs.
• Connect the output of each of the generateSine.vi subVIs to the inputs of the Compound Arithmetic block.
• Connect the output of the Compound Arithmetic block to the output terminal of the Case Structure on the bottom right.

Your finished “Bass” case should look similar to Figure 2.

![Figure 2: Bass case](image)

To test your note, open Synthesizer.vi. This VI has a small keyboard and a time and frequency graphs. When you run the VI, you can choose an instrument, set the duration of the note and play the keyboard. It will show the time and frequency response of the note played.

• Repeat this whole process of measuring the recoding and filling in the case for one of the remaining three instruments, Alto Sax, French Horn or Clarinet. If time permits, you are welcome to model more than one.
  – For some of these instruments, the Harmonic Information panel on the measureWave.vi VI may show several tones at very close frequencies when in reality there is only one tone, for these cases only use the frequency, amplitude and phase of the one having the largest amplitude.
– Musical instruments have frequency multipliers at half or whole integer multiples of the primary frequency. Your analysis may not give exact multipliers, you should round them to the nearest half integer as we did for the bass case.

• The last case, “Custom” case, is left for you to experiment and create a note of your own creation. If time permits, fill in the “Custom” case as you please.
• Using Synthesizer.vi VI, play the melody from the prelab (A2, B2, G2, G1, D2) or a melody of your choosing for your TA using an instrument besides the bass and, if you created it, using the Custom instruments (required for Exit Ticket).

Parting Thought

The prelab question suggests how sheet music is a compressed way of describing a sound wave. As we’ve just seen, instruments produce more complex waveforms than pure tones. However, we’ve also seen that we can build a generative model of the sounds they produce. With this additional model, we can produce the richer sounds of musical instruments while still using the compact description of sound as a sequence of musical notes.
Lab Writeup Guidelines

Theory

Pre-lab Question:

Assume:

- each quarter note plays for 0.5 seconds
- each whole note plays for 2 seconds
- Each line on the staff represents a frequency, there are 96 distinct note frequencies (corresponding to potential vertical positions of notes on staff).

1. What is the duration (in seconds) of the sound represented?
2. Assuming 16-bit samples and 48KHz sampling frequency (similar to Lab 2), how many bytes would it take to produce this sound?
3. Consider, instead, encoding the notes on the staff into a digital representation that the computer can understand. How would you represent the different frequencies? The different types of notes? How many bytes does it take to represent the information on the musical staff shown above? (State any further assumptions you need to make.)
4. What is the compression ratio achieved by modeling the notes as in question 3 vs capturing the raw audio as in question 2?

Analysis

Exploring the Frequency Domain

Question 1: What happens to the sine wave as you change the frequency, amplitude and phase? Describe what you observe in both the time and frequency domain as well as what you hear.

Question 2: How does the sine wave differ from the other three (triangle, square and sawtooth) waves. I.e. What do the other three waves have in common that the sine wave does not?

Question 3: The triangle wave shows multiple spikes in the frequency domain graph.
• For a triangle wave at a frequency of 180Hz, what are the ratios of the higher frequency spikes to the frequency of the first spike (you should use the cursor to get estimates of the frequency of each spike)? Round your ratios to the nearest integer value. Remember to look at the frequency (on the abscissa) and not the amplitude (on the ordinate).
• What happens to the relative horizontal distance between spikes as you sweep the frequency slider back and forth between 180Hz and 380Hz (going above 380 will show a strange “mirroring” effect in the frequency domain, the reason for this will become clear in week 6)?
• How are the frequency values of the higher frequencies changing with respect to the value of the primary frequency defined by the frequency slider?

Analyzing Sound Waves

**Question 4:** Using what you’ve learned from lecture and the harmonic information, quantitatively describe the four .wav files.

Building Sounds

**Question 5:** For the two pure tones you created:

a. What happens when the frequencies of the two pure tones are close together?
b. What happens when the frequencies of the two pure tones are far apart?
c. How does the wave change when you change the phase of one of the two tones?

Building a Synthesizer

**Question 6:** Present the values that you wrote down for the different instruments in this section.

Conclusion

**Question 7:** In lecture 3, we saw the idea of saving space by having a lookup table that allows the Kindle to store the shape of each character once and look it up whenever it needs to draw a character. How can you use a similar space saving idea when talking about notes on a staff and synthesized instruments?

**Question 8:** In this question, you will extend your model from the prelab (question 3) to include what you need to represent the synthesized instruments we created so that a computer can take
the digital representation of the music and the digital representation of the instrument and play
the song using the instrument.

a. How would you encode the instruments you modeled above? How many bytes do you need
to describe your model? (State your assumptions; the course has not given you a way to give
a unique, precise answer, however, it has given you enough background that you can make
ballpark estimates).
b. Adding this additional model information to the information you identified in question 3 of the
prelab, how does this impact the compression ratio as compared to the results from question
4 in the prelab?
c. How would it impact the compression ratio for an entire 3 minute song? (State necessary
assumptions.)

Further Thought

This section is not part of your required assignment. Along with each week, we will offer directions
and questions for further thought. Due to the nature of this course, we can only begin to glimpse the
depth and richness of each of the topic areas. These questions offer some headings to contemplate
further depth. These will often be open-ended questions.

How could you produce sheet music from a sampled recording of a piano (or orchestra)?