Connectionist Models for Real-Time Control of Synthesis and Compositional Algorithms

Michael Lee and David Wessel

Center for New Music and Audio Technologies
University of California, Berkeley
Berkeley, CA 94710
e-mail: wessel@cnmat.berkeley.edu & lee@cnmat.berkeley.edu
telephone: 1 510 643 9990

ABSTRACT

Connectionist models provide rich and trainable control structures for generative algorithms. In this report we describe how multilayer neural networks trained by back-propagation can be effectively used to transform performance gestures into control parameters. We have enhanced the MAXNet neural network simulator by giving it the capability to construct networks from a graphical specification. This capability facilitates experimenting with networks with architectures richer than the conventional feed forward fully connected networks. With this new description language, we have built controller networks based on forward models for control. Forward modeling derives the controller by back-propagating errors through an emulator, thus reducing the search space from that of a direct inverse technique. We show how a user can map parameters obtained from a personalized gesture space to the control parameters of a synthesis engine. We discuss these forward modeling network architectures and training strategies. We also wish to emphasize that intelligent preprocessing of gestural data and perceptually based representations of sound are critical determinants of the performance of such network based control structures. Our examples include live performance control where the performer makes gestures in a low dimensional perceptually based timbre space and controls either FM, Resonant Synthesis, or Waveguide Synthesis.

1. INTRODUCTION

Advances in modern control theory suggest that neural networks are effective for the identification and control of nonlinear systems (Narendra & Parthasarathy 1990). This report explores some of the general features of musical control systems based on neural networks that can be trained to respond in specific ways to specific gestures of the performing musician.

1.1 A control theory framework

In our view, controllers map musical intentions to the parameters of a synthesis or compositional algorithm. To place things in a practical context, we present a brief overview of the components of a control system involved in an actual performance. The performer interacts with sensors that capture features of his gestures. These are transformed into system parameters that control the musical process, be it synthesis or a more temporally expansive procedure. At the control level concerned with the synthesis of sound the performer behaves much like a traditional instrumentalist whereas the control of compositional procedures is more akin to driving, guiding, or conducting. The general control paradigm in Fig. 1 shows that a significant part of the control system lies within the motor program of the performer. In fact, the development of a mapping of musical intention to instrument control parameters traditionally requires large doses of motor skill learning and, as yet, has not been facilitated by adaptive automatic control mechanisms within the instrument itself. In this report we will concentrate on the development of control components contained within the music making device. We will assume that proper attention has been paid to the ergonomic aspects of the gestural interface and will examine the problem of mapping the data from the gesture sensors to the parameters of the generative algorithm. We hasten to point out that any real system put to use for musical performance will involve motor learning. The examples that we have examined thus far are ones in which the performer specifies a mapping between intention and his gestures. For example, the performer could indicate a place in timbre space with a pointing device or specify the significance of hand movements. Our emphasis here is on the machine adapting to the specified gestures of the performer rather than on the traditional situation of the performer adapting to the instrument.

The generative algorithm typically has a large number of input parameters. This is true of both synthesis and compositional procedures. The gestural interface typically supplies fewer parameters and so we require that the controller map from a low to a high dimensional space. This low to high dimensional mapping does not necessarily pose any particular complication and is a prominent aspect of the computer music medium.

The generative algorithm itself is most often poorly understood in spite of the fact that it may be well specified and deterministic. This is due first to the fact that most complex synthesis and compositional algorithms are highly nonlinear dynamical systems. Second, outcomes are evaluated by hearing and so the generative algorithm must be characterized in terms of how it transforms its control parameters to perceptually relevant properties. The characterization of the generative system is a system identification problem and, in fact, a model of the generative system can play a critical role in determining its controller. In the contemporary control theory literature (Miller *et al* 1990, Jordan and Rumelhart 1992) this model of the generative component, or plant as it is called, is the forward model. The inverse of the forward model is the controller.

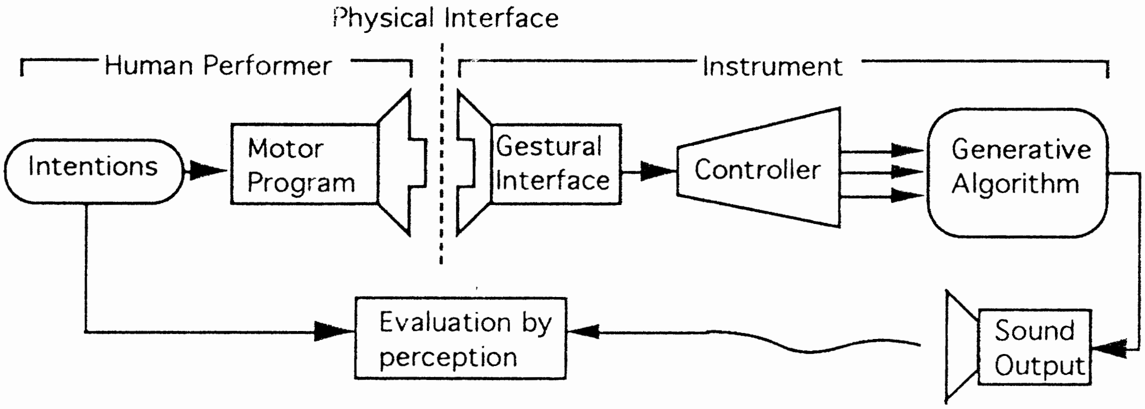

Figure 1: A general control theory framework for computer based musical instrumentation. The notion of intention can be interpreted as the desired outcome of the generative algorithm. The evaluation of the correspondence between intention and outcome is perceptual.

1.2. Neural networks and control systems

Referring again to Fig. 1, a multilayer or recurrent network can be used as the controller and another network can be used as the forward model of the generative algorithm. While our work stresses the use of multilayer networks trained by the back propagation supervised learning algorithm and its variants (see (Hertz *et al* 1991) for a review), other network architectures and learning procedures should be explored. We have implemented multilayer and recurrent networks in the MAXNet neural network simulator (Lee *et al 1991)* that runs in the MAX programming environment (Puckette & Zicarelli 1990). The real-time scheduling and the ease with which we can introduce neural networks into MAX patches that handle data-acquisition from gestural input devices and control synthesis running on multiprocessor DSP hardware interfaced directly to the Macintosh Nu-bus has made it possible to explore the use of neural network control in real musically viable live performance contexts.

1.3. An elementary vowel singer example

For purposes of explanation we present a concrete though quite elementary example of a model singer that produces vowels with articulatory synthesis like that developed by Perry Cook (Cook 1990). As illustrated in Fig. 2, the model singer states intention by indicating the vowel to the controller. The controller configures the vocal tract and operates the articulators to produce the acoustic output corresponding to the desired vowel. Evaluation of the performance is carried out by using a perceptually based norm to measure the difference between the ideal spectrum for the intended vowel and the spectrum of the synthesized vowel. In this example both the forward model of the vocal tract and the controller are implemented as feed-forward neural networks that learn by back propagation of error. The forward model of the vocal tract synthesis procedure is learned by presenting input-output pairs to a supervised learning procedure. The input to the forward model network consists of vocal tract configuration parameters and the corresponding output is a spectrum. The controller is an inverse of this forward model in that it takes vowel specification as input and generates the corresponding vocal tract configuration parameters.

There are two problems that arise in the determination of the inverse model. First, the forward model is many-to-one in that there are different configurations of the vocal tract that map to the same vowel spectrum. Second, there is no guarantee that the set of vocal tract configurations that map to the same vowel is convex. The many-to-one mapping in the forward model makes it difficult to determine a unique inverse model. If we were to attempt to determine the controller by direct inverse modeling, that is, by training the controller with input-output pairs consisting of spectrum as input and vocal tract configurations as output, back propagation learning would provide an average of the non-unique vocal tract configurations as output to a given spectrum input. And since the set of vocal tract configurations may be non-convex, this average vocal tract configuration will not necessarily be in the set of configurations that produce the given spectrum as output. Back propagating the error through the forward model goal directs the optimization of the controller (Jordan and Rumelhart 1992).

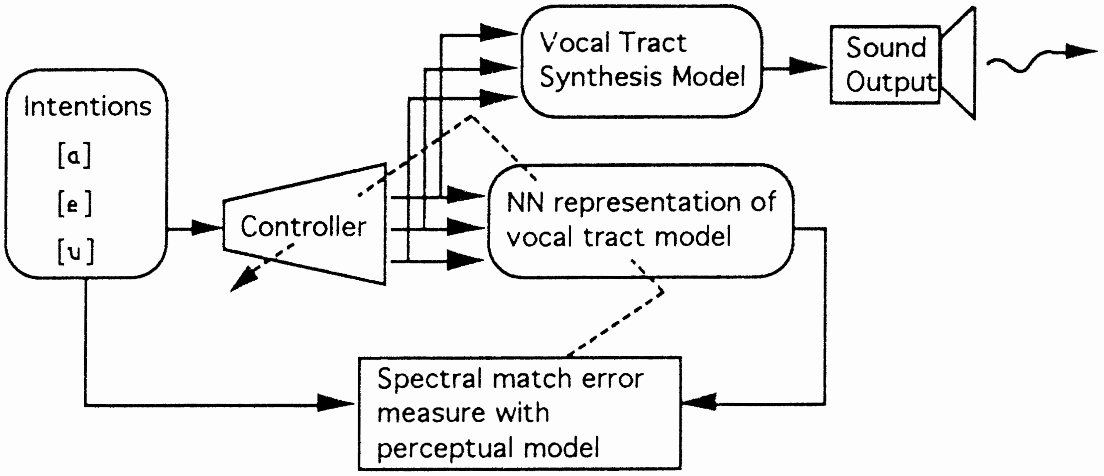

Figure 2: Intentions are specified by passing vowel names to the neural network controller which produces vocal tract configuration control parameters. A neural network forward model emulates the vocal tract behavior and is used to generate outcomes that can be matched against the desired outputs or intentions. The dotted line indicates back propagation of error through the forward model to constrain the controller.

Though we have not experimented extensively, at the time of writing, with neural networks for articulatory control of the vocal tract, Jordan (personal communication) has obtained impressive results using forward modeling of the sort just described for the development of an articulatory model of speech production.

2. DETERMINING THE FORWARD MODEL

The musical imagination can effectively be brought into play in specifying the nature of what is to constitute the forward model of a musical process. The forward model need not have a direct sonic referent like the spectrum in the previous example. At CNMAT we have experimented with forward models of rhythm pattern generators that provide perceptually based metrical abstractions to be evaluated against the same metrical abstractions as input intentions. Music theoretic concepts were used to characterize how the rhythms would map to the metrical abstractions. We turn now to another form of the forward model based on the timbre space model.

2.1. Timbre space as a forward model

Considerable attention has been paid to the psychoacoustics of timbre using geometric models as perceptual spaces (Wessel 1973, 1985, Grey 1975, Krumhansl 1989). The idea of this approach is to represent the various timbres as points in some geometric space, typically Euclidean. Proximate timbres in the space sound similar and the timbres that are far apart sound different. The space is usually generated by using a multidimensional scaling technique (Shepard 1974) that begins with a matrix of dissimilarity judgments made by comparing the members of all pairs of tones drawn from the set. These dissimilarity judgments are modeled as distances in the perceptual space and a configuration of timbres is sought that minimizes an error measure between the subjective dissimilarities and the distances. The musical motivation of this work is to produce a model of timbre that provides navigational advice about the compositional manipulation of timbre, and guides the development of a low dimensional control strategy for synthesis by determining the perceptually salient dimensions common to the set of timbres. Viewed as a forward model, as illustrated in Fig. 3, a timbre space provides a mapping between synthesis control parameters and coordinates of a given timbre in the perceptual space. For control of timbre, the inverse of the forward model is required. That is to say, the musician indicates the intended timbre by specifying its location in timbre space and the inverse model or controller generates the appropriate parameters for the synthesis algorithm. Timbral interpolation is thus provided by the controller network.

It is helpful to add additional constraints during the training of the controller. A particularly useful additional constraint for the timbre space controller is maintenance of constant loudness as one moves about.

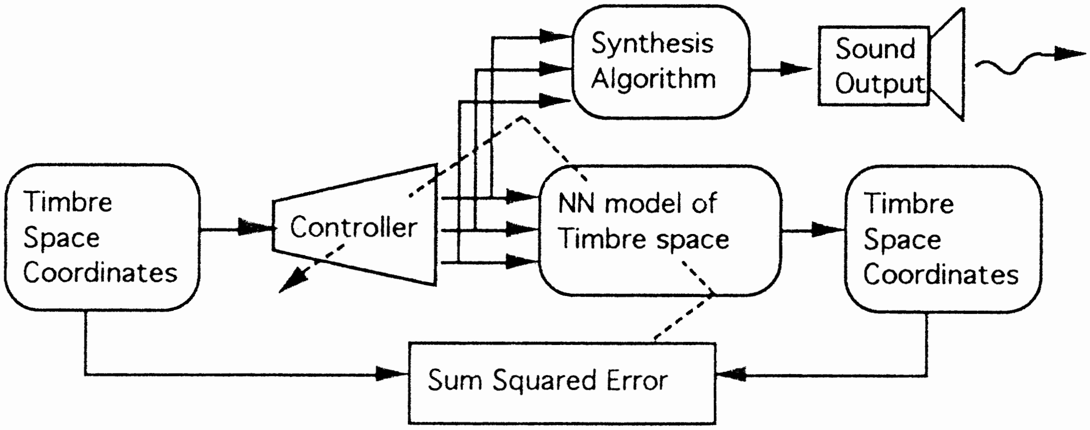

Figure 3: A controller based on timbre space as a forward model. Intentions are expressed as coordinates in timbre space. The error measure used in training the controller is like that of an auto associator. The dotted line indicates back propagation of error through the forward model to constrain the controller.

3. ACKNOWLEDGMENTS

This work was supported in part by grant INV 9004-011 from Apple Computer Inc. We would also like to thank Silicon Graphics Inc. for the Iris Indigo that greatly speeded network learning. Thanks as well to Guy Garnett and Leslie Delehanty for their support.

4. REFERENCES

Cook, P. R. (1990) "Identification of Control Parameters in an Articulatory Vocal Tract Model, with Application to the Synthesis of Singing", Ph.D. Dissertation, Elec. Eng. Dept., Stanford University.

Grey, J. M. (1975) An Exploration of Musical Timbre. Ph.D. Dissertation., Department of Psychology, Stanford University. CCRMA Report STAN-M-2.

Hertz, J., Krough, A. & Palmer, R. G. (1991) *Introduction to the Theory of Neural Computation.* Menlo Park, CA: Addison-Wesley.

Jordan, M. I. & Rumelhart, D. E. (1992) "Forward models: Supervised learning with a distal teacher", *Cognitive Science* (in press).

Krumhansl, C.L. (1989) "Why is Musical Timbre so hard to understand?" in Nielzen, S. & Olsson, O. (eds) *Structure and Perception of Electroacoustic Sound and Musicc.* NY: Excerpta Medica.

Lee, M., Freed, A., & Wessel, D., (1991) "Real-time neural network processing of gestural and acoustic signals". *Proceeding of the International Computer Music Conference 1991.* International Computer Music Association.

Miller, W. T. III, Sutton, R. S., & Werbos, P. J. (1990) *Neural Networks for Control.* Cambridge MA: MIT Press.

Narendra, K. & Parthasarathy, K. (1990) "Identification and control of dynamical systems using neural networks", *IEEE Transactions on Neural Networks,* Vol. 1, no 1, March.

Puckette, M. & Zicarelli, D., (1990) *MAX - An Interactive Graphic Programming Environment,* Opcode Systems, Menlo Park, CA.

Shepard, R. N. (1974) "Representations of structure in similarity data: Problems and prospects. " *Psychometrika* 39: 373-421.

Wessel, D. (1973) "Psychoacoustics and music." *Bulletin of the Computer Arts Society* 30: 1-2.

Wessel, D. (1985) "Timbre space as a musical control structure." in Roads, C. & Strawn, J. (eds) *Foundations of Computer Music* . Cambridge, MA: MIT Press.

Wessel, D. (1991) "Instruments that learn, refined controllers, and source model loudspeakers." *Computer Music Jouranal,* Vol. 15, No. 4, Winter , 82-86.