Choosers: The design and evaluation of a visual algorithmic music composition language for non-programmers

Matt Bellingham
Department of Music
University of Wolverhampton
matt.bellingham@wlv.ac.uk

Simon Holland
Music Computing Lab
The Open University
s.holland@open.ac.uk

Paul Mulholland
Knowledge Media Institute
The Open University
p.mulholland@open.ac.uk

Abstract

Algorithmic music composition involves specifying music in such a way that it is non-deterministic on playback, leading to music which has the potential to be different each time it is played. Current systems for algorithmic music composition require the user to have considerable background knowledge of programming and/or music theory. However, much of the potential user population are self-taught music producers without the required background in either programming or music. To investigate how this gap between tools and potential users might be better bridged we designed Choosers, a prototype algorithmic programming system centred around a new abstraction (of the same name) designed to allow non-programmers access to algorithmic music composition methods. Choosers provides a graphical notation that allows structural elements of key importance in algorithmic composition (such as sequencing, choice, multi-choice, weighting, looping and nesting) to be foregrounded in the notation in a way that is accessible to non-programmers. In order to test design assumptions a Wizard of Oz study was conducted in which seven pairs of undergraduate music technology students used Choosers to carry out a range of rudimentary algorithmic composition tasks. Feedback was gathered using the Programming Walkthrough method. All users were familiar with Digital Audio Workstations, and as a result they came with some relevant understanding, but also with some expectations that were not appropriate for algorithmic music work. Users were able to successfully make use of the mechanisms for choice, multi-choice, looping, and weighting after a brief training period. The 'stop' behaviour was not as easily understood and required additional input before users fully grasped it. Some users wanted an easier way to override algorithmic choices. These findings have been used to further refine the design of Choosers.