Unit 5: Algorithmic Music and Perception

Course Notes

MA375TGR - Mathematics of Music
Semester VII - Institutional Elective

Faculty: Dr. Venugopal K.

Contents

1 Introduction to Algorithmic Composition **4**
1.1 Overview . . . 4
1.2 Historical Context . . . 4
1.3 Deterministic vs. Stochastic Methods . . . 4
1.4 Questions and Answers . . . 5

2 Fractals and Recursion in Music **5**
2.1 Fundamental Concepts of Fractals . . . 5
2.1.1 Why Logarithms? The Mathematical Necessity . . . 6
2.2 Examples of Fractals . . . 7
2.2.1 The Koch Snowflake . . . 7
2.2.2 The Weierstrass Function: A Mathematical "Monster" . . . 8
2.3 Fractals in Music . . . 10
2.3.1 1/f Noise and Musical Structure . . . 10
2.3.2 Self-Similarity in Musical Form . . . 11
2.4 Generating Fractal Music . . . 11
2.4.1 Deterministic Fractal Signals . . . 11
2.4.2 Random Fractals . . . 11
2.5 Questions and Answers . . . 12

3 Stochastic and Probabilistic Methods **14**
3.1 Randomness in Composition . . . 14
3.2 Markov Chains . . . 15
3.2.1 Markov Chain Orders . . . 15
3.2.2 Markov Process Formulation . . . 15
3.3 Transition Matrices . . . 16
3.4 Information Theory in Music . . . 16
3.4.1 Entropy . . . 16
3.4.2 Musical Meaning and Expectation . . . 16
3.5 Monte Carlo Methods . . . 17
3.6 Questions and Answers . . . 17

4 Mathematical Models of Musical Perception **20**
4.1 Psychoacoustics Overview . . . 20
4.2 Pitch Perception . . . 20
4.2.1 Just Noticeable Difference (JND) . . . 20
4.2.2 Missing Fundamental . . . 20
4.2.3 Duration and Pitch . . . 20
4.3 Critical Bands . . . 20
4.3.1 Critical Bandwidth . . . 21
4.3.2 Bark Scale . . . 21
4.3.3 Quality Factor . . . 21
4.4 Consonance and Dissonance . . . 22
4.4.1 Theories of Consonance . . . 22
4.4.2 Critical Band Theory of Dissonance . . . 22
4.4.3 Dissonance Metric . . . 22
4.5 Loudness Perception . . . 22
4.5.1 Acoustic Uncertainty Principle . . . 22
4.5.2 Duration and Loudness . . . 23
4.6 Temporal Integration and Masking . . . 23
4.6.1 Types of Masking . . . 23
4.6.2 Frequency Masking . . . 23
4.7 Questions and Answers . . . 23

5 Neural Networks and Genetic Algorithms **27**
5.1 Artificial Neural Networks . . . 27
5.1.1 Basic Architecture . . . 27
5.1.2 Learning Process . . . 27
5.1.3 Musical Applications . . . 28
5.2 Genetic Algorithms . . . 28
5.2.1 GA Process . . . 28
5.2.2 Musical Representation . . . 29
5.2.3 Genetic Operators . . . 29
5.3 Comparative Advantages . . . 30
5.4 Questions and Answers . . . 30

6 Advanced Topics in Algorithmic Composition **31**
6.1 L-Systems and Formal Grammars . . . 31
6.2 Constraint-Based Composition . . . 32
6.3 Cellular Automata . . . 32
6.4 Chaos and Strange Attractors . . . 33
6.5 Questions and Answers . . . 33

7 Conclusion **40**
7.1 Key Insights . . . 40
7.2 Connections Across Topics . . . 41
7.3 Limitations and Future Directions . . . 41
7.4 Final Reflection . . . 41

A Quick Reference Formulas **42**
A.1 Fractal Mathematics 42
A.2 Information Theory 42
A.3 Psychoacoustics 42
A.4 Neural Networks 43
A.5 Markov Chains 43

B Glossary of Key Terms **43**

1 Introduction to Algorithmic Composition

1.1 Overview

Algorithmic composition represents the intersection of mathematics, computer science, and musical creativity. It involves the use of mathematical procedures, formal rules, and computational algorithms to generate musical structures. This approach has roots extending back to ancient times, but has flourished with the advent of computing technology.

Definition 1.1 (Algorithmic Composition)**.** Algorithmic composition is the technique of using algorithms—formal sets of operations, procedures, or instructions—to create musical material. These algorithms may range from simple deterministic rules to complex stochastic processes.

Definition 1.2 (Generative Music)**.** Generative music refers to music that is created by a system in real-time, where the system has been initialized with parameters that allow it to produce novel, non-repeating musical structures. The composer designs the system and its rules rather than the specific musical output.

1.2 Historical Context

The use of systematic methods in composition predates computers by centuries:

- **Medieval Era**: Guido d'Arezzo (c. 991-1033) developed systematic methods for composing plainchant melodies
- **Renaissance**: Composers used canons and mathematical transformations (inversion, retrograde, augmentation, diminution)
- **18th Century**: Mozart's *Musikalisches Würfelspiel* (Musical Dice Game) used randomness to generate minuets
- **20th Century**: Schoenberg's twelve-tone technique, Cage's chance operations, Xenakis' stochastic music
- **Modern Era**: Computer-assisted composition, artificial intelligence in music

1.3 Deterministic vs. Stochastic Methods

Algorithmic composition can be classified broadly into two categories:

1. **Deterministic Methods**: Given the same initial conditions, the algorithm always produces the same output. Examples include:
   - Strict serial techniques
   - L-systems and formal grammars
   - Deterministic fractals
2. **Stochastic Methods**: Incorporate randomness or probability, producing different outputs from the same initial conditions. Examples include:

- Markov chains
- Monte Carlo methods
- Random fractals
- Genetic algorithms

1.4 Questions and Answers

Question 1 (Short Answer): Define algorithmic composition and provide two examples of algorithmic techniques.

Answer: Algorithmic composition is the use of formal procedures, rules, or algorithms to create musical material. Two examples are: (1) Twelve-tone serialism, where all 12 chromatic pitches are arranged in an ordered row and systematically manipulated through transformations like inversion and retrograde; (2) Markov chains, which use probability tables based on previous states to determine subsequent musical events.

Question 2 (Short Answer): What is the key difference between deterministic and stochastic compositional methods?

Answer: Deterministic methods produce the same output given the same initial conditions and always follow fixed rules without randomness. Stochastic methods incorporate probability and randomness, producing potentially different outputs each time even with identical starting conditions. Deterministic methods are predictable and repeatable, while stochastic methods introduce controlled unpredictability.

2 Fractals and Recursion in Music

2.1 Fundamental Concepts of Fractals

Definition 2.1 (Fractal)**.** A fractal is a geometric or mathematical object that exhibits self-similarity at different scales. Fractals often have fractional (non-integer) dimensions and reveal similar patterns regardless of the level of magnification.

Definition 2.2 (Self-Similarity)**.** A structure is self-similar if, when magnified or examined at different scales, its structure remains similar to or statistically related to the original scale. This property can be exact (deterministic) or statistical (random).

Definition 2.3 (Fractal Dimension)**.** The fractal dimension D is a measure of how a fractal's detail changes with the scale at which it is measured. It is given by:

$$D = \frac{\ln s}{\ln L} \tag{1}$$

where s is the characteristic size (number of self-similar pieces) and L is the scaling coefficient (scaling factor).

For Euclidean objects, $D \in \{1, 2, 3, \ldots\}$ (lines, surfaces, volumes). For fractals, D can take non-integer values, indicating structures that exist "between" traditional dimensions.

2.1.1 Why Logarithms? The Mathematical Necessity

The use of the logarithm (ln) in the dimension formula is to solve for dimensions.

1. Dimensions Act as Exponents

In geometry, "dimension" tells us how the "mass" (or number of pieces) of an object grows when we scale it.

Let's examine standard shapes to see the pattern:

- **1D Line**: If you scale it by 3, you get 3 pieces. ($3^1 = 3$)
- **2D Square**: If you scale the sides by 3, you get 9 small squares. ($3^2 = 9$)
- **3D Cube**: If you scale the sides by 3, you get 27 small cubes. ($3^3 = 27$)

It can be observed that the fundamental relationship is:

$$s = L^D \tag{2}$$

This reads as: *(Number of Pieces) = (Scaling Factor) raised to the power of (Dimension)*

2. Solving for the Exponent

Consider the Koch curve: when we scale by a factor of 3, we get 4 identical pieces. So we know $s = 4$ and $L = 3$, but we don't know D:

$$4 = 3^D \tag{3}$$

We are asking: "To what power must I raise 3 to get 4?"

In mathematics, the operation that isolates an exponent is the **logarithm**. Just as division is the inverse of multiplication, logarithms are the inverse of exponentiation. Therefore:

$$D = \log_3(4) \tag{4}$$

Fortunately, the **Change of Base Formula** allows us to calculate this using any log base, as long as we are consistent:

$$\log_L(s) = \frac{\ln s}{\ln L} \tag{5}$$

Remark 2.1. It doesn't strictly have to be ln (base e). You could use $\log_{10}$ and get the exact same answer:

$$\frac{\ln 4}{\ln 3} \approx 1.2619$$
$$\frac{\log_{10} 4}{\log_{10} 3} \approx 1.2619$$

The natural logarithm is preferred in mathematics because of its special calculus properties and its relationship to exponential growth processes.

Summary: Why We Use Logarithms

1. The relationship between size and scale is a **power law**: $s = L^D$

2. To solve for the exponent D, we **must take the logarithm**
3. We use the ratio of logs ($\ln s/\ln L$) because it makes the calculation easy on any standard calculator
4. The logarithm is not a choice—it is the mathematical necessity for extracting exponents from exponential relationships

Example 2.1 (Computing the Koch Curve Dimension)**.** For the Koch curve:

- Scaling factor: $L = 3$ (each segment divided into thirds)
- Number of pieces: $s = 4$ (four segments replace each original segment)

Using the formula:

$$\begin{aligned} D &= \frac{\ln s}{\ln L} \\ &= \frac{\ln 4}{\ln 3} \\ &\approx 1.2619 \end{aligned}$$

This non-integer dimension (between 1 and 2) reflects that the Koch curve is "more than a line but less than a surface"—a true fractal.

2.2 Examples of Fractals

2.2.1 The Koch Snowflake

The Koch snowflake is a classic example of a deterministic fractal:

1. Start with an equilateral triangle with sides of length 1
2. For each side, divide it into three equal segments
3. Replace the middle segment with two sides of an equilateral triangle pointing outward
4. Remove the base of this new triangle
5. Repeat infinitely

Properties:

- As iterations increase, the perimeter approaches infinity
- The enclosed area remains finite
- Exhibits exact self-similarity at all scales
- Fractal dimension: $D = \frac{\ln 4}{\ln 3} \approx 1.262$

2.2.2 The Weierstrass Function: A Mathematical "Monster"

The Weierstrass function (also called the Weierstrass-Mandelbrot function) is one of the most important examples in mathematical history. Before its discovery in 1872, mathematicians believed that if a function was continuous (no breaks), it had to be smooth (differentiable) at some points. The Weierstrass function shattered this intuition: it is **continuous everywhere, but smooth nowhere**. No matter how much you zoom in, it remains jagged—a true mathematical "monster."

$$W(t) = \sum_{n=0}^{N-1} r^{nH} \cos(2\pi r^n t) \tag{6}$$

The "Recipe": Building a Fractal Through Layering

The summation symbol ($\sum$) tells us this is a "recipe" for layering many different waves on top of each other. Think of it like a music synthesizer or geological stratification:

1. **Layer 0 ($n = 0$):** The base layer—a slow, large-amplitude wave
2. **Layer 1 ($n = 1$):** A faster, smaller wave added on top
3. **Layer 2 ($n = 2$):** An even faster, even smaller wave
4. **Layer 3 and beyond:** Progressively faster and smaller waves

By adding infinite layers, we get a shape that appears random and "spiky," like a mountain horizon or a coastline.

The Parameters: Understanding the "Knobs"

Parameter r: Lacunarity (The Frequency Multiplier)

- **Definition**: "Lacunarity" comes from Latin *lacuna* (gap). It controls how far apart the different layers are in frequency.
- **In the formula**: Appears in $\cos(2\pi r^n t)$
- **Typical values**: $r > 1$ (e.g., $r = 2$ or $r = 3$) to create higher frequencies
- **Effect**:
  - $r = 2$: Each layer vibrates twice as fast as the previous
  - $r = 3$: Each layer vibrates three times as fast
  - Larger r creates more detailed, finer structure

Remark 2.2 (Note on Convention)**.** In the formula as typically written, r should be > 1 to create progressively higher frequencies. Some sources use the reciprocal notation where $r < 1$, representing wavelength ratios rather than frequency ratios. For fractal generation, we need increasing frequencies (decreasing wavelengths) at each iteration.

Parameter H: Hurst Exponent (The Roughness Control)

- **Definition**: Controls how quickly the wave amplitudes diminish as frequencies increase

- **In the formula**: Appears in r^{nH} (the amplitude coefficient)
- **Range**: $0 < H \leq 1$
- **Effect**:
  - **High H (close to 1)**: Small, fast waves diminish very quickly
    - Result: Smoother curve, like rolling hills
    - Less roughness, more predictable
  - **Low H (close to 0)**: Small waves remain relatively large
    - Result: Very jagged, noisy appearance
    - Like static or a harsh cliff face
    - Maximum roughness and unpredictability

Parameter D: Fractal Dimension

- **Definition**: Single number quantifying the curve's "complexity"
- **Relationship**: $D = 2 - H$
- **Interpretation**:
  - A normal smooth line: $D = 1$ (truly one-dimensional)
  - A completely filled square: $D = 2$ (two-dimensional)
  - Fractal curves: $1 < D < 2$ (fractional dimension)
- **Trade-off**:
  - Low H (rough) $\Rightarrow$ High D (close to 2): Line so jagged it almost fills space like a surface
  - High H (smooth) $\Rightarrow$ Low D (close to 1): Line behaves more like standard 1D curve

Visual Examples and Applications
Example A: Mountain Terrain (Geology/Computer Graphics)
Imagine designing a video game landscape:

- **Base Layer ($n = 0$):** General mountain shape (large triangle)
- **Detail Layer ($n = 1$):** Add smaller boulders on slopes
- **Fine Layer ($n = 2$):** Add jagged rocks on boulders
- **Micro Layer ($n = 3$):** Add pebbles and grit

The Weierstrass function mathematically models this exact process. The result looks natural because nature follows this fractal pattern—each level of magnification reveals similar complexity.

Example B: Pink Noise (Music and Audio)
The Weierstrass function can generate **pink noise** (or $1/f$ noise), which sounds more "natural" than white noise:

- **White noise**: Every frequency equally loud (sounds harsh: "shhhhh")
- **Pink noise**: Higher frequencies have lower volume
  - The r^{nH} term creates this frequency-dependent amplitude
  - Sounds like rushing water, wind, or distant rain
  - Mimics natural sound spectra
- **Musical relevance**: As discussed in research by Voss and Clarke, much music exhibits $1/f$ spectral characteristics, balancing entropy and redundancy optimally

Mathematical Properties

Theorem 2.1 (Continuity and Non-Differentiability)**.** The Weierstrass function is:

1. **Continuous**: No breaks or jumps anywhere
2. **Nowhere differentiable**: Has no smooth tangent line at any point
3. **Self-similar**: Zooming in reveals similar structure at all scales

This makes it a pathological function that challenged 19th-century mathematical intuitions about continuity and smoothness.

Remark 2.3 (Historical Significance)**.** Before the Weierstrass function, mathematicians believed continuous functions must be differentiable "almost everywhere." This function proved that intuition wrong and opened the door to fractal geometry, leading eventually to Mandelbrot's formalization of fractals in the 1970s.

2.3 Fractals in Music

2.3.1 1/f Noise and Musical Structure

Richard Voss and John Clarke discovered that much music, when analyzed spectrally over long time spans, exhibits a $1/f$ spectral tendency, where $1 < v < 2$.

Definition 2.4 (1/f Spectrum)**.** A signal has a $1/f$ spectrum if the power at frequency nf is $1/n$ times the power at frequency f. This corresponds to an energy roll-off of approximately -3 dB per octave.

This spectral characteristic appears in:

- Rhythmic patterns (frequencies below 1 Hz)
- Phrase structures
- Motivic development
- Large-scale formal organization

The $1/f$ spectrum suggests an optimal balance between:

- **Redundancy** (predictability, order)
- **Entropy** (unpredictability, surprise)

2.3.2 Self-Similarity in Musical Form

Musical works often exhibit self-similar structures across multiple time scales:

1. **Micro-level**: Ornaments, grace notes, melodic contours within a phrase
2. **Meso-level**: Phrases, periods, sections
3. **Macro-level**: Movements, entire works, cycles of works

Example 2.2 (Bach's Goldberg Variations)**.** The Goldberg Variations exhibit self-similarity through:

- Each variation transforms the bass line and harmonic structure of the theme
- Variations are grouped in threes, with every third being a canon
- The work begins and ends with the same aria, creating large-scale symmetry
- Canonic techniques create self-similar structures at multiple levels

2.4 Generating Fractal Music

2.4.1 Deterministic Fractal Signals

Using the Weierstrass function to generate musical material:

$$f(t) = \sum_{n=0}^{N-1} a^n \cos(2\pi b^n t) \tag{7}$$

$$\text{where } a = r^H, \quad b = r \tag{8}$$

This creates a waveform with self-similar characteristics that can be mapped to:

- Pitch sequences
- Rhythmic patterns
- Dynamic envelopes
- Timbral evolution

2.4.2 Random Fractals

Random fractals use stochastic processes to generate self-similar structures:

1/f Noise Generation:

$$\text{Power Spectrum: } P(f) \propto \frac{1}{f^\alpha} \tag{9}$$

where α controls the spectral slope:

- $\alpha = 0$: White noise (completely random)
- $\alpha = 1$: Pink noise (1/f, optimal for music)
- $\alpha = 2$: Brownian noise (too correlated)

2.5 Questions and Answers

Question 3 (Medium Answer): Explain the concept of self-similarity in music with at least two specific examples from different structural levels.

Answer: Self-similarity in music means that patterns and structures repeat or show statistical similarity across different time scales.

Example 1 (Motivic Level): In Beethoven's Fifth Symphony, the famous four-note motif (short-short-short-long) appears at multiple scales—as a melodic cell, extended into phrases, and governing entire sectional structures.

Example 2 (Harmonic Level): Many Bach fugues exhibit self-similarity through the subject appearing in different voices, at different pitch levels, and in various transformations (inversion, augmentation, diminution), creating fractal-like patterns in the contrapuntal texture.

Example 3 (Formal Level): Rondo form (ABACADA) shows self-similarity where the A section returns multiple times, creating recursive patterns at the large-scale formal level.

Question 4 (Medium Answer): What is the significance of 1/f noise in music? Explain its relationship to entropy and redundancy.

Answer: 1/f noise (pink noise) represents an optimal balance between randomness and predictability in music. Research by Voss and Clarke showed that music across different styles and cultures tends to exhibit 1/f spectral characteristics when analyzed over long time spans.

Significance:

- 1/f noise balances entropy (surprise, unpredictability) with redundancy (pattern, predictability)
- White noise ($1/f^0$) has maximum entropy but no structure—purely random
- Brownian noise ($1/f^2$) has maximum redundancy but is too predictable—overly correlated
- Pink noise ($1/f^1$) achieves the "sweet spot" that maintains listener interest

This explains why purely random music sounds chaotic, highly repetitive music becomes boring, but music with 1/f characteristics maintains optimal engagement through balanced predictability and surprise.

Question 5 (Long Answer): Describe how the Koch snowflake construction process could be adapted to create a musical composition. Include specific mappings from geometric properties to musical parameters.

Answer:

Geometric-to-Musical Mapping Strategy:

1. Initial Triangle → Opening Phrase

Start with a three-note motif representing the three sides of the equilateral triangle:

- Side 1: C (fundamental)
- Side 2: E (major third)
- Side 3: G (perfect fifth)

Duration: 4 beats each = 12 beats total

2. First Iteration → Subdivision

Each side is divided into 3 segments and the middle third is replaced with two new segments:

- Original sides (3) become 4 segments each
- Total segments: 12
- Duration per segment: 1 beat

Mapping:

- Segment 1: Original pitch
- Segment 2: Step up (new triangle point)
- Segment 3: Step up again (triangle apex)
- Segment 4: Return to next original pitch

For C-E-G:

- C → D → E♭ → E (replacing C)
- E → F♯ → G♯ → G (replacing E)
- G → A → B♭ → C (replacing G)

3. Second Iteration → Further Subdivision

Each of the 12 segments undergoes the same process:

- 12 segments become 48 segments
- Duration per segment: 0.25 beats
- Total duration remains 12 beats

4. Musical Parameters

Apply fractal properties to multiple dimensions:

Pitch: Each iteration adds passing tones following the triangular pattern

Rhythm: Note durations decrease by factor of 4 each iteration ($4 \to 1 \to 0.25$ beats)

Dynamics: Follow the "altitude" of each point in the geometric construction:

$$\text{Dynamic} = \text{Base\_Level} + \alpha \times \text{Height}$$

Timbre: Increasingly complex timbres as iterations proceed (more partials added)

5. Self-Similarity Property

The resulting composition exhibits:

- Same melodic contour at different scales
- Increasing rhythmic complexity while maintaining pattern
- Ever-finer ornamentation without losing structural coherence

6. Termination Criteria
Stop iterations when:

- Duration becomes too short to perceive (< 50ms)
- Pitch changes become smaller than JND (Just Noticeable Difference)
- Computational or performance complexity limits reached

Mathematical Properties
Total notes after n iterations: $N_n = 3 \times 4^n$
Perimeter (melodic span) ratio: $P_n = 3 \times (4/3)^n \to$ approaches infinity
This creates music with infinite detail in finite time, exemplifying fractal characteristics while remaining musically coherent.

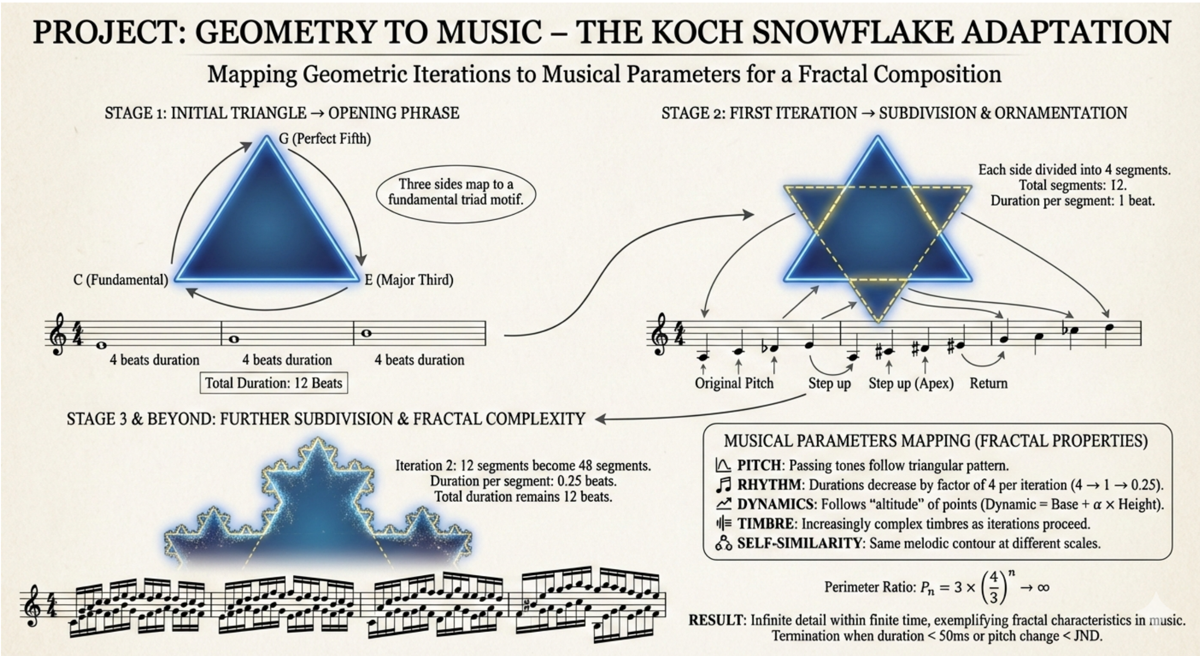

Figure 1: Visual representation of the Koch snowflake construction adapted to musical composition parameters.

3 Stochastic and Probabilistic Methods

3.1 Randomness in Composition

Definition 3.1 (Random Variable)**.** A random variable is a variable whose value is determined by the outcome of a random process. In music, random variables can determine pitch, rhythm, dynamics, or any other musical parameter.

Definition 3.2 (Probability Distribution)**.** A probability distribution describes the likelihood of different outcomes for a random variable. Common distributions in music include:

- Uniform distribution: All outcomes equally likely

- Normal (Gaussian) distribution: Outcomes cluster around a mean
- Weighted distribution: Some outcomes more likely than others

3.2 Markov Chains

Definition 3.3 (Markov Chain)**.** A Markov chain is a stochastic process where the probability of transitioning to any particular state depends only on the current state and not on the sequence of events that preceded it.

3.2.1 Markov Chain Orders

1. **Zeroth-Order Markov Process** (M_0):
   - No memory of previous states
   - Each choice independent
   - Equivalent to weighted random selection
2. **First-Order Markov Process** (M_1):
   - Considers one previous state
   - Transition probability: $P(X_n|X_{n-1})$
   - Most commonly used in music
3. **Second-Order Markov Process** (M_2):
   - Considers two previous states
   - Transition probability: $P(X_n|X_{n-1}, X_{n-2})$
   - Creates more contextual coherence
4. **Nth-Order Markov Process** (M_N):
   - Considers N previous states
   - Transition probability: $P(X_n|X_{n-1}, X_{n-2}, \ldots, X_{n-N})$
   - Higher orders → more complex patterns but risk of verbatim repetition

3.2.2 Markov Process Formulation

Analysis Phase:

$$X = M_N(x) \tag{10}$$

where x is the sequence to analyze and X is the set of probability distribution functions.

Synthesis Phase:

$$y = M_N^{-1}(X) \tag{11}$$

where y is the synthesized sequence generated from the probability distributions.

3.3 Transition Matrices

For a first-order Markov chain, transitions are represented by a matrix:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \tag{12}$$

where p_{ij} is the probability of transitioning from state i to state j.
Properties:

- Each row sums to 1: $\sum_{j=1}^{n} p_{ij} = 1$
- All entries are non-negative: $0 \leq p_{ij} \leq 1$

3.4 Information Theory in Music

3.4.1 Entropy

Definition 3.4 (Shannon Entropy)**.** The entropy H of a discrete random variable X with possible outcomes $x_1, x_2, \ldots, x_n$ and probability mass function $P(X)$ is:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i) \tag{13}$$

Entropy measures the average amount of information (in bits) in a message.

Definition 3.5 (Maximum Entropy)**.** Maximum entropy occurs when all outcomes are equally likely:

$$H_{\max} = \log_2 N \tag{14}$$

where N is the number of possible outcomes.

Definition 3.6 (Redundancy)**.** Redundancy measures how much actual entropy differs from maximum entropy:

$$R(X) = 1 - \frac{H(X)}{\log_2 N} \tag{15}$$

where $R(X) \in [0, 1]$.

3.4.2 Musical Meaning and Expectation

Musical meaning arises from the balance between entropy and redundancy:

- **Too much redundancy** ($R \to 1$): Music becomes too predictable $\to$ boredom
- **Too much entropy** ($H \to H_{\max}$): Music becomes too random $\to$ confusion
- **Optimal balance**: Creates interest through manipulation of expectation

Theorem 3.1 (Expectation and Musical Meaning)**.** Musical meaning is a product of expectation. Expectation is a prediction based on current and past musical experiences. The neocortex continuously anticipates future events, and music engages this faculty through controlled violations and fulfillments of expectation.

3.5 Monte Carlo Methods

Definition 3.7 (Monte Carlo Method)**.** Monte Carlo methods use repeated random sampling to generate solutions or make decisions in compositional processes. They are particularly useful when deterministic algorithms are impractical.

Applications in music:

1. Random note generation with constraints
2. Stochastic exploration of pitch space
3. Generate-and-test approaches to composition
4. Simulating natural variation in performance

3.6 Questions and Answers

Question 6 (Short Answer): What is a Markov chain and how does it differ from purely random selection in music composition?

Answer: A Markov chain is a stochastic process where the probability of each event depends on the current state. Unlike purely random selection (zeroth-order), Markov chains have memory. In first-order chains, the next note depends on the current note; in higher orders, it depends on several previous notes. This creates musical coherence because choices are contextually related rather than independent, resulting in more stylistically appropriate sequences than pure randomness.

Question 7 (Medium Answer): Explain the concepts of entropy and redundancy in music. How do they relate to listener engagement?

Answer: Entropy measures unpredictability or information content in music. High entropy means less predictability, more surprise. It is calculated using Shannon's formula:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i)$$

Redundancy measures predictability or pattern recognition. It is defined as:

$$R(X) = 1 - \frac{H(X)}{\log_2 N}$$

Relationship to Engagement:

- *Too much entropy* (R$\to$0): Music sounds random, chaotic $\to$ listener loses orientation and interest
- *Too much redundancy* (R$\to$1): Music is too predictable $\to$ listener becomes bored
- *Optimal balance*: Maintains interest through balance of familiarity (redundancy) and surprise (entropy)

Composers manipulate this balance to control listener attention and emotional response. Meyer's affect theory states that musical meaning emerges from the manipulation of expectation, which depends on this entropy-redundancy balance.

Question 8 (Long Answer): Construct a first-order Markov chain for a simple 4-note melody (C, D, E, F). Create a plausible transition matrix and explain how to use it to generate a new 8-note melody. Calculate the entropy of your system.

Answer:

Step 1: Define the State Space

States: $S = \{C, D, E, F\}$ (4 states)

Step 2: Construct Transition Matrix

Based on common melodic patterns (stepwise motion preferred, leaps less common):

$$P = \begin{array}{c|cccc} & C & D & E & F \\ \hline C & 0.1 & 0.6 & 0.2 & 0.1 \\ D & 0.3 & 0.1 & 0.5 & 0.1 \\ E & 0.2 & 0.4 & 0.1 & 0.3 \\ F & 0.1 & 0.2 & 0.5 & 0.2 \end{array}$$

Interpretation:

- From C: 60% likely to move to D (step up), 10% to stay on C
- From D: 50% to E, 30% to C (stepwise motion favored)
- From E: 40% to D, 30% to F (stepwise motion)
- From F: 50% to E (step down), 20% to stay on F

Step 3: Generate 8-Note Melody

Starting note: C

Step 1: Current = C. Generate random number $r_1 = 0.42$

- Cumulative probabilities: C[0,0.1), D[0.1,0.7), E[0.7,0.9), F[0.9,1.0]
- $r_1 = 0.42$ falls in range [0.1, 0.7) $\rightarrow$ Next note = **D**

Step 2: Current = D. $r_2 = 0.15$

- Cumulative: C[0,0.3), D[0.3,0.4), E[0.4,0.9), F[0.9,1.0]
- $r_2 = 0.15 \rightarrow$ Next note = **C**

Step 3: Current = C. $r_3 = 0.75$

- Next note = **E**

Step 4: Current = E. $r_4 = 0.35$

- Next note = **D**

Step 5: Current = D. $r_5 = 0.62$

- Next note = **E**

Step 6: Current = E. $r_6 = 0.85$

- Next note = **F**

Step 7: Current = F. $r_7 = 0.45$

- Next note = **E**

Step 8: Current = E. $r_8 = 0.12$

- Next note = **C**

Generated Melody: C - D - C - E - D - E - F - E - C

Step 4: Calculate Entropy

For each state, calculate the entropy of its transition probabilities:

State C:

$$H(C) = -(0.1 \log_2 0.1 + 0.6 \log_2 0.6 + 0.2 \log_2 0.2 + 0.1 \log_2 0.1)$$

$$H(C) = -(-0.332 - 0.442 - 0.464 - 0.332) = 1.57 \text{ bits}$$

State D:

$$H(D) = -(0.3 \log_2 0.3 + 0.1 \log_2 0.1 + 0.5 \log_2 0.5 + 0.1 \log_2 0.1)$$

$$H(D) = 1.68 \text{ bits}$$

State E:

$$H(E) = -(0.2 \log_2 0.2 + 0.4 \log_2 0.4 + 0.1 \log_2 0.1 + 0.3 \log_2 0.3)$$

$$H(E) = 1.85 \text{ bits}$$

State F:

$$H(F) = -(0.1 \log_2 0.1 + 0.2 \log_2 0.2 + 0.5 \log_2 0.5 + 0.2 \log_2 0.2)$$

$$H(F) = 1.76 \text{ bits}$$

Average System Entropy:

$$H_{\text{avg}} = \frac{H(C) + H(D) + H(E) + H(F)}{4} = \frac{6.86}{4} = 1.72 \text{ bits}$$

Maximum Possible Entropy:

$$H_{\text{max}} = \log_2 4 = 2 \text{ bits}$$

Redundancy:

$$R = 1 - \frac{1.72}{2.00} = 0.14 \text{ or } 14\%$$

Interpretation:

The system has moderate entropy (1.72 out of 2.0 bits possible), meaning it balances predictability with surprise. The 14% redundancy indicates the system is not purely random but maintains some predictable patterns (preference for stepwise motion). This would likely create musically coherent melodies that are interesting without being chaotic.

4 Mathematical Models of Musical Perception

4.1 Psychoacoustics Overview

Definition 4.1 (Psychoacoustics). Psychoacoustics is the scientific study of sound perception—how physical sound waves are detected, processed, and interpreted by the auditory system and brain. It relates objective physical properties (Φ variables) to subjective psychological experiences (Ψ variables).

Key Φ (physical) and Ψ (perceptual) relationships:

Φ **(Physical)**	Ψ **(Perceptual)**
Frequency (Hz)	Pitch
Intensity (dB SPL)	Loudness
Spectrum	Timbre
Envelope	Texture

4.2 Pitch Perception

4.2.1 Just Noticeable Difference (JND)

Definition 4.2 (Just Noticeable Difference for Pitch). The JND for pitch is the smallest frequency difference that can be perceived. For pure tones in the mid-frequency range (around 1000 Hz), the pitch JND is approximately 3-5 Hz, or about 1% of the frequency.

The pitch JND varies with frequency:

- At 100 Hz: JND $\approx$ 1-2 Hz
- At 1000 Hz: JND $\approx$ 3-5 Hz
- At 5000 Hz: JND $\approx$ 20-30 Hz

4.2.2 Missing Fundamental

Theorem 4.1 (Missing Fundamental). When a complex tone contains harmonics $2f_0, 3f_0, 4f_0, \ldots$ but lacks the fundamental frequency f_0, the auditory system nevertheless perceives pitch at f_0. This demonstrates that pitch perception is a cognitive construct based on harmonic relationships rather than simply detecting the lowest frequency component.

4.2.3 Duration and Pitch

Pitch perception depends on duration:

- Tones $<$ 10 ms: Sound like clicks regardless of frequency
- Tones 15-30 ms: Pitch begins to solidify
- Optimal pitch recognition: Requires 4-8 cycles of the waveform

4.3 Critical Bands

Definition 4.3 (Critical Band). A critical band is a frequency bandwidth within which the ear cannot distinguish separate frequency components. Sounds within a critical band interact and can produce beats, roughness, and masking effects.

4.3.1 Critical Bandwidth

The critical bandwidth varies with frequency, approximated by Zwicker and Fastl (1990):

$$BW_c(f) = 25 + 75\left[1 + 1.4\left(\frac{f}{1000}\right)^2\right]^{0.69} \text{ Hz} \tag{16}$$

Properties:

- Below 500 Hz: Critical bandwidth relatively constant ($\approx$ 100 Hz)
- Above 500 Hz: Critical bandwidth $\approx$ 20% of center frequency
- The ear can be modeled as having approximately 24 critical bands

4.3.2 Bark Scale

Definition 4.4 (Bark Scale)**.** The Bark scale is a psychoacoustical scale that divides the audible frequency range into 24 critical bands numbered 0-24. One Bark corresponds to one critical bandwidth.

Conversion from frequency to Bark number:

$$z(f) = 13\arctan(0.00076f) + 3.5\arctan\left(\frac{f}{7500}\right)^2 \tag{17}$$

Bark No.	**Center Freq (Hz)**	**Bandwidth (Hz)**
0	50	80
5	570	120
10	1,370	210
15	2,900	450
20	7,000	1,300
24	19,500	—

Table 1: Selected Critical Bands on the Bark Scale

4.3.3 Quality Factor

The quality factor Q of a bandpass filter is:

$$Q = \frac{f_c}{BW} \tag{18}$$

where f_c is the center frequency and BW is the bandwidth.

For critical bands, Q remains relatively constant at 4-6 across most of the audible range, demonstrating constant-Q behavior similar to musical intervals.

4.4 Consonance and Dissonance

4.4.1 Theories of Consonance

Multiple theories explain consonance/dissonance:

1. **Cultural theories**: Consonance determined by learned social and stylistic norms
2. **Acoustic theories**: Based on harmonic relationships and simple frequency ratios
3. **Psychophysical theories**: Related to critical bands and neural processing
4. **Cognitive theories**: Involve expectation and categorical perception

4.4.2 Critical Band Theory of Dissonance

Definition 4.5 (Tonotopic Dissonance)**.** Tonotopic dissonance occurs when two frequencies fall within the same critical band, causing roughness due to beats and interference. Dissonance is maximized when frequencies differ by 5-50% of a critical bandwidth.

Relationship between interval and perception:

- **Difference < JND**: Perfect consonance (unison)
- **Difference = 5-50% of critical band**: Maximum dissonance (beats, roughness)
- **Difference > critical band**: Consonance (separate pitches clearly perceived)

4.4.3 Dissonance Metric

For two pure tones with frequencies f_1 and f_2:

$$D(f_1, f_2) = \begin{cases} 0 & \text{if } |f_2 - f_1| < \text{JND} \\ \text{high} & \text{if } 0.05 \cdot BW_c < |f_2 - f_1| < 0.5 \cdot BW_c \\ \text{low} & \text{if } |f_2 - f_1| > BW_c \end{cases} \tag{19}$$

For complex tones, sum the dissonance contributions of all partial pairs.

4.5 Loudness Perception

4.5.1 Acoustic Uncertainty Principle

Theorem 4.2 (Acoustical Uncertainty Principle)**.** There is a fundamental trade-off between frequency resolution and temporal resolution:

$$\Delta f \cdot \Delta t = k \tag{20}$$

where Δf is frequency uncertainty, Δt is temporal uncertainty, and k is a constant.

This principle has important implications:

- Better frequency precision requires longer time windows
- Shorter sounds have inherently poorer frequency discrimination
- Critical bands allow the ear to optimize this trade-off

4.5.2 Duration and Loudness

Loudness perception depends on duration:

- The ear averages intensity over approximately 200 ms
- Sounds shorter than 200 ms must be proportionally more intense to sound equally loud
- Loudness grows by 10 dB as duration increases by a factor of 10 (up to 200 ms)

4.6 Temporal Integration and Masking

4.6.1 Types of Masking

Definition 4.6 (Masking)**.** Masking occurs when one sound (the masker) makes another sound (the maskee) less audible or completely inaudible.

Types of masking:

1. **Simultaneous Masking**: Masker and maskee occur at the same time
2. **Forward Masking**: Masker precedes maskee (temporal masking, up to 200 ms)
3. **Backward Masking**: Maskee precedes masker (up to 50 ms before)

4.6.2 Frequency Masking

Within a critical band:

- Lower frequencies mask higher frequencies more effectively than vice versa
- A loud tone can raise the threshold for detecting nearby frequencies by 20-40 dB
- Masking spreads asymmetrically upward in frequency

4.7 Questions and Answers

Question 9 (Short Answer): What is a critical band and why is it important in music perception?

Answer: A critical band is a frequency bandwidth (approximately 100-4000 Hz depending on center frequency) within which the ear cannot separately process frequency components. Sounds within a critical band interact, producing beats and roughness. Critical bands are important because they determine consonance and dissonance—intervals smaller than a critical band sound rough and dissonant, while larger intervals sound clearer and more consonant. They also affect timbre perception, loudness summation, and masking effects.

Question 10 (Medium Answer): Explain the relationship between critical bands and the perception of consonance and dissonance. Provide specific frequency examples.

Answer: Critical band theory explains consonance and dissonance through tonotopic interactions:

Mechanism:

- When two frequencies fall within the same critical band, they cannot be separately resolved
- This causes beats and roughness—perceived as dissonance
- Maximum dissonance occurs at 15-30% of critical bandwidth
- Beyond the critical bandwidth, tones are clearly separate—consonant

Example at 1000 Hz:
Critical bandwidth at 1000 Hz $\approx$ 160 Hz (from Zwicker's formula)
Case 1: 1000 Hz + 1015 Hz

- Difference: 15 Hz
- $15/160 = 9.4\%$ of critical band
- Result: Moderate dissonance, beats at 15 Hz

Case 2: 1000 Hz + 1040 Hz

- Difference: 40 Hz
- $40/160 = 25\%$ of critical band
- Result: Maximum dissonance (roughness)

Case 3: 1000 Hz + 1200 Hz

- Difference: 200 Hz
- $200/160 = 125\%$ of critical band
- Result: Consonant (minor third interval)

This explains why minor seconds (dissonant) span less than a critical band, while major thirds and larger intervals (consonant) exceed a critical band.

Question 11 (Long Answer): Describe how the human auditory system processes sound from the physical wave to perception. Include discussion of the basilar membrane, critical bands, and the missing fundamental phenomenon. How do these mechanisms enable music perception?

Answer:
1. Sound Wave Reception (Outer/Middle Ear)
Process:

- Sound waves enter the ear canal and vibrate the eardrum (tympanic membrane)
- Middle ear bones (malleus, incus, stapes) amplify vibrations by $\approx$ 30 dB
- Impedance matching from air to cochlear fluid

2. Frequency Analysis (Inner Ear - Cochlea)
Basilar Membrane:

- Tapered structure: narrow/stiff at base, wide/flexible at apex

- Base responds to high frequencies (20 kHz)
- Apex responds to low frequencies (20 Hz)
- Acts as a mechanical spectrum analyzer

Tonotopic Organization:

$$\text{Position on basilar membrane} \leftrightarrow \text{Frequency response}$$

Each point vibrates maximally at its characteristic frequency, creating a frequency-to-place mapping.

3. Critical Band Formation

Mechanism:

- Adjacent regions on basilar membrane respond to nearby frequencies
- Critical bandwidth: $BW_c(f) = 25 + 75[1 + 1.4(f/1000)^2]^{0.69}$ Hz
- Approximately 24 critical bands span audible range
- Neural connections pool activity within each band

Function:

- Resolves acoustical uncertainty principle trade-off
- Each band has moderate frequency resolution with good time resolution
- Enables simultaneous processing of multiple frequencies

4. Neural Encoding (Cochlear Nerve)

Place Code:

- Which neurons fire indicates frequency (tonotopic map)
- Dominant mechanism for frequencies > 5 kHz

Temporal Code:

- Firing rate and phase-locking encode periodicity
- Dominant mechanism for frequencies < 5 kHz
- Can encode periodicities not present in stimulus (missing fundamental)

5. Missing Fundamental Phenomenon

Demonstration:

- Complex tone: harmonics at 400, 600, 800, 1000 Hz
- Fundamental 200 Hz is absent
- Perceived pitch: 200 Hz!

Explanation:

- Auditory system detects pattern of harmonics: $n \times 200$ Hz
- Neural pattern recognition reconstructs fundamental
- Demonstrates pitch is cognitive construct, not just frequency detection

Mathematical Model:

$$f_0 = \text{GCD}(f_1, f_2, f_3, \ldots) = \text{GCD}(400, 600, 800, 1000) = 200 \text{ Hz}$$

The brain essentially computes the greatest common divisor of the harmonic series.

6. Higher-Level Processing (Auditory Cortex)

Integration:

- Pitch, loudness, timbre, location synthesized
- Pattern recognition and memory comparison
- Expectation generation based on musical context

7. Implications for Music Perception

Harmony and Consonance:

- Consonant intervals separate harmonics into different critical bands
- Dissonant intervals create roughness within critical bands
- Explains psychoacoustic basis of consonance theory

Timbre Recognition:

- Complex tones analyzed into harmonics across critical bands
- Pattern of harmonic amplitudes determines timbre
- Missing fundamental allows bass notes to be implied by harmonics

Polyphony:

- Multiple voices separated by critical band analysis
- Temporal patterns (rhythm) processed independently from pitch
- Enables perception of simultaneous melodic lines

Musical Pitch:

- Pitch is logarithmic perception of frequency: $p = 12 \log_2(f/f_0)$
- JND of $\approx 1\%$ corresponds to about 1/10 semitone
- Categorical perception of pitches within scale systems

Conclusion:

The auditory system is a sophisticated multi-stage analyzer optimized for music perception. The basilar membrane provides initial frequency decomposition, critical bands enable parallel processing of spectral regions, and neural pattern recognition constructs cognitive representations of pitch, including missing fundamentals. This hierarchical processing allows us to perceive complex musical structures—harmony, melody, rhythm, and timbre—from physical sound waves, demonstrating that music perception is fundamentally a mathematical and neurophysiological process.

5 Neural Networks and Genetic Algorithms

5.1 Artificial Neural Networks

Definition 5.1 (Artificial Neural Network)**.** An artificial neural network (ANN) is a computational model inspired by biological neural networks, consisting of interconnected processing units (artificial neurons) that can learn to perform tasks by adjusting connection strengths (weights) through experience.

5.1.1 Basic Architecture

Components:

1. **Input Layer**: Receives external data
2. **Hidden Layer(s)**: Performs intermediate processing
3. **Output Layer**: Produces results
4. **Connections**: Weighted links between neurons

Activation Function:

$$y_j = f\left(\sum_i w_{ij}x_i + b_j\right) \tag{21}$$

where:

- x_i = input from neuron i
- w_{ij} = weight from neuron i to neuron j
- b_j = bias term
- f = activation function (sigmoid, tanh, ReLU, etc.)

5.1.2 Learning Process

Supervised Learning:

1. Present input pattern and desired output
2. Network produces actual output
3. Calculate error: $E = \frac{1}{2}\sum_k (t_k - y_k)^2$
4. Adjust weights to reduce error (backpropagation)
5. Repeat until error minimized

Weight Update Rule:

$$w_{ij}^{(\text{new})} = w_{ij}^{(\text{old})} + \eta\delta_j x_i \tag{22}$$

where:

- η = learning rate
- δ_j = error signal for neuron j

5.1.3 Musical Applications

1. Style Recognition:

- Input: Musical features (pitch, rhythm, harmony)
- Output: Composer/style classification
- Networks can learn to distinguish Bach from Mozart

2. Harmonization:

- Input: Melody
- Output: Chord progressions or bass line
- HARMONET system harmonizes chorale melodies in Bach's style

3. Composition:

- Networks trained on corpus of music
- Generate new material in learned style
- Context units remember recent musical events

5.2 Genetic Algorithms

Definition 5.2 (Genetic Algorithm)**.** A genetic algorithm (GA) is an optimization technique inspired by biological evolution. It maintains a population of solutions that evolve over generations through selection, crossover (recombination), and mutation operations.

5.2.1 GA Process

Algorithm Structure:

1. **Initialize**: Create random population of solutions
2. **Evaluate**: Assess fitness of each individual
3. **Select**: Choose individuals for reproduction based on fitness
4. **Crossover**: Combine pairs of individuals to create offspring
5. **Mutate**: Randomly modify some offspring
6. **Replace**: Form new generation from offspring and possibly some parents
7. **Repeat**: Until termination condition met

5.2.2 Musical Representation

Genotype (Encoding):

Musical material must be encoded as strings that can be manipulated:

- *Pitch sequences*: [60, 62, 64, 65, 67] (MIDI numbers)
- *Rhythm patterns*: [0.25, 0.25, 0.5, 0.25, 0.75] (durations)
- *Rule sets*: Parameters controlling generative processes

Fitness Function:

Critical component—must evaluate musical quality:

$$F(\text{individual}) = w_1 f_{\text{consonance}} + w_2 f_{\text{variety}} + w_3 f_{\text{contour}} + \cdots \tag{23}$$

where:

- $f_{\text{consonance}}$ = measures harmonic consonance
- f_{variety} = measures pitch/rhythm diversity
- f_{contour} = evaluates melodic shape
- w_i = weighting factors

5.2.3 Genetic Operators

Crossover:

Combine two parent melodies at a crossover point:

```
Parent 1: C D E | F G A B
Parent 2: G A B | C D E F
          ------+----------
Child 1:  C D E | C D E F
Child 2:  G A B | F G A B
```

Mutation:

Randomly modify elements:

- *Pitch mutation*: Change a note (C $\rightarrow$ D)
- *Rhythm mutation*: Alter duration (0.25 $\rightarrow$ 0.5)
- *Insertion/Deletion*: Add or remove notes

Selection Methods:

- *Fitness-proportionate*: Probability $\propto$ fitness
- *Tournament*: Best of random subset selected
- *Elitism*: Always keep best individuals

5.3 Comparative Advantages

Aspect	Neural Networks	Genetic Algorithms
Learning Method	Learn from examples through gradient descent	Evolve through selection and variation
Knowledge Representation	Distributed in connection weights	Explicit in chromosome encoding
Best For	Pattern recognition, style imitation	Optimization, creative exploration
Interpretability	"Black box"—difficult to interpret	More transparent—can analyze solutions
Computation	Requires training data	Requires fitness function

5.4 Questions and Answers

Question 12 (Short Answer): What are the main components of an artificial neural network and how do they relate to music composition?

Answer: An artificial neural network has three main components: (1) **Input layer** receives musical features (pitches, rhythms, harmonies), (2) **Hidden layer(s)** process this information through weighted connections, and (3) **Output layer** produces results (next note, chord, style classification). The network learns by adjusting connection weights based on training examples. In music, networks can learn stylistic patterns from composers' works and generate new music in similar styles.

Question 13 (Medium Answer): Explain how a genetic algorithm could be used to compose music. Describe the encoding, fitness function, and genetic operators.

Answer:

Encoding (Genotype): Musical material encoded as chromosome, e.g., melody as MIDI pitch sequence: [60, 62, 64, 65, 67, 69, 71, 72]

Fitness Function: Evaluates musical quality through multiple criteria:

$$F = w_1(\text{consonance}) + w_2(\text{variety}) + w_3(\text{contour}) - w_4(\text{dissonance})$$

Example metrics:

- Consonance: Favor intervals of 3,4,5,7,12 semitones
- Variety: Penalize excessive repetition
- Contour: Reward arch-shaped melodies
- Dissonance: Penalize minor 2nds, tritones

Genetic Operators:

Selection: Choose melodies with higher fitness for reproduction

Crossover: Combine two parent melodies:

```
Parent 1: [60,62,64,65] | [67,69,71,72]
Parent 2: [64,62,60,59] | [57,60,62,64]
Child:    [60,62,64,65] | [57,60,62,64]
```

Mutation: Random changes (5% probability):

- Change pitch: $64 \to 65$
- Transpose segment: $[60,62,64] \to [62,64,66]$

The population evolves over generations, with fitter melodies surviving and producing offspring, gradually improving musical quality.

6 Advanced Topics in Algorithmic Composition

6.1 L-Systems and Formal Grammars

Definition 6.1 (L-System (Lindenmayer System)). An L-system is a parallel rewriting system that starts with an initial string (axiom) and repeatedly applies production rules to generate complex structures. Originally developed to model plant growth, L-systems have been adapted for music generation.

Components:

1. **Alphabet**: $V = \{A, B, C, \ldots\}$ (symbols)
2. **Axiom**: ω (initial string)
3. **Production Rules**: $p : A \to BC$ (rewriting rules)
4. **Iterations**: Number of times rules applied

Example 6.1 (Simple L-System). **Alphabet:** $\{A, B\}$
Rules:

$$A \to AB$$
$$B \to A$$

Axiom: A
Generations:

$$\begin{aligned} n = 0 &: \quad A \\ n = 1 &: \quad AB \\ n = 2 &: \quad ABA \\ n = 3 &: \quad ABAAB \\ n = 4 &: \quad ABAABABA \\ n = 5 &: \quad ABAABABAABAAB \end{aligned}$$

This generates the Fibonacci sequence in string lengths: 1, 2, 3, 5, 8, 13, ...

Musical Mapping:

- $A \to$ pitch C, duration 0.25
- $B \to$ pitch E, duration 0.5

Generated rhythm has self-similar fractal structure.

6.2 Constraint-Based Composition

Definition 6.2 (Constraint Satisfaction Problem)**.** A constraint satisfaction problem (CSP) consists of variables, domains for those variables, and constraints that restrict possible combinations. In music, this can model rules of counterpoint, harmony, and voice leading.

Example: Species Counterpoint
Variables:

- $P_1, P_2, \ldots, P_n$ = pitches in counterpoint line

Domains:

- $P_i \in$ {C3, D3, E3, ..., C5} (two-octave range)

Constraints:

1. No parallel perfect intervals: $(P_i - C_i) \neq (P_{i+1} - C_{i+1})$ for 5ths, 8ves
2. Avoid tritones: $|P_{i+1} - P_i| \neq 6$ semitones
3. Begin and end on perfect consonances
4. Conjunct motion preferred: $|P_{i+1} - P_i| \leq 2$ (¿70% of time)

Search Algorithm:

1. Start with first note
2. For each subsequent note, try values from domain
3. Check all constraints
4. If constraint violated, backtrack and try different value
5. Continue until valid solution found

6.3 Cellular Automata

Definition 6.3 (Cellular Automaton)**.** A cellular automaton is a discrete model consisting of a grid of cells, each in one of a finite number of states. The state of each cell evolves according to rules based on the states of neighboring cells.

Musical Application:
1D CA for Rhythm:

- State 1 = note played
- State 0 = rest
- Rule: Cell's next state depends on its current state and immediate neighbors

Example 6.2 (Rule 30 for Rhythm)**.** Rule 30 (binary: 00011110):
Current pattern → Next state:

```
111 → 0    110 → 0    101 → 0    100 → 1
011 → 1    010 → 1    001 → 1    000 → 0
```

Starting with single 1 in center creates complex, aperiodic rhythmic patterns.

2D CA for Composition:

- x-axis = time
- y-axis = pitch
- Cell state = note on/off
- Rules create evolving melodic patterns

6.4 Chaos and Strange Attractors

Logistic Map:

$$x_{n+1} = rx_n(1 - x_n) \tag{24}$$

where r is a parameter and $x_n \in [0, 1]$.
Behavior depends on r:

- $r < 3$: Converges to fixed point (too predictable)
- $3 < r < 3.57$: Periodic oscillations
- $r \approx 3.57$: Onset of chaos
- $r > 3.57$: Chaotic behavior (deterministic but unpredictable)

Musical Mapping:

$$\text{Pitch} = \text{round}(x_n \times 24) + 60 \quad \text{(MIDI)} \tag{25}$$

With $r = 3.8$, generates complex but structured melodic sequences.

6.5 Questions and Answers

Question 14 (Medium Answer): Design an L-system to generate a simple melody and explain how the production rules create musical structure over multiple iterations.

Answer:
L-System Design for Melody:
Alphabet: $V = \{A, B, C, D\}$
Musical Mapping:

- A = C4 (60), duration = 0.5 beats
- B = E4 (64), duration = 0.5 beats
- C = G4 (67), duration = 1.0 beat
- D = C5 (72), duration = 0.25 beats

Production Rules:

$$A \to AB$$
$$B \to CA$$
$$C \to D$$
$$D \to BA$$

Axiom: $\omega = A$
Evolution:
Generation 0:

$$A$$

Melody: [C4(0.5)] Duration: 0.5 beats
Generation 1: Apply $A \to AB$:

$$AB$$

Melody: [C4(0.5), E4(0.5)] Duration: 1.0 beat
Generation 2: Apply rules to each symbol:

- $A \to AB$
- $B \to CA$

$$ABCA$$

Melody: [C4(0.5), E4(0.5), G4(1.0), C4(0.5)] Duration: 2.5 beats
Generation 3:

- $A \to AB$: twice
- $B \to CA$: once
- $C \to D$: once
- Result: $AB\ CA\ D\ AB$

$$ABCADAB$$

Melody: [C4, E4, G4, C4, C5, C4, E4] Durations: [0.5, 0.5, 1.0, 0.5, 0.25, 0.5, 0.5] Total: 3.75 beats
Generation 4:

$$ABCADABCADBABCA$$

15 notes, 7.75 beats
Musical Analysis:
Self-Similarity:

- The pattern AB appears at multiple scales
- CA motif recurs throughout
- Each generation contains previous generation plus new material

Growth Pattern:

- Length follows Fibonacci-like sequence: 1, 2, 4, 7, 15, 26, ...
- Expansion ratio $\approx 1.7 - 1.9$ per generation

Harmonic Structure:

- Tonic (C) recurs through rule $A \to AB$ and $D \to BA$
- Mediant (E) and dominant (G) provide harmonic motion
- Upper tonic (C5) creates climax points

Rhythmic Interest:

- Mix of durations (0.25, 0.5, 1.0) creates rhythmic variety
- Longer notes (C) provide anchors
- Shortest notes (D) create momentum

This demonstrates how simple rewriting rules create complex, self-similar musical structures with emergent properties—hallmark of algorithmic composition using L-systems.

Question 15 (Long Answer): Compare and contrast three different algorithmic composition techniques: Markov chains, genetic algorithms, and neural networks. For each method, discuss: (1) fundamental principles, (2) strengths and weaknesses, (3) a specific musical application, and (4) how it models or deviates from human compositional processes.

Answer:

1. MARKOV CHAINS

Fundamental Principles:

- Stochastic process with "memory" of recent events
- Next state probability depends only on current state(s)
- Transition matrix encodes style: $P(X_n|X_{n-1}, \ldots, X_{n-k})$
- Analysis of existing music $\to$ probability tables $\to$ synthesis

Strengths:

1. *Easy to implement*: Simple transition tables
2. *Style capture*: Learns from corpus automatically
3. *Local coherence*: Context-sensitive choices
4. *Controllable*: Order adjusts memory depth

Weaknesses:

1. *No long-term planning*: Reactive, not teleological
2. *Wandering*: Lacks goal-directed structure
3. *Plagiarism risk*: High orders reproduce original

4. *Degenerate cycles*: Can get stuck in loops

Musical Application Example:
Melodic improvisation in jazz style:

- Analyze corpus of Charlie Parker solos
- Build second-order transition matrix for pitch sequences
- Generate improvisation over given chord changes
- Result: Stylistically appropriate phrases but lacks overarching narrative

Relation to Human Composition:
Similar to:

- Intuitive improvisation (reacting to what was just played)
- Learning through imitation and absorption of style

Different from:

- Human composers plan large-scale structure
- Humans use teleological thinking (working backward from goal)
- Composers balance multiple simultaneous constraints

2. GENETIC ALGORITHMS
Fundamental Principles:

- Population of solutions evolves through selection
- Fitness function evaluates quality
- Genetic operators: crossover, mutation, selection
- Survival of the fittest drives optimization

Mathematical Framework:

$$\begin{aligned}
\text{Population}_t &= \{I_1, I_2, \ldots, I_n\} \\
\text{Fitness} &: I \to \mathbb{R}^+ \\
\text{Selection} &: \text{Population}_t \to \text{Parents} \\
\text{Crossover} &: \text{Parents} \to \text{Offspring} \\
\text{Mutation} &: \text{Offspring} \to \text{Offspring}' \\
\text{Population}_{t+1} &= \text{Replace}(\text{Population}_t, \text{Offspring}')
\end{aligned}$$

Strengths:

1. *Global exploration*: Searches solution space broadly
2. *Creative novelty*: Combines existing material in new ways
3. *Multi-objective*: Fitness can balance multiple criteria

4. *No training data*: Only needs fitness function

Weaknesses:

1. *Fitness function challenge*: Hard to quantify "good music"
2. *Computational cost*: Many generations needed
3. *Parameter tuning*: Crossover/mutation rates affect results
4. *Evaluation bottleneck*: Human listening required?

Musical Application Example:
Four-part chorale harmonization:

- Genotype: 4 voices $\times$ 16 notes = 64 integers (MIDI pitches)
- Fitness function: $F = w_1 f_{\text{voice-leading}} + w_2 f_{\text{consonance}} + w_3 f_{\text{cadence}}$
- Population: 100 harmonizations
- Evolution: 50 generations
- Result: Produces valid harmonizations, some quite musical

Fitness components:

- $f_{\text{voice-leading}}$: Penalize large leaps, parallel 5ths/8ves
- $f_{\text{consonance}}$: Reward consonant intervals, complete triads
- f_{cadence}: Ensure proper V-I or IV-I ending

Relation to Human Composition:
Similar to:

- Compositional revision process (generate, evaluate, revise)
- Cultural evolution of musical styles
- "Try different things and keep what works" approach

Different from:

- Humans don't compose purely by trial and error
- Human creativity involves intentional innovation, not just mutation
- Composers use explicit theoretical knowledge, not just fitness evaluation

3. NEURAL NETWORKS
Fundamental Principles:

- Interconnected processing units (neurons)
- Learning through weight adjustment
- Pattern recognition in training data

- Distributed representation of knowledge

Architecture:

$$\text{Input}: \vec{x} = (x_1, x_2, \ldots, x_n)$$
$$\text{Hidden}: h_j = f\left(\sum_i w_{ij}x_i + b_j\right)$$
$$\text{Output}: y_k = g\left(\sum_j w_{jk}h_j + b_k\right)$$
$$\text{Learning}: \Delta w_{ij} = \eta\delta_j x_i \quad \text{(backpropagation)}$$

Strengths:

1. *Pattern learning*: Automatically extracts features
2. *Generalization*: Produces novel outputs similar to training data
3. *Non-linear relationships*: Captures complex dependencies
4. *Adaptive*: Can continue learning with new data

Weaknesses:

1. *Black box*: Difficult to interpret learned knowledge
2. *Training data dependency*: Quality depends on corpus
3. *Overfitting risk*: May memorize rather than generalize
4. *Architecture design*: Network topology affects results

Musical Application Example:
Bach chorale harmonization (HARMONET):

- Input: Soprano melody + current chord context
- Hidden layers: 2 layers, 50 neurons each + context units
- Output: Next chord (root, inversion, doubling)
- Training: 150 Bach chorales
- Result: "Level of an improvising organist"

Architecture details:

- Context units remember recent chords (provides memory)
- Separate networks for chord selection and voice assignment
- Final network adds passing tones and ornaments
- Multi-stage pipeline mimics compositional process

Relation to Human Composition:
Similar to:

- Implicit learning through extensive listening/study
- Internalized style knowledge (”knowing more than we can tell”)
- Intuitive decision-making based on experience
- Pattern matching against mental models

Different from:

- Humans use explicit theoretical knowledge alongside intuition
- Composers consciously manipulate abstract musical concepts
- Human creativity involves intentional rule-breaking
- Neural networks lack ”understanding” of music meaning

COMPARATIVE SUMMARY

Aspect	**Markov Chains**	**Genetic Algs**	**Neural Nets**
Learning Method	Analyze transition frequencies	Evolve through selection	Learn from examples via backprop
Memory	Short-term (N previous states)	None (each individual independent)	Long-term (in weights) + short (context units)
Planning	None (reactive)	Implicit (fitness guides evolution)	None directly (learned patterns)
Novelty	Low-moderate (recombines training material)	High (crossover creates new combinations)	Moderate (interpolates training space)
Interpretability	High (transition probabilities clear)	Moderate (can inspect individuals)	Low (distributed weights)
Best For	Style imitation, improvisation	Optimization, exploration	Pattern recognition, style transfer
Computational Cost	Low	High (many generations)	High (training), Low (use)

PHILOSOPHICAL IMPLICATIONS

Each method reflects different aspects of musical creativity:

Markov Chains model the reactive, moment-to-moment aspect of improvisation—similar to jazz musicians responding to what was just played.

Genetic Algorithms model the evolutionary aspect of cultural musical development—styles evolve as successful innovations are preserved and combined.

Neural Networks model the implicit, embodied knowledge musicians develop through extensive practice and listening—style internalized at a sub-symbolic level.

However, none fully captures human composition because:

- All lack genuine understanding of musical meaning
- Missing teleological thinking (goals, intentions, narratives)

- No emotional or semantic content
- Cannot truly innovate beyond training/fitness constraints
- Lack consciousness and intentionality

CONCLUSION
These three methods are complementary:

- **Markov chains** excel at local coherence and style mimicry
- **Genetic algorithms** excel at global optimization and exploration
- **Neural networks** excel at learning complex patterns and generalizing

The most sophisticated algorithmic composition systems often combine multiple techniques, leveraging the strengths of each to overcome individual limitations. The future may involve hybrid systems that integrate rule-based knowledge, statistical learning, evolutionary search, and perhaps even models of musical semantics and emotion.

7 Conclusion

Unit 5 has explored the fascinating intersection of mathematics, computation, and musical creativity through algorithmic composition and perception. We have examined how mathematical models can both generate music and explain how we perceive it.

7.1 Key Insights

1. **Fractals and Self-Similarity**: Musical structures exhibit self-similar patterns across multiple time scales. The 1/f spectral characteristic found in much music represents an optimal balance between entropy and redundancy, maintaining listener interest through balanced predictability and surprise.
2. **Stochastic Methods**: Markov chains and other probabilistic techniques can capture musical style through analysis of existing works, though they lack the long-term planning characteristic of human composition.
3. **Information Theory**: Musical meaning emerges from the manipulation of expectation, which depends on the balance between information (entropy) and redundancy. Shannon's entropy provides a mathematical framework for understanding how music maintains interest.
4. **Psychoacoustics**: The perception of music is fundamentally a mathematical and neurophysiological process. Critical bands explain consonance and dissonance, while phenomena like the missing fundamental demonstrate that pitch is a cognitive construct based on pattern recognition.
5. **Computational Intelligence**: Neural networks and genetic algorithms offer powerful approaches to learning musical style and optimizing musical structures, though they operate quite differently from human compositional processes.
6. **Generative Systems**: L-systems, cellular automata, and chaotic dynamics provide ways to create musical structures with complexity emerging from simple rules.

7.2 Connections Across Topics

The unit reveals deep connections:

- *Fractals bridge composition and perception*: Self-similar structures in composition (1/f noise) correspond to optimal processing in perception (critical bands with constant-Q characteristics)
- *Information theory unifies style and perception*: Entropy explains both why certain compositional patterns work and how the brain processes musical information
- *Algorithms model both creation and cognition*: Computational methods can generate music and model aspects of musical understanding

7.3 Limitations and Future Directions

While mathematical and computational approaches have illuminated many aspects of music, fundamental questions remain:

- Can algorithmic systems achieve genuine creativity, or only sophisticated imitation?
- How can we model the semantic and emotional content of music mathematically?
- What role does consciousness play in musical composition and perception?
- Can we develop metrics that truly capture musical quality and aesthetics?

Future research directions include:

- Integration of multiple AI techniques (hybrid systems)
- Incorporation of musical semantics and narrative structure
- Real-time interactive composition systems
- Better models of long-term musical planning and structure
- Cross-cultural studies of universal vs. learned aspects of perception

7.4 Final Reflection

The study of algorithmic music and perception demonstrates that while mathematics can describe and even generate musical structures, the essence of music may transcend purely formal description. The most profound musical experiences involve emotional, cultural, and personal dimensions that resist reduction to algorithms.

However, by understanding the mathematical foundations of music—from the fractal structures in composition to the neural processing in perception—we gain deeper insight into this most human of art forms. The interplay between order and disorder, expectation and surprise, pattern and variation reveals music as a sophisticated mathematical dialogue between composer, performer, and listener.

As technology advances, the boundary between human and algorithmic composition continues to evolve. The tools developed through mathematical music research enhance human creativity rather than replace it, offering new possibilities for musical expression while deepening our understanding of how and why music moves us.

A Quick Reference Formulas

A.1 Fractal Mathematics

Fractal Dimension:

$$D = \frac{\ln s}{\ln L}$$

Weierstrass Function:

$$W(t) = \sum_{n=0}^{N-1} r^{nH} \cos(2\pi r^n t)$$

Brownian Noise Power Spectrum:

$$P(f) \propto \frac{1}{f^\alpha}, \quad \alpha \in [0, 2]$$

A.2 Information Theory

Shannon Entropy:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i) \text{ bits}$$

Maximum Entropy:

$$H_{\max} = \log_2 N$$

Redundancy:

$$R(X) = 1 - \frac{H(X)}{\log_2 N}$$

A.3 Psychoacoustics

Critical Bandwidth:

$$BW_c(f) = 25 + 75 \left[1 + 1.4 \left(\frac{f}{1000}\right)^2\right]^{0.69} \text{ Hz}$$

Bark Number:

$$z(f) = 13 \arctan(0.00076f) + 3.5 \arctan \left(\frac{f}{7500}\right)^2$$

Quality Factor:

$$Q = \frac{f_c}{BW}$$

Acoustical Uncertainty Principle:

$$\Delta f \cdot \Delta t = k$$

A.4 Neural Networks

Activation:

$$y_j = f\left(\sum_i w_{ij}x_i + b_j\right)$$

Error:

$$E = \frac{1}{2}\sum_k (t_k - y_k)^2$$

Weight Update:

$$\Delta w_{ij} = \eta\delta_j x_i$$

A.5 Markov Chains

Transition Probability:

$$P(X_n = j|X_{n-1} = i) = p_{ij}$$

Row Constraint:

$$\sum_{j=1}^{n} p_{ij} = 1$$

B Glossary of Key Terms

Algorithm A finite sequence of well-defined instructions for accomplishing a task

Artificial Neural Network Computational model inspired by biological neural networks

Bark Scale Psychoacoustic scale dividing hearing into 24 critical bands

Critical Band Frequency bandwidth within which sounds interact perceptually

Entropy Measure of uncertainty or information content in a system

Fitness Function Evaluation metric for genetic algorithm solutions

Fractal Geometric object exhibiting self-similarity at different scales

Generative Music Music created by systems rather than fixed compositions

Genetic Algorithm Optimization technique based on natural selection principles

L-System Parallel rewriting system for generating complex structures

Markov Chain Stochastic process with memory of previous state(s)

Psychoacoustics Study of psychological and physiological responses to sound

Redundancy Measure of predictability or pattern in information

Self-Similarity Property of appearing similar at different scales

Stochastic Involving randomness or probability

Tonotopic Relating to spatial organization by frequency

References and Further Reading

1. Benson, D. J. (2006). *Music: A Mathematical Offering*. Cambridge University Press.

2. Loy, G. (2006). *Musimathics: The Mathematical Foundations of Music* (Vols. 1 & 2). MIT Press.

3. Fauvel, J., Flood, R., & Wilson, R. (Eds.). (2003). *Music and Mathematics: From Pythagoras to Fractals*. Oxford University Press.

4. Mandelbrot, B. B. (1977). *The Fractal Geometry of Nature*. W. H. Freeman.

5. Shannon, C. E., & Weaver, W. (1949). *The Mathematical Theory of Communication*. University of Illinois Press.

6. Meyer, L. B. (1956). *Emotion and Meaning in Music*. University of Chicago Press.

7. Voss, R. F., & Clarke, J. (1978). "1/f Noise in Music: Music from 1/f Noise." *Journal of the Acoustical Society of America*, 63(1), 258-263.

8. Zwicker, E., & Fastl, H. (1990). *Psychoacoustics: Facts and Models*. Springer-Verlag.

9. Roads, C. (1996). *The Computer Music Tutorial*. MIT Press.

10. Miranda, E. R. (2001). *Composing Music with Computers*. Focal Press.

11. Nierhaus, G. (2009). *Algorithmic Composition: Paradigms of Automated Music Generation*. Springer.

12. Peitgen, H. O., & Richter, P. H. (1986). *The Beauty of Fractals*. Springer-Verlag.

13. Hiller, L., & Isaacson, L. (1959). *Experimental Music*. McGraw-Hill.

14. Cope, D. (2001). *Virtual Music: Computer Synthesis of Musical Style*. MIT Press.