



Robert Rowe

# MACHINE MUSICIANSHIP



# ***Machine Musicianship***

*Robert Rowe*

*The MIT Press  
Cambridge, Massachusetts  
London, England*

# ***Machine Musicianship***

© 2001 Massachusetts Institute of Technology

All rights reserved. No part of this book may be reproduced in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.

This book was set in Melior by Achorn Graphic Services, Inc., and was printed and bound in the United States of America.

Library of Congress Cataloging-in-Publication Data

Rowe, Robert.

Machine musicianship / Robert Rowe.

p. cm.

Includes bibliographical references and index.

Contents: Machine musicianship—Symbolic processes—Sub-symbolic processes—Segments and patterns—Compositional techniques—Algorithmic expression and music cognition—Interactive improvisation—Interactive multimedia—Installations—Directions.

ISBN 0-262-18206-8 (hc. : alk. paper)

1. Artificial intelligence—Musical applications. 2. Music—Computer programs.

3. Real-time programming. 4. Computer composition. I. Title.

ML74.R68 2001

780'.285—dc21

00-038699

# Contents

Acknowledgments	ix
<b>1 Machine Musicianship</b>	<b>1</b>
1.1 The Motivation for Machine Musicianship	3
1.2 Algorithmic Composition	6
1.3 Algorithmic Analysis	8
1.4 Structure of the Text	12
1.5 Machine Musicianship Library	14
<b>2 Symbolic Processes</b>	<b>17</b>
2.1 Chord Theory	18
2.1.1 <i>Triad Classifier</i>	19
2.1.2 <i>Representations</i>	28
2.1.3 <i>MIDI Chord Recognizer</i>	33
2.1.4 <i>Chord Spelling</i>	42
2.2 Context Sensitivity	46
2.2.1 <i>Virtual Pitch</i>	47
2.2.2 <i>Determination of Chord Type</i>	58
2.3 Key Induction	60
2.3.1 <i>Interaction with Key Induction</i>	60
2.3.2 <i>Parallel Processing Model</i>	66
2.4 C++ and Object Orientation	77
2.4.1 <i>The Note Class</i>	77
2.4.2 <i>The Event Class</i>	80
2.4.3 <i>The Listener Class</i>	85
2.4.4 <i>The ListenProp Class</i>	89

<b>3</b>	<b>Sub-symbolic Processes</b>	<b>93</b>
3.1	Neural Networks	93
	3.1.1 <i>Neural Network Key Induction</i>	98
	3.1.2 <i>Sequential Neural Networks</i>	101
3.2	Time Structures	110
	3.2.1 <i>Quantization</i>	112
3.3	Beat Tracking	122
	3.3.1 <i>Multiple Attractors</i>	125
	3.3.2 <i>Adaptive Oscillators</i>	130
	3.3.3 <i>Meter Induction</i>	135
3.4	Max Externals	139
<b>4</b>	<b>Segments and Patterns</b>	<b>145</b>
4.1	Segmentation	145
	4.1.1 <i>Grouping Preference Rules</i>	146
	4.1.2 <i>Gestalt Segmentation</i>	161
4.2	Pattern Processing	168
	4.2.1 <i>Dynamic Programming</i>	170
	4.2.2 <i>Intervallic Representation</i>	176
	4.2.3 <i>Segmentation and Matching</i>	182
	4.2.4 <i>Pattern Processors</i>	187
4.3	Auditory Models	191
	4.3.1 <i>Auditory Input and Self-Organizing Maps</i>	192
<b>5</b>	<b>Compositional Techniques</b>	<b>201</b>
5.1	Generation Techniques	203
	5.1.1 <i>Selection Principles</i>	204
5.2	Score Following and Algorithmic Signal Processing	212
	5.2.1 <i>Pitch Tracking</i>	214
	5.2.2 <i>Jupiter</i>	215
	5.2.3 <i>Spectral Analysis and Control of Interaction</i>	221
	5.2.4 <i>Rhythmic Effects in Improvisation</i>	227
5.3	Standard MIDI Files	232

<b>6</b>	<b>Algorithmic Expression and Music Cognition</b>	<b>235</b>
6.1	Music Cognition and Machine Musicianship	235
6.1.1	<i>Expert Systems</i>	237
6.2	Knowledge Representation	241
6.2.1	<i>Schemata</i>	241
6.2.2	<i>Scripts and Plans</i>	245
6.3	Learning	246
6.3.1	<i>Genetic Algorithms</i>	248
6.3.2	<i>Artificial Agent Societies</i>	258
6.4	Expressive Performance	264
6.4.1	<i>Rule-Based Systems</i>	265
6.4.2	<i>Learning Expression</i>	270
<b>7</b>	<b>Interactive Improvisation</b>	<b>277</b>
7.1	Sequences in Improvisation	279
7.1.1	<i>SeqSelect</i>	280
7.2	Influence on Improvisation Processes	287
7.3	Transformation in Improvisation	297
7.4	Player Paradigm Systems	301
7.5	Ensemble Improvisation	308
7.5.1	<i>Multiple Cypher</i>	310
<b>8</b>	<b>Interactive Multimedia</b>	<b>317</b>
8.1	Intimate Immensity	319
8.1.1	<i>Interactor</i>	321
8.2	A Flock of Words	325
8.2.1	<i>Woids</i>	330
8.3	In Transit	334
8.3.1	<i>Style Recognition</i>	334
8.3.2	<i>Chord Selection and Voicing</i>	338
8.4	Multimedia Improvisation	343
8.4.1	<i>Controllers</i>	343



8.4.2	<i>Composition Techniques</i>	346
8.4.3	<i>Continuous Controls</i>	351
<b>9</b>	<b>Installations</b>	<b>355</b>
9.1	Multimedia Installations	355
9.1.1	<i>Audio and Imagery</i>	356
9.1.2	<i>Large-Scale Interaction</i>	360
9.2	Animated Improvisation	362
9.2.1	<i>Improvisation with Scales</i>	365
9.2.2	<i>Improvising Melodic Lines</i>	370
9.3	Multimodal Environments	372
<b>10</b>	<b>Directions</b>	<b>377</b>
10.1	Research Synergy	377
10.2	Research Directions	378
	References	381
	Index	393

## ***Acknowledgments***

A text like this is never complete—I encounter new material that relates to it daily. My only comfort is that such a book is not and could never be comprehensive. To my knowledge, easily twice as many projects could have been included. I cannot claim methodological rigor in selecting which things received considerable attention, or little or none. Of course, I wrote most about the things I know best. Another main reason for including something was that I had access to materials—articles, source code, Max patches—that allowed me to document more than my impressions.

All of the work referred to here was supported by material provided by composers and researchers, and I am grateful to all of them for their help. The list of names is too long to recount here; the index probably best reflects its extent. Beyond providing grist, many of these colleagues also read what I had written and helped amend my errors. Mistakes that remain, of course, are all my doing.

I owe a great debt to the Music, Mind, Machine research group at the University of Nijmegen in the Netherlands. Group directors Peter Desain and Henkjan Honing invited me there for my sabbatical from New York University. I spent that time conducting research and writing this book; indeed, I could not have written it without the work I accomplished there. Many stimulating discussions with students and faculty helped hone my ideas and presentation. Particular thanks go to Richard Ashley, Dirk-Jan Povel, Piet Vos, Renee Timmers, Paul Trilsbeek, Yvonne Schouten, Hank Heijink, Huub van Thienen, Rinus Aarts, Chris Jansen, and of course Peter Desain and Henkjan Honing.

The sabbatical year that led to this book was granted to me by New York University, and my experience in the Music Technology

Program of the Department of Music and Performing Arts Professions there provided the preparation necessary to write it. Many people helped me by reading and commenting on various drafts. In particular I would like to thank Kenneth Peacock, George Fisher, Carlos Guedes, Paul Berg, Roger Dannenberg, Lawrence Ferrara, and Ilya Smulevich.

Earlier versions of certain parts of the text were published in *Contemporary Music Review*, Volume 13, Part 2, in an article entitled “Incrementally Improving Interactive Music Systems,” *Music Perception*, Volume 17, No. 4, in an article entitled “Key Induction in the Context of Interactive Performance,” and in a chapter in *Readings in Music and Artificial Intelligence*, edited by Eduardo Reck Miranda (1999).

I would like to thank my editor at The MIT Press, Douglas Sery, for his support, guidance, and patience through the whole process.

Thanks, too, to my wife Tamara, who managed the move of our family of five across an ocean and back, took care of the kids, took care of me, and gave me time even to think about undertaking this project. *Machine Musicianship* is dedicated with love to my three daughters, Charlotte, Abigail, and Miranda.

# ***Machine Musicianship***



# 1 *Machine Musicianship*

*Machine Musicianship* is both an exploration of the theoretical foundations of analyzing, performing, and composing music with computers, and a tutorial in writing software to pursue those goals. The theoretical foundations are derived from the fields of music theory, computer music, music cognition, and artificial intelligence. The intended audience includes practitioners in those fields, as well as composers and interested performers.

The training of musicians begins by teaching basic musical concepts, a collection of knowledge commonly known as musicianship. These concepts underlie the musical skills of listening, performance, and composition. Computer programs designed to implement any of these skills—that is, to make sense of what is heard, perform music expressively, or compose convincing pieces—can similarly benefit from a musician’s fundamental level of musicianship.

To be sure, there are many worthy computer music programs that have no basic musical knowledge at all. The usual technique is to implement thoroughly that part of musicianship required for the task at hand. Notation programs must know how many beats belong in a bar; sequencers must be able to transpose enharmonically. In this text we will explore how a more systematic foundation of musical knowledge can further extend such programs’ range of use as well as improve their communication with human musicians.

Consider a simple example of this level of functionality: music sequencers can transpose enharmonically, but they cannot execute a command such as “transpose the selected measures to the subdominant.” The reason for this limitation is that typical programs have

## 1.1 The Motivation for Machine Musicianship

Designing computer programs that will recognize and reason about human musical concepts enables the creation of applications for performance, education, and production that resonate with and reinforce the basic nature of human musicianship. Access to functions such as phrase boundary recognition makes possible operations that simply cannot be accomplished without such capabilities. The realization of norms for the expressive shaping of a phrase by a machine performer, for example, can only be applied once a phrase has been identified as a phrase in the first place. Further, realizing these concepts algorithmically allows us to augment human musicianship with processes and representations that only a computer could implement. A complete record of the program's "listening experience" is immediately available and can be used both to evaluate the algorithm's performance and to direct further analysis.

Beyond the pedagogical and practical value, I believe that there are compelling musical reasons to emulate human musicianship with computers. Readers may determine for themselves on the basis of extensive existing repertoire whether or not computer music programs have contributed to enduring compositions. Those who dismiss machine musicianship tend to argue that algorithmic composition programs (as one example) are more interesting technologically than they are musically. Another prominent source of dissatisfaction with the enterprise derives from a belief that the development of increasingly musical programs forms a real and growing threat to the livelihood of human musicians.

The need for better musicianship in music processing is relatively self-evident when contrasted with the aesthetic and ethical questions surrounding the use of automated composition and performance programs. Computers in music have made possible new kinds of creation at the same time that they have caused upheaval in the social and cultural practice of music making. Music programs are cheap, easy to use, and tireless. These attributes make it attractive to use them for many tasks that previously were performed by human

no access to a description of the music in terms of relative harmonic function. Such an extension would be quite straightforward for current sequencers—and there may even be some that do it—though I have not seen any. Even better would be a sequencer that could transpose to the subdominant without the user having to inform the program as to which tonal center is current. That facility is more computationally demanding, but still well within the reach of established algorithms. Examples such as these can be generated at will and doubtless have occurred to anyone who has used music software in any depth. The point is that such programs can become more useful simply by better accommodating the practices of fundamental musicianship.

This book explores the technology of implementing musical concepts in computer programs and how resulting applications can be used to accomplish tasks ranging from the solution of simple musical problems through live performance of interactive music compositions to the design and implementation of musically responsive installations and web sites. These concepts are programmed using both C++ and Max, a graphic programming environment developed by Miller Puckette and David Zicarelli (Dobrian 1997). Some experience with one or both of these is assumed if readers wish to extend the example programs on their own. The accompanying CD-ROM includes working versions of the examples, as well as source code and a hypertext document showing how the code leads to the programs' musical functionality.

*Machine Musicianship* is not intended as a programming tutorial, however. The processes described in these pages constitute a computational approach to music analysis, composition, and performance that may engage practitioners in those fields whether they are programmers or not. I present the practical examples with programming information in order to help those who wish to write their own, but they can also be used as stand-alone applications by those who do not. It is my hope that interested musicians may even profit from simply reading the text without any use of a computer at all.



musicians. None of these properties, however, have anything to do with the nature of the music being performed. In other words, a large part of the motivation for making music with computers is that computers are less troublesome to employ than people. This situation has had a dramatic effect on the economic prospects for musicians almost as profound as the proliferation of television and sound recording equipment since the 1940s. One can condemn this trend in a hand-wringing Luddite reflex, but the situation is unlikely to change except in the direction of ever greater reliance on machines.

There are other reasons to use computers in music, however, that have everything to do with the nature of the music performed. My own interest in computer music generally, and interactive music systems in particular, stems from the new compositional domains they open up. Composers have used algorithms in the creation of music for centuries. The speed with which such algorithms can now be executed by digital computers, however, eases their use during the performance itself. Once they are part of a performance, they can change their behavior as a function of the musical context going on around them. For me, this versatility represents the essence of interaction and an intriguing expansion of the craft of composition.

An equally important motivation for me, however, is the fact that interactive systems require the participation of humans making music to work. If interactive music systems are sufficiently engaging as partners, they may encourage people to make music at whatever level they can. I believe that it is critical to the vitality and viability of music in our culture that significant numbers of people continue (or begin) to engage in active music making, rather than simply absorbing reproduced music bombarding them from loudspeakers on every side.

Tod Machover stresses a similar point:

*Traditional instruments are hard to play. It takes a long time to [acquire] physical skills which aren't necessarily the essential qualities of making music. It takes years just to get good tone quality on a violin or to play in tune. If we could find a way to allow people to spend the same amount of concentration and effort on listening and*

*thinking and evaluating the difference between things and thinking about how to communicate musical ideas to somebody else, how to make music with somebody else, it would be a great advantage. Not only would the general level of musical creativity go up, but you'd have a much more aware, educated, sensitive, listening, and participatory public. (1999)*

We are at an inflection point in the technology of our culture as the trajectories of television and computer usage cross. Already more computers than television sets are sold each year in the United States. Televisions themselves are due to become digital within a matter of years and households are already becoming wired to receive a much higher bandwidth of information than they currently get from a telephone connection. None of this comes as a revelation anymore and has been thoroughly discussed elsewhere. The interesting question for this discussion is whether people using the new computer/televisions will simply look at these devices or be moved to interact with them. I believe that if computers interact with people in a musically meaningful way, that experience will bolster and extend the musicianship already fostered by traditional forms of music education. Ultimately, the goal must be to enrich and expand human musical culture. Certainly, music will continue to be produced in any case, but without the ferment of an actively engaged audience it will lapse into yet another form of consumerism.

Philippe Manoury makes this assessment of the relationship between music and its society:

*I am convinced that a certain culture is being lost. Music is increasingly playing the role of a diversion and that scares me. I don't have anything against music as a diversion, but I have the impression that our society, faced with numerous problems and no resolutions in sight, considers diversion as an antidote to those problems. . . . The more society stagnates, the more it distributes this antidote of diversion, in which music plays an important role. There is an overconsumption of the music of diversion and people don't see that music can also be the fruit of a reflection and an internal process, something they recognize more easily in literature. (Derrien 1995, 19–20 [my trans.])*

Although it is tempting to believe that one's own approach to music-making will lead to a more engaged society and more fully developed art form, I make no claims of special aesthetic or social virtue inherent to interactive music. However, as computer music is so often accused of leading us to a day when machines will listen only to machines, I feel compelled to observe that many of us are motivated by a much different vision of the computer's potential connection to the community of human musicians.

## **1.2 Algorithmic Composition**

The formalization of processes for generating music has a long and distinguished history in Western art music. From Guido d'Arezzo's chant generation method through the isorhythmic motet to serial techniques and Xenakis' "formalized music," interest in processes that produce music has waxed and waned through several centuries of composition (Loy 1989). Such algorithms move the compositional act to a meta-level where the evolution of the music's character is controlled over time by the manipulation of a limited number of parameters. Computers can now execute these processes so quickly that they can be realized on stage as part of an ongoing performance (Chadabe 1989). Interactive systems change the values of compositional parameters using information from a variety of inputs, including live performance data from multiple members of an ensemble.

Because these systems derive control parameters from a real-time analysis of performance, they can generate material based on improvised input as easily as they can on interpretations of notated music. They become a kind of ligature connecting improvisation to notated composition, just as the same processes used to govern the response to notated music can be employed to generate new improvisations in performance. This possibility expands the domain of composition. By delegating some of the creative responsibility to the performers and some to a computer program, the composer pushes composition up (to a meta-level captured in the processes executed by the computer) and out (to the human performers improvising within the logic of the work).

An interesting effect of this delegation is that the composer must give very detailed instructions to the computer at the same time that she gives up such precise direction of the human improviser. The resulting music requires a new kind of performance skill as much as it enables a new kind of composition. The human player working with an interactive system must learn how to perform with it much as he would learn to play with another human. The very real differences between computer performers and human performers mean, however, that the human also acquires a new degree of freedom in invoking and directing real-time algorithms through different styles of performance. An interactive composition changes and matures as the human and computer performances increasingly intertwine.

Another possibility, of course, is that the composer will take to the stage to perform with the system herself (figure 1.1). One of the notable characteristics of the field is the resurgence of the composer/



**Figure 1.1** Mari Kimura improvising

improviser, those musicians who design interactive systems and then improvise with them and/or other players in performance (e.g., Richard Teitelbaum, George Lewis, Chris Chafe, Mari Kimura, David Wessel, Ed Campion, Laetitia Sonami, and many others).

There is very seldom something new under the composition sun. Algorithmic thought is certainly not new, having been in evidence in Western music composition from the beginnings of its notation. Using processes in performance that change their behavior according to an analysis of other players's music, however, was never possible before the advent of computers and interactive music systems. Such systems therefore engender a realm of composition that was unknown only a few decades ago. I believe that this music, however, should not be described as being "in its infancy" or passing through an "experimental" phase. Doing so belittles the very real aesthetic credibility many of these works have achieved and gives composers an excuse to present works that still belong in the studio.

The musical values evinced in interactive compositions are ultimately the same as those underlying a string quartet. By transferring musical knowledge to a computer program and compositional responsibility to performers onstage, on the other hand, the composer of interactive works explores the creative potentials of the new technology at the same time that he establishes an engaging and fruitful context for the collaboration of humans and computers.

### 1.3 Algorithmic Analysis

There is a certain paradox at the heart of the transfer of musical knowledge to a machine. We must labor mightily to make a computer program perform the analysis required of a freshman music student. Once the work is done, however, the program can make analyses more reliably and certainly much more quickly than the freshman. The computer can deliver complete descriptions of each chord in a dictation within milliseconds of its performance, for example.

The purely quantitative difference of a very great acceleration produces a qualitative difference in the kinds of tasks a machine musi-

cian can reasonably be asked to perform. We would not set a novice musician in front of an ensemble with no idea of the piece of music to be played, its key, tempo, character, or form, and expect that apprentice player to follow what was going on very well, let alone contribute to the performance in more than a perfunctory way. Interactive systems whose knowledge of music theory does not go much beyond that of our hypothetical novice are often put into just such situations, however. Because these systems always do what they do correctly and very quickly, a little musical knowledge goes a long way.

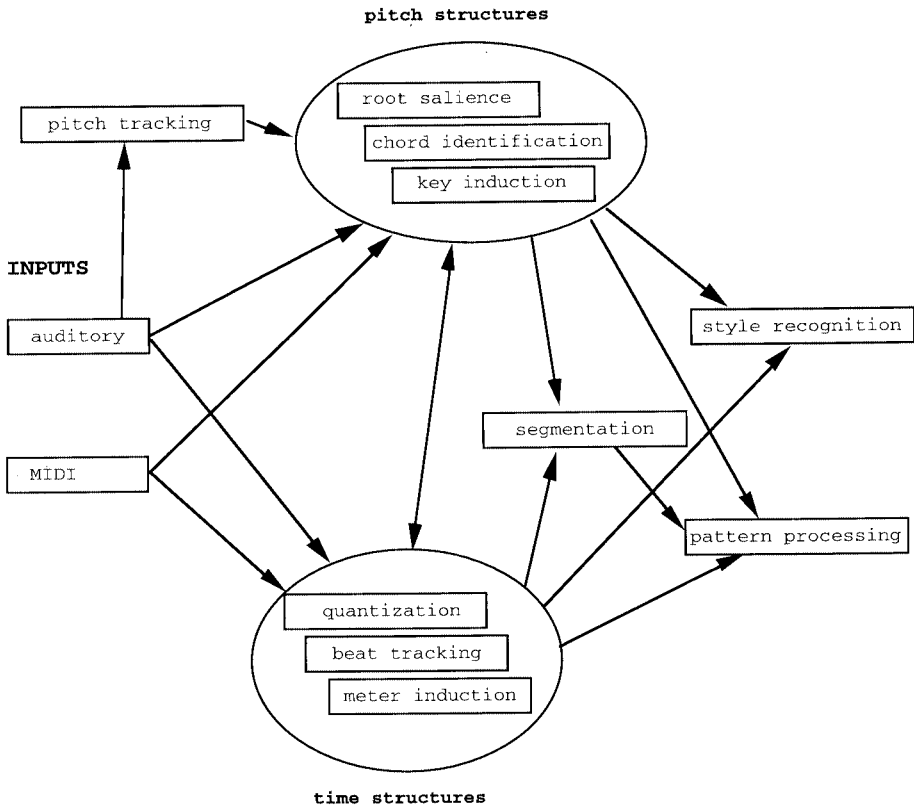
The formalization of musical concepts is proceeding apace through research in several fields, including music theory, music cognition, and artificial intelligence. So much work has been done in recent years that it would be inconceivable to document it all in one volume. The work reviewed in this text, then, is delimited by the requirement that the algorithms discussed be able to work in real time as part of a musical performance involving human players. Even with that restriction, this text in no way forms a comprehensive overview of the field.

There is a particularly interesting convergence between the fields of music cognition and interactive composition: as music cognition research becomes increasingly concerned with processes that could account for musical competence in a real musical environment, it gives rise to algorithms that can be adapted and used by composers and improvisers in performance. Whether or not it was a concern of the great variety of developers whose algorithms are described in these pages, all of these programs also pass the minimum threshold of psychological plausibility: they are all capable of execution in real time using only the information that becomes available as it is presented in sequence.

Certainly, some aspects of musicianship do not require such demanding performance in time; analysis is usually carried out over a period of days, not milliseconds, with an open score that allows the analyst to consult the music in any desired sequence. Many excellent systems of algorithmic analysis model this situation, and a suspen-

sion of the real-time requirement often allows them to produce better results than their performance-oriented counterparts. To maintain a more manageable scope, however, some such systems will be considered only to the extent that they can be adapted to real-time use.

Figure 1.2 illustrates some of the main processes extant in the literature that can be applied to real-time analysis of musical input. Space from left to right in the figure corresponds roughly to movement from low- to high-level processes in the algorithms. The arrows approximate the flow of information between stages: pitch input is forwarded to root and key identifiers, for example, while segmentation



**Figure 1.2** Machine musicianship processes

and style recognition rely in turn on the output of those lower-level analyses. This list is not exhaustive, but every element in it is manifested by one or more published algorithms.

My interest in this field is twofold: (1) to implement certain published processes so that they can be executed in performance; and (2) to design control structures within which these components can be combined to produce a more complete account of musical context and to make the cooperating components work better. This figure sketches only those processes related to analysis; there are similar collections that pertain to algorithmic composition and to the generation of expressive performance.

In their article “On the Thresholds of Knowledge,” Douglas Lenat and Edward Feigenbaum propose the Empirical Inquiry Hypothesis: “The most profitable way to investigate AI [artificial intelligence] is to embody our hypotheses in programs, and gather data by running the programs. The surprises usually suggest revisions that start the cycle over again. Progress depends on these experiments being able to falsify our hypotheses. Falsification is the most common and yet most crucial of surprises. In particular, these programs must be capable of behavior not expected by the experimenter” (Lenat and Feigenbaum 1992, 187).

The Empirical Inquiry Hypothesis—clearly related to Sir Karl Popper’s observations on the nature of science (1992)—suggests that machine musicianship programs should be built to exhibit behaviors that observers can recognize as correct or incorrect. Many, if not most interactive music systems are written to function in an environment of new music performance and improvisation. Their output can certainly be evaluated by their makers and those familiar with the idiom. My previous book, *Interactive Music Systems* (Rowe 1993) and several sections of this one deal with just such examples. A still broader class of musicians can evaluate the performance of algorithms that process standard works, however, and in accordance with the Empirical Inquiry Hypothesis many examples in this text will treat the mainstream classical and jazz repertoires as well.



I should make clear that this is not a psychology text, though the techniques I describe could be used to implement music cognition models or experiments. Psychological theories must address the question of how the processes they propose are realized in humans. My measure of success, however, is not whether these programs match empirical data from research with human subjects, but whether they output structures that make musical sense. I will gauge their performance in those terms by comparing their output with the answers expected from students studying introductory texts in music theory. The software may produce an acceptable answer by using processes similar to those of humans, or by using others that are wildly different. All else being equal, I would prefer that the machine processes resemble the human ones. Whether or not they do is a side effect, however. Ultimately I am concerned with machine musicianship and not a strict emulation of human music cognition.

#### 1.4 Structure of the Text

The programming examples in *Machine Musicianship* are written using two languages: C++ and Max. C++ is an object-oriented programming language that is widely available, well documented, and firmly established as one of the main vehicles for developing computer music applications. As examples are described in the text, I will develop a library of C++ objects that can be used as the basis for the reader's custom programs. This book is not an introduction to C++ programming. As examples are introduced I will summarize a few features of object orientation that are particularly valuable in developing a library for machine musicianship. Beyond that, any computer store will have a shelf full of introductory C++ books to which the reader is referred.

The fact that I will be illustrating concepts with C++ programs does not mean, however, that one must be or become a programmer to follow the text. C++ programs are a compact and complete way of notating algorithms. The algorithms themselves are the topic of interest here and will be explained in the text as they are imple-

mented in code. All of the applications described are included on the accompanying CD-ROM, but only a very small portion of the associated source code is printed in the text. Non-programmers approaching *Machine Musicianship* can then read the theory of the algorithms in question and run the associated applications to test their operation. Programmers can run the applications and modify the source, recorded in its entirety on the CD-ROM, to produce their own variations. C++ fragments in the text are concentrated at the end of each chapter so that non-programmers can skip over the code if they wish.

Max is a graphic programming language developed by Miller Puckette and David Zicarelli. There are a number of compelling reasons to include it as a development language here. First of all, Max has spawned a user community that is the most active and prolific group of interactive music designers working in the world today. There is no good reason to port Max patches to another language when there are probably more readers who know Max than know C++. In fact, I will demonstrate how C++ code is translated into a Max external to suggest how the Max community might make use of the C++ applications introduced here. Another good reason to use Max is the library of objects that has already been written for it and for MSP, a set of digital signal processing extensions. Programmers can quickly write powerful applications by building on the work of others.

Following this introduction, chapter 2 focuses on symbolic representations and algorithms directed toward the processing of pitch material. Issues such as root salience, chord identification, and key induction are addressed there. Chapter 3 continues with sub-symbolic processes, notably neural networks, and expands the field of application to include rhythm. Chapter 4 moves to higher-level musical constructs including segments and patterns and discusses systems whose input consists of a digital audio stream. Chapter 5 begins to look at compositional techniques, including score following and algorithmic digital signal processing. In chapter 6 processes for the automatic application of expressive performance techniques are reviewed.

In the remaining chapters I look in detail at some distinctive interactive systems that have been used in performances or installations. The particular techniques of interactive improvisation are the nucleus of chapter 7, leading to a discussion of how such machine performers can collaborate with an ensemble of other players. Chapter 8 looks at extensions of interactive environments to include other media, most prominently graphics, in live performance situations. Chapter 9 presents several interactive installations, where the inherent ability of such systems to deal with unpredictable input contributes to responsive environments exhibiting a variety of behaviors. A presentation of research directions form the conclusion in chapter 10.

## 1.5 Machine Musicianship Library

The CD-ROM enclosed in this book contains a library of C++ objects that can be used to build interactive programs. The source code of the examples, also listed on the CD-ROM, provides a set of templates for users to follow in writing their own applications. The library includes the files listed in table 1.1.

Many more files are included with their associated projects: the library is a repository of base classes from which specializations are built for almost every program in the book. In object-oriented programming, a *base class* is a generalized encapsulation of data and processes concerned with a particular subset of some application area. Specializations refine those general classes into *derived classes* that address the details of some specific application. All of the analysis processes depicted in figure 1.2, for example, have examples detailed in the text and included on the CD-ROM that are made from a combination of base classes, derived classes, and custom code.

Max programs are referenced as stand-alone applications, but are not included in the *Machine Musicianship* library as the necessary objects already form the heart of Max itself. One fully developed example is ported from the C++ environment into Max as a Max external, and many of the other C++ examples could similarly produce useful

**Table 1.1** Machine Musicianship Library

---

Clock.cp	timing routines
Event.cp	representation of a group of notes
EventBlock.cp	representation of a group of events
File.cp	file handling
Listener.cp	analysis of incoming MIDI events
ListenProps.cp	analysis processes
Mac.cp	macintosh I/O
MMerrors.cp	error reporting
Note.cp	representation of notes
OMSInPort.cp	OMS input routines
OMSOutPort.cp	OMS output routines
OMSSystem.cp	OMS communication
Scheduler.cp	scheduling facilities
Segment.cp	representation of segments
Utilities.cp	miscellaneous

---

Max objects. Taking programs in the other direction, from Max to C++, should be facilitated by the *Machine Musicianship* base classes.

The danger in producing a set of classes like this, particularly when it includes such entries as `Note`, `Event`, and `EventBlock`, is that it can be taken as a general representation of music. My intention is precisely the opposite—I do not believe that there is a simple and general way to represent all of the aspects of music we might want to process. The representations suggested by the *Machine Musicianship* library emerged from the particular collection of applications described in this book. Other tasks will demand other representations, or at the very least, modifications of these. Rather than a proposal for a generalized solution, the classes described here should be seen as an example of how programmers might design their own. After getting our feet wet with an initial application, in fact, I will discuss the issues of representation design more thoroughly in chapter 2.



## References

- Allen, P., and R. Dannenberg. 1990. Tracking musical beats in real time. In *Proceedings of the 1990 International Computer Music Conference*. San Francisco: International Computer Music Association, 140–43.
- Arcos, J., L. Mántaras, and X. Serra. 1998. Saxex: A case-based reasoning system for generating expressive musical performances. *Journal of New Music Research* 27(3):194–210.
- Ashley, R. 1989. Modeling music listening: General considerations. *Contemporary Music Review* 4:295–310.
- Baisnée, P., J.-B. Barrière, O. Koechlin, and R. Rowe. 1986. Real-time interactions between musicians and computer: Live performance utilisations of the 4X musical workstation. In *Proceedings of the 1986 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Baker, D. 1983. *Jazz improvisation: A comprehensive method for all musicians*. Van Nuys, CA: Alfred Publishing Company.
- Bent, I. 1980. *Analysis*. New York: W.W. Norton & Company.
- Berg, P. 1998. *Using the AC Toolbox*. Distributed by the author at <http://www.koncon.nl/ACToolbox/>
- Berliner, P. 1994. *Thinking in jazz: The infinite art of improvisation*. Chicago: The University of Chicago Press.
- Berry, W. 1987. *Structural functions in music*. New York: Dover Publications.
- Bharucha, J. 1987. Music cognition and perceptual facilitation: A connectionist framework. *Music Perception* 5:1–30.
- Bharucha, J. 1999. Neural nets, temporal composites, and tonality. In *The psychology of music*. 2nd Ed. Ed. D. Deutsch. London: Academic Press.
- Bharucha, J., and P. Todd. 1989. Modeling the perception of tonal structure with neural nets. *Computer Music Journal* 13(4):44–53.
- Bigand, E. 1993. The influence of implicit harmony, rhythm, and musical training on the abstraction of “tension-relaxation schemas” in tonal musical phrases. *Contemporary Music Review* 9(1,2):123–137.
- Biles, J. 1994. GenJam: a genetic algorithm for generating jazz solos. In *Proceedings of the 1994 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Biles, J. 1998. Interactive GenJam: Integrating real-time performance with a genetic algorithm. In *Proceedings of the 1998 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Bilmes, J. 1993. *Timing is of the essence: Perceptual and computational techniques for representing, learning, and reproducing expressive timing in percussive rhythm*. Master’s thesis, Massachusetts Institute of Technology.
- Bishop, C. 1995. *Neural networks for pattern recognition*. Cambridge, England: Clarendon Press.

- Bloch, J., and R. Dannenberg. 1985. Real-time computer accompaniment of keyboard performances. In *Proceedings of the 1985 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Bobrow, D., and T. Winograd. 1977. An overview of KRL, a knowledge representation language. *Cognitive Science* 1(1):3–46.
- Boden, M. 1994. What is creativity? In *Dimensions of creativity*. Ed. Margaret A. Boden. Cambridge, MA: The MIT Press.
- Bongers, B. 1998. An interview with Sensorband. *Computer Music Journal* 22(1):13–24.
- Bregman, A. 1990. *Auditory scene analysis*. Cambridge, MA: The MIT Press.
- Bresin, R. 1998. Artificial neural networks based models for automatic performance of musical scores. *Journal of New Music Research* 27(3):239–270.
- Brown, H., D. Butler, and M. Riess Jones. 1994. Musical and temporal influences on key discovery. *Music Perception* 11:371–407.
- Butler, D. 1989. Describing the perception of tonality in music: A critique of the tonal hierarchy theory and a proposal for a theory of intervallic rivalry. *Music Perception* 6:219–242.
- Butler, D. and W. Ward. 1988. Effacing the memory of musical pitch. *Music Perception* 5: 251–260.
- Cambouropoulos, E. 1997. Musical rhythm: A formal model for determining local boundaries, accents and metre in a melodic surface. In *Music, gestalt, and computing: Studies in cognitive and systematic musicology*. Ed. M. Leman. Berlin: Springer.
- Camurri, A., A. Catorcini, C. Innocenti, and A. Massari. 1995. Music and multimedia knowledge representation and reasoning: The HARP system. *Computer Music Journal* 19(2):34–58.
- Camurri, A., and M. Leman. 1997. Gestalt-based composition and performance in multimodal environments. In *Music, gestalt, and computing: Studies in cognitive and systematic musicology*. Ed. M. Leman. Berlin: Springer.
- Camurri, A., and P. Ferrentino. 1999. Interactive environments for music and multimedia. *Multimedia Systems* 7:32–47.
- Camurri, A., M. Ricchetti, and R. Trocca. 1999. EyesWeb: Toward gesture and affect recognition in dance/music interactive systems. In *Proceedings of IEEE multimedia systems 1999, Florence*.
- Canazza, S., G. De Poli, G. Di Sanzo, and A. Vidolin. 1998. A model to add expressiveness to automatic musical performance. In *Proceedings of the 1998 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Canazza, S., G. De Poli, A. Rodà, and A. Vidolin. 1997. Analysis by synthesis of the expressive intentions in musical performance. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Canetti, E. 1984. *Crowds and power*. trans. Carol Stewart. New York: Farrar Straus Giroux.
- Carreras, F., M. Leman, and M. Lesaffre. 1999. Automatic description of musical signals using schema-based chord decomposition. *Journal of New Music Research*.
- Chadabe, J. 1989. Interactive composing: An overview. In *The music machine*. Ed. C. Roads. Cambridge, MA: The MIT Press.
- Chadabe, J. 1997. *Electric sound: The past and promise of electronic music*. Upper Saddle River, NJ: Prentice-Hall.
- Chafe, C., B. Mont-Reynaud, and L. Rush. 1989. Toward an intelligent editor of digital audio: Recognition of musical constructs. In *The music machine*. Ed. C. Roads. Cambridge, MA: The MIT Press.

- Clarke, E. 1985. Structure and expression in rhythmic performance. In *Musical structure and cognition*. Ed. P. Howell, I. Cross, and R. West. London: Academic Press.
- Clarke, E. 1987. Levels of structure in the organization of musical time. *Contemporary Music Review* 2:211–238.
- Clarke, E. 1988. Generative principles in music performance. In *Generative processes in music*. Ed. J. Sloboda. Oxford: Clarendon Press.
- Clarke, E. 1999. Rhythm and timing in music. In *The psychology of music. 2nd ed.* Ed. D. Deutsch. London: Academic Press.
- Cook, N. 1987. *A guide to musical analysis*. Oxford: Oxford University Press.
- Cook, N. 1998. *Analysing musical multimedia*. Oxford: Clarendon Press.
- Cope, D. 1990. Pattern matching as an engine for the computer simulation of musical style. In *Proceedings of the 1990 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Cope, D. 1991. *Computers and musical style*. Madison, WI: A-R Editions, Inc.
- Cope, D. 1993. A computer model of music composition. In *Machine models of music*. Ed. S. Schwanauer and D. Levitt. Cambridge, MA: The MIT Press.
- Craik, F., and R. Lockhart. 1972. Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior* 11:671–684.
- Cuddy, L. 1993. Melody comprehension and tonal structure. In *Psychology and music: The understanding of melody and rhythm*. Ed. T. J. Tighe and W. J. Dowling. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Dannenberg, R. 1984. An on-line algorithm for real-time accompaniment. In *Proceedings of the 1984 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Dannenberg, R. 1989. Real-time scheduling and computer accompaniment. In *Current directions in computer music research*. Ed. M. Mathews and J. Pierce. Cambridge, MA: The MIT Press.
- Dannenberg, R. 1993. Music representation issues, techniques, and systems. *Computer Music Journal* 17(3):20–30.
- Dannenberg, R. 1998. In *Transit* program note. Program of the 1998 International Computer Music Conference.
- Dannenberg, R. 1999. Personal communication.
- Dannenberg, R., and B. Mont-Reynaud. 1987. Following an improvisation in real time. In *Proceedings of the 1987 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Dannenberg, R., B. Thom, and D. Watson. 1997. A machine learning approach to musical style recognition. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Deliège, I., M. Mélen, D. Stammers, and I. Cross. 1996. Musical schemata in real-time listening. *Music Perception* 14(2):117–160.
- De Poli, G., A. Piccialli, and C. Roads, eds. *Representations of musical signals*. Cambridge, MA: The MIT Press.
- Derrien, J. 1995. Entretien avec Philippe Manoury. In *Philippe Manoury: Les cahiers de l'Ircam*. Paris: Editions Ircam—Centre Georges-Pompidou.
- Desain, P. 1992. A (de)composable theory of rhythm perception. *Music Perception* 9(4): 439–454.
- Desain, P. 1993. A connectionist and a traditional AI quantizer: Symbolic versus sub-symbolic models of rhythm perception. *Contemporary Music Review* 9(1,2):239–254.



- Desain, P., and H. Honing. 1989. The quantization of musical time: A connectionist approach. *Computer Music Journal* 13(3).
- Desain, P., and H. Honing. 1994a. Foot-tapping: A brief introduction to beat induction. In *Proceedings of the 1994 International Computer Music Conference*. San Francisco: International Computer Music Association, 78–79.
- Desain, P., and H. Honing. 1994b. Rule-based models of initial-beat induction and an analysis of their behavior. In *Proceedings of the 1994 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Desain, P., and H. Honing. 1994c. Advanced issues in beat induction modeling: Syncopation, tempo and timing. In *Proceedings of the 1984 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Desain, P., and H. Honing. 1999. Computational models of beat induction: The rule-based approach. *Journal of New Music Research* 28(1):29–42.
- Deutsch, D. 1999a. Grouping mechanisms in music. In *The psychology of music*. 2nd Ed. Ed. D. Deutsch. London: Academic Press.
- Deutsch, D. 1999b. The processing of pitch combinations. In *The psychology of music*. 2nd Ed. Ed. D. Deutsch. London: Academic Press.
- Deutsch, D., and J. Feroe. 1981. The internal representation of pitch sequences in tonal music. *Psychological Review* 88:503–522.
- Dobbins, B. 1994. *A creative approach to jazz piano harmony*. Advance Music.
- Dolson, M. 1991. Machine tongues XII: Neural networks. In *Music and connectionism*. Ed. P. Todd and G. Loy. Cambridge, MA: The MIT Press.
- Dowling, W. and D. Harwood. 1986. *Music cognition*. New York: Academic Press.
- Duckworth, W. 1998. *A creative approach to music fundamentals*. 6th Ed. Belmont, CA: Wadsworth Publishing Company.
- Ebcioğlu, K. 1992. An expert system for harmonizing chorales in the style of J.S. Bach. In *Understanding music with AI: Perspectives on music cognition*. Ed. O. Laske and M. Balaban. Cambridge, MA: The AAAI Press/The MIT Press.
- Forde, A. 1973. *The structure of atonal music*. New Haven, CT: Yale University Press.
- Friberg, A. 1991. Generative rules for music performance: A formal description of a rule system. *Computer Music Journal* 15(2):56–71.
- Gabrielsson, A. 1988. Timing in music performance and its relations to music experience. In *Generative processes in music: The psychology of performance, improvisation, and composition*. Ed. J. Sloboda. Oxford: Clarendon Press.
- Gabrielsson, A. 1995. Expressive intention and performance. In *Music and the mind machine*. Ed. R. Steinberg. Berlin: Springer-Verlag.
- Garton, B., and M. Suttor. 1998. A sense of style. In *Proceedings of the 1998 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Garton, B., and D. Topper. 1997. RTcmix: Using CMIX in real time. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Gjerdingen, R. 1988. *A classic turn of phrase: Music and the psychology of convention*. Philadelphia: University of Pennsylvania Press.
- Gjerdingen, R. 1990. Categorization of musical patterns by self-organizing neuronlike networks. *Music Perception* 8:339–370.
- Gjerdingen, R. 1999. Apparent motion in music? In *Musical networks: Parallel distributed perception and performance*. Ed. N. Griffith and P. Todd. Cambridge, MA: The MIT Press.

- Goldman, C., D. Gang, J. Rosenschein, and D. Lehmann. 1999. NetNeg: A connectionist-agent integrated system for representing musical knowledge. *Annals of Mathematics and Artificial Intelligence* 25:69–90.
- Goto, M., and Y. Muraoka. 1994. A beat-tracking system for acoustic signals of music. In *ACM multimedia 1994 proceedings*. San Francisco: Association of Computing Machinery.
- Goto, M., and Y. Muraoka. 1997a. Real-time rhythm tracking for drumless audio signals: Chord change detection for musical decisions. *Working notes of the IJCAI-97 workshop on computational auditory scene analysis*, 135–144.
- Goto, M., and Y. Muraoka. 1997b. Issues in evaluating beat tracking systems. *Working notes of the IJCAI-97 workshop on issues in AI and music*, 9–16.
- Gresham-Lancaster, S. 1998. The aesthetics and history of The Hub: The effects of changing technology on network computer music. *Leonardo Music Journal* 8:39–44.
- Griffith, N., and P. Todd, eds. 1999. *Musical networks: Parallel distributed perception and performance*. Cambridge, MA: The MIT Press.
- Handel, S. 1989. *Listening: An introduction to the perception of auditory events*. Cambridge, MA: The MIT Press.
- Hewlett, W., E. Selfridge-Field, D. Cooper, B. Field, K. Ng, and P. Sitter. MIDI. In *Beyond MIDI: The handbook of musical codes*. Ed. E. Selfridge-Field. Cambridge, MA: The MIT Press, 41–72.
- Hillis, W. 1992. Co-evolving parasites improve simulated evolution as an optimization procedure. In *Artificial life II*. Ed. C. Langton, C. Taylor, J. Farmer, and S. Rasmussen. Reading, MA: Addison-Wesley.
- Holloway, R. 1989. Word—Image—Concept—Sound. *Contemporary Music Review* 5:257–265.
- Honing, H. 1993. Issues on the representation of time and structure in music. *Contemporary Music Review* 9(1,2):221–238.
- Howell, P., R. West, and I. Cross. 1991. *Representing musical structure*. San Diego: Academic Press.
- Huron, D. 1992. Design principles in computer-based music representation. In *Computer representations and models in music*. Ed. A. Marsden and A. Pople. London: Academic Press.
- Huron, D. 1994. *The Humdrum Toolkit reference manual*. Menlo Park, CA: Center for Computer Assisted Research in the Humanities.
- Huron, D. 1997. Humdrum and Kern: selective feature encoding. In *Beyond MIDI: The handbook of musical codes*. Ed. E. Selfridge-Field. Cambridge, MA: The MIT Press, 375–401.
- Jackendoff, R. 1992. Musical processing and musical affect. In *Cognitive bases of musical communication*. Ed. M.R. Jones and S. Holleran. Washington, DC: American Psychological Association.
- Järvinen. 1985. Tonal hierarchies in jazz improvisation. *Music Perception* 12:415–437.
- Johnson-Laird, P. 1991. Jazz improvisation: A theory at the computational level. In *Representing musical structure*. Ed. P. Howell, R. West, and I. Cross. London: Academic Press.
- Jordan, M. 1986. Attractor dynamics and parallelism in a connectionist sequential machine. *Proceedings of the 8th Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Cognitive Science Society.
- Kendall, R., and E. Carterette. 1990. The communication of musical expression. *Music Perception* 8(2):129–164.

- Kohonen, T. 1984. *Self-organization and associative memory*. Berlin: Springer.
- Kosslyn, S., and J. Pomerantz. 1977. Imagery, propositions, and the form of internal representations. *Cognitive Psychology* 9:52–76.
- Knuth, D. 1973. *The art of computer programming. Vol. 3, Searching and sorting*. Reading, MA: Addison-Wesley.
- Krumhansl, C. 1990. *Cognitive foundations of musical pitch*. Oxford: Oxford University Press.
- Krumhansl, C. 1992. Internal representations for music perception and performance. In *Cognitive bases of musical communication*. Ed. M.R. Jones and S. Holleran. Washington, DC: American Psychological Association.
- Krumhansl, C., and E. Kessler. 1982. Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys. *Psychological Review* 89:334–368.
- Kuivila, R., and D. Behrman. 1998. Composing with shifting sand: A conversation between Ron Kuivila and David Behrman on electronic music and the ephemerality of technology. *Leonardo Music Journal* 8:13–16.
- Large, E. and J. Kolen. 1994. Resonance and the perception of musical meter. *Connection Science* 6:177–208.
- Lee, C. 1985. The rhythmic interpretation of simple musical structures. In *Musical structure and cognition*. Ed. P. Howell, I. Cross, and R. West. London: Academic Press.
- Lee, M., and D. Wessel. 1992. Connectionist models for real-time control of synthesis and compositional algorithms. In *Proceedings of the 1992 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Leman, M. 1992. Artificial neural networks in music research. In *Computer representations and models in music*. Ed. A. Marsden and A. Pople. London: Academic Press.
- Leman, M. 1995. *Music and schema theory: Cognitive foundations of systematic musicology*. Heidelberg: Springer-Verlag.
- Leman, M., and F. Carreras. 1997. Schema and gestalt: Testing the hypothesis of psychoneural isomorphism by computer simulation. In *Music, gestalt, and computing: Studies in cognitive and systematic musicology*. Ed. Marc Leman. Berlin: Springer.
- Lenat, D., and E. Feigenbaum. 1992. On the thresholds of knowledge. In *Foundations of artificial intelligence*. Ed. D. Kirsh. Cambridge, MA: The MIT Press.
- Lerdahl, F., and R. Jackendoff. 1983. *A generative theory of tonal music*. Cambridge, MA: The MIT Press.
- Lindemann, E., F. Dechelle, B. Smith, and M. Starkier. 1991. The architecture of the IRCAM musical workstation. *Computer Music Journal* 15(3):41–49.
- Lippe, C. 1997. Music for piano and computer: A description. In *Proceedings of the U.S.A./Japan computer music conference 1997*. Tokyo: Kunitachi Conservatory.
- Longuet-Higgins, H., and C. Lee. 1982. The perception of musical rhythm. *Perception* 11: 115–128.
- Longuet-Higgins, H., and C. Lee. 1984. The rhythmic interpretation of monophonic music. *Music Perception* 1(4):424–441.
- Loy, D. 1985. Musicians make a standard: The MIDI phenomenon. *Computer Music Journal* 9(4). Reprinted in *The music machine*. Ed. C. Roads. Cambridge, MA: The MIT Press.
- Loy, D. 1989. Composing with computers: A survey of some compositional formalisms and music programming languages. In *Current directions in computer music research*. Ed. M. Mathews and J. Pierce. Cambridge, MA: The MIT Press.
- Maes, P. 1995. Modeling adaptive autonomous agents. In *Artificial life: An overview*. Ed. C. Langton. Cambridge, MA: The MIT Press.

- Machover, T. 1999. Technology and the future of music. Interview by F. Oteri, August 18, 1999. *NewMusicBox* 6. <http://www.newmusicbox.org>.
- Manoury, P. 1984. The role of the conscious. *Contemporary Music Review* 1:147–156.
- Manoury, P. 1991. *Jupiter*. Paris: Éditions Musicales Amphion.
- Marsden, A. 1992. Modelling the perception of musical voices. In *Computer representations and models in music*. Ed. A. Marsden and A. Pople. London: Academic Press.
- Marsden, A., and A. Pople, eds. 1992. *Computer representations and models in music*. London: Academic Press.
- Marslen-Wilson, W., and L. Tyler. 1984. The temporal structure of spoken language understanding. *Cognition* 8:1–71.
- Martin, K., E. Scheirer, and B. Vercoe. 1998. Music content analysis through models of audition. *ACM multimedia 1998 workshop on content processing of music for multimedia applications*. San Francisco: Association of Computing Machinery.
- Maxwell, H. 1992. An expert system for harmonic analysis of tonal music. In *Understanding music with AI*. Ed. M. Balaban, K. Ebcioglu, and O. Laske. Cambridge, MA: The MIT Press.
- McAdams, S. 1987. Music: A science of the mind? In *Contemporary Music Review* 2(1):1–62.
- McClelland, J., and D. Rumelhart. 1988. *Explorations in parallel distributed processing*. Cambridge, MA: The MIT Press.
- McEwan, I. 1998. *Amsterdam*. London: Jonathan Cape.
- Meehan, J. 1980. An artificial intelligence approach to tonal music theory. *Computer Music Journal* 4(2):64.
- Messiaen, O. 1997. *Traité de Rythme, de Couleur, et d'Ornithologie* (Tome IV). Paris: Alphonse Leduc.
- Minsky, M. 1985. A framework for representing knowledge. In *Readings in knowledge representation*. Ed. R. Brachman and H. Levesque. Los Altos, CA: Morgan Kaufmann Publishers.
- Miranda, E., ed. 1999. *Readings in music and artificial intelligence*. The Netherlands: Harwood Academic Publishers.
- Mitchell, M. 1996. *An introduction to genetic algorithms*. Cambridge, MA: The MIT Press.
- Mozer, M. 1991. Connectionist music composition based on melodic, stylistic, and psychophysical constraints. In *Music and connectionism*. Ed. P. Todd and D.G. Loy. Cambridge, MA: The MIT Press.
- Mozer, M. 1993. Neural net architectures for temporal sequence processing. In *Predicting the future and understanding the past*. Ed. A. Weigend and N. Gershenfeld. Reading, MA: Addison-Wesley.
- Myers, C., and L. Rabiner. 1981. A level building dynamic time warping algorithm for connected word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-29(2):284–297.
- Narmour, E. 1977. *Beyond Schenkerism: The need for alternatives in music analysis*. Chicago: University of Chicago Press.
- Narmour, E. 1990. *The analysis and cognition of basic melodic structures: The implication-realization model*. Chicago: University of Chicago Press.
- Narmour, E. 1999. Hierarchical expectation and musical style. In *The psychology of music*. 2nd ed. Ed. D. Deutsch. London: Academic Press.
- Nigrin, A. 1993. *Neural networks for pattern recognition*. Cambridge, MA: The MIT Press.
- Nordli, K. 1997. MIDI extensions for musical notation (1): NoTAMIDI meta-events. In *Beyond MIDI: The handbook of musical codes*. Ed. E. Selfridge-Field. Cambridge, MA: The MIT Press, 73–79.

- Odiard, P. 1995. De la confrontation à la conjonction: a propos de *Sonus ex machina*. In *Philippe Manoury: Collection «compositeurs d'aujourd'hui»*. Paris: Editions IRCAM—Centre Georges-Pompidou.
- O Mardin, D. 1992. Representation of music scores for analysis. In *Computer representations and models in music*. Ed. A. Marsden and A. Pople. London: Academic Press.
- Ossowski, S. 1999. *Co-ordination in artificial agent societies: Social structure and its implications for autonomous problem-solving agents*. Berlin: Springer.
- Owens, T. 1974. *Charlie Parker: Techniques of improvisation*. Ph.D. diss., University of California, Los Angeles.
- Palisca, C. 1996. *Norton Anthology of Western Music*. 3rd ed. Vol. 2. Ed. C. Palisca. New York: W.W. Norton & Company.
- Palmer, C. 1989. Mapping musical thought to music performance. *Journal of Experimental Psychology* 15:331–346.
- Paradiso, J. 1999. The Brain Opera technology: New instruments and gestural sensors for musical interaction and performance. *Journal of New Music Research* 28(2):130–149.
- Parncutt, R. 1988. Revision of Terhardt's psychoacoustical model of the roots of a musical chord. *Music Perception* 6:65–94.
- Parncutt, R. 1989. *Harmony: A psychoacoustical approach*. Berlin: Springer.
- Parncutt, R. 1994. A perceptual model of pulse salience and metrical accent in musical rhythms. *Music Perception* 11:409–464.
- Parncutt, R. 1997. A model of the perceptual root(s) of a chord accounting for voicing and prevailing tonality. In *Music, gestalt, and computing: Studies in cognitive and systematic musicology*. Ed. Marc Leman. Berlin: Springer.
- Pennycook, B., D. Stammen, and D. Reynolds. 1993. Toward a computer model of a jazz improviser. In *Proceedings of the 1993 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Perkis, T. 1995. Liner notes to the audio CD, *Wreckin' Ball*. Artifact Recordings ART 1008.
- Perlin, K. 1995. Real-time responsive animation with personality. *IEEE transactions on visualization and computer graphics*. New York: IEEE.
- Perlin, K., and A. Goldberg. 1996. IMPROV: A system for scripting interactive actors in virtual worlds. In *Proceedings of SIGGRAPH 96, Annual Conference Series*. New York: ACM SIGGRAPH.
- Pope, S., ed. 1991. *The well-tempered object: Musical applications of object-oriented software technology*. Cambridge, MA: The MIT Press.
- Popper, K. 1992. *The logic of scientific discovery*. London: Routledge.
- Porter, L. 1998. *John Coltrane: His life and music*. Ann Arbor, MI: The University of Michigan Press.
- Povall, R. 1995. Compositional methods in interactive performance environments. *Journal of New Music Research* 24(2):109–120.
- Pressing, J. 1988. Improvisation: Methods and models. In *Generative processes in music: The psychology of performance, improvisation, and composition*. Ed. J. Sloboda. Oxford: Clarendon Press.
- Puckette, M. 1993. Contributions to “Putting Max in perspective.” Ed. R. Rowe and B. Garton. *Computer Music Journal* 17(2):3–11.
- Rai, T. 1998. Program note to *Kinetic Figuration: Five works with computers*. Audio CD, Toshiba-EMI DAC-1210.
- Reis, B. 1999. Simulated music learning: On-line, perceptually guided pattern induction of context models for multiple-horizon prediction of melodies. In *Proceedings of the 1999 workshop on pattern processing in music*.

- Repp, B. 1992. Diversity and commonality in music performance: An analysis of timing microstructure in Schumann's "Träumerei." *Journal of the Acoustical Society of America* 92(5):2546–2568.
- Reynolds, C. 1987. Flocks, herds, and schools: A distributed behavioral model. In *Computer Graphics, SIGGRAPH 87 Conference Proceedings*. New York: ACM SIGGRAPH.
- Richter, M., C. Smith, R. Wiehagen, and T. Zeugmann. 1998. Editor's Introduction. In *Algorithmic learning theory*. Ed. M. Richter, C. Smith, R. Wiehagen, and T. Zeugmann. Berlin: Springer.
- Ritter, D. 1996. The intersection of art and interactivity. In *Ars Electronica Festival 96: Mimesis*. Vienna: Springer.
- Rogers, J. 1997. *Object-oriented neural networks in C++*. New York: Academic Press.
- Rolland, P., and J. Ganascia. 1996. Automated motive-oriented analysis of musical corporuses: A jazz case study. In *Proceedings of the 1996 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Rosenboom, D. 1992. Parsing real-time musical inputs and spontaneously generating musical forms: Hierarchical form generator (HFG). In *Proceedings of the 1992 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Rosenboom, D. 1997. Propositional music: On emergent properties in morphogenesis and the evolution of music. Part II: Imponderable forms and compositional methods. *Leonardo Music Journal* 7:35–39.
- Rosenthal, D. 1989. A model of the process of listening to simple rhythms. *Music Perception* 6(3):315–328.
- Rowe, R. 1993. *Interactive music systems: Machine listening and composing*. Cambridge, MA: The MIT Press.
- Rowe, R., and T. Li. 1995. Pattern processing in music. In *Proceedings of the fifth biennial symposium for arts and technology*. New London, CT: Connecticut College.
- Rowe, R., and E. Singer. 1997. Two highly integrated real-time music and graphics performance systems. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Rowe, R., E. Singer, and D. Vila. 1996. *A Flock of Words: Real-time animation and video controlled by algorithmic music analysis*. In *Visual Proceedings, SIGGRAPH 96*. New York: ACM SIGGRAPH.
- Sankoff, D., and J. Kruskal, eds. 1983. *Time warps, string edits, and macromolecules: The theory and practice of sequence comparison*. Reading, MA: Addison-Wesley.
- Sawada, H., N. Onoe, and S. Hashimoto. 1997. Sounds in hands: A sound modifier using datagloves and twiddle interface. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Scarborough, D., B. Miller, and J. Jones. 1991. Connectionist models for tonal analysis. In *Music and connectionism*. Ed. P. Todd and G. Loy. Cambridge, MA: The MIT Press.
- Schank, R., and R. Abelson. 1977. *Scripts, plans, goals and understanding: An inquiry into human knowledge structures*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Scheirer, E. 1996. Bregman's chimerae: Music perception as auditory scene analysis. *Proceedings of the Fourth International Conference on Music Perception and Cognition*. Montreal: McGill University.
- Schiphorst, T. 1996. Felt histories (re: The fact of a doorframe). Artist's description.
- Schwanauer, S. 1988. Learning machines and tonal composition. In *Proceedings of the first workshop on artificial intelligence and music*. Minneapolis/St. Paul, MN: AAAI–88.

- Selfridge-Field, E., ed. 1997a. *Beyond MIDI: The handbook of musical codes*. Cambridge, MA: The MIT Press.
- Selfridge-Field, E. 1997b. Describing musical information. In *Beyond MIDI: The handbook of musical codes*. Ed. E. Selfridge-Field. Cambridge, MA: The MIT Press, 3–38.
- Selfridge-Field, E. 1998. Conceptual and representational issues in melodic comparison. *Computing in musicology* 11:3–64.
- Settel, Z. 1993. *Hok Pwah*. Program Note.
- Shannon, C., and W. Weaver. 1949. *The mathematical theory of communication*. Urbana: The University of Illinois Press.
- Shepard, R. 1964. Circularity in judgments of relative pitch. *Journal of the Acoustical Society of America* 36:2346–2353.
- Shepard, R. 1999. Cognitive psychology and music. In *Music, cognition, and computerized sound: An introduction to psychoacoustics*. Ed. P. Cook. Cambridge, MA: The MIT Press.
- Siegel, W., and J. Jacobsen. 1998. The challenges of interactive dance: An overview and case study. *Computer Music Journal* 22(4):29–43.
- Simon, H., and K. Kotovsky. 1963. Human acquisition of concepts for sequential patterns. *Psychological Review* (70):534–546.
- Simon, H., and R. Sumner. 1993. Pattern in music. In *Machine models of music*. Ed. S. Schwanauer and D. Levitt. Cambridge, MA: The MIT Press.
- Sims, K. 1991. Artificial evolution for computer graphics. *Computer Graphics* 25(4):319–328.
- Singer, E., K. Perlin, and C. Castiglia. 1996. Real-time responsive synthetic dancers and musicians. In *Visual proceedings, SIGGRAPH 96*. New York: ACM SIGGRAPH.
- Singer, E., A. Goldberg, K. Perlin, C. Castiglia, and S. Liao. 1997. Improv: Interactive improvisational animation and music. *ISEA 96 proceedings: Seventh international symposium on electronic art*. Rotterdam, Netherlands: ISEA96 Foundation.
- Sloboda, J. 1985. *The musical mind: The cognitive psychology of music*. Oxford: Clarendon Press.
- Smoliar, S. 1992. Representing listening behavior: Problems and prospects. In *Understanding music with AI*. Ed. M. Balaban, K. Ebcioglu, and O. Laske. Cambridge, MA: The MIT Press.
- Stammen, D. 1999. *Timewarp: A computer model of real-time segmentation and recognition of melodic fragments*. Ph.D. diss., McGill University.
- Stammen, D., and B. Pennycook. 1993. Real-time recognition of melodic fragments using the dynamic timewarp algorithm. In *Proceedings of the 1993 International Computer Music Conference*. San Francisco: International Computer Music Association, 232–235.
- Subotnick, M. 1997. *Intimate Immensity*. Program note.
- Sundberg, J. 1988. Computer synthesis of music performance. In *Generative processes in music: The psychology of performance, improvisation, and composition*. Ed. J. Sloboda. Oxford: Clarendon Press.
- Sundberg, J., A. Askenfelt, and L. Frydén. 1983. Music performance: A synthesis-by-rule approach. *Computer Music Journal* 7:37–43.
- Sundberg, J., A. Friberg, and L. Frydén. 1991. Common secrets of musicians and listeners—an analysis-by-synthesis study of musical performance. In *Representing musical structure*. Ed. P. Howell, R. West, and I. Cross. London: Academic Press.
- Tanimoto, S. 1990. *The elements of artificial intelligence: Using Common LISP*. New York: W.H. Freeman and Company.

- Tarabella, L. 1993a. Introduction. *Interface* 22:179–182.
- Tarabella, L. 1993b. Real-time concurrent PascalMusic. *Interface* 22:229–242.
- Tarabella, L., and M. Magrini. 1998. GALileo, a Graphic ALgorithmic music language. In *Proceedings of the XII colloquium of musical informatics*.
- Tarabella, L., M. Magrini, and G. Scapellato. 1998. A system for recognizing shape, position, and rotation of the hands. In *Proceedings of the 1998 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Teitelbaum, R. 1982. Digital piano music. Unpublished manuscript, quoted in Chadabe, J. 1997. *Electric sound: The past and promise of electronic music*. Upper Saddle River, NJ: Prentice-Hall.
- Tenney, J., and L. Polansky. 1980. Temporal gestalt perception in music. *Journal of Music Theory* 24:205–241.
- Temperley, D. 1997. An algorithm for harmonic analysis. *Music Perception* 15(1):31–68.
- Temperley, D., and D. Sleator. 1999. Modeling meter and harmony: a preference-rule approach. *Computer Music Journal* 23(1):10–27.
- Terhardt, E., G. Stoll, and M. Seewann. 1982. Algorithm for extraction of pitch and pitch salience from complex tonal signals. *Journal of the Acoustical Society of America* 71: 679–688.
- Thomasson, J. 1982. Melodic accent: experiments and a tentative model. *Journal of the Acoustical Society of America* 71.
- Thompson, W. 1993. The harmonic root: A fragile marriage of concept and percept. *Music Perception* 10:385–416.
- Thompson, W., and M. Stainton. 1996. Using Humdrum to analyze melodic structure: An assessment of Narmour's implication-realization model. *Computing in Musicology* 10: 24–33.
- Todd, N. 1985. A model of expressive timing in tonal music. *Music Perception* 3(1):33–58.
- Todd, N. 1992. The dynamics of dynamics: A model of musical expression. *Journal of the Acoustical Society of America* 91(6):3540–3550.
- Todd, N. 1994. The auditory “primal sketch”: A multi-scale model of rhythmic grouping. *Journal of New Music Research* 23(1):25–70.
- Todd, P. 1991. A connectionist approach to algorithmic composition. In *Music and connectionism*. Ed. P. Todd and D. Loy. Cambridge, MA: The MIT Press.
- Todd, P. and D. Loy. 1991. *Music and connectionism*. Cambridge, MA: The MIT Press.
- Toiviainen, P. 1998. An interactive MIDI accompanist. *Computer Music Journal* 22(4):63–75.
- Van Immerseel, L., and J. Martens. 1992. Pitch and voiced/unvoiced determination with an auditory model. *Journal of the Acoustical Society of America* 91:3511–3526.
- Vercoe, B. 1984. The synthetic performer in the context of live performance. In *Proceedings of the 1984 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Vercoe, B. 1997. Computational auditory pathways to music understanding. In *Perception and cognition of music*. Ed. I. Deliège and J. Sloboda. East Sussex, GB: Psychology Press.
- Vos, P. 1999. Key implications of ascending fourth and descending fifth openings. *Psychology of Music* 27:4–17.
- Vos, P., and E. Van Geenen. 1996. A parallel-processing key-finding model. *Music Perception* 14(2):185–224.
- Watkinson, J. 1994. *The art of digital audio*. 2d ed. Oxford: Focal Press.



- West, R., P. Howell, and I. Cross. 1985. Modelling perceived musical structure. In *Musical structure and cognition*. Ed. P. Howell, I. Cross, and R. West. London: Academic Press.
- West, R., P. Howell, and I. Cross. 1991. Musical structure and knowledge representation. In *Representing musical structure*. Ed. P. Howell, R. West, and I. Cross. London: Academic Press.
- Widmer, G. 1992. Qualitative perception modeling and intelligent musical learning. *Computer Music Journal* 16(2):51–68.
- Widmer, G. 1995. Modeling the rational basis of musical expression. *Computer Music Journal* 19(2):76–96.
- Widmer, G. 1996. Learning expressive performance: The structure-level approach. *Journal of New Music Research* 25:179–205.
- Widrow, B. 1963. ADALINE and MADALINE. *IEEE-ICNN* 1(1):143–158.
- Wiggins, G., E. Miranda, A. Smaill, and M. Harris. 1993. A framework for the evaluation of music representation systems. *Computer Music Journal* 17(3):31–42.
- Winkler, T. 1998. *Composing interactive music: Techniques and ideas using Max*. Cambridge, MA: The MIT Press.
- Winograd, T. 1968. Linguistics and the computer analysis of tonal harmony. *Journal of Music Theory* 12(1):2–49.
- Winsor, P. 1989. *Automated music composition*. Denton: University of North Texas Press.
- Winston, P. 1984. *Artificial Intelligence. 2d ed.* Reading, MA: Addison-Wesley Publishing Company.
- Witten, I., L. Manzara, and D. Conklin. 1994. Comparing human and computational models of music prediction. *Computer Music Journal* 18(1):70–80.
- Woideck, C. 1996. *Charlie Parker: His music and life*. Ann Arbor: The University of Michigan Press.
- Wright, M., and A. Freed. 1997. Open Sound Control: A new protocol for communicating with sound synthesizers. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Wright, M., A. Chaudhary, A. Freed, D. Wessel, X. Rodet, D. Virolle, R. Woehrmann, and X. Serra. 1998. New applications of the Sound Description Interchange Format. In *Proceedings of the 1998 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Wright, M., and D. Wessel. 1998. An improvisation environment for generating rhythmic structures based on North Indian “Tal” patterns. In *Proceedings of the 1998 International Computer Music Conference*. San Francisco: International Computer Music Association.
- Zicarelli, D. 1996. *Writing external objects for Max*. Palo Alto: Opcode Systems.