INTRODUCTION TO COMPUTER MUSIC
REVIEW

Roger B. Dannenberg
Professor of Computer Science, Art, and Music

COURSE REVIEW

What did we learn?
What are the most important lessons to take away?

1. Digital Representation of Continuous Signals

- Sampling Theory
- You can represent EVERY instant of a CONTINUOUS signal with FINITE samples (if the signal is band limited)
- In practice, quantization introduces noise, but every bit reduces the noise by a factor of 2 (6dB / bit)
- You should know
  - How A-to-D and D-to-A work
  - Impact of sample rate and sample size

2. Unit Generators

- Synthesis by combining unit generators
- Unit generators model processing on input signals to produce output signals
- Unit generators have internal state
- You should know
  - How unit generators and connections are expressed in Nyquist
  - How UGs and connections are expressed in Max/MSP

3. Frequency Domain

- Our perception is mainly based on intensity at different frequencies: the spectrum.
- (Linear) Filters are described as multiplication of the spectrum.

- You should know
  - How to interpret a spectrum as a sum of sines, noise, etc.
  - How to interpret filter frequency response expressed as a magnitude spectrum
  - Multiplication in frequency is convolution in time

4. FM Synthesis

- Musical tones have many frequencies (partials)
- Tones usually have harmonic partials: f_0, $2f_0$, $3f_0$, $4f_0$, etc.
- FM generates carrier frequency (f_0) + multiples of the modulator frequency
- Rich, variable spectra for the cost of 2 oscillators!

- You should know
  - How to design an FM instrument to (qualitatively) achieve a desired spectrum
  - Modulation Index and its relationship to number of partials

5. Algorithmic Composition

- Patterns and combinations of patterns
- Random selection from distributions and lists
- Guiding overall form, letting computer make detailed choices

- You should know
  - How pattern objects work
  - Form and interpretation of Nyquist scores
  - How to make a score with SCORE-GEN
  - Using SREF to incorporate global tendencies into algorithmic choices

6. Sampling Synthesis

- High quality sound from recordings
- Limits to what you can do with the sounds:
  - Pitch
  - Duration

- You should know
  - How sampling changes duration of samples
  - How sampling changes pitch of samples

7. Granular Synthesis

- Chopping up sounds to generate rich textures
- Deterministic processing for time stretching
- Random processing for turbulent textures

- You should know
  - Grain parameters: duration, density, time offset in source file, time offset in output sound, pitch shift
  - How to use fixed step sizes to accomplish time stretching

8. Spatial Audio

- HRTF - Head-Related Transfer Function
  - Idea: filter sound the same way your head would, use headphones to eliminate room effects
- Cues for direction and distance
  - Panning (inter-aural amplitude difference)
  - Inter-aural time difference
  - Attenuation with distance
  - Reverberation to direct sound ratio

- You should know
  - How to use HRTFs: select left and right HRTFs and convolve
  - Constant power panning

9. Voice, Source-Filter Model

- Human voice:
  - Source (Vocal folds) +
  - Filter (Vocal tract)
- Formants:
  - Resonances that boost the spectrum around certain frequencies
  - First (lowest) 2 formants mainly determine vowel sounds
- You should know
  - How to use LPC or Spectral Processing for cross synthesis
  - How are vowel sounds made?

10. Physical Models

- Attempt to *simulate* the physics of vibration
- Waveguide
  - 2-directional delay line (left-going, right-going wave)
  - Sum to get simulated signal, e.g. air pressure in a tube
- You should know
  - Waveguide is efficient (just two delay lines = circular buffers)
  - Lumped filter to represent losses in round-trip
  - Non-linear elements are used to produce oscillation

11. Spectral Interpolation Synthesis

- Spectrum as a function of frequency and amplitude controls
- Controls like amplitude and frequency depend strongly on phrases and musicianship

- You should know
  - Implementation is modified table-lookup oscillator (2 tables)
  - Limited to harmonic spectra because of table-lookup

12. Data Compression

- Coding redundancy
- Inter-sample redundancy
- *Perceptual redundancy*

- You should know
  - uLaw = quasi-logarithmic encoding, why?
  - Temporal masking
  - Frequency masking
  - Encode in frequency domain – why?
  - Role of quantization in audio compression:
    - Reduce sample size -> fewer bits -> more quantization
    - Quantization is acceptable when it is masked

13. Music Understanding

- Computer Accompaniment
- Score Alignment
- Style Recognition
- Onset Detection
- Human-Computer Music Performance:
  - Responsive, listening computers in live performance

- You should know
  - What all the terms mean

What did we leave out?

- Real-time systems:
  - Scheduling events, managing/representing time
  - How do audio and MIDI systems and interfaces work?
- Music Information Retrieval
  - Audio Features,
  - Genre Classification,
  - Music Fingerprinting
- Instruments and Interfaces
  - Human factors, ergonomics
  - Sensors
  - Gestures and mapping to musical control

Conclusion

- Why Computer Music?
  - Precision – the ability to
- I hope you learned some CS and some Music skills you can use in the future.