

Collaborating with the Behaving Machine: Simple Adaptive Dynamical Systems for Generative and Interactive Music

Alice Eldridge

Submitted for the degree of D.Phil.

University of Sussex

February, 2008

Declaration

I hereby declare that this thesis has not been submitted, either in the same or different form, to this or any other university for a degree.

Signature:

Acknowledgements

This is perhaps the smallest of many projects which was spawned by the energy and enthusiasm of Drew Gartland-Jones. I must thank him for support, encouragement and inspiration. To Phil Husbands I am indebted not only for a powerfully zen approach to supervision, but for being an inspirationally Renaissance man in modern academia. Thanks must go to Jon Bird for useful discussion and priceless support in times of need. Thanks also to Chris Thornton for lending his ear in the absence of Phil, and to Thor Magnusson and Peter Copley for discussion of less than conventional ideas. Beyond Sussex, I am indebted to the Live Algorithms for Music group which has provided a rare community for exchange and debate of matters core to this thesis and in particular to Ollie Bown whose intelligent enthusiasm encouraged me to pursue this project and continues to inspire its future moves. Finally I would like to thank members of blip, and all the people and places that have hosted performances of the work described here.

Preface

Much of the material in this thesis has been previously published and presented in various Journals, conferences, festivals and concerts.

In particular, parts of Chapter 6 have appeared in:

- Eldridge, A. (2002b). Adaptive systems music: Algorithmic process as musical form. In *Proceedings of the 2002 Generative Art Conference*, Milan.

Some of the ideas in Chapter 3 and models from Chapter 5 were presented in:

- Eldridge, A. (2005b). Extra-music(ologic)al algorithms for automated composition. In Rothlauf, F. et. al., editors, *Evomusart workshops 2005, LNCS 3449*, Heidelberg. Springer- Verlag Berlin.

The work presented in Chapter 8 has been previously published in:

- Eldridge, A. (2005c). Fond punctions: Generative processes in live improvised performance. In Edmonds, E., Brown, P., and Burraston, D., editors, *Proceedings of Generative Art Practice*, pages 41-51, Sydney, Australia. Creativity and Cognition, Creativity and Cognition Studios Press. andChapter
- Eldridge, A. (2005a). Cyborg dancing: generative systems for man-machine musical improvisation. In Innocent, T., editor, *Proceedings of third Iteration.*, pages 129-142, Mel- bourne, Australia. Centre for Electronic Media Art (CEMA).

Results from the study presented in Chapter 4 were discussed in a different context in:

- Eldridge, A. (2006). Issues in auditory display. *Artificial Life*, 12(2):259-274.

All of these papers should be considered as my own work. In addition, a summary of the ideas presented here are due for publication in a co-written book chapter:

- Husbands, P., Copley, P., Eldridge, A. and Mandelis, J. (2007). An introduction to evolutionary computing for musicians. In Miranda, E. R. and Biles, A., editors, *Evolutionary Computer Music*. Springer-Verlag.

Many of the works presented here have appeared in public exhibitions, festivals and gigs as follows:

AdSyMII (Chapter 6)

- July 2002: Blip5, Bar Sumo, Brighton, UK
- December 2002: Concert of Generative Art 2002, Politecnico di Milano University, Milan, Italy

Organised Entry (Chapter 6)

- October 2005: The Big Blip 05, Brighton Fringe Basement, Brighton, UK

Self-Karaoke Pond (Chapter 8)

- October 2005: The Big Blip 05, Brighton Fringe Basement, Brighton, UK

Ashby's Grandmother's Footsteps (Chapter 7)

- April 2006: Process Revealed at the Artpool Gallery, Budapest, Hungary (in association with the EvoMusArt workshop at EuroGP 2006.)

Performances of *Fond Punctions*, which uses the **Self-karaoke Machine** described in Chapter 8, include the following events and venues:

- July 2005: *Interactive Mind and Art(efacts)*, The Sussex Arts Club, Brighton, UK
- August 2005: The Great Escapade, Sussex, UK
- September 2005: Live Algorithms for Music meeting, The Great Hall, Goldsmiths college, London, UK
- October 2005: Wrong music, The Volks, Brighton, UK
- November 2005: University of Sussex Lunchtime Concert, Friends Meeting House, Brighton, UK
- November 2005: Third Iteration. Monash University, Melbourne, Australia
- December 2005: Generative Arts Practice. University of Technology, Sydney, Australia
- December 2005: Lan Franchis memorial discotheque, Sydney, Australia
- April 2006: Process Revealed, Artpool, Budapest, Hungary
- May 2006: Sunday Relay, The Albert, Brighton, UK

Compositions made with some of the systems here have been publicly played or distributed as follows:

- *Sines* (2002) a piece composed using the basic homeostat described in Chapter 5 was commissioned for the Lux OPEN 2002, Royal College of Art, London, UK
- *It Didn't Happen at Lan Franchis*, *Picket Fence Study 2: The Ant's ear view* and *Picket Fence Study 4: The Larvae's ear view* are due to appear on the cd accompanying *Evolutionary Computer Music*. (Miranda, E.R. and Biles, A.J. 2007). Springer-Verlag

Finally, the 'Behavioural Objects' project outlined in Chapter 9 was presented at:

- Improvising with Computers: IRCAM, Paris as part of NIME 2006

Collaborating with the Behaving Machine: Simple Adaptive Dynamical Systems for Generative and Interactive Music

Alice Eldridge

Summary

Situated at the intersection of interactive computer music and generative art, this thesis is inspired by research in Artificial Life and Autonomous Robotics and applies some of the principles and methods of these fields in a practical music context. As such the project points toward a paradigm for computer music research and performance which complements current mainstream approaches and develops upon existing creative applications of Artificial Life research.

Many artists have adopted engineering techniques from the field of Artificial Life research as they seem to support a richer interactive experience with computers than is often achieved in digital interactive art. Moreover, the low level aspects of life which the research programme aims to model are often evident in these artistic appropriations in the form of bizarre and abstract but curiously familiar digital forms that somehow, despite their silicon make-up, appear to accord with biological convention.

The initial aesthetic motivation for this project was very personal and stemmed from interests in adaptive systems and improvisation and a desire to unite the two. In simple terms, I wanted to invite these synthetic critters up on stage and play with them. There has been some similar research in the musical domain, but this has focused on a very small selection of specific models and techniques which have been predominantly applied as compositional tools rather than for use in live generative music. This thesis considers the advantages of the Alife approach for contemporary computer musicians and offers specific examples of simple adaptive systems as components for both compositional and performance tools.

These models have been implemented in a range of generative and interactive works which are described here. These include generative sound installations, interactive installations and a performance system for collaborative man-machine improvisation. Public response at exhibitions and concerts suggests that the approach taken here holds much promise.

Submitted for the degree of D.Phil.

University of Sussex

February, 2008

Contents

1	Introduction	17
1.1	Summary of contributions	21
2	All Computers are Interactive: But Some Are More Interactive Than Others.	23
2.1	Interaction in Live Computer Music	25
2.1.1	Metaphors from Traditional Performance Practice	26
2.1.2	New Musical Instruments	27
2.1.3	Responsive Accompanists	31
2.1.4	Virtual Improvisers	32
2.1.5	Audible Ecosystems	33
2.1.6	Summary	34
2.2	Interaction in Alife Installation Arts	34
2.2.1	Breeding Artificial Forms: Interacting with Evolution	35
2.2.2	Artificial Ecosystems: Interacting with an Environment	38
2.2.3	Single Synthetic Agents: Interacting with an Autonomous Other	40
2.2.4	Summary	41
2.3	Frameworks for Understanding and Implementing Interactive Systems	42
2.3.1	Design Schemes: Pipeline vs Circular Causality	42
2.3.2	Heteronomy vs Autonomy	45
2.3.3	Cognitive Science and the Computationalist Approach	45
2.3.4	Autonomy	46
2.3.5	Dynamical Approaches	47
2.3.6	Bottom Up Design	49
2.4	Summary and Implications for Design	49
3	Computer Automated Algorithmic Composition in Research and Practice	53
3.1	Techniques	55
3.1.1	AI Methods of Embodying Musical Knowledge	55
3.1.2	Techniques Employed in Exploration of New Musics	60
3.2	Aims and Inspirations: Ambiguity in Cross-disciplinary Enterprises	65
3.2.1	Implications for Research Methodologies	66
3.2.2	Implications for Software Design	67
3.3	Performance Issues	73
3.3.1	Aims: Automation – Augmentation – Autonomy	74
3.3.2	Adaptability vs Dependability: Inspiration vs Freedom	76
3.4	Discussion	78
4	Mimesis, Alife Art and Music	80
4.1	Seeing Artificial Life	81
4.2	Hearing Alife	84
4.3	Testing the Auditory Perception of CA States	85
4.3.1	Auditory Perception and Auditory Display	85
4.3.2	The Effect of Musical Experience on Perceptual Acuity	87
4.3.3	Design Rationale	87
4.3.4	Method	88

4.3.5	Results	91
4.3.6	Musical Experience Effects	92
4.4	On Mapping and Model Selection	93
4.5	Summary	96
5	Studies in Simple Adaptive Dynamical Systems	98
5.0.1	Models	98
5.0.2	Mappings	99
5.1	Adaptation and Homeostasis	101
5.1.1	A Model of the Homeostat	104
5.1.2	Homeostat Behaviour	104
5.1.3	Example Mappings from the Homeostat	105
5.1.4	Summary of Homeostat Features	108
5.2	Entrainment in Neural Oscillators	108
5.2.1	A Neural Oscillator Model	109
5.2.2	Neural Oscillator Behaviour	110
5.2.3	Example Mappings from the Neural Oscillator	110
5.2.4	Summary of Neural Oscillator Features	112
5.3	Pattern Propagation in Cellular Automata	113
5.3.1	Summary of CA Features	114
5.4	Ecology Models	114
5.4.1	N-species Lotka-Volterra Model	114
5.4.2	A Simple Agent-based Model of Spectral-temporal Organisation	117
5.5	Implementation	119
5.6	Discussion	120
6	Composing Generative Systems	122
6.1	AdSyMII	122
6.1.1	A Self-regulating Homeostatic Network	123
6.1.2	System overview	124
6.1.3	AdSyMII System Components	125
6.1.4	AdSyMII Reactions and Discussion	128
6.2	Organised Entry	129
6.2.1	Organised Entry: System Overview	129
6.2.2	Organised Entry: System Components	129
6.2.3	Organised Entry: Reactions and Discussion	130
6.3	Summary	132
7	Ashby's Grandmother's Footsteps: an Interaction Installation	133
7.1	Power and Play in Responsive Environments	134
7.1.1	Responsive Environments	134
7.1.2	Designing Feedback	136
7.2	Design and realisation	137
7.2.1	Overview	138
7.2.2	Composition and Implementation	138
7.2.3	Technical Details	141
7.3	Reactions and Discussion	142

8	Self-karaoke Machines: Collaborative Man-Machine Improvisations	144
8.1	Generative Processes in Live Improvisation	145
8.2	Fond Punctions	146
8.2.1	System Overview	148
8.2.2	Component Details	148
8.2.3	Personal Reflections and Audience Reactions	152
8.2.4	Self-karaoke Pond in Installation	153
8.3	Live Improvisation in Generative Systems	156
8.3.1	Creative Constraints in Generative Systems	156
9	Recapitulation and Future Development	160
A	Supplementary Results for Chapter 5	176
A.1	Neural Oscillator	176
A.2	Self-regulating homeostat	177
B	Technical set up for installations and performances	178
B.1	Set up for Ashby's Grandmother's Footsteps	178
B.2	Technical Set up for Self-karaoke in installation and performance	179

Track Listings

The attached DVD contains:

- Max/ MSP externals for the main algorithms described in Chapter 5. Also included are help files using mappings similar to those described in this chapter.
- A video of one performance made with the Self-karaoke Machine described in Chapter 8.
- Audio examples described in Chapters 5, 6, 7 and 8 are provided on the following tracks:

1	[5:1]	hom-perturb
2	[5:2]	hom-stabilise
3	[5:3]	hom-sines
4	[5:4]	planting-trees-excerpt
5	[5:5]	hom-sam-remix
6	[5:6]	hom-filter
7	[5:7]	hom-wrongbeats1
8	[5:8]	hom-wrongbeats2
9	[5:9]	nosc-change-weights
10	[5:10]	nosc-minima
11	[5:11]	nosc-change-inputs
12	[5:12]	nosc-perc-simple
13	[5:13]	nosc-perc-delta-tau
14	[5:14]	CA-chaotic
15	[5:15]	CA-complex
16	[5:16]	CA-ordered
17	[5:17]	GLV-inuit-pitch
18	[6:2]	AdSyMII
19	[6:3]	Organised Entry
20	[7:1]	Ashby's Grandmother's Footsteps
21	[8:1]	It Didn't Happen at Lan Franchis
22	[8:2]	Picket Fence Study One: The Blackbeetle's Ear View
23	[8:4]	Picket Fence Study Three: The Ant's Ear View
24	[8:3]	Picket Fence Study Two: The Larvae's Ear View
25	[8:5]	Self-karaoke Pond-harp

Glossary: Abbreviations and resources

1D	- One Dimensional
2D	- Two Dimensional
AI	- Artificial Intelligence
ANN	- Artificial Neural Network
CA	- Cellular Automata
CTRNN	- Continuous Time Recurrent Neural Network
DSP	- Digital Signal Processing
EC	- Evolutionary Computation
GA	- Genetic Algorithm
GM	- General MIDI
MIDI	- Musical Instrument Digital Interface
OSC	- OpenSound Control (see below)
USB	- Universal Serial Bus

Arduino is an open-source physical computing platform based on a simple i/o board, and a development environment for writing Arduino software. The Arduino programming language is an implementation of Wiring, itself built on Processing.

<http://www.arduino.cc/en/>

Chuck is a concurrent, strongly-timed audio programming language for real-time synthesis, composition, and performance, which runs on Mac OS X, Linux, and Windows. Code can be added, removed and modified on the fly, while the program is running making it an ideal language for live coding. It was originated by Perry Cook and Ge Wang of Princeton University.

<http://chuck.cs.princeton.edu/>

Csound is a text based music programming language written in the C programming language. A typical Csound program will include an *orchestra* file describing the nature of the instruments and a *score* file describing the parameters of the material (pitch, duration, amplitude etc). Csound then renders these files to produce an audio file or real-time audio stream.

<http://www.csounds.com/>

Jitter extends the Max/MSP programming environment to support realtime manipulation of video, 3D graphics and other data sets within a unified processing architecture.

<http://www.cycling74.com>

- Max/MSP is a graphical development environment for music and multimedia. The program is highly modular and allows the development of third-party externals as objects which can be fully integrated with the native libraries. A typical Max programme, called a 'patch' is based on multiple graphical objects connected into a data flow. Control rate MIDI messages can be combined with a DSP network. Max was originally developed by Miller Puckette and is now developed and maintained by Cycling'74.
<http://www.cycling74.com>
- Processing is an open source programming language and integrated development environment (IDE) built for the electronic arts and visual design communities. It builds on the graphical side of Java, simplifying some features and adding new ones. It is developed by Casey Reas and Ben Fry
<http://www.processing.org>
- Pure Data (Pd) is a graphical programming language developed by Miller Puckette in the 1990s for the creation of interactive computer music and multimedia works. Though Puckette is the primary author of the software, Pd is an open source project and has a large developer base working on new extensions to the program. It is released under a license similar to the BSD license.
<http://puredata.info/>
- OpenSound Control is a protocol for communication among computers, sound synthesisers and other multi-media devices. It is optimised for networking technology allowing very fast data sharing between machines. It can transport over many protocols but is commonly used with UDP or TCP/IP. It can be compared to MIDI, but does not suffer the same time lags and allows an open-ended url-style symbolic naming scheme.
<http://www.cnmat.berkeley.edu/OpenSoundControl/>
- SuperCollider is a real time audio synthesis programming language. The Language combines the object oriented structure of Smalltalk and features from functional programming languages with a C programming language family syntax. Originating as proprietary software, it was released in 2002 by its author James McCartney under the free software GPL license.
<http://www.audiosynth.com/>
- Wiring is a programming environment and electronics i/o board for exploring the electronic arts, tangible media, teaching and learning computer programming and prototyping with electronics. It is an open project initiated by Hernando Barragàn and builds on Processing.
<http://wiring.org.co/>

List of Figures

1.1	Screen shot from the Game of Life, a CA rule set devised by John Conway.	19
2.1	Sonic Wire sculpture by Amit Pitaru, an AV interface exhibited at Ars Electronica 2005. (photo: Thomas Petersen)	29
2.2	Laetitia Sonami's Lady's Glove	30
2.3	Alex McClean (left) and Perry Cook and Ge Wang (right) live coding	31
2.4	Screen shots of the 'stages of evolution' of one form in a biomorph environment. Each image (left to right, top to bottom) represents a mutation of the previous.	35
2.5	Karl Sims' Genetic images on display at the Pompidou Centre, Paris (1993).	36
2.6	Examples of forms evolved in William Latham's Mutator.	37
2.7	Christa Sommerer and Laurent Mignonneau's A-Volve.	38
2.8	Jon McCormack's <i>Eden</i> in installation.	39
2.9	Simon Penny's Petit Mal in action.	40
2.10	Schematic of system design protocol in interactive computer music.	43
2.11	Basic design of Di Scipio's Audible Eco-Systemic Interface.	44
2.12	Mari Kimura performing with Eric Singer's Guitar Bot at NIME 2005	50
3.1	Hiller with colleagues in the Electronic Music Studios, University of Illinois (left), at the controls of the console (middle), and in front of the main-frame ILLIAC I (right).	54
3.2	Graphical representation of a one-dimensional CA and its rule set (bottom).	62
3.3	L-system generated plant forms.	63
3.4	A Natural harmonisation of a phrase from a simple folk tune using <i>I</i> (tonic), <i>V</i> (dominant) and <i>IV</i> (sub-dominant) (top) and appropriate chordal inversions (<i>IV^b</i> etc) (bottom).	71
3.5	Standard harmonisation of a phrase from a Chorale (A) and examples from Toch's alternative harmonisations (B) and (C).	72
3.6	Screen shot of an Autechre Max patch.	74
4.1	Visualisation of simulated foraging behaviour. Dale (2000).	82
4.2	One of a set of 28 prints made with the <i>Tissue</i> software. Casey Reas 2002.	83
4.3	Examples of visual stimuli for chaotic (left) complex (middle) and ordered (right) rule sets.	88
4.4	Rhythmic mapping: cell states are transformed to musical events: 1 = play, 0 = rest. Four lines are voiced simultaneously	89
4.5	Harmonic mapping for the CA.	90
4.6	Mean scores and standard deviations for each group across all conditions	91
4.7	Mappings used in CAMus. The cartesian coordinates of a live cell are mapped to a triple (top). Each iteration of the rule set produces a number of such chords (bottom).	95
4.8	Evolution of CA used in Chaosynth: initial random distribution of cells (right) evolves to an oscillatory pattern (left).	96
4.9	3D wire sonogram showing evolution of spectra from initial white noise to 2nd, 3rd and 4th harmonic for a Mridangam (east Indian drum) stroke.	96

5.1	Outline of the mapping techniques explored. Outputs of the model are used to: generate material directly creating either single lines (a), multiple different but related lines (b); trigger pre-existing sonic material (c); or to parameterise some other DSP process (d).	100
5.2	Ashby's electro-mechanical homeostat.	101
5.3	Diagram of part of the homeostat circuitry from Ashby's notebook.	102
5.4	Schematic of a fully connected four unit homeostat. Each unit is represented as a square box, its output being the deviation of the small arrow from the centre. Weighted connections between units are represented by the uni-directional arrows which link each unit.	105
5.5	Outputs of a four unit homeostat demonstrating stability to minor perturbation (a) and re-stability after critical perturbation (A).	106
5.6	Change in stability as a function of number of units and $1/\text{viscosity}$ (left) and as a function of connectivity (right). Stability is taken as the point at which all units remain inside limits (and therefore weights remain constant), and measured as the number of iterations taken to achieve this state, averaged over 200 runs.	106
5.7	Schematic of a neural oscillator node. The oscillator equations simulate two neurons in mutual inhibition as shown here. Black circles correspond to inhibitory connections, open to excitatory. The mutual inhibition is through the $\gamma[x_i]^+$ connections ($[x]^+ = \max(x, 0)$), and the βv_i connections correspond to self-inhibition. The input g_j is weighted by a gain h_j , and then split into positive and negative parts. The positive part inhibits neuron 1, and the negative part neuron 2. The output of each neuron y_i is taken to be the positive part of the firing rate x_i , and the output of the oscillator as a whole is the difference of the two outputs.	109
5.8	Mapping from a continuous output to quantised notes. The section of score represents the graphed output within the box and <i>above</i> the horizontal line only. Notes are only re-voiced if they have changed by more than a semi-tone across timesteps, creating the spaces shown here as rests.	111
5.9	Outputs of three nodes in series (left) and detail, showing beats triggered (right): sinusoidal input and node three are triggered at local maxima, node two at zero-crossings (falling and rising) and node one at local minima.	112
5.10	Graphical representation of 1D CAs: chaotic (left), complex (middle) and ordered (right).	113
5.11	Period doublings in a three-species Lotka-Volterra system: phase space on the left and x_i on the right.	116
5.12	Figures showing distribution of agents in each pitch class (left) and across time slots (right) for system with no lifetime pitch variation.	118
5.13	Figures showing initial distribution of agents in pitch-time space (left) and final distribution (right).	119
5.14	Figures showing distribution of agents in each pitch class (left) and across time slots (right) for system with lifetime pitch variation.	119
5.15	Figures showing distribution of agents in each pitch slot (left) and across time slots (right) for system with lifetime pitch variation. At iteration 250, four agents of pitch class five, onset time 80 were introduced	120
6.1	Diagram of homeostat network showing full intraconnections and 10% interconnectivity	123
6.2	Schematic of the three process loops in AdSyMII and outputs produced.	124

6.3	Transcript of the simple random loop. In this example the notes G \sharp , E, A and G are voiced every 4,4,2 and 11 semiquavers respectively. In the actual system these pitches are defined by specific homeostat units.	125
6.4	Graphical depiction of the 1D CA used (above) and rule set (below)	126
6.5	Transcription of the three separate lines generated by the system. The top lines shows the sustained microtonal chords generated directly by the outputs of four units the homeostat network. The second line show the 'melody' line picked out of these chords at fixed time intervals. The bottom four staves show the rhythm part generated from four lines of the CA shown in Figure 6.4	127
6.6	Schematic of the organisation of models and mappings in Organised Entry.	129
6.7	Three-species Lotk-Volterra equation as mixer. The horizontal lines represent the hand-set cut-off values. As the population density of each species exceeds the cut-off, the species-density is used to specify the amplitude of the associated part. Shaded areas signify the times when each part is audible.	131
7.1	Visitors treading carefully in Ashby's Grandmother's Footsteps at Artpool, Budapest, 2006.	133
7.2	Artificial Reality space of Myron Krueger's <i>Critter</i> from <i>Videoplace</i>	135
7.3	David Rokeby in Very Nervous System in the street in Potsdam (left) and a schematic of the basic operational loop (right).	136
7.4	Schematic showing basic feedbacks between user, sensors, mediating devices and sound.	138
7.5	A screen shot from Jitter showing the visual effect of the frame differencing algorithm which is used for motion detection.	141
8.1	Fond Punctions performance at Third Iteration, Melbourne (2005).	144
8.2	Setting up for a Fond Punctions performance at Artpool. Budapest	146
8.3	Performance of Fond Punctions in the Friends Meeting House, University of Sussex.	147
8.4	Schematic showing the network of influences between components in the whole performance system.	148
8.5	The outputs of each unit in the homeostat (O_n) are used to determine the point p in the sample that the grain is taken from.	149
8.6	Screen shot of the video projections in the Self-karaoke Machine.	151
8.7	Visitors to the The Big Blip 05 playing in the Self-karaoke Pond.	154
8.8	Diagrammatic instructions for Self-karaoke Pond.	155
8.9	Some very small people playing in the Self-karaoke Pond at The Big Blip 05.	156
8.10	Overview of the interactions and influences in the generative process. (with kind permission from McCormack (2004)). The user's influence on the final outcome is constrained by the designer's decisions over the genotype, enaction process and material from which the phenotype is formed.	158
A.1	Plots showing the output of a neural oscillator pair and sinusoidal input signal for varying amplitudes of input. (from Williamson(2002))	176
A.2	Outputs of a ten-unit self-regulated homeostatic network demonstrating stability to minor perturbation and re-stability after critical interference.	177
A.3	Change in stability as a function of interconnectivity for an auto-regulated network	177
B.1	Set up for Ashby's Grandmothers Footsteps installation	178

B.2	Set up for Fond Punctions Performances	179
B.3	Set up for Self-karaoke Pond in installation	180