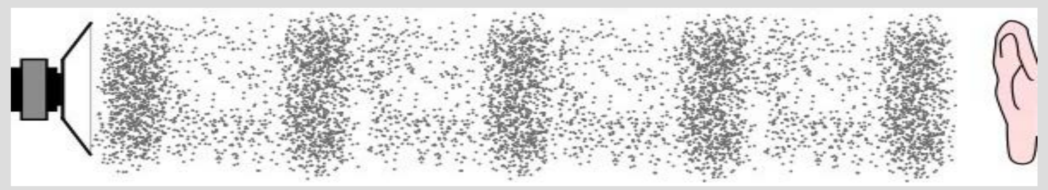
Converting Music from Audio to Visual Representation

Yuval Medina[1], Malcolm Slaney[2]
Henry M. Gunn High School[1], Google Inc.[2]

Background

What is sound?

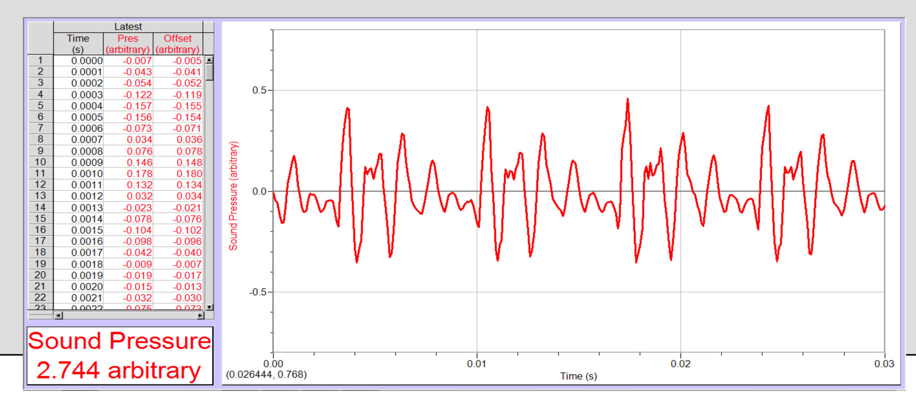
- *A sound is a wave of air pressure*
- *Just like any wave, it has both amplitude and frequency*
- *When a sound reaches our ears, we hear its amplitude (the amount of pressure) as its loudness, and its frequency (how frequent the waves are) as pitch*



In the diagram above, a sound is played by a speaker, and sound waves travel through air to the ear (the individual points represent single particles of air)

- *The way our ear hears sound is highly complex, involving thousands of delicate hair cells that vibrate deep within the ear canal*
- *In fact, the way we perceive sound is so complex that it cannot be reproduced, not even by a computer; a computer can only approximate the human perception of sound*
- *The computer does this by storing the amount of air pressure felt by a microphone, or any other recording device, at time intervals that are pre-specified*
- *A computer doesn't 'hear' pitch*

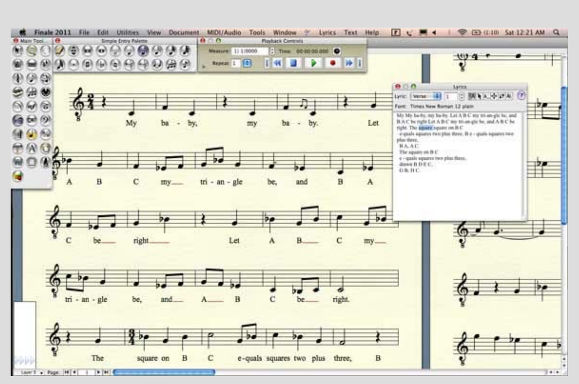
A visual representation of what the computer 'hears'.



Definition of the Problem

- Because the computer perceives sound differently from the way we perceive sound, some challenges arise in software design that analyzes sound, especially in applications that relate to musicians and composers.
- Music composers who wish to use technology to simplify the music writing process (which can be very time-consuming, as the author can attest), have very few options available today.

An typical MIDI keyboard



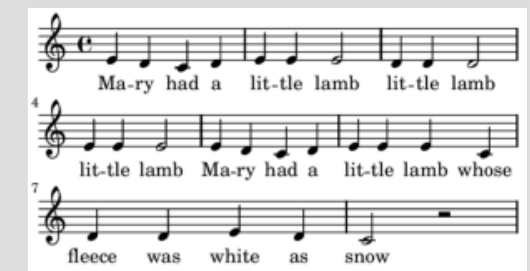
A commonly used program for manual notation is Finale, pictured to the right

- Two of the most commonly used genres of music writing software are MIDI recording and Manual Notation
  - MIDI recording: a specially designed electronic version of an instrument is plugged into a computer and records musical notes in a machine-readable format called MIDI
    - Pros: easy to use, recording the music is not very time consuming
    - Cons: equipment can be expensive, the process can be very constricting and error prone- electronic instrument most likely doesn't sound and feel like the real instrument, which can hinder the recording quality of a song
  - Manual Notation: a program in which the composer manually inputs every note of his composition, using either the mouse or the keyboard
    - Pros: less time-consuming than writing music by hand, straightforward
    - Cons: can still take up a lot of time, hard to get used to the process- when just starting to use manual notation, composition can actually be more time consuming than even writing by hand
- Evidently, writing music using technology can be unusually hard when compared to recording objects in other fields, like photography, which simply uses digital images, graphic design, which utilizes Photoshop, novel and story writing, which can use anything from Word to Google Docs, etc.

The goal of this project, then, was to research information from various sources that could be of use in the end goal: writing a computer program capable of writing music simply through playing a song.

Finding the Solution

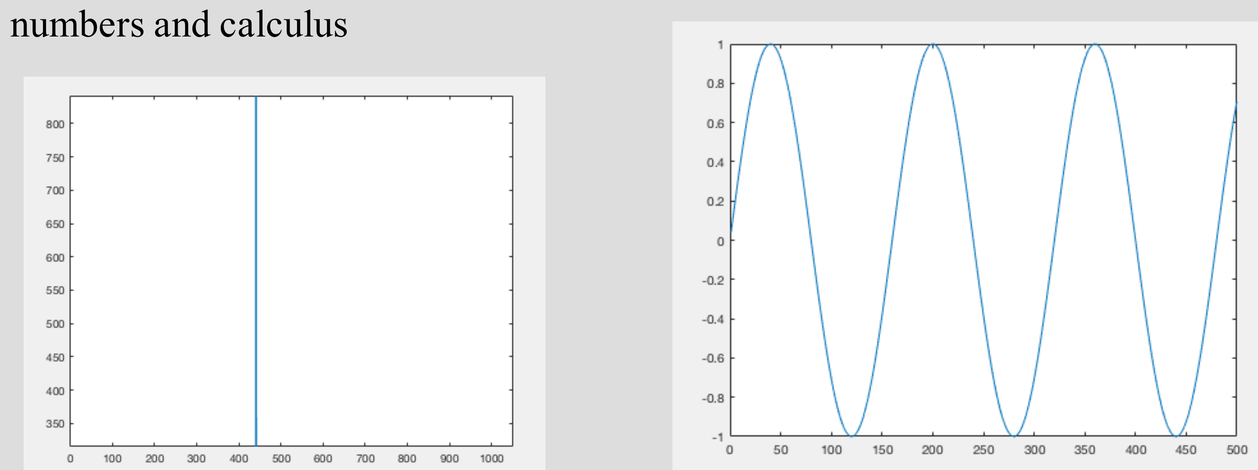
The best way to achieve the goal of this project was to write a program that would record a piece of music, and then write it down in a human-readable format, A.K.A sheet music format, as pictured below:



To convert the computer's individual pressure readings into pitch and rhythm, the program will have to do the following:

1. Finding the different notes, or pitches, within the song

- The computer records sound as individual bits of pressure which form a sinusoid, or repeating curves, when graphed
- To figure out the pitch of the sound signal, the computer must figure out how frequent the curves are, or in other words what the frequency is
- Because the pitch, and thus the frequency, changes throughout a song, the computer can only apply this analysis to tiny bits of the song
- To figure out what the frequency of the sound signal is, a process called "Fourier Transform" is applied to the signal, which uses math like complex numbers and calculus



After a Fourier Transform is applied, the computer can figure out the pitch of a sound by seeing on which 'x' value the highest peak is placed on the graph

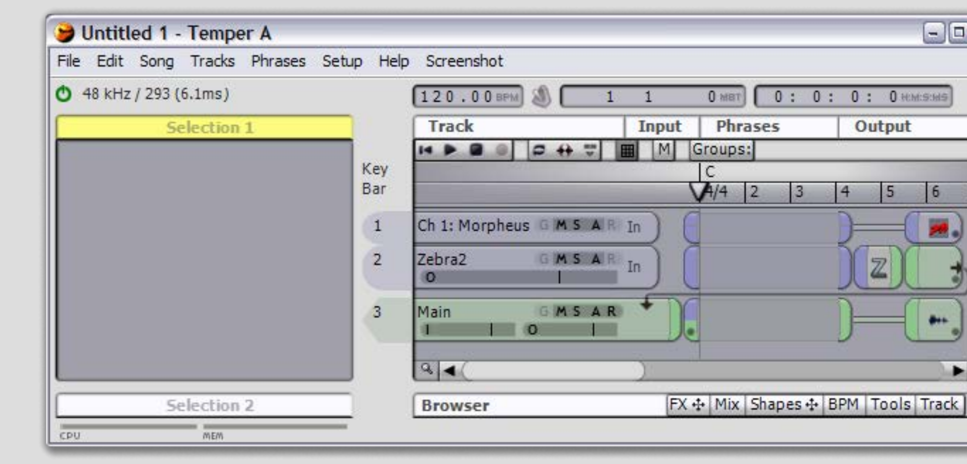
- However, this process is much more complicated than it sounds, particularly because of the overtone series and the missing fundamental problem
- Overtone series- a series of frequencies that play together with the original note and are two times, three times, four times, etc. higher in frequency than the original note- all instruments and voices contain overtones
- Missing fundamental problem- when some notes are played, especially very low notes, the original, or fundamental, note (the one that is played, and subsequently heard) doesn't actually sound, and only the note's overtones are played

2. Rhythm

- The rhythm is the duration of each note within the song
- Because the program analyzes the different pitches of bits of the song one at a time, the song will be stored as a myriad of notes that represent each bit of the song
- If the bits were all of equal length, for example 1 second, and, for example, the 4 first bits represented middle C, or C4, the program can assume that the song starts with the note middle C, of duration 4 seconds
- Then, if the user is asked to specify what is the tempo of his or her song, for example 60 quarter notes per minute, the program could figure out that the first note was a whole note, which constitutes four quarter notes

Implementation

- The program was written in MATLAB and is called Magic Notation
- It is given 5 parameters- the length of the song to be recorded in seconds, the beats per minute, the value of the first note played (quarter note, half note, etc.), the time signature of the song, and the key signature of the song
- It returns a table of note values, and their duration
- It also saves an MIDI file to memory, which can be played back by the computer or converted to sheet music in applications like GarageBand



a MIDI track, a MIDI track feeding into a VST instrument, and an audio track

Evaluation

Except for the fact that the program I wrote is missing an easy-to-use user interface, all of it functions very smoothly. As long as the user doesn't sing or play notes that are shorter than 15 milliseconds (a feat that can only be accomplished in extremely virtuosic piano pieces), the program can precisely notate every voice of a song and it can later be expanded to combine separate voices of a song into one MIDI file in which multiple notes play at once.

Thus, the problem addressed by this project was nearly fully resolved, and a great alternative to writing music with technology was created.

I am extremely proud of the accomplishments I made during this program, and I will continue to try to improve my algorithm and to create a pleasing user interface as this year comes to a close.

ACKNOWLEDGEMENTS / REFERENCES

Mu̇ller, Meinard. *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. N.p.: n.p., n.d. Print.

Thank you to Dr. Slaney for making my project possible! I gained a lot of insight and know-how in both my programming skills and my knowledge in a field I long considered infinitely interesting but just as infinitely unapproachable!