

# Analysis of Generative Music

Nick Collins  
University of Sussex



## Generative music

- Good old fashioned algorithmic composition?
- planting seeds, Goethe and Eno's gardening metaphors
- designing realtime generative systems which produce products on demand, not just a single fixed product
- Humans may design the systems and consume/modify the outputs, but are not typically 'within' the running of the algorithm (no IGA or IMS, live coding, text music etc). Typically computer programs without *human computation*, output only

## Examples

- More deterministic: isorhythm and Music for Airports, Jem Finer's LongPlayer
- Works on rand()%, the Algorithmic Stream, Infinite CD, Morpheus CD
- Karlheinz Essl's Lexicon Sonata (non-interactive version)
- Brian Eno's Generative Music 1 installation with Koan (1996)
- 'Infinite Length' Pieces (examples: *iDAB*, *pythcirc*, *Canonic Hill Loss*, *Decison*, *Mother Fuga*, *infno*)
- Existing as standalone computer programs

## Problem: How to Analyse?

- Music's innate combinatoriality of material (Prokofiev 1978, pp. 46-9)
- How many runs of the work are representative? (Statistical sampling problems: can't guarantee random sampling, normal distributions)
- Mathematical versus psychological space
- Testing against composer's statements as specification; but artists often obfuscate or hide technical details

## **Wading into the program code**

- May not be available! Black box testing/reverse engineering
- White box testing procedures: hard task to design tests supplying representative coverage
- Archiving is a general issue in digital media

## **Automated critics**

- May never be available or remain slower than RT (and the original generative program might only run at RT or slower!)
- Certain gross timbral measures and transcriptions with success rates from 40-80% or so are currently available in computational audio analysis. But transcription is ill-defined and most auditory mechanisms remain beyond the state of the art

## **Other tactics**

- Reductionism - dangerous! Changing the program can cripple emergence...
- Strategies inspired by other work- analysis of jazz improvisation, speech literature, musical style analysis, rule-based art

## **Counter-examples**

- ‘To establish a system for representing any aspect of music is almost a challenge to a composer to invent music which the system is incapable of representing.’ (Marsden 2000, p.168)
- Why do we need more music, let alone automatic music generating programs? Or even worse, computational creativity giving us automatic GM program generation!
- Perhaps we should solve problems of analysis first just for single fixed works...



## **What next?**

- Generative music already exists: as musicologists, we should engage with it
- Certainly, generative music poses novel problems (and rewards) for the analyst



## **Example analysis**

- Two short SuperCollider demo examples by James McCartney, ‘toy’ problems, white box testing possible