Chapter 9

Recapitulation and Future Development

This thesis aimed to demonstrate the appeal and application of simple adaptive systems as mechanisms for composing and improvising with computers. The work here is inspired by the principles and practices of Alife and autonomous robotics research and suggests a novel approach to the design and implementation of computer music software which complements mainstream AI-inspired approaches.

The choice of models stemmed from a consideration of the interactive and generative potential of existing creative applications of Alife models tempered by the requirements of the musician. The main projects centred around a model of an Ashbian homeostat. Aside from being somewhat of an Iconic device, the model implemented here is one of the simplest that can be seen to exhibit goal-directed and self-modulated adaptive dynamics, achieving what we might describe as minimally autonomous behaviour. The projects presented here in no way 'prove' this, but it is felt that these characteristics imbue a certain liveliness and coherence in both its dynamical output and its response to external events which come across when the streams of floating point numbers are used to specify sonic events. These attributes support an aesthetic which lies at the meeting point of digital and acoustic idioms to produce musical outputs which are "weird and surprising but strangely familiar".

Whether or not this claim is acceded to or this aesthetic is appealing to the reader, these types of adaptive dynamical systems seem to answer the callings of those wishing to indulge in live generative performance whilst retaining some modicum of respect, i.e. to achieve a balance of adaptability and dependability, of inspiration and obedience. The homeostat for example offers constant variation within a known behavioural field: it will invariably change dramatically when you poke it hard, and deviate only temporally when you prod it lightly. But what it will change to, or how long it will take to return are unknowns. Similarly you cannot control its behaviour directly, but the viscosity variable offers a form of global-control which allows you to reign it in or stoke it up as necessary. This is not meant to imply that the homeostat itself is the future of computer music, but it is a convenient device with which to illustrate a move toward a collaborative approach to algorithmic composition and performance.

The basic behaviours of the homeostat are indeed quite limited and there are many other possibilities yet to be explored. The relative simplicity of the device was another of its attractions as a starting point. Unlike many complex dynamical systems it can be readily parameterised by hand. One of the problems with the neural oscillator networks as implemented here was that the parameter space is so vast that whilst it is possible to discover some parameter sets which generate dynamics of musical use this was a slightly arduous process which relied largely on serendipity. The obvious next move in terms of systems design then is to explore more powerful systems and employ some form of artificial evolution to aid parameter search.

Hand designing complex systems in some ways goes against the principles of Alife. The observation that complex environments play a significant role in the creation of apparently complex behaviour leads to the use of artificial evolution for engineering the agent-side mechanisms necessary to deal with the environment. An agent can be given eyes and legs and told to follow walls, and artificial evolution will duly contrive the requisite sensori-motor connections. But as noted in Chapter 2, framing musical attributes in a formal fitness function is hard: 'play something interesting and relevant' is a decidedly ill-posed problem. The common alternative, interactive evolution, where a formal fitness function is replaced by a subjective choice on the part of the listener, is similarly fraught with difficulties, leading to a body of research on the 'fitness bottleneck'. Never-the-less, EC remains an attractive search method, particularly for setting parameter values for the sorts of complex systems explored here. The neural oscillator or broader CTRNNs models are full of musical promise, but exploring their parameter space by hand as done here, we can only get a peek at their full behavioural repertoire. An exciting possibility is that having engineered a minimally autonomous system, its response characteristics *could* be privy to formalisation. Rather than having to specify minutiae of musical details, evolution could be used to design fields of behaviour.

To develop these ideas further, it is also necessary to go beyond personal implementations and test the utility of these tools by getting other musicians to play with them. These needs are met in the form of a burgeoning collaborative project, *Behavioural Objects*. Working with computer scientist and musician, Ollie Bown and composer and improviser, Sebastien Lexer both of Goldsmiths College, UK, the project builds upon many of the ideas presented here and aims to present a compendium of adaptive dynamical objects for use in interactive computer music performance systems. Bown has recently built a CTRNN and has been using it in performance with trumpeter Tom Arthurs. Currently he has been using very simple fitness functions to ensure for example that outputs remain active. Working at this level, it seems entirely possible to be able to engineer simple relations with a performer. For example if pitch and/or onset analyses were fed into the network as inputs, it seems feasible that one could easily evolve a system to respond differentially to the performer using fitness functions such as 'play when I play', or 'remain active when I stop'. Such investigations are planned for the near future.

The project has also provided the opportunity to organise a focus group to explore other computer musician's and improviser's responses to these models. We plan to demonstrate ready made implementations of full interactive systems such as the basic Self-karaoke Machine, but more importantly to hand over the models in their raw form as Max/MSP objects and encourage other's experimentation. Feedback from this focus group will feed into planned future research which aims to move beyond the use of handcrafted behaviours and investigate methods for training and evolving behaviours.

At the start of this thesis were a list of reasons why it was an exciting time to be a musician. The last one of these is that it is now common practice to share ideas in the form of lines of code. At the end of the last chapter it was suggested that one answer to Dorin's concerns over the creative limitations of some generative systems for non-programmer users was to introduce the performer – and thus the vagaries of the real-world – into the enaction process, as in the Self-karaoke Machines. Programmes like Max/MSP and the social structures of its community of users mean that we no longer need to specify any form of enaction mechanism to share a generative idea. We can just post an ant and people can make their own eyes and legs and design their own beaches. Generative mechanisms are not just a new way of writing and performing music, but a new way of sharing and expressing musical ideas: a new musical currency.