# Analyzing and composing music with algorithms and machine learning

| Author: | Dimo Chanev |
|---|---|
| Supervisiors: | Emil Kelevedjiev |
| | Zornitsa Dzhenkova |

August 15, 2018

# Outline

Tuning
MIDI
Markov chains
Tested algorithms

Pythagorean tunning
Equal temperment and 12-TET

# Tuning

Tuning
MIDI
Markov chains
Tested algorithms

Pythagorean tunning
Equal temperment and 12-TET

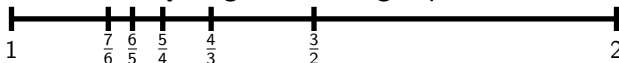# Pythagorean tuning

Pythagorean fractions:

$$f = \frac{i+1}{i}; \ i = 1, 2, ..., n$$

The Pythagorean string separation:

Tuning
MIDI
Markov chains
Tested algorithms

Pythagorean tunning
Equal temperment and 12-TET

# Equal temperment and 12-TET

| Name | 12-TET | Pythagorean scale |
|------|--------|-------------------|
| Unison ($C$) | $2^{\frac{0}{12}} = 1$ | $\frac{1}{1} = 1$ |
| Minor second ($C\sharp/D\flat$) | $2^{\frac{1}{12}} \approx 1.05946$ | $\frac{16}{15} \approx 1.06666$ |
| Major second ($D$) | $2^{\frac{2}{12}} \approx 1.12246$ | $\frac{9}{8} = 1.125$ |
| Minor third ($D\sharp/E\flat$) | $2^{\frac{3}{12}} \approx 1.18920$ | $\frac{6}{5} = 1.2$ |
| Major third ($E$) | $2^{\frac{4}{12}} \approx 1.25992$ | $\frac{5}{4} = 1.25$ |
| Perfect fourth ($F$) | $2^{\frac{5}{12}} \approx 1.33484$ | $\frac{4}{3} \approx 1.33333$ |
| Tritone ($F\sharp/G\flat$) | $2^{\frac{6}{12}} \approx 1.41421$ | $\frac{7}{5} = 1.4$* |
| Perfect fifth ($G$) | $2^{\frac{7}{12}} \approx 1.49830$ | $\frac{3}{2} = 1.5$ |
| Minor sixth ($G\sharp/A\flat$) | $2^{\frac{8}{12}} \approx 1.58740$ | $\frac{8}{5} = 1.6$* |
| Major sixth ($A$) | $2^{\frac{9}{12}} \approx 1.68179$ | $\frac{5}{3} \approx 1.66666$* |
| Minor seventh ($A\sharp/B\flat$) | $2^{\frac{10}{12}} \approx 1.78179$ | $\frac{16}{9} \approx 1.77777$* |
| Major seventh ($B$) | $2^{\frac{11}{12}} \approx 1.88774$ | $\frac{15}{8} = 1.875$* |
| Octave ($C$) | $2^{\frac{12}{12}} = 2$ | $\frac{2}{1} = 2$ |

Note: the values with * can't be represented like Pythagorean fractions with decent accuracy but the human ear can't differentiate this (in the most cases)

Tuning
MIDI
Markov chains
Tested algorithms

Header chunk
Track chunks

# MIDI

MIDI is an acronym to "Musical Instrument Digital Interface". It:

- Developed by MIDI Manufacturers Association
- Targets compact representation
- Splited into chunks

This makes MIDI a perfect way to store musical data in form of notes that enables easy manipulation

Tuning
MIDI
Markov chains
Tested algorithms

Header chunk
Track chunks

## Header chunk

- Contains data about the file
- Always one

$$(M)(T)(h)(d)\ (0)(0)(0)(6)\ (0)(f)\ (n)(n)\ (d)(d)$$

f - the file type
nn - the number of track chunks
dd - the way of division of the time

Tuning
MIDI
Markov chains
Tested algorithms

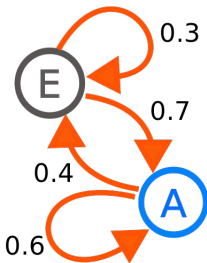Header chunk
Track chunks

# Track chunks

Each track chunk is built up by events (messages). Each event has delta time and data bytes. The most important events are:

- NOTE_ON - start playing a note
- NOTE_OFF - stop playing a note

The events described above have common structure:
(type;channel) (note number) (velocity)

$$note\ number = 69 + 12\log_2 \frac{f}{440}\ where\ f\ is\ the\ frequency$$

# Markov chains



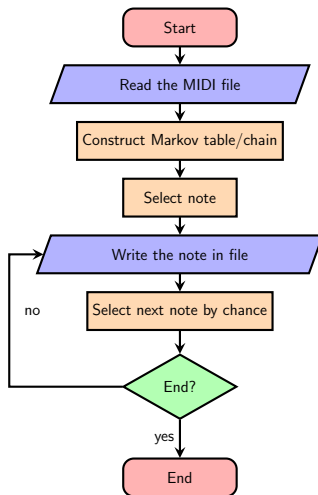Markov chain diagram          Andrey Markov

# Markov chains

|   | Next event | | |
|---|---|---|---|
|   | A | B | C |
| A | 33% | 22% | 45% |
| B | 81% | 9% | 10% |
| C | 30% | 60% | 10% |

|   |   | Next event | | |
|---|---|---|---|---|
|   |   | A | B | C |
| A | A | 16% | 16% | 68% |
| A | B | 100% | 0% | 0% |
| A | C | 12% | 75% | 13% |
| B | A | 37% | 25% | 38% |
| B | B | 0% | 0% | 100% |
| B | C | 100% | 0% | 0% |
| C | A | 33% | 33% | 34% |
| C | B | 83% | 17% | 0% |
| C | C | 100% | 0% | 0% |

Example Markov chains for the string
"AABAACBABACBABACCACAACBAAACBACBBCABAACBA"

Tuning
MIDI
Markov chains
**Tested algorithms**

**Common**
Different
Results

# Common block diagram

Tuning
MIDI
Markov chains
**Tested algorithms**

Common
**Different**
Results

# Diffrences between the algorithms

- First algorithm - uses only the pitch of the previous note for constructing Markov chain table
- Second algorithm - uses the pitch and the length of the previous note for constructing Markov chain table
- Third algorithm - uses only the pitch of the previous two notes for constructing Markov chain table

Tuning
MIDI
Markov chains
Tested algorithms

Common
Different
Results

# Results

Tuning
MIDI
Markov chains
Tested algorithms

Common
Different
Results

# Future development

Tuning
MIDI
Markov chains
Tested algorithms

Common
Different
Results

# Questions

Tuning
MIDI
Markov chains
Tested algorithms

Common
Different
Results

# Thank you for the attention!