Composing Interactive Music:
A Pedagogical Approach to Teaching Interactive Composition

Todd Winkler
Brown University
Todd_Winkler@Brown.Edu

Abstract
This paper will suggest a pedagogical approach to teaching interactive composition, drawing upon the
author's teaching experience and research conducted for a forthcoming book to be published by AR-Editions,
titled Composing Interactive Music. The technical and aesthetic possibilities of interactive composition
are explored using Max, with a sequence of modules intended to provide musicians with the programming
skills and conceptual framework needed to produce interactive works.

Introduction
Interactive music has emerged as an interesting and prominent subset of computer music research and
composition. The evolving field is maturing thanks to continuing research at major centers, and the wide
availability of software that allows composers to experiment with interactive systems using inexpensive
and portable MIDI equipment. A survey and evaluation of a large number of programs and techniques is
well-documented in Robert Rowe's Interactive Music Systems. Of all these programs, Max in particular has
attracted the attention of a large number of composers, and the increase of its use for music production has
been reflected in its increased use for teaching purposes.

The adoption of Max as the demonstration language for interactive composition is a logical choice. Max
was developed expressly for the purpose of real-time interactive work. Its ease of use and intuitive graphical
interface provides a comfortable starting point for composers new to programming, while its large
collection of objects, and its extendibility using the C programming language gives it the flexibility more
advanced programmers need. Max software is easy to read and lends itself well to study, making it a highly
effective tool for introducing composers to interactive techniques.

Musicians with little computer experience, may not be interested in mastering the complexities of a
standard computer language in order to experiment with computer music processes. Thus, Max may serve to
introduce composers to fundamental programming concepts, while allowing them to explore musical
processes. The majority of examples cited in Composing Interactive Music are taken directly from the
author's compositions, with a few contributions from other composers, presenting working examples of
compositional techniques together with programming techniques.

How one gets from an introduction to Max to writing sophisticated interactive algorithms for composition
is not at all clear, especially since the modular nature of Max defies easy organization into a coherent
whole. What is clear is that Max offers composers using MIDI systems a higher level of control and
experimentation than most commercially available music software, within a sophisticated and challenging
environment for developing compositional ideas.

Pedagogical approach
The pedagogical approach contains a clear goal: to equip composers with the skills and concepts, both
musical and technical, needed to write interactive compositions. Presenting these basic techniques to
composers will hopefully lead to more advanced, independent work. As in most compositional studies, new
musical concepts are often inspired by newfound technical skills. Similarly, new techniques are created to
meet the demands of innovative musical ideas.
Programming should be viewed as one of the many skills a composer needs to effectively create computer music. Composers must be provided with the general concepts common to most programming languages along with a specific working knowledge of Max gained through musical examples and projects. It is beneficial to take advantage of a composer's interest and knowledge by presenting programming concepts and techniques that demonstrate musical phenomenon. For example, the obligatory exercise to convert Celsius temperatures to Fahrenheit that seems to accompany every programming textbook holds little interest to composers. More relevant, would be an exercise to create software that takes a melody and creates a canon at the fifth, with augmented or diminished intervals, thus introducing the concepts of offset and scaling reinforced in sound.

Outline of Study
Below is an abbreviated outline of study that more or less follows the structure of *Composing Interactive Music*. The foundation section begins with a theoretical look at interactive music systems, followed by an introduction to programming principles and their practical use in Max. These two areas ought to be examined simultaneously, so that musical concepts develop alongside programming skills.

The heart of the syllabus is in three large sections examining interactive composition techniques: performance analysis, compositional algorithms, and score structures. Listening algorithms demonstrate techniques for analyzing a performer's input. This data is collected and in turn used to influence musical processes. Various types of compositional algorithms are described, with parameters responsive to listener data. These are the techniques used to create the materials of the composition and in smaller works, they may also generate the entire structure. Building up to larger forms, compositional strategies will suggest associated score structures that contain the mechanisms for advancing through multi-sectional works.

Finally, in this quickly evolving field, it is important to look at new developments that have a direct impact on interactive techniques. These include a description of new interactive systems under development, performing with new controllers, and a look at interactive music applications in multimedia forms.

**OUTLINE FOR TEACHING INTERACTIVE COMPUTER MUSIC**

I. Foundation
   A. Background and Theory - definitions, history and background, models of interaction
   B. Programming Fundamentals - introduction to programming, Max basics, algorithms, data structures, program structure and design, interface design

II. Interactive Composition Techniques
   C. Performance Analysis - listener objects, time analysis, pitch analysis, changes over time
   D. Compositional Algorithms - response to performer input, pitch and rhythm, color and sound
   E. Score Structures - compositional strategies for large interactive works, linear and non-linear structures, scheduling mechanisms, examples of score structures

III. New Developments
   New systems, new controllers, multimedia extensions

Conclusion
This cumulative method provides a conceptual framework for teaching interactive music composition, beginning with small, self contained algorithms, and working up to large multi-sectional works. Programming principles are learned through the creation and analysis of musical examples. The skills gained offer great potential for new musical discourse, one where computer music processes are informed and shaped by a performer who, in turn responds to the results.

240