Musical Metacreation
AI for Generative Music

Philippe Pasquier
Associate Professor
SIAT, SFU, Vancouver, Canada.
About me
About Us

• International Workshop on Musical Metacreation:
  – MUME 2012@AAAI AIIDE - Stanford U.
  – MUME 2013@AAAI AIIDE - Northeastern U.
  – MUME 2014@AAAI AIIDE - Univ. North Carolina
  – MUME 2016@ICCC - Paris
  – MUME 2017@ICCC – Georgia Tech
  – MUME 2018@ICCC - Salamanca

• Generative Music Concerts:
  – MUME-WE 2013@ISEA2014, Sydney
  – MUME-WE 2014@NIME2014 (Oto), London
  – MUME-WE 2015@ISEA2015, Vancouver
  – MUME@ICCC2017, Georgia Tech
  – MUME@ICCCC2018, Salamanca

• Records / releases / performances

• Tutorials:
  – MUME-tut@NIME2015
  – MUME-tut@IJCAI2015
About you

- Computer scientists?
- AI, ML?
- Computer music?
- Musicians?
Outline of the Tutorial

Part I: "An Introduction to MuMe" (1.5 hours)

- Name that MuMe: Introduction to Musical Metacreation
- MuMe and Variation: Classification, Ontology, and History

[Coffee Break, 30 mins]

Part II: "MuMe Systems and Evaluation" (1.5 hours)

- Fruits of the MuMe: Current approaches, including Evolutionary Computation, Neural Networks, Multi-agent Systems
- A Kind of MuMe: Evaluation of MuMe Systems, Past, Present and Future
- MuMe Over: Critical discussion of societal issues
Name that MuMe
Introduction to Musical Metacreation

Philippe Pasquier
Associate Professor
School for Interactive Arts + Technology,
Simon Fraser University
Artificial Intelligence

Artificial intelligence is the science of having machine solve problems that do require intelligence when solved by human.

Adapted from Simon (1960).
AI is Ubiquitous

- AI has been tremendously successful at rational problem solving (optimality is well defined).
  - AI systems:
    - Fly planes (goals),
    - Regulate nuclear plants (constraints),
    - Design electric circuits (objective function),
    - Automated negotiation (maximizing utility function and finding Pareto dominant solutions),
    - Diagnose diseases (probability distribution)
    - Play chess (win/lose)
    - Play Jeopardy (good/bad answer).
    - ...
  - The list is seemingly endless, but can machines be creative?
Computational Creativity

• Computational Creativity is a new and fast growing scientific field that is exploring the partial or complete automation of creative processes.

• A.k.a artificial creativity: endowing machines with creative behaviors.

• As a field, it investigates:
  – creativity as it is: striving to understand and simulate human creativity (cognitive science)
  – creativity as it could be: processes that we know humans to be incapable of (at least without machines).
Computational creativity departs from AI when the notion of optimality is ill-defined:
- No definitive answer, goal states, Pareto dominance, objective function, utility function, preference relations, ...

Creative tasks as those for which there is no clear “best” outcomes.
- No such thing as the best design, choreography, music composition, interpretation of a piece, level for a video game, drawing, painting, narrative, poetry, joke, ...
Metacreation

"the use of an autonomous system for art making" Philip Galanter (2003)

Scientific domain that focuses on the modeling and study of computational processes that achieve creative tasks.
MUME: Musical Metacreation

- Partially or completely automate musical creative tasks:

Artistic / Specific

Generative Music

Musical Metacreation

Simulation of Musical Creativity

Generic / Scientific
Generative Music as Art

• Generative Music 1 (1996) was released as a floppy disk in 1996 by Brian Eno

• Icarus (Ollie Bown and Sam Britton)  
  An album in 1000 variations (2012)  
  Fake Fish Distribution
Interdisciplinary MUME

Musicology
Philosophy
Sociology
Design

Generative Arts
Interactive Arts

Computer Music,
Digital Signal Processing

Music (cognition,
composition,
interpretation,
production)

Artificial Intelligence,
Artificial Life
Machine Learning

Musical Metacreation

Metacreation Lab
SFU
MuMe and Variations
Classification, Ontology, and History

Philippe Pasquier
Associate Professor
School for Interactive Arts + Technology,
Simon Fraser University
Simulation of Musical Creativity

• Partially or completely automate musical creative tasks.
• What do we mean by creative/creativity?
Defining Creativity

• **Creativity** is the ability to come up with ideas or artifacts that are **original, and valuable** (adapted from Margareth Boden, 2004)

• **P-creativity**: psychological creativity (novel and valuable for the individual), a.k.a mundane or everyday creativity

• **H-creativity**: historical creativity (novel and valuable for the group, i.e. humanity), a.k.a eminent creativity
Defining Creativity

• Three types of creativity (Boden, 2006):
  1. Exploratory creativity
Defining Creativity

• Three types of creativity (Boden, 2006):
  1. Exploratory creativity
  2. Combinatorial
Defining Creativity

• Three types of creativity (Boden, 2006):
  1. Exploratory
  2. Combinatorial
  3. Transformational
MUME Problems

• Partially or completely automate musical creative tasks
• Musical Metacreation addresses a variety of problems:
  – Classic cognitive science and computer music issues:
    – Music perception, recognition, classification
    – Music representation
    – Music cognition
  – Composition: generating a score
  – Interpretation: audio rendering of a composition
  – Improvisation: composition and interpretation
  – Accompaniment: playing along with a composition or an interpretation/improvisation
  – Continuation: taking over when interpretation/composition stops.
There are (too) many MuMe problems...

- Harmonic progressions [Eigenfeldt and Pasquier 2010; Whorley et al. 2010; Groves 2013; Manaris et al. 2013; Pachet and Roy 2014];
- Rhythm generation [Eigenfeldt 2008; Chordia and Rae 2010];
- Melodic generation [Bosley et al. 2010; Sarwate and Fiebrink 2013];
- Orchestration [Handelman et al. 2012];
- Harmonization [Pachet and Roy 2001; Simon et al. 2008; Pachet and Roy 2014];
- Affective interpretation [Kirke and Miranda 2009];
- Affective composition [Birchfield 2003; Wallis et al. 2011; Eigenfeldt et al. 2015];
- Automatic mixing [Reiss and Perez Gonzalez 2008; Reiss 2011];
- Soundscape composition [Eigenfeldt and Pasquier 2011; Thorogood et al. 2012].
- Automatic Mastering
Level of autonomy

Purely reactive systems (no autonomy, no pro-activity)

• Many systems are interactive.
• Enable computer-assisted creativity, creativity support tools, computer-assisted composition, ...

Purely generative (On/Off)
Musical Metacreation

• Characteristics of the systems:
  – Music representation: symbolic vs. audio signal
  – Online: various levels of real-time
  – Offline: generated ahead of time (the generation itself can occur slower or faster than real-time)
  – Corpus-based: the system has been exposed to music (symbolic notation or audio signal).
  – Non corpus-based: generated from scratch
Computational Creativity

• **Style imitation:** Given a corpus $C = \{C_1, \ldots, C_n\}$ representative of style $S$, generate new instances that would be classified as belonging to $S$ by an unbiased observer.

• The Metacreation Lab produce corpus-based style machines:
  – style imitation,
  – style interpolation,
  – style combination,
  – style transformation,
  – style extrapolation,...
Typology of MuMe

• We distinguishes the following elements for a typology of generative system in generative art and computational creativity:
  – Domain (symbolic or audio)
  – Creative Tasks
  – Level of autonomy
  – Genericity/specificity of the system
  – Levels of interactivity and type of inputs
  – Relation to time
  – Architecture and algorithms
Why does it matter?

1. Fundamental research on creative process / AI / ML
2. Rational problem solving is not the main use of computers (anymore): Creative and entertainment computing is.
3. The move from linear to non-linear media entails an explosion of the number of assets needed:
   – Ex: World of Warcraft: 12 millions players, playing 20 hours per week on average!
   – Music for game: copyright free, adaptive, personalized,…
   – Visuals, animations, story lines, levels, …
4. Software are mostly inert (no IHCI).
A brief History of MuMe

• It does not start with computers.
• Guido d’Arezzo (one of the pioneer of musical notation) had the idea of an algorithmic composition associating a note to each vowels of a text as early as 1026.
• Conceptual machines aside, it starts with early automaton.
Early Automaton

– With the development of energy sources some processes start to be automatized, and more and more machines are being built.
– Very early, water was used and hydraulic energy started to be exploited.
– The hydraulic organ or Hydrolis was conceived 3rd century BC in ancient greece. It does not need the human to blow air anymore.
– Fountains, which seem to defy the laws of gravity become a trend. The siphon that makes water travel upward is attracting curiosity (as it is magic to those that are not in the know)
– This is the emergence of automaton
– The polymath and mechanical genius, al Jazari (12th century), is as known for his hydraulic automaton, than for his ingenuous engineering
– He produced a band of musical automaton.
– Al-Jazari created a boat with four automatic musicians that floated on a lake to entertain guests at royal drinking parties. It was programmable so that each automatic musician could could play different patterns.
Around the 14th century, and with the development of physics hydrolic energy is supplemented with mecanical energy and steam powered systems.

Automatons become more common:

- A wide variety of automaton are produced ranging from pieces of furniture and instruments like the barrel organ, to androids and animal automaton like Vaucanson flute player, tambour player and duck.
- The duck, for example is made of over 400 moving parts, allowing the automaton to eat, digest and defecate.
Besides the cam, the pin cylinder was invented. Although it was not thought of in terms of information and programming at the time, it did inspire the automatic loom, which in turn influenced the design of the first computers. Kircher, hydraulic organ with dancing skeleton from 1650.
Cypher
EMI
Voyager
Continuator
HPSCHD
Cybernetic
Serendipity
Senster
LAM @ Goldsmiths
CSIRAC’s first melody
Musicolour
Iliac Suite
Xenakis founds EMAMu
Cybernetic Serendipity
HPSCHD
9 Evenings: Theatre and Engineering
IRCAM
EMI
Continuator
Deep Blue beats Kasparov
Cypher
Machines 12@CBC
LAM @ Goldsmiths
MuMe
Weiner defines cybernetics
Electronics Computation Personal Computation Big Data…
Walking on the MuMe

Families of approaches with examples
MUME Algorithms

• Chance operation
• Chaotic systems, Fractals, Cellular Automata
• Substitution Systems: Grammars, L-Systems, Augmented Transition Networks
• Stochastic / Markovian approaches
• Search-based systems
• Agent and MultiAgent Systems
• Evolutionary Computing
• Neural Networks
Chance Operations

• Generative art implies for the artist to sacrifice (or more accurately “delegate”) some control in favour of a process.

• Randomness, noise, and weighted randomness (probability distributions, density functions) have been extensively used in music.

• These allow to generate variety:
  – The artist defines a space with various parameters that can take values within certain ranges.
  – These values are then randomly selected and an instantiation of the artwork is completed.
Aleatoric Music

- **Aleatoric music** exemplifies the use of randomness in music:
  - “a process is said to be aleatoric [...] if its course is determined in general but depends on chance in detail” (German physicist, Werner Meyer-Eppler 1957, 55).
  - Became popular with the **musical dice games** of the late 18th and early 19th century.
  - Marcel Duchamp composed aleatoric pieces as early as 1913, but John Cage's *Music of Changes* (1951) is often considered the first piece to be conceived largely through random procedures (Randel 2002, 17).
  - Aleatoric music thrived throughout the 20th century and to these days with composers like Charles Ives, Henry Cowell, Pierre Boulez, Karlheinz Stockhausen, and many more.
Franco-greek composer and architect Iannis Xenakis was a fervent user of randomness and stochastic systems in music and architecture. Applying probability theory and probability distribution to music composition.

In Metastasis (1954) or Pithoprakta (1956) using probability theory for composition, Xenakis made all the calculations by hand and became convinced that computers would be useful ;)}
Almost everything is stochastically decided. The pitch themselves are chosen at random, there is no link between the notes. The density of events is also determined stochastically...as indicated by the colors in this part of the composition. This pointillist style sounded a bit like the serialist composition of the time.
Buying some time on the only IBM computer in Paris at the time, he went on to develop his first computer generated compositions. ST series (1962) are compositions entirely generated by an algorithms which lie down all the stochastic choices that will generate the piece:
- ST/4 for quartet,
- ST/10 for mixed ensemble,
- ST/48 for orchestra.
Eventually, Xenakis turned himself to random walks first for digital synthesis and then for composition.

For example, the piece MIKKA (1971) is a solo for violin that is a direct mapping of a random walk on pitch. It is interpreted as a giant glissando.

This is the score of Evryali (1973), where you can clearly see some kind of random walk type function.
Conclusion on Chance as generator.

- In modern days, artists use randomness to bring up the tension between choice (or control) and chance, question intentionality, and actively explore the creative possibilities of the arbitrary and the accidental.
- In most cases, randomness is deployed as a strategy for exploratory creativity.
- It can be used to simply bring variability ("canned chance"), or when more dimensions are left un-fixed, as a tool for liberating creativity from rational thought.
Chaotic Systems

- **Chaos theory** is the field of mathematics that study dynamical systems that are sensitive to initial conditions.
- Chaotic systems are not to be confused with randomness, as chaotic system typical do not rely on any stochastic or otherwise non-deterministic operation. They are always deterministic. They appear to be linked with randomness simply because the future states of chaotic systems can only be predicted in the short term (a few iterations, or time steps).
- Unpredictability is not randomness. Many phenomenon that were thought to be random are actually chaotic. The good news is that short term predictions are possible.
Chaos theory in music

• In the 80s, Chaos theory was applied to music composition: applied to pitch, duration, dynamic level, orchestration,... Chaotic, or non-linear dynamic systems are useful because they can easily generate repetition and variations in periodic and quasi-periodic modes, or break out to more unpredictable behaviors in chaotic mode.
• Four pioneers of these methods are Jeff Pressing, Michael Gogins, Rick Bidlack, and Jeremy Leach.
• Chaotic systems have also been applied at the micro-level in the context of sound synthesis: Composers Barry Truax and Agostino DiScipio did experiment with applying chaos to granular synthesis so that a non-linear dynamic system is applied to the re-ordering of the grains of a given sound file. The different modes of the system allow to navigate textures that are very stable and close to the original sound, to more chaotic ones that deviate completely.
• To illustrate this, here is an excerpt of “Piccoli-ritmi” (1996), by Agostino Scipio
Cellular Automata

• **Cellular automata** were developed in the 1950s by Konrad Zuse, Stanislav Ulam, and John Von Neumann

• A cellular automaton consists in:
  
  – A **universe**: A n-dimensional grid of cells that can be in a finite number of states.

  – A **transition rule** that indicates for a given cell what state it should be in at the next time step given: (a) its current state and (b) the states of the cells in its **neighborhood**.
1-Dimensional Cellular Automata

- The neighborhood of a cell consists in the cell and its adjacent cells. We consider a radial neighborhood of radius $r$. Typically, $r=1$.

- Example, rule number 30:

<table>
<thead>
<tr>
<th>Neighborhood</th>
<th>111</th>
<th>110</th>
<th>101</th>
<th>100</th>
<th>011</th>
<th>010</th>
<th>001</th>
<th>000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resulting state</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

- There are $2^8 = 256$ different rules for 1-dimensional automata with $r=1$. 
Example

Rule 30:

<table>
<thead>
<tr>
<th>Neighborhood</th>
<th>111</th>
<th>110</th>
<th>101</th>
<th>100</th>
<th>011</th>
<th>010</th>
<th>001</th>
<th>000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resulting state</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Cellular automaton:

<table>
<thead>
<tr>
<th>Initial State</th>
<th></th>
<th></th>
<th></th>
<th>X</th>
<th></th>
<th></th>
<th></th>
<th>t₀</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generation 1</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>t₁</td>
</tr>
<tr>
<td>Gen 2</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>t₂</td>
</tr>
<tr>
<td>Gen 3</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>t₃</td>
</tr>
<tr>
<td>Gen 4</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>t₄</td>
</tr>
<tr>
<td>Gen 5</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>t₅</td>
</tr>
<tr>
<td>Gen 6</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>t₆</td>
</tr>
</tbody>
</table>
Class 3 Automaton - rule 30

<table>
<thead>
<tr>
<th>Neighborhood</th>
<th>111</th>
<th>110</th>
<th>101</th>
<th>100</th>
<th>011</th>
<th>010</th>
<th>001</th>
<th>000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resulting state</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Class 4 Automaton – rule 90

Cellular automata can generate self-similar structures (e.g., Sierpinski triangle)
Printout of CA states used in Iannis Xenakis composition *Horos* in 1986.
Noisesquare, Mo Zareei, Del Carnegy and Ajay Kapur, 2015.
Example of mapping of 1-dimentional CA to music.

- **Pitch**: Each cell corresponds to a note
- **Rhythm**: Each generation corresponds to a time unit (say a $1/16^{th}$ of a beat).

Screenshot of early CA-based system by Peter Beyls in the early 1980s.
Screenshot of CAMUS 3D (2001), Eduardo Miranda and his team.

– Play 36 seconds of this track:
  https://cataclyst.bandcamp.com/album/automata-48
CA-based Software for Music

• Amongst the many software allowing to use CA for music production:
  – Cellular Automata Music, 2000
  – FractMus, 2000
  – York’s Cellular Automata Workstation, 2005
  – Cellular Grid Machine, 2008
In **LASy (Linear Automata Synthesis)**, by Jacques Chareyron, 1990:

- A 1D CA of 512 cells is viewed as a wavetable in which the cell values are the sample values.
- 4096 values/states per cell: \(2^{12}\) corresponding to the 12-bit depth encoding of the soundwave.)
Pros and Cons

- **Randomness:**
  - **Pro:** cheap, versatile, discrete or continuous, scale to all media and dimensions
  - **Cons:** limited to be applied to existing parametric spaces

- **Chaos, Fractals, and Cellular Automaton**
  - **Pro:** cheap, several modes to explore
  - **Cons:** hard to control, deterministic (but there ways around this)
Grammars and state machines
Generative Grammar in music

<table>
<thead>
<tr>
<th>Eight measures</th>
<th>Second four</th>
</tr>
</thead>
<tbody>
<tr>
<td>First four</td>
<td>Opening cadence</td>
</tr>
<tr>
<td></td>
<td>Opening cadence'</td>
</tr>
<tr>
<td>Second four</td>
<td>Opening cadence</td>
</tr>
<tr>
<td></td>
<td>Middle cadence</td>
</tr>
<tr>
<td></td>
<td>Middle cadence</td>
</tr>
<tr>
<td>Opening cadence</td>
<td>I</td>
</tr>
<tr>
<td>Opening cadence'</td>
<td>I</td>
</tr>
<tr>
<td>Middle cadence</td>
<td>I</td>
</tr>
<tr>
<td>Closing cadence</td>
<td>I</td>
</tr>
</tbody>
</table>

Philippe Johnson-Laird’s context-free grammars (type 2) rules to generate 8-measures chord sequences in Jazz (2002)
Generative Grammar in music

Structure underlying the piece Off Mirror by Thelemonius Monk
Impro-Visor

Bob Keller et al. (2005-now)
Impro-Visor

Based on a jazz chord structure for 'Hit the road Jack' by Percy Mayfield
Solo generated by Impro-Visor and then modified by using the 'draw' tool

For this version, the STYLE chosen was 11-4
David Cope EMI (Experiments in Musical Intelligence) is a system that does style imitation using a recombinant approach based on ATN (1996-).
L-system

- The Koch curve as a
- L-system rule: 
  \[ F \rightarrow F + F - - F + F \]
- Initiator: \( F \)
- \( \phi = 60^\circ \) (angle),
- \( L=2 \) (length)
“Cells”, Hanspeter Kyburz, 1993. For saxophone and ensemble

Play audio from link.
Conclusion on Substitution Systems

— Pro:
  • Capture contextual and hierarchical aspects
  • Capture pre-existing knowledge
  • Human readable
  • Can be learned

— Con:
  • Not adapted to multidimensional sequences (NLP)
Markov Models

- **Markov assumption**: the future only depends on the present or a limited part of the past, say the $d$ past events.

$$ P(X_t | X_{t-1}, X_{t-2}, ..., X_1) = P(X_t | X_{t-1}, ..., X_{t-d}) $$

- $d$ is the **order** of the Markov model:
  - Order 0: probability distribution of the events, $P(X_t)$
  - Order 1: conditional probability distribution, $P(X_t | X_{t-1})$
  - Order 2: conditional probability distribution $P(X_t | X_{t-1}, X_{t-2})$
  - Order $d$: conditional distribution (transition table) over the $d$ previous events
Example using the corpus of 11 Stephen Foster songs from [Olson, 1967]
# Markov Model

Example using the corpus of 11 Stephen Foster songs from [Olson, 1967]

<table>
<thead>
<tr>
<th>Note</th>
<th>B2</th>
<th>C4</th>
<th>D4</th>
<th>E4</th>
<th>F4#</th>
<th>G4</th>
<th>G4#</th>
<th>A4</th>
<th>B4</th>
<th>C5#</th>
<th>D5</th>
<th>E5</th>
</tr>
</thead>
<tbody>
<tr>
<td>B2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>1/16</td>
<td>1/16</td>
<td>2/16</td>
<td>5/16</td>
<td>3/16</td>
<td>1/16</td>
<td></td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>1/16</td>
<td>3/8</td>
<td>3/16</td>
<td>1/4</td>
<td></td>
<td>1/16</td>
<td></td>
<td>1/16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F#</td>
<td>1/8</td>
<td>1/4</td>
<td>5/16</td>
<td>1/8</td>
<td>1/8</td>
<td></td>
<td>1/8</td>
<td>1/16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td>1/4</td>
<td>3/16</td>
<td></td>
<td>3/8</td>
<td>3/16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>1/16</td>
<td></td>
<td>5/16</td>
<td>1/16</td>
<td>1/16</td>
<td>4/16</td>
<td>3/16</td>
<td></td>
<td>1/16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9/13</td>
<td>2/13</td>
<td>2/13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1/2</td>
<td></td>
<td>1/2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1/4</td>
<td>7/16</td>
<td>3/16</td>
<td>1/16</td>
<td>1/16</td>
</tr>
<tr>
<td>E5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6/16</td>
<td></td>
<td>10/16</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Each line sums to 1
Markov models have been extensively used for music generation:

– The **ILLIAC Suite**, Lejaren Hiller and Leonard Isaacson, Fourth movement, 1956. Know to be the first computer generated composition!

– Xenakis “Analogique A” (1958)

– Brooks et al. (1993)


– Pachet “Continuator” (2002)

– ...
BeatBack: Interactive Percussion System

- **BeatBack** for interactive or augmented drumming:
  - Uses drum zoning
  - Variable Order Markov Models (VOMM)
  - Call-response and accompaniment
META-MELO

• Challenge:
  • Qualifying the bias of corpus-based systems

• Solution:
  • Implement three models:
    • VOMM
    • Factor Oracle
    • MusiCog
  • Test with three corporuses

• Experiment:
  • These systems work!
    • https://soundcloud.com/pournam/ambient

• Ongoing:
  – Fixed point / iterative drift?
  – Combinatorial creativity?
Harmonic Progression Generator

- It is doing style imitation, at human-competitive levels.
- The system is available online (open source and free!).
"Dead Slow / Look Left"

Composed by Arne Eigenfeldt

Performed by

The Yaletown String Quartet

and Daniel Jones, percussion

December 2, 2011
StyleMachine Lite

- corpus: recalculating...
- complexity
- density
- length
- gear
- done!
- generate phrase
- generate all

Help View
The Help View provides access to Lessons, which are short, step-by-step tutorials that are a great way to learn about Live interactively.
Style Machine

• Generative EDM

• How?
  • Manual analysis of corpus by experts (composers, producers)
  • Our machine learning algorithm
    Genetic algorithm / VOMM

• Validation: ongoing!
  • Confuses classifiers: pieces gets classified properly!
  • Confuses listeners
  • Public shows since 2013: Algorave, …
  • Album on ChordPunch (UK)
Markov Models

• Advantages:
  – Intuitive and easy to understand
  – Computationally cheap

• Issues:
  – Randomness in the output, with clear lack of overall structure
  – Worse with low orders, limited choices with higher orders (e.g., AFGBBF#FGDFG#EFG would be regenerated with order 3)
  – “equivalence/transient/recurrent classes”: strong internal connection, few connections between classes (one gets stuck for a while, or even fail to leave them at all) [Kevin Jones, 1981]
  – Limited to one-dimensional symbolic sequences (e.g., natural language processing)
  – Limited to style imitation (although Xenakis was using it for computer-assisted composition)
Hidden Markov Models (HMM)

Used to learn coupling:
- Accompaniment, harmonization (on a given melody, in a given style)
- Interpretation (of a given score)
Substitution systems

- Rule-based system, substitution systems or production systems:
  - Generative grammars
  - L-systems
  - Shape grammars
  - Automaton:
    - Transition networks
    - Augmented transition networks
    - Petri nets
  - Markov Chains
  - Hidden Markov Chains
Search Based systems
Creativity as a search

• Modeling creativity as a search assumes:
  – A search space or conceptual state that corresponds to all the possible artifacts, behavior, or candidates that would be the product of a creative task.
  – A representation and structure for the search space: set of parameters, list, tree, graph, ...
  – A search strategy for exploratory creativity:
    • Generate and test
    • Enumeration
Heuristic search

Online accompaniment interpretation system, Roger Danenberg, 1984.
Database search

• **Audio Metaphor: Soundscape generation engine**

• **Approach:**
  – User input: an expression + desired affect (pleasantness, eventfulness) + duration
  – Sounds retrieval from tagged db (WSP, freesound)
  – Segmentation and classification of background and foreground sounds
  – Pleasantness and eventfulness classification
  – Mixing and audio rendering

www.AudioMetaphor.ca
Audio Metaphor

A waterfall in Thailand
Audio Metaphor

A city in the bush
Audio Metaphor

A quenching rain drenched my burning head
MUME Algorithms

• Chance operation
• Chaotic systems, Fractals, Cellular Automata
• Substitution Systems: Grammars, L-Systems, Augmented Transition Networks
• Stochastic / Markovian approaches
• Search-based systems
• Agent and MultiAgent Systems
• Evolutionary Computing
• Neural Networks
Agents and Multiagents Systems
Agents and Multiagents Systems

• An artificial **agent** is a computer system that is capable of **autonomous** action on behalf of its user or designer.

• A **multiagent system** is one that consists of a number of agents, which **interact** with their environment (including with one-another)
Agent architectures

• Three types of agent architectures:
  – **Cognitive**: maintain internal symbolic representations
    • Deliberative architectures: reasoning and planning
  – **Reactive**: no explicit representation of the environment and focus on behavioural rules
    • **Reflex**: no internal states (just mapping inputs to outputs)
    • **Reactive**: with internal states (but not cognitive)
  – **Hybrid**: mixing reactive and cognitive components to balance Reactiveness and deliberativeness
Musical Agents: Voyager

- Early example of “cognitive agent” working online, and interacting with live musician in the context of Jazz improvisation (free Jazz).
- The system was programmed in Forth in 1986
- Voyager Duo 4, George Lewis, 1986
- Play 38 seconds of: https://www.youtube.com/watch?v=hO47LiHsFtc&list=RDhO47LiHsFtc&t=12
Musical Agents: Voyager

Interactive Trio - George Lewis (2011)

Musical MultiAgents Systems

- Coming Together, Arne Eigenfeldt, 2010
  - Using the BDI architecture
  - Play 20s from 4:25 to 4:45
Musical Metacreation

• Closure-based Cueing Model (CbCM)
• Challenge: learning and generating music (symbolic)
• Solution:
  • Another attempt at a hierarchical, deep learning, model of musical cognition
  • Based on notions from the musical perception and cognition literature
• Validation: it actually works!
  • Applied in the ManuScore computer-aided composition software
  • Used for actual compositions (instrumental contemporary music): presented in concerts.
  • Empirical evaluation with 42 participants: could not segregate
MusiCog

MUSEBOTS framework
Musical cognitive agents

• Musical cognitive agents:
  – Performing on their own
  – Performing alongside with humans
  – Helping humans to create new material
Agent architectures

• Three types of agent architectures:
  – Cognitive: maintain internal symbolic representations
    • Deliberative architectures: reasoning and planning
  – Reactive: no explicit representation of the environment and focus on behavioural rules
    • Reflex: no internal states (just mapping inputs to outputs)
    • Reactive: with internal states (but not cognitive)
  – Hybrid: mixing reactive and cognitive components to balance reactiveness and deliberativeness
Subsumption Architecture
BeatBender: multi-agent rhythm generation

- **Challenge:** non-corpus based generation of rhythmic patterns
- **Our approach:**
  - Using reactive agents to create rhythmic patterns
  - Using subsumption agent architecture

Experiments on a sample of 10+10 rhythms show that:
- Humans prefer BeatBender rhythms over human composed ones
- They find them more natural (less artificial)
Boids and Swarms

A basic boid agent is implementing three simple behavioral rules:

1. **Avoidance**: move away of a flock that is too close.
2. **Imitate**: fly in the average direction/speed of the flock by averaging the velocity and direction of the other boids in the neighborhood.
3. **Center**: Minimize exposure to the flock exterior by drifting towards the perceived center of the flock.
Reactive agents – Swarm Music

Musical MAS – BeatBender

• The system models a drum circle with agents based on the subsumption architecture with four types of behavioral rules:
  – Neighborhood rules react to the status of the neighbors agents
  – Directed rules react to the status of specific agents
  – Collective rules react to the global activity of all the active agents
  – Temporal rules that use the history of the agent state

• Experiments show that complex rhythmic structures can be generated this way.
Musial agents – Porto Actors with Eargram

Porto actors with Eargram, Peter Beyls, Gilberto Bernardes, and Marcelo Caetano, 2015.
Layered Hybrid Architectures

Horizontal Layering

sensor input

action output

Vertical Layering

one-pass control

sensor input

action output

two-pass control

action output

sensor input
Generic Musical Agent Architecture

Input (audio or symbolic)

- Perception module

- Learning and generation module

- Interpretation module

Output (audio or symbolic)
Musical agent – Odessa

The Odessa musical agent, Adam Linson, Chris Dobbyn, George Lewis, and Robin Laney, 2012.
Musical agent – Odessa

• Here is an excerpt of the system in an improvisation with Adam Linson playing the double bass.
Musical agent – OMAX

Excerpt of a recording a variant of the system using the Variable Order Audio Oracle algorithm by Cheng-I Wang and Shlomo Dubnov, 2013.
MASOM - Live performance

iOTA
(collaboration with OUCHHH and Audiofil)
Conclusion on Agents and MAS

• **Cognitive approaches** propose a **top-down** solution to:
  – **Agent design**: the agent architecture, and its decision process.
  – **Society design**: The organization of the MAS uses roles, conventions and protocols. Group goals, are broken down into individual goals, themselves broken down in sub-goals, reified as intentions, achieved through planning sequences of actions.

• **Reactive AI** proposes a **bottom-up** emergent solution to:
  – **Agent design**: the agent behavior emerges from the interaction between its behavioral rules
  – **Society design**: the MAS behavior emerges from interaction the agents with their environment.

• **Hybrid architecture** marry both approaches
Pro and Cons

• Pro:
  – A natural framework
  – Online, interactive, ...
  – Possibly distributed: group creativity, hybrid systems,…

• Con:
  – ????
Evolutionary Computing

• Genetic algorithms
• Genetic Programming

• Types of fitness functions:
  – Interactive fitness, in which humans are judges
  – Automatic fitness functions:
    1. Data-driven fitness based on target, targets or target’s properties,
    2. Data-driven fitness based on machine learning of human preferences or physiological data,
    3. Analytical and theoretical formulations of fitness functions.
In order to improvise Jazz solos, GenJam is co-evolving two populations of melodic ideas:

- A **measure population** of 64 individuals: chromosomes are made of 8 genes that each map to an 8th notes. Each gene in a measure is encoded by four bits, with value 0 for rest, 15 being a hold, and 1-14 being the notes events that are mapped to an actual MIDI note through a set of scales that corresponds to the chord being played during that measure.

- A **phrase population** with 48 individuals: A phrase is made of 4 measures each encoded by 6 bits.

Musically meaningful operators:

- The **measure mutations** operate at the note level and include transposition, rotation, sorting, inversion, retrograde, ...

- The **phrase mutations** operate at the measure-pointer level and include reverse, rotation, sequencing, ...
Data-driven fitness function

- **Data-driven or examples-driven fitness function** can be derived from existing pieces:

  1. Define a number of dimensions normalized in [0;1]: pitch variety, dissonance, contour direction, contour stability, rhythmic variety, rhythmic range, ...
  2. Select a corpus of existing pieces, and calculate statistics (typically average and standard deviation) for the selected features and use these values as a multidimensional fitness goals.
  3. Use a distance of similarity measure as a fitness function.
The **StyleMachine Lite**, by Metacreative Technologies, does corpus-based style imitation of Electronic Dance Music (EDM) since 2014.
Style Machine

• Generative EDM

• How?
  • **Manual analysis** of corpus by experts (composers, producers)
  • **Our machine learning algorithm**
    Genetic algorithm / VOMM

• Validation: ongoing!
  • Confuses classifiers: pieces gets classified properly!
  • Confuses listeners
  • Public shows since 2013: Algorave, …
  • Album on ChordPunch (UK)
Preset Generation

PresetGen

- **Automatic preset generation**
- **Challenge:** finding the set of parameters that gets us as close as possible to a target sound
- **Our approach:**
  - Algorithm: NSGA-II
  - Fitness: FFT, SFFT, temporal envelope
- **Evaluation:**
  - Reverse engineering
  - Other sounds
  - Example: piano (c5)
  - Ongoing empirical study
- **Ongoing / Future Works:**
  - Online deployment for TE
  - Synthesizer generation (Pure Data)
Musical Metacreation

• Problem: 
  Sound synthesizer generation

• Our approach:
  – Mixed-type Cartesian programming to evolve Pure Data patches
  – Fitness function based on perceptual sound similarity

• Results:
  – Promising?
  – After 7 years of work!
Reverse Engineering Sine Waves

Play each column in turn. For each column play top and then bottom. Print “Target” when you play top, and “Approximation” when you play bottom.

Spectra comparisons between approximated and target sounds.
Generating synthesizers for real-world sounds.

Play each column in turn (starting at 42:27).
For each column play top and then bottom
Print “Target” when you play top,
and “Approximation” when you play bottom.

Spectra comparisons between target and generated sounds.
Pros and cons of IGA

• Challenges with IGA include:
  - **Difficult to control**:
  - Mating operations often result in offspring which resemble just one (or often neither) of the parents.
  - The user cannot specify which changes are desired.
  - **Low population size**: interactive systems require design spaces with higher average fitness.
  - **Time required**: the time needed to review individuals is the bottleneck of the system leading to user fatigue.

• **Pros**:
  - **Parallel search**: IGA allows to navigate the search space explicitly and “zoom in” or “zoom out” on a particular design and its neighborhood.
  - **Genetic engineering interfaces** allow for manual refinement of the genome of an individual. However, genes often interact with each other when the phenotype is expressed (i.e., epistasis).
  - **Crowdsourcing** allow for collective creativity and cumulative progress.
Evolutionary ecosystem

- Mixing agents, ecosystems, and evolutionary computing
**Purely sonic ecosystems**

**Sonomorphs**: An application of genetic algorithms to the growth and development of musical organisms, Gary L. Nelson, 1993.
ElektroPlancton, Indieszero, Nintendo DS, 2005.
Evolutionary Ecosystems

• Evolution needs to be steered:
  1. Put the human in the loop with IGA
  2. Use a computational analytic evaluation: an analytic fitness function, or a data-driven one
  3. Use the ecosystemic approach using an indirect fitness function emerging from the system’s dynamic.

• Characteristics of evolutionary ecosystems:
  – More complex strategies for the genome expression through the agent’s lifetime behavior.
  – Harness co-evolution and has emergent dynamic fitness function
  – The termination criteria is unclear.
  – Hard to control and experimental.
  – Often evolve their own aesthetics that are likely irrelevant to human.
“Generative art practice focuses on the production and composition of the genotype and the media in which it produces the phenotype. When run, interpreted, or performed, the genotype produces the phenotype – the work to be experienced and the realization of the process encoded by the genotype.”

Neural Networks
Artificial Neural Networks (ANN)

- A family of connectionist approaches:
  - SOM (Self Organised Map)
  - Perceptron
  - MLP
  - Recurrent Neural Networks
  - Hopfield Networks
  - ART (Adaptive Resonance Theory)
  - LTST memory
  - Boltzmann machines
  - Deep learning
Unsupervised learning

Supervised learning:

Inputs → Supervisor → Desired output → Error

Learner → Output

Unsupervised learning:

Inputs → Learner → Output
A SOM running the learning algorithms on the color domain.
SOM in musical metacreation

• Examples of SOM used for musical agents include:
  – Benjamin Smith and Scott Deal (2014) present a musical agent architecture that utilizes a SOM as part of the agent short term memory.

• Other musical use include:
  – Phon-Amnuaisuk (2007) has used SOM to extract musical structures and use it as a critic or fitness function for an evolutionary system.
  – In order to capture the hierarchical dimension of music, the work has been extended using Hierarchical SOM or HSOM.
SOM for sound organization

- The **sonic SOM** is a project by Arne Eigenfeldt and Philippe Pasquier, 2009.
- It is a **computer-assisted creativity system** helping musicians and sound designer navigating and selecting audio samples.

**Diagram:**
- Repository of sound samples
- Audio feature extraction
- The color of the neuron represents the spectral dominant of the sound (red is mapped to low frequencies, green to mid frequency and blue to high).
ALVINN (Autonomous Land Vehicle In a Neural Network), Dean A. Pomerleau, Todd Jochem, 1989.
Dealing with time series with ANN

• **Music is sequential** so we need to encode time with MLP.
• The first solution is to use a **sliding window**: we assume discrete time and all the data are shifted right at each new instant.
Recurrent Neural Network

- Memory can be captured by recurrent connexions also called feedback connection through which the state or emission of a neuron is being kept by being transmitted as input somewhere else.
Recurrent Neural Network

- Recurrent connexions allow to represent and learn a wild range of sequential behaviour.
- RNN have the most general representational power.

A recurrent collection and a tap delay line allows this network to represent and learn a sequential behaviour.
Peter Todd at al., early use of RNN for music, 1989.
Example: HARMONET

- HARMONET by Wolfram Menzel et al., 1992
- Creative task: to harmonize melodies in the style of J.S. Bach.
- It is a multilayer perceptron with simple recurrent links such that it has for input:
  - The **harmonic context** made of its previous outputs $H_{t-1}$, $H_{t-2}$, $H_{t-3}$
  - The **melodic context** (both past and future): $s_{t-1}$, $s_t$, $s_{t+1}$
  - Phr$_1$ indicates the position in the musical phrase and the str$_1$ indicates whether the current harmony is a stressed quarter.

- All together, the network has 106 input nodes, 70 nodes in the hidden layer and 20 output nodes.
- Trained on 40 Bach chorales, using back propagation for learning algorithm.

Evolution of AI and machine learning systems.
Unsupervised greedy layer-wise training

Pre-learning: train the network with a lot of unlabeled data (unsupervised learning)

Supervised learning on a small(er) set of labeled data.
• **ALICE (A LSTM-Inspired Composition Experiment),** Andreas Brandmaier, 2008.
Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pascal Vincent are modeling temporal dependencies in high-dimensional musical polyphonic sequences with RTRBM and RNN-RBM, 2012.
DeepBach (2017)

Allows for incremental and interactivity generation. Any of the four voices can be given/modified.
Deep learning of the relations between music audio features and dance movements.
Dance generation in MAVI, Sunny Yang, Philippe Pasquier, 2016.
ANN in sound synthesis

**NEURAL SYNTHESIS** (Nos. 6-9), David Tudor in collaboration with engineers Forrest Warthman, Mark Holler, and Scot Gresham-Lancaster. It features an analog neural network synthesizer used to generate complex oscillations, 1992-94.
Oliver Bown musical agent uses a CTRNN (continuous time recurrent neural network) driving a sound synthesis system, 2009.
Deep Learning

- WaveNet generate sounds signal (sample by sample)
- Developed for text to speech, it can generate music.
- Using Dilated Convolutional Network
There is plenty!

- Other "deep Learning approaches to audio music generation:
  - SampleRNN (Mehri et al. 2017),
  - DeepVoice (Arik et al. 2017),
  - TacoTron 2 (Shen et al. 2017),
  - WaveRNN (Kalchbrenner et al. 2018)
Improved SampleRNN (MuMe 2018)
Neural networks as fitness function

– Researchers used ANN as fitness functions in musical systems:
  • Pazos et al. (1999) used this approach for their GA-based rhythmic generation system.
  • Marcus Pearce (2000) used a MLP for evaluating the fitness of drum and bass rhythms generated using GA.
  • Al Biles et al. (1999) used ANN for fitness ratings in GenJam.
Pro and Cons of ANN for generative art.

• Disadvantages and limitations:
  – Learning is slow, and computationally demanding.
  – The right type and quantity of data needs to be available.
  – ANN are difficult to design, and have very many parameters.
  – ANN are black boxes: opaque learning

• Advantages:
  – ANN are fast for generation.
  – They are very flexible to capture multi-dimensional domains with a variety of inputs.
  – They can generalize beyond the corpus.
The Evaluation Problem

How good is this system?
Evaluation of metacreation

Metacreations can be evaluated by:

1. Their authors: artists, designers, computer scientists,…
2. Users, peers and experts: composers, musicians, sound designers,…
3. The audience: popularity, concert and album sales,…
4. Media: critics, journalists,…
5. Peer reviewers, curators and jury: papers, concerts, festivals, grants, …
6. Theoretical and analytic measures: of the process, of the input/output relationship, …
7. Empirical studies: qualitative or quantitative user/audience study, …
Evaluation of Musical Metacreation

• **Evaluating** creative systems is a difficult task:
  
  • Theoretical reasons:
    • No notion of optimality
    • Subjective/cultural impressions/judgments are involved
    • It is multidimensional and framing can play a role: humans seem biased against computational creativity (Moffat and Kelly, Comp. Crea., 2006)
  
  • Practical reasons:
    • Choice of the corpus
    • Choice of the parameters (user study)
    • The system needs to be evaluated on a sample output: generative systems can create ad infinitum
    • The various uses of the system needs to be taken into account
    • Composition and interpretation are dependent
Evaluation of Musical Metacreation

• This study is part of a series of 8 studies exploring a range of qualitative and/or quantitative methods (on various systems).
• The goal is to explore existing and craft new research instruments (because we need them)
• Methods often include indirectness:
  • Instead of: “Is the system creative?”
  • Researchers look at comparative studies:
    • “Are the system’s productions comparable to human productions (from the corpus or not)?”
    • “Can the audience identify which outputs are system generated?”
  • Using deception:
    • How engaging was the piece?
    • How boring-enjoyable, simple-complex, organic-mechanical, ...
Methodology

• Our methodology builds on previous work on Turing test-like proposals.

• For 20 excerpts, we ask the user to determine the likely provenance of the source (no deception) using a 4-point scale:
  1. “definitely human”,
  2. “probably human”,
  3. “probably computer”
  4. “definitely computer”.

• This way:
  – We get both guess and confidence.
  – We do not provide an “I don’t know” option to keep the participants engaged (perceptual studies show that participants tend to underestimate their capabilities).
Corpus

- **Twenty 8 bars excerpts**: 2 human-composed progressions and 2 system-generated ones from each of the following 5 classical and romantic music corpuses:
  
  
  - Antonín Dvořák: *Humoresque*, Legend; *Slavonic Dance No. 1*; Slavonic Dance No. 2, Symphony No. 9 “From The New World” Second Movement, Valse Gracieuse.
  
  - Johannes Brahms: Symphony No. 1 In C Minor 3rd Movement, *Symphony No. 2 In D 3rd Movement, Symphony No. 3 in F 2nd Movement*, Symphony No. 3 in F 3rd Movement, Symphony No. 4 In E minor 3rd Movement, Hungarian Dance No. 5.
  
  - Felix Mendelssohn: *Consolation*, If With All Your Hearts, Spinning Song, O Rest In The Lord, *Scherzo in E Minor*, Venetian Boating Song (from Songs Without Words).
  
Participants

- Two independent groups: formal training in classical music analysis (the “target language”) versus no/informal musical training
- Total participant count: 87
  - 2+ years of bachelor’s degree in music: 9
  - Royal Conservatory (≥ 5th grade): 11
  - Some experience: 37
  - No experience: 30

“Musicians”
“Non musicians”
Results: Discrimination choice

• Statistical analysis of the results shows that:
  – Participants were not able to identify computer-composed pieces above chance level, thus “validating” our system
  – Expertise does not make a difference: no significant difference between groups could be observed.
  – Human competitive system: but only for a short sequence (4-32 bars)
Evaluation of the Creative Output

• Generative art and computational creativity consists in encoding, at least partially, a creative practice into a process.

• This creative process can be:
  a. Modeled after of an existing practice, addressing an existing creative task and can be compared with humans products at the same task.
  b. A deliberately new process the outcome of which can not be produced by a human without modern computers.

• Comparative evaluation methods do not really evaluate the creative process, just the creative outcome.
  – Weak computational creativity: only the system’s product is deemed creative, sometime called “mere generation”
  – Strong computational creativity: the process itself is deemed creative.

• Framing, intentions, explanations and justifications are important and influence perception. This has been lacking in MuMe (while it does exist in other domains: e.g., The Painting Fool, Angelina).
The Bias Against Metacreation

• The bias against computational creativity is the hypothesis that computationally-generated artifacts are often judged to be less interesting, valuable, and less creative than human generated ones.

• Anecdotal evidence include David Cope’s Experience in Musical Intelligence (EMI, 1981), an early style imitation system that was received with controversy.
"Most musicians, academic or composers, have always held this idea that the creation of music is innately human, and somehow this computer program was a threat in some way to that unique human aspect of creation,"

"I have always refuted that by saying that a human built the machine, listens to the output, and chooses what's the best. What's less human about that than if I had taken years and years to just compose the whole thing myself?"

David Cope, 1986.
The bias against MuMe?

• Empirical study of the bias against computational creativity: David Moffat and Martin Kelly (2006)
MuMe Future
“Human musicians routinely jam with cybernetic musicians.”

“Virtual artists in all of the arts are emerging and are taken seriously.”

“Many of the leading artists are machines.”

“The reverse engineering of the human brain appears to be complete.”

“The algorithmic revolution lies behind us and nobody noticed it. That has made it all the more effective—there is no longer any area of social life that has not been touched by algorithms. Over the past 50 years, algorithmic decision-making processes have come very much to the fore as a result of the universal use of computers in all fields of cultural literacy—from architecture to music, from literature to the fine arts and from transport to management. The algorithmic revolution continues the sequencing technology that began with the development of the alphabet and has reached its temporary conclusion with the human genome project. No matter how imperceptible they may be, the changes this revolution has wrought are immense.”

Industry interest

• The industry is getting interested:
  – (New) Spotify Creator Lab (computer assisted composition)
  – JukeDeck (composition for video)
  – Metacreative Inc. (computer assisted composition)
  – Melodrive (adaptive generative music for games)
  – Google Magenta (?)
  – Many more...
The audience is on it

• CDs, vinyl, k7, and concert tickets are selling.
Poster of the musical “Beyond the fence”, 2016.
Live Coding

- Programming consists in instructing computers to do what we known to ask them.
- **Live coding** explores the idea of using code itself as a live generative instrument.
- Some computer music or creative coding platforms can be used for live coding: E.g., Supercolider, Chuck, Vvvv, Pure Data and Max.
- Specialized languages emerged for the purpose of live coding: Impromptu, Extempore, TidalCycles, Gibr, ...
- In effect, **Live coding is the coding of a generative system in real time**:  
  - The code can be deterministic or not.
  - Pre-programmed generative operations or algorithms can be used.
The Algorave movement
Live coding for visuals is possible with LiveCodeLab, Fluxus, Cyril, MAX or Pure Data.
The Fear of Technologic Unemployment

Luddites were English rioters in favor of destructing machinery.
The Fear of Technologic Unemployment

Advertisement from the American Federation of Musicians, Syracuse Herald, September 2, 1930.

IS ART TO HAVE A TYRANT?
“The time is coming fast when the only living thing around a motion picture house will be the person who sells you your ticket. Everything else will be mechanical. Canned drama, canned music, canned vaudeville. We think the public will tire of mechanical music and will want the real thing. We are not against scientific development of any kind, but it must not come at the expense of art. We are not opposing industrial progress. We are not even opposing mechanical music except where it is used as a profiteering instrument for artistic debasement.”

President of the American Federation of Musician, 1930.

A robot grinding up musical instruments, Syracuse Herald, November 3, 1930.
The Debate between Technophobia and Technophilia

- **Technoscience** refers to any concrete or abstract technology capitalizing on scientific methods and methodologies.
<table>
<thead>
<tr>
<th>Technophobia</th>
<th>Anti - Technophobia</th>
<th>Humanist Technophilia</th>
<th>The Post-modern Position</th>
<th>Evolutionist Technophilia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Independent development of technoscience</td>
<td>The autonomy of technoscience is a myth. Technoscientific development is Anthropocentric instrumentalism is a myth.</td>
<td>Technoscience is reticular</td>
<td>Technoscience achieves the project of modernity and is the metanarative of the western world.</td>
<td>Our nature is a product of evolution</td>
</tr>
<tr>
<td>Technical determinism</td>
<td>Technoscientific Instrumentalism is a myth.</td>
<td>Ideal of co-evolution of mankind and technique</td>
<td>Culture and technique have to grow together</td>
<td>Scientific knowledge is a factor of emancipation</td>
</tr>
<tr>
<td>Technoscience is a totalitarian ideology.</td>
<td>Ideal of diversity, conserving traditional, symbolic cultures vs. barbarism.</td>
<td>Ideal of dialogue, reintroducing democratic debate over pre-scientific issues.</td>
<td>We master nature through technoscientific means.</td>
<td>Ideal of techno-symbolic diversification</td>
</tr>
</tbody>
</table>

**Modern Humanism.**

- Tristram Engelhardt (1941 - )
- Jean Francois Lyotard (1924-1998)
- Gilbert Simondon (1926-1987)
- Jürgen Habermas (1929 - )
- Martin Heidegger (1889-1976)
- Jacques Ellul (1912-1994)
Community and Resources

• Online:
  – MuMe site: http://musicalmetacreation.org/
  – MuMe group: musicalmetacreation@googlegroups.com

• Academic venues:
  – International Workshop on Musical Metacreation
  – International Conference on Computational Creativity
  – International workshop on machine learning and music
  – International conference on generative art
  – EvoMusArt
  – ISMIR, ICMC, SMC, Audio Mostly
  – IJCAI, AAAI, ICML, GECCO
Written resources

• Numerous texts:
  – Book Neuhaus
  – Double issue ACM computer in Tentetainmet (including our introduction)
  – MuMe proceedings (110 papers on generative music) and all other proceedings (1k+ papers).
www.Kadenze.com

- A new MOOC for art education
- Focusing on the theory and practice of:
  - New Media
  - Digital Art
  - Creative computing
  - Computer Music
  - Interactive Art

Coming soon – Available For Credit Course opens Fall 2016

Introduction to Generative Arts and Computational Creativity
Simon Fraser University
Philippe Pasquier

VIEW  ENROLL
Conclusion
Lessons and Challenges

- Symbolic generation is better than audio generation (=interpretation is hard).
- Controllable factors need to be further explored (affective computing)
- Moving beyond style imitation
- Like everywhere, ANN are making a foray.
- More needs to be done (cognitive modelling, agent learning).
Computational Creativity

• **Computational creativity** is exploring the automation of creative tasks (as opposed to strict rational problem solving).

• Key research areas/questions are:
  – **Computational models of human creativity**: What is creativity? Can we model it?
  – **Artificially creative systems (metacreations)**: How to automate creative tasks? How do we evaluate creative machines? Can we go beyond human capabilities?
  – **Applications as Art/Design practices**: generative art, generative music, generative design, procedural content generation in games, …
  – **Applications in computer-assisted creativity**: computational systems for supporting human creativity (IHCI needed!)
Why does it matter?

• There are many reasons why the automation of creative tasks is a relevant research topic.

• 1. Scientific / academic:
  – Research in CC is fundamental research on creative processes.
  – It contributes to cognitive sciences, and to IHCI in the context of computer-assisted creativity, computer assisted design.
Why does it matter?

2. Pragmatic and economical reasons
   - Rational problem solving is not the main use of computers (anymore), creative and entertainment computing is.
   - There is a demand from the market as we move from linear to non-linear media.

• This entails an explosion of the number of assets needed:
  • Take gaming as an example: World of Warcraft currently has 12 millions players, playing 20 hours per week on average!
  • Music for game: copyright free, adaptive, personalized,…
  • Visuals, animations, story lines, levels, …
Why does it matter?

3. Efficiency and HCI software design: Creative software are mostly inert.
  • No, or not enough, intelligence and automation in creative software. There is a need in the creative industry
  • More generally in HCI, we are still pressing keys, and buttons, selecting functions in menus, ..., in a very repetitive way. We would like more fluid, high-level interactions with the machine.
  • This requires more machine autonomy, and generative system to completely or partially automate creative processes.
Why does it matter?

4. Societal:

– As the industrial revolution of the 19\textsuperscript{th} and early 20\textsuperscript{th} century was about automation of the mechanical labor, the digital/information revolution is about the automation of the treatment of information.

– Before the computer, only human brains could do the algebraic operations allowed by Excel or any basic computer program...

– Many tasks are and have been automatized.

– There is no reason why creative tasks would stay out of the reach of this revolution.

– There is no reason why art would stay out of this revolution and the accompanying dialogues.

– Some are afraid or less enthusiastic about the idea of artificial creativity, but the idea is not to replace artist.

– Artists and creative industries have always used the tools of their time to disseminate new forms, and the accompanying reflexions.
Why does it matter?

5. The final set of reasons are Cultural / artistic:
   - Generative art is an ancestral cultural practice in which artists are transferring (some of) their (creative) autonomy to a process (often a machine).
   - This is a process by which artists can free themselves from their own limitations, and go further in the conceptualization of their pieces (at the cost of a work of formalization through the writing of procedures and algorithms and their implementation).
   - Now a days generative art is at the forefront of digital art, using computers (the most common media of our time) for art making and culture-making.
   - This is one of the ways by which artists can engage into the dialogue around new technologies and the computerization of society.
Conclusion

• From the invention of the wheel to the development of the most advanced artificial intelligence, machine learning and artificial life, technology has continually shaped us.

• Away from the fears of AI taking over, I believe in the humanist tradition of anthropomorphistic instrumentalism: we design and make the machines we need that we think will serve us.

• The computerization of society and the rise of autonomous machines has deep implications, and the future is generative but can we harness the power of machines to expand our creativity?
Metacreation Lab
• **MUTEK 2018**, August 22, Montreal (CA)
• **MUME 2019**, June 2019, Charlotte (USA)
  – 6th International Workshop on Musical Metacreation
  – in conjunction with the International Conference on Computational Creativity (ICCC)
• **MOCO 2019**, October 2019, ASU (USA)
  – 4th International ACM Conference on Movement Computation