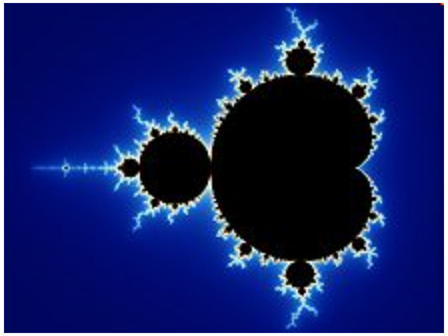


Algorithmic art and music composition

Ch 8.4

Generative and algorithmic art

- Generative art refers to art that has been
  - generated, composed, or constructed in an algorithmic manner
  - through the use of
    - systems defined by computer software algorithms,
    - or similar mathematical or mechanical or randomized autonomous processes
- Algorithmic art:
  - the creative design is the result of an algorithmic process
  - It is usually digital art
- The artist's self-made algorithms are
  - an integral part of the authorship,
  - a medium through which their ideas are conveyed.

The role of the algorithm

- Creation must include a process based on an algorithm devised by the artist.
  - detailed recipe for the design and possibly execution of an artwork

- Input may be mathematical, computational, or generative in nature
  - some random factor is usually introduced
  - some artists also work with gestural input which is then modified by an algorithm.

Algorithms for music composition

- Modeling traditional, non-algorithmic compositional procedures.
- Modeling new, original compositional procedures, different from those known before.
- Selecting algorithms from extra-musical disciplines.

- Approaches
  - Integrated tools and languages that will cover all possible composing desiderata;
  - Personalized micro-programs written in small languages

Algorithmic processes

- Mathematical models
  - Markov processes
  - Stochastic processes
  - Fractals

Algorithmic processes

- **Algorithm** `GenerateAndTest`

  while composition is not terminated
    generate raw materials
    modify according to various functions
    select the best results according to rules

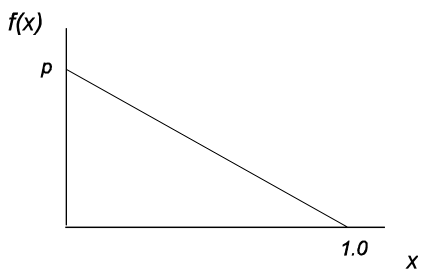
- **Algorithm** `RandomWalk`

  Get events distribution by analysing a music repertoire
  while composition is not terminated
    sample a random event from the distribution of events
    add to the piece

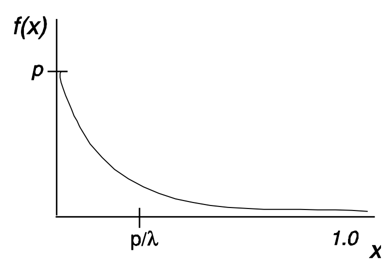
Stochastic - Probabilities

- Statistical distributions have been used which can be thought of as a means of constraining random processes.

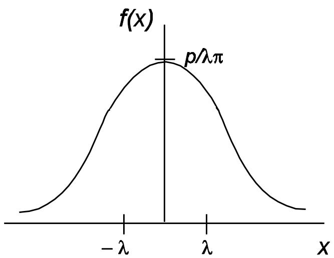
- For an even distribution function, probability P of event E occuring is the ratio between no of events in E and no of all possible events S.
- E.g. for a dice:
  - S = {1, 2, 3, 4, 5, 6}
  - If E is 'get a number higher than 3', then E = {4,5,6}
  - So P(e) = 3 / 6 = 0.5
- Distribution functions can be defined to make certain events more or less likely



Linear



exponential



concave

Early examples 2

- Iannis Xenakis (1971) – Stochastic composition.
  - eg Morsima-Amorsima for 4 instruments
- Approach
  - Stochastic music programme “deduced” a score from a list of note densities and probabilistic weights defined by the composer.
- Aims
  - Development of a mathematical language for describing and manipulating music.
  - First to conceive of a composing program as a utility capable of generating many pieces, rather than achieving one specific compositional goal.

Stochastic - Markov Chains

- Conditional probability system where probability of future events is defined by past events.
- Applied in music e.g. for probabilistic construction of melodic lines in tonal music
- No of events taken into consideration defines the 'order' of the chain
  - I.e. first order takes 1 previous event into consideration, n-th order considers n previous events.
- Represented by a transition probability matrix of n+1 dimensions

Input notes and probability matrix for 'Mary had a little lamb'

Set of input notes:

E D C D E E E D D D E G G E D C D E E E E D D E D C

Transition Probability Matrix:

Next Event

Current Event	C	D	E	F	G
C		2/3	1/3		
D	3/10	3/10	4/10		
E		5/11	5/11		1/11
F					
G			1/2		1/2

Example

- Miranda's CAMUS 3D uses markov chains to control rhythm and temporal organisation of note groups
- Several of Cope's systems are based on nth order Markov chains

Advantage

- Captures note-by-note structure of input in short term

Disadvantage

- Lacks long term structure

Markov processes

- HMM as music generator

1. Choose initial state $x(1) = S_1$ according the initial state distribution π.
2. Set $t = 1$
3. Choose $o(t)$ according the symbol probability distribution in state $x(t)$ described in matrix $\mathbf{B}$
4. Transit to new state $x(t+1) = S_j$ according to the state transition probability for state $x(t) = i$, i.e. $a_{i,j}$
5. Set $t = t + 1$ and return to step 2

- Markov chain
  - Without step 3 and output *x(t)*

Example Markov chain

- ”Hymn tunes” generated by computer from an analysis of the probabilities of notes occurring in various hymns. (1957)

Disadvantages of stochastic processes

- Someone needs to find the probabilities by analyzing many pieces.
  - The resulting models will only generate music of similar style to the input.
- For higher order Markov models, transition tables become unmanageably large for the average computer.
- The deviations from the norm and how they are incorporated in the music is an important aspect.
- They also provide little support for structure at higher levels

Knowledge based systems

- Many early systems focused on taking existing musicological rules and embedding them in computational procedures.
  - systems which are symbolic and use rules or constraints.

- Advantages
  - have explicit reasoning;
  - can explain their choice of actions.
- Disadvantages
  - Knowledge elicitation is difficult and time consuming,
  - depend on the ability of the ”expert”
  - become too complicated if we try to add all the ”exceptions to the rule and their preconditions,

Grammars

- The idea that there is a grammar of music
- Linguistics is an attempt to identify how language functions:
  - what are the components,
  - how do the components function as a single unit,
  - how do the components function as single entities within the context of the larger unit.
- Generative grammar
  - formal system of principles or rules which describes (or ’generates’) the possible sentences of the language.

- Composition algorithms: based upon more abstract representations of musical structures.

KBS 2 - Grammars – methods

- A grammar is a collection of prescriptive and/or descriptive rules for analysing or generating sequences of symbols
  - can be applied to natural and computer languages ... and musical symbols
  - used computationally to define valid symbol strings

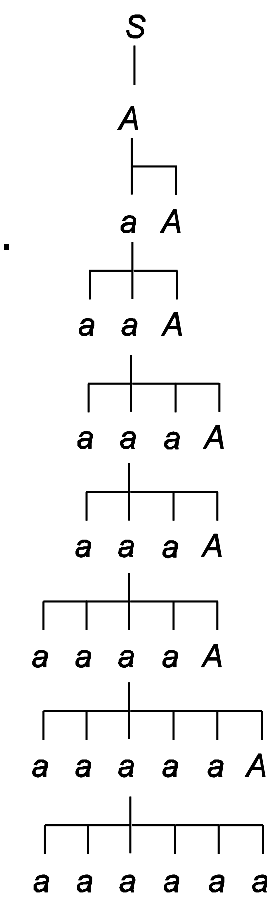
Example of simple grammar:

S -> A | B
A -> aA | a
B -> bB | b

Where:

Hi case = non terminal node
lo case = terminal node
S = start point
| = or
-> = 'is defined as'

so possible sequence =
S => A => aA => aaA => aaaA => aaaaA => aaaaa



Tree representation of grammar

Grammars – natural language example

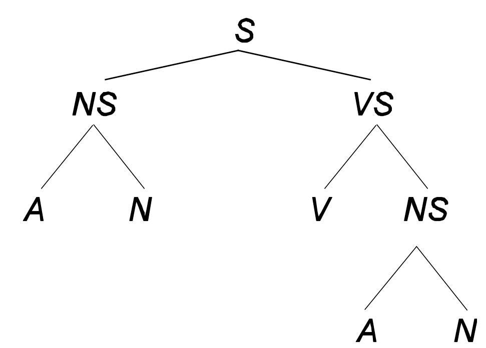
- If lexicon of possibilities is provided, can be used to generate syntactically correct sentences.
- Simple lexicon for natural language:
  - A = {the, a , an}
  - V = {composes, makes, hears}
  - N = {dog, computer, music, musician, coffee}
- Does not ensure semantic sense in linguistics (“the coffee hears the dog” is equally valid).

grammar:

S -> NS + VS
NS -> A + N
VS -> V + NS

Tree representation of

‘a musician composes the music’



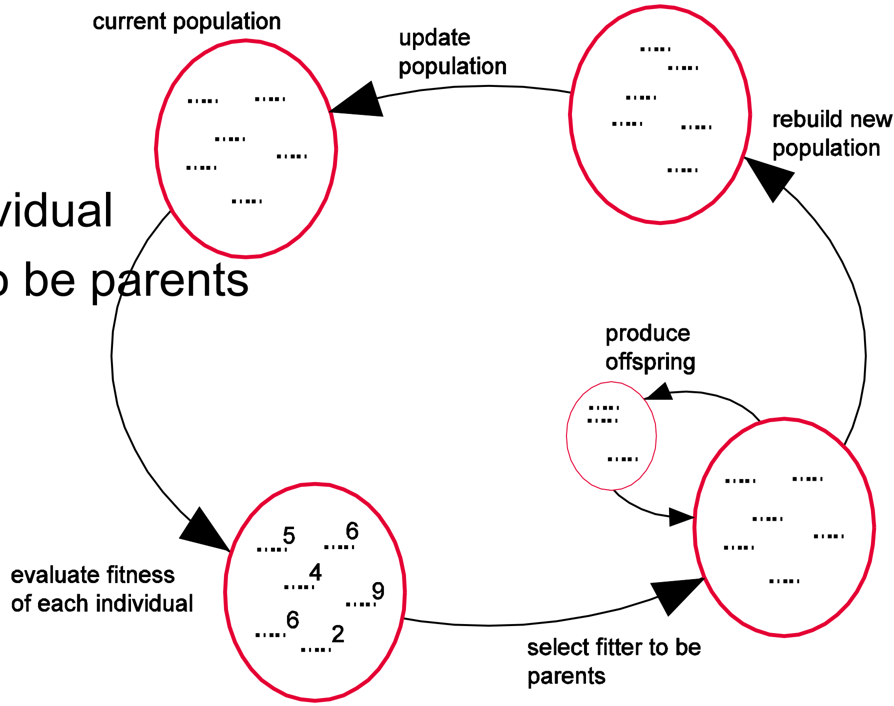
Grammars: problems

- They are hierarchical structures while much music is not (i.e. improvisation).
  Therefore ambiguity might be necessary since it ”can add to the representational power of a grammar”.
- Most, if not all, musical grammar implementations do not make any strong claims about the semantics of the pieces.
- Usually a grammar can generate a large number musical strings of questionable quality.
- Parsing is, in many cases, computationally expensive especially if we try to cope with ambiguity.

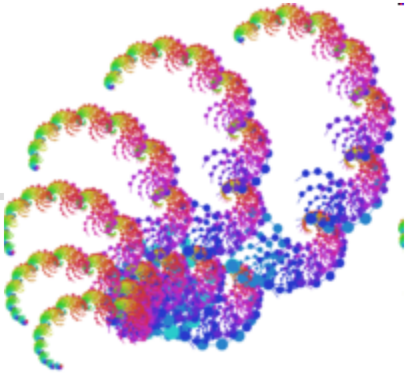
Evolutionary methods

- Genetic algorithms (GAs)
  - very efficient search methods, especially when dealing with problems with very large search spaces.
  - ability to provide multiple solutions

- Algorithm **`GeneticAlg`**
  - Initialize population
  - while not finished evolving
    - Calculate fitness of each individual
    - Select preferred individuals to be parents
    - for N != populationSize
      - Breed new individuals
      - (cross over + mutation)
    - Build next generation
  - Render output



Evolutionary methods



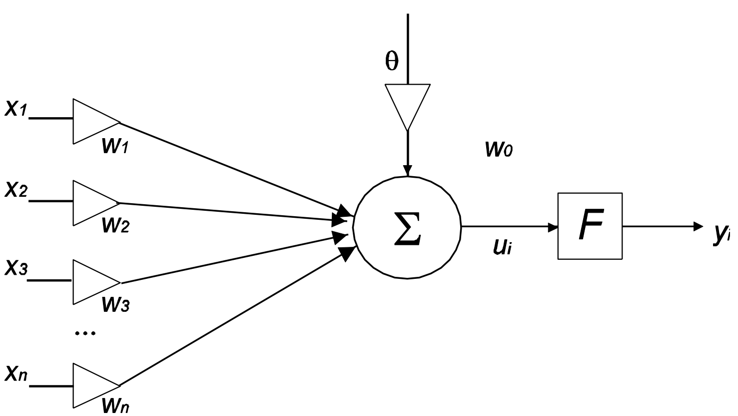
- Fitness function
  - Use of an objective fitness function.
  - Use of a human as a fitness function. ➔ Interactive GA
- Mutation ensures new individuals are introduced into the population
- Reproduction operator (cross-over) mixes 'good' solutions of 2 parents

- Drawback
  - Subjectivity
  - Efficiency, the ”fitness bottleneck, boredom effects

  - this approach tells us little about the mental processes
  - Often: very simple representations

Learning systems

- Learning systems,
  - do not have a priori knowledge (e.g. production rules, constraints) of the domain,
  - but instead learn its features by examples.
- Stored information
  - subsymbolic/distributive (Artificial Neural Networks, ANN)
  - symbolic (Machine Learning, ML).



Schematic of a simple artificial neuron

- Inputs (**x**) are weighted (**w**) (+ve or -ve)
  - Weighted inputs are summed

$$u_i = \Sigma\ w_{in}\, x_n$$

  - And passed through an activation function
    $y_i = F(u_i)$
  - Bias term (θ) models sensitivity/ resistivity

Artificial Neural Networks – training

- Three basic types
  - Supervised learning
    - Requires training data of input-output sets.
    - Error correction mechanism used. Weights altered to minimise discrepancy between actual and desired output
  - Reinforcement learning
    - Requires training data of input-output sets.
    - Binary (right-wrong) feedback given
  - Unsupervised learning
    - Input-output sets not required
    - Network self-organises

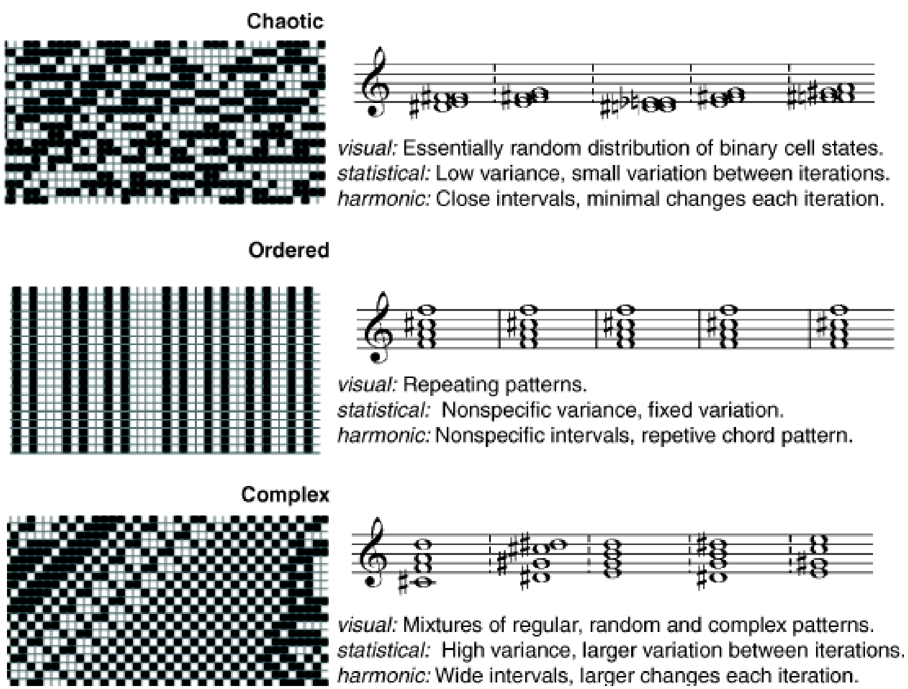
Learning systems: Subsymbolic

- Composition as compared with cognition is a much more highly intellectual process (more ”symbolic”).
  - successfully capture the surface structure of a melodic passage
  - they mostly fail to pick up the higher-level features of music
- The representation of time can not be dealt efficiently even with ANNs which have feedback.
- Usually they solve toy problems, with many simplifications,
- Need to avoid training set examples which conflict.

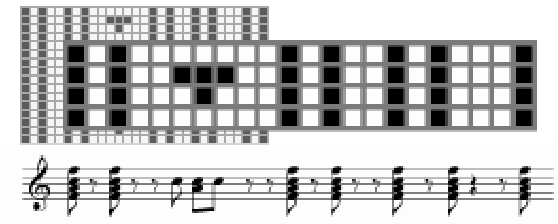
- Learn from examples things which can’t be represented symbolically using rules (i.e. the ”exceptions”)

'Extramusical' approaches

- Models that exhibit musically-relevant dynamics or structures
  - a more experimental, or metaphorical approach.
  - Rather than learning directly from existing music, or attempting to formalise musicological concepts,
- E.g. Cellular automata



Harmonic mappings



Rhythmic mappings

Extramusical

- Rationale
  - Structures in certain mathematical systems seen to model musical structures/ compositional or performance processes.
- Advantages
  - No need for explicit formalisation of musical ‘rules’
  - Encourages exploration of ‘new’ music rather than attempts to recreate existing music
- Disadvantages
  - Arguably difficult for listeners to engage with as no cultural reference.
  - Relies heavily on selecting good mappings (ie defining which musical parameters are controlled)

References

- George Papadopoulos and Geraint Wiggins. *Ai methods for algorithmic composition: A survey, a critical view and future prospects*. In Proceedings of the AISB'99 Symposium on Musical Creativity, 1999.
  `www.doc.gold.ac.uk/~mas02gw/papers/AISB99b.pdf`