Computer Science Workshop (AINT101): Computer-Aided Composition (2)

Torsten Anders
Interdisciplinary Centre for Computer Music Research (ICCMR)
University of Plymouth
http://cmr.soc.plymouth.ac.uk/

University of Plymouth, AINT101 Lectures d,
27 November 2007

Outline

Markov Chain Introduction

Pitches from a well-known song

$[d4, e4, d4, g4, fs4,$
$d4, e4, d4, a4, g4,$
$d4, d5, b4, g4, fs4, e4,$
$c5, b4, g4, a4, g4]$

Task

Write a list: which pitch can be followed by which pitch in the pitch sequence above.

Markov Chain

- Stochastic model expressing ‘what follows what’
- Used in algorithmic composition since the 50s.
  Examples: Foster songs model (Olson), Illiac suite (Hiller & Issacson)
- A Markov chain can be freely created by hand
- Alternatively, it can be an analysis of a given value sequence (analysis/synthesis model)

Markov Chain Example I

Second order Markov chain for "Happy Birthday" (created with Common Music, probabilities left out)

$a4, g4 \longrightarrow d4$

$b4, g4 \longrightarrow a4|fs4$

$c5, b4 \longrightarrow g4$

$d4, a4 \longrightarrow g4$

$d4, d5 \longrightarrow b4$

$d4, e4 \longrightarrow d4$

$d4, g4 \longrightarrow fs4$

$d5, b4 \longrightarrow g4$

$e4, c5 \longrightarrow b4$

$e4, d4 \longrightarrow a4|g4$

$fs4, d4 \longrightarrow e4$

$fs4, e4 \longrightarrow c5$

$g4, a4 \longrightarrow g4$

$g4, d4 \longrightarrow d5|e4$

$g4, fs4 \longrightarrow d4|e4$

Markov Chain Example II

Example

Little melody created from “Happy Birthday” Markov chain with Common Music

Advanced Use of Markov Chains

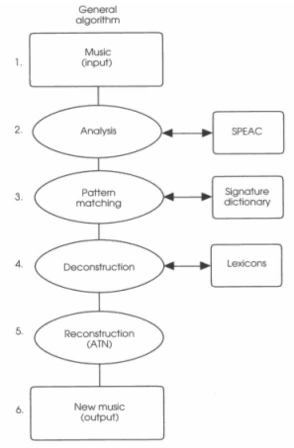
- Addressing multiple parameters (requires many Markov chain clauses)
- Composite musical segments (e.g., motifs, chords) are units (cf. Mozart's Würfelspiel)
- Markov chains arranged in a hierarchy (e.g., some Markov chain creates sequence of musical segments, and the notes of these segments are again created with a Markov chain)

David Cope: EMI

Example

David Cope's EMI. *5000 Bach Chorales, No. 1*

The overall structure of the EMI algorithm I



(Cope, 1996, p. 27)

The overall structure of the EMI algorithm II

- SPEAC analysis: tags denote what musical function each segment (e.g., each bar) serves (SPEAC means Statement, Preparation, Extension, Antecedent, Consequent)
- Signature pattern matching: detection of characteristic musical segments (which may be varied)
  - Composer specific phrases
  - Earmarks: markers of formal structure
  - Unifications: piece-specific segments
- Reconstruction: newly combining musical segments so that result complies with SPEAC restrictions, and places signatures appropriately

SPEAC analysis of a Bach Chorale (Cope, 2001, p. 132)

Versions of a Mozart signature (Cope, 1996, p. 149)

David Cope: EMI

EMI can create compositions in many different musical styles.

Example

David Cope's EMI. *Chopin Mazurka*

What is Constraint Programming?

Definitions

A *constraint satisfaction problem* (CSP) states *constrains* (mathematical relations) between *variables* (unknowns) with a specific *domain* (a set of possible variable values). Constrains specify properties of a solution, which is found by search.

CSP example

$$X + Y = 7$$
$$\wedge\ X < Y$$
$$\text{where } X \in \{1, \ldots, 10\} \wedge Y \in \{1, \ldots, 10\}$$

possible solution: $X = 3$, $Y = 4$

Advantages of Constraint Programming

Advantages

- Complex problems simple to model: model states only *what*, not *how*
- There exist efficient solvers to search for solution(s)

Music Constraint Programming

Definition

A *musical constraint satisfaction problem* is a set of rules (constraints) applied to a music representation where some aspects are unknown (variables).

- Executable implementation of a music theory model: the user states a music theory and the computer generates music which complies with this theory
- Declarative definition: similar to the way music theory is traditionally expressed: by a set of rules (although a musical CSP is more formal)

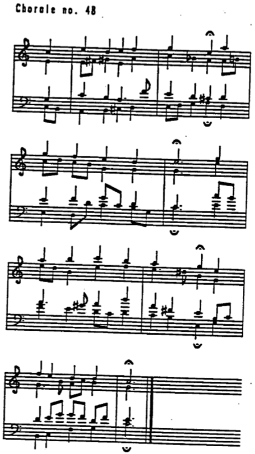
Constraint Programming: Modeling an Existing Style I

Example

Kemal Ebcioglu. *Composition system CHORAL*

- System creates four-part harmonisations of given choral melody in the style of Bach
- Implements about 350 rules
- Two subtasks
  - Harmonisation: creating of chord skeleton, style-appropriate modulation (!) and cadencing
  - Melody generation, with special care of the outer voices
- System implemented in constraint programming language BSL (Backtracking Specification Language), designed for CHORAL

Constraint Programming: Modeling an Existing Style II



CHORAL example output, transposed into C-major [Ebcioglu, 1993]

Constraint Programming: Modeling New Music

Example

Örjan Sandred. *Whirl of Leaves*

We now study the rule set for one section of the piece (A') with slides kindly provided by the composer.

Combination of Techniques

- Algorithmic composition provides a large toolbox of techniques – we touched upon a few
- Each technique can be applied in many ways
  Example: a Markov chain can be created by hand or by analysis, and can be applied to various musical parameters
- Different techniques can be combined freely

Example: combining the techniques Lindenmayer-system (a fractal) and constraint programming

`http://strasheela.sourceforge.net/strasheela/doc/Example-HarmonisedLindenmayerSystem.html`

Different Musical Levels

Algorithmic composition can address different musical levels

- Note level: creation of individual notes, an approach often applied
- Macro level: creation of formal building blocks
  Examples: Mozart's Würfelspiel, Cope's EMI, example combining L-system and constraint programming
- Micro level: controlling sound features (examples below)
- Different levels can be addressed at the same time (see example combining L-system and constraint programming which combines note level and macro level)

Micro Level: Generation of Control Data for Sound Synthesis

Example

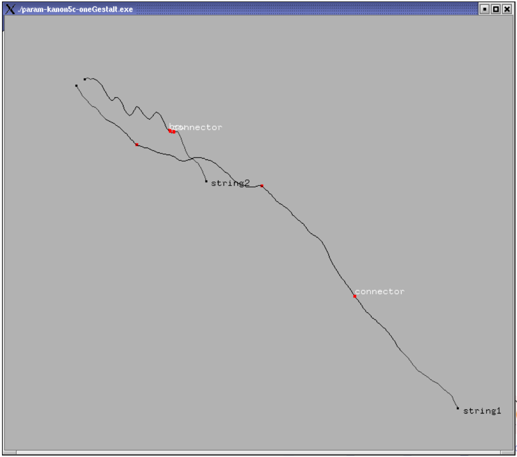
Torsten Anders. *With Shifting Joints*

The following slides explain the approach taken in *With Shifting Joints*.

Design of Physical-Modeling-Instrument (with Software Tao)

Virtual instrument example

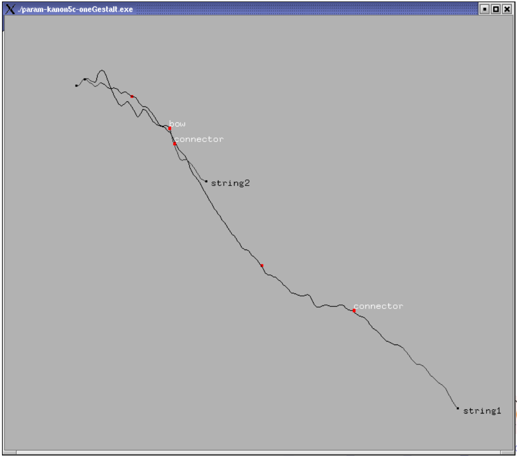
- Two vibrating strings
- First string is bowed
- Second string is connected to contact microphone
- Both strings coupled by a spring (!)



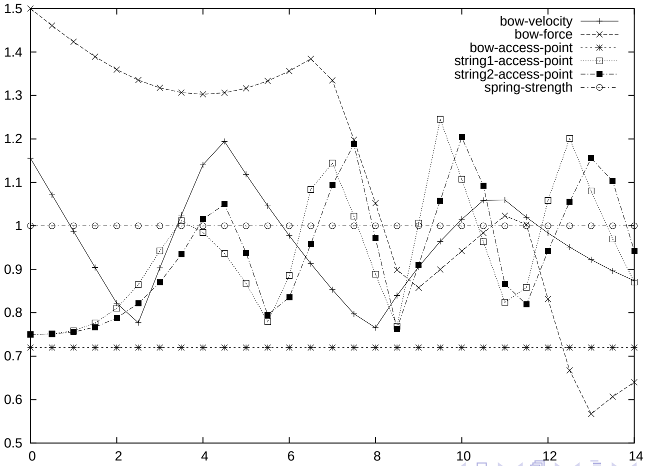
Design of Physical-Modeling-Instrument (with Software Tao)

Virtual instrument example

- Two vibrating strings
- First string is bowed
- Second string is connected to contact microphone
- Both strings coupled by a spring (!)



"Score"-Excerpt for this Instrument



A Performance Programming Language

- Algorithmic composition created with programming language for expressing music performances (designed for this piece)
- Low-level statements (original Tao statements)
  - Performance statements. Example
    `pluck my-string <where> <how-strong>`
  - Control structures. Example
    `at <time> do <statement>`
- High-level statements, abstracting several low-level statements
  - Numeric functions used as envelopes: rich library for generating, combining and transforming envelopes
  - Higher-level performance statements: parameters can be envelopes
  - Hierarchic music representation

Micro Level: Algorithmic Sound Processing

Example

Paul Lansky. *Smalltalk*

- Material is recording of conversation
- Recording is analysed (e.g., extraction of pitches, rhythm, and contours) – analysis expresses micro level
- The analysis information together with the original recording used for further processing
- Software: Cmix

Realtime Composition

Example

Karlheinz Essl. *Lexikon-Sonate*

`http://www.essl.at/works/Lexikon-Sonate.html`

- Based on serial composition and random processes
- Note use of tempo, dynamics, and pauses – creates a lively performance

Summary

- Computer-aided composition (CAC): musical intentions are formalised, and implemented in computer program
- Formalisation of various musical aspects has long tradition in music composition
- CAC can apply both musical knowledge (e.g., generated by music analysis, or encoded 'by hand') and abstract processes (e.g., mathematical models such as fractals)
- CAC can be applied to various musical levels (micro level, note level, macro level)