Strategies for interaction in *construction 3*

TONY MYATT

Department of Music, University of York, Heslington ,York, UK
E-mail: am12@york.ac.uk

This paper describes the interactive strategies adopted during the composition and realisation of *construction 3* for soprano saxophone and multiple media, by Tony Myatt and Peter Fluck. This work is an interactive multiple media composition for computer, computer-enhanced saxophone and computer graphics projected onto irregularly shaped screens. Derivation of the performer sensing system and mapping strategies to control signal processing, computer generated materials and communication gestures between the performer and computer are described. These processes include generative mapping techniques using neural networks for the recognition of gestural information and the development and application of wireless, wearable sensing technology. This work is described as a holistic approach to the derivation of performer sensing, data mapping and the application of reactive and interactive processes in the context of the creation of a new musical and multiple media composition.

1. INTRODUCTION

This paper describes the combination of contemporary technologies and techniques for the generation of a real-time[1] multiple media composition. It discusses issues related to the hardware and software used to create performer–computer interaction for a multiple media work with a solo instrument. Strategies adopted to map data from the performer and from a performer sensing system to the generation of audio material, graphic materials and control parameters will also be discussed.

One of the primary aims of this work was to investigate the control of a complex series of sound and graphic parameters to create a multiple media composition in a musical context. A holistic approach was adopted to define the performer's interaction with the computer. This embraced the concepts of the conductor/chamber music performer/improvisation model of interaction (Winkler 1999: 23–7) and the combination of a variety of sensing technologies and mapping strategies to establish an intimate control of the sound and graphic elements of the composition.

This work was pursued in a practical context – the realisation of *construction 3* for soprano saxophone and multiple media. The guiding principles were:



Figure 1. Performance configuration for *construction 3*.

- to establish performer control of the work;
- to eliminate physical connections between the performer and computer, establishing performer independence;
- to implement a functional (playable) hardware interface that was sensitive to the performer's instrumental technique;
- to define appropriate mappings of performer data for sound/graphic control and generation;
- to allow a flexible interpretation of the composition during performance;
- to allow performer control of the surround-sound presentation of the work.

The realisation of *construction 3* was carried out by the Real-time Interactive Multiple Media project (RIMM).[2] The project consisted of a realisation team from the University of York who were supported by IRCAM's Real-time and Room Acoustics Groups. The first performance of the work was presented by Staatliches Institut für Musikforschung PK in Berlin (SIMPK).

[1]Real-time is used here to denote computer audio performance where audio sample rate outputs can be sustained and action between performer and the reaction of audio/graphic systems occurs without perceivable time delay.

[2]The Real-time Interactive Multiple Media project (RIMM) was funded under the 5th Framework of the European Commission Information Society Technologies programme, IST-1999–21022 RIMM.

Organised Sound **7**(2): 157–169 © 2002 Cambridge University Press. Printed in the United Kingdom. DOI:10.1017/S1355771802002078

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

construction 3, by Tony Myatt and Peter Fluck, is an audio composition with computer graphics which are projected on to a sculpted projection surface (figure 1). The piece is performed by a soprano saxophonist who plays and also controls the signal processing of the saxophone sound, sound materials generated by the computer, sound locations within the performance space and the colours, shapes and motions of the computer graphics. The graphics are projected onto a series of irregularly shaped screens which form a sculptured, three-dimensional surface (designed to eliminate some of the visual cues associated with the 'view-port' or 'window' notion of traditional visual display units and screen projections, and to generate shadows which interact with the projected graphics). The musical and visual components of this work were composed and developed simultaneously to establish aural/visual correlations and dialogue.

Video tapes of the performer were analysed to determine the methods used to sense and derive control information. This analysis generated a specification for the sensing devices used and their application. The integration of technologies which were capable of producing a work of this type was a significant aspect of the work described below.

A wireless data acquisition system was developed for this project to integrate a variety of hardware sensing devices: accelerometers, digital compasses, force sensing resistors and switches. The data acquisition system allowed a flexible word length to represent data from analogue sensors. This was designed to provide an improved dynamic range for continuous controllers over MIDI-based controller devices.

jMax (Déchelle *et al*. 1999) with *Spatialisateur* (Jot and Warusfel 1995) was used exclusively for all audio generation, sound processing and data manipulation. Several *jMax* objects were also written during the project for data communications and the implementation of neural networks. Graphics were generated using an *OpenGL*[3] application which responded to performance control data from *jMax* via *UNIX* sockets.

The realisation of the work required multiple instances of *jMax* (seven) and *Spatialisateur* (nine). These were executed on a high performance parallel computer (HPC); a Silicon Graphics *Origin 2000* with eight R10000 MIPS processors and 4 GB memory. The graphic generation program ran on a networked Silicon Graphics *O2* computer. The audio soundfield was generated in Ambisonic B-format (Malham and Myatt 1995) and reproduced over a cubic loudspeaker array.

The project also evaluated the hardware described above for its suitability in applications of this type and produced benchmarks to illustrate its capability. The techniques and strategies adopted in this work are also applicable to multiple single-processor computer platforms, with an appropriate inter-computer network bandwidth.

The focus of this paper is the application of sensing technologies and the mapping strategies which were derived for performer–computer interaction with audio and graphic processes. Many of the signal processing, synchronisation, score-following etc. techniques applied were straightforward and well defined in the literature. The use of simple processes allowed the project to concentrate on the combination of technologies and the holistic approach to their application.

2. PERFORMER ANALYSIS

In addition to the control of the audio and graphic parameters in *construction 3*, the saxophonist was required to perform a complex and sometimes difficult acoustic instrumental part. It was important that any system of sensors used to acquire information from the performer did not interfere with the complex task of playing the instrument or the natural movements and gestures inherent in performance (Cook 2001).

The guiding principle for this investigation was that any mechanism which adversely affected breathing, fingering, stance, embouchure, or reading music was unacceptable. I have observed that the effect of using foot-operated controls, often requiring a performer to stand on one leg, not only influences balance but has repercussions for breath control and many other important physiological actions. Instrumental performers develop physical skills over many years which allow the intimate control of sound production. Stance plays a very important role in allowing physical actions to take place. A strategy was adopted to work with the performance skills of the player, to adopt elements of instrumental performance technique which could produce control data with appropriate mapping and to integrate additional controllers in a way which had a minor impact on instrumental performance techniques.

A performer analysis was carried out to develop the sensing system. This was divided into two elements: (i) video analysis of the performer; (ii) discussion with the performer about performance and ensemble communication techniques. The performer was filmed simultaneously by two cameras while playing a straight soprano saxophone. The cameras were at ninety degrees to each other, one recording a front view of the performer and the other a view from the right side (figure 2). While the recordings were being made, the general stance and motion of the performer were also recorded and noted. Primary data was gathered from the two cameras aligned to record the performer from head to waist and to encompass the full range of the saxophone's motion during the experiment. Information about his movement under certain conditions was recorded from observations

[3]http://www.opengl.org (checked 22 February 2002)

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

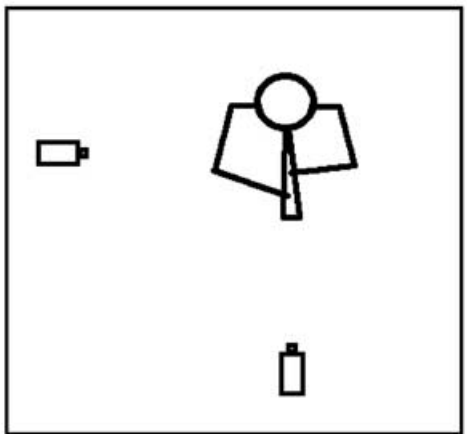


Figure 2. Camera positions for performer analysis (plan view).

of the video and additional tests were also carried out using a computer video-tracking system (producing (x, y) position vectors within the video frame) to generate data for off-line analysis. A series of small coloured dots were attached to the player (who was otherwise wearing black clothes) to provide sources which could be video-tracked for analysis. The dots were placed at the shoulders, chest, arms, hips and on the instrument. The performer was asked to play and then make a series of movements, reporting when the movements inhibited his performance technique.

The mechanism produced by the instrument, the arms of the performer, and the fixed position of the hands and embouchure – the *set-up* – restricted the motion of the instrument and performer. The head of the performer generally has to follow any up–down movement of the instrument to maintain a correct embouchure for sound production, and so movement is restricted to torso/neck-based left–right motions and torso/neck/arm based up–down movements of the instrument.

During the analysis the saxophonist was asked to perform naturally, to make a series of geometric motions with the instrument, a series of time variable gestures and to play whilst making communication gestures of a type that would be used by a lead instrumentalist in a chamber music context.

In the natural performing context, the analysis showed the saxophonist to be remarkably balanced and centred with almost no body movement other than his arm/instrument/head adjustments to create the correct embouchure for different registers of the instrument, his fingering techniques and his breathing. Geometric movements – circles, squares and lines with the bell of the instrument – were all influenced by the geometry of the *set-up*. Only arcs which allow this *set-up* to be correctly maintained are possible. Left–right arcs, circles and up–down motions can be performed without difficulty as long as they are relatively smooth and do not jar the embouchure. Data was also gathered from a series of natural and exaggerated communication gestures. Two types of communication gesture were considered – gestures to signal a cue (*start*) and gestures used to synchronise the end of note events (*stop*). These are gestures which contain meaning for other performers. The cue gestures recorded for *start* were largely variable-tempo up–down gestures. *Stop* gestures can be broadly categorised as variable-tempo circles with a small accelerating 'flick' at the end of the gesture.

Movement of the arms, raising the elbow up and down, had little effect on performance or overall performance technique. Both arms could be moved independently or together without restricting fingering action (although it was clear that there may be undesirable theatrical implications in using both arms simultaneously).

In a normal performance stance, the weight of the player was distributed between the balls and heels of the feet. While toes provide a stabilising mechanism, there appeared to be some potential for the use of toe pressure to control switches without adversely affecting the stance or balance.

The implications of applying finger switches, finger-based controllers (data gloves), air pressure sensors and other hardware sensors (e.g. pressure-sensitive resistors attached to the instrument) were considered but eliminated for this application since we felt that a significant proportion of the data they might provide was already contained within the acoustic signal of the saxophone and could be extracted with appropriate hardware and software.

The above considerations defined three categories of performer-derived data which were appropriate for this application: audio data, motion data and communication signals. These are, of course, not mutually exclusive. They are defined in this form to distinguish the distinct mechanisms required to detect, process and map them in the context of this work.

The instrument's movement, the movement of feet (toes) and arms were defined as potential controls in addition to the acoustic audio signal. All of these data sources were used independently and in combination to provide information about the performer's movements and gestures.

Playing an instrument at the highest level requires extremely sophisticated skills on the part of a performer. If performers are in complete control of timbre production, via a complex series of muscular controls, intonation, fingering, breathing technique, etc., it may be impossible for them to fully engage with additional control functions. While there are clear indications that high-level concepts which encompass multiple performance actions can be learnt over long periods of time, certain elements must remain conscious for performers. For example, it cannot be assumed that intonation does not require constant attention by the performer because an experienced player 'plays in tune'; an experienced

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

performer will be sensitive to context-dependent tuning, altering intonation according to harmonic context, spectral context, scale, temperament, etc. Equally, a single additional parameter control will have an impact on all other instrument control processes and can therefore require considerable assimilation.

The sensing system and mapping strategies in *construction 3* establish a relationship between performer and computer by combining several data-sensing and communication strategies simultaneously. This is an attempt to develop a simple model that incorporates some of the inter-performer communication channels which exist during performer–performer interaction, combining communication gestures, aural information and visual cues.

3. SENSOR APPLICATION / DATA SOURCES

Audio data streams were sent from the saxophone to the computer using radio microphone transmission. The saxophone-mounted microphone system was carefully adjusted to produce reliable signals over the range of the instrument. Microphones of this type allow good amplitude repeatability following the calibration of input gain and can eliminate the need for audio signal compression. Many processes which extract information from audio streams, e.g. pitch-tracking programs, are amplitude sensitive and pitch detection error rates were reduced by the careful consideration of the audio signal source.

The capability of contemporary pitch tracking (Puckette 1998) and score following techniques (Déchelle *et al*. 2000) can generate a number of data streams from an audio signal. Continuous pitch, discrete pitch, note onset and rms amplitude were extracted from the audio signal for this application. These were made available to the mapping processes and used for synchronisation and score following.

In-ear audio monitoring was also provided from the computer to the performer via a radio link. The transducers used for movement detection were designed to be discrete and unobtrusive.

3.1. Sensors

The *RIMM* project developed a Programmable Interface Controller (PIC) based system to accept data from a range of analogue and switch sensors. The sensors were connected to a belt-mounted PIC device. Sensor data output by this PIC was sent by radio transmission (433 MHz) to a receiving system connected to the computer. The prototype of the PIC and transmission hardware was mounted in a small belt pack, approx. 120 mm x 80 mm. The receiver and PIC hardware were interfaced directly to an RS232 serial connection on the main computer system.

Arm movement was sensed by two surface-mount accelerometers attached to arm bands and mounted above the elbow on both arms. These were extremely sensitive and could be oriented to provide positional information by sensing the local gravitational field. An electronic compass was attached to the upper microphone mount on the instrument which, along with an accelerometer, was used to provide left–right–up–down data[4] (figure 3). All of these devices were connected to the belt-mounted PIC device and transmitted to the computer system.



Figure 3. Sensor system attached to performer before it was encapsulated by a lounge suit (saxophonist: Jonathan Eato).

In addition to audio and movement signals, two pressure-sensitive resistors (PSRs) were mounted in a pair of shoe insoles. The PSR devices were positioned directly beneath the first toes of both feet. Small switch-clicking elements were added to provide some haptic feedback to the performer and again these were transmitted to the computer from the PIC via the belt-worn radio pack.

The small wires connecting each sensor to the PIC unit were built into a lounge suit. This allowed all the transducers and wiring to be mounted on the performer quickly. All data- sensing technology was small and sufficiently unobtrusive to provide a good prototype for wearable technology.

The implications of wireless technology appeared to be more significant than we had imagined. Trailing wire connections can inhibit performers but also establish a

[4]The compass system was developed late in the project. Much of the original work and the initial realisation of the composition used an infra-red video tracking system to produce up-down-left-right data. A cluster of infra-red LEDs were mounted on the upper microphone mount for this configuration.

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

clear two-element process; the acoustic instrument at one end of the cable and the computer at the other. Removing this connection had a remarkable impact on the perception of performer control. The performer's perspective was reported to be one of isolation from the means of sound production and sound modification while a connection was in place. Removing these connections appeared to establish a more powerful cause and effect relationship between the performer and resulting computer-generated sound. Without a connection, the control of computer-generated and computer-modified sounds seemed more closely allied to the acoustic instrument's control gestures, movements and sound. This seems to be the result of reactions to the acoustic feedback caused by performer actions. A reliance upon audio cues rather than an assumption that electrical signals will produce a desired affect appeared to create a more intimate relationship between the performer and computer. Freedom to move over a large area also had a significant effect on rehearsal practice. To be able to pick up an instrument and play without going through a ritual of connection and the organisation of wires also appears to be significant.

4. MAPPING

construction 3 has a notated score that includes directions for the performer's movements and computer parameter control. The audio and graphic elements of the work occupy clearly delineated, episodic, sections within the piece. The synchronisation methods of performer and computer vary throughout the work; different techniques are applied according to the available signals and the type of musical and graphic material at any instant. The overall structure of the musical score is encoded in a controlling patch. This engages the correct components within each section of the composition and signals the position in the score to the graphics generation computer. A separate score existed for the graphics elements of the piece, and its progression was controlled by the music control patch. The graphics score took the form of a script listing the graphic objects used within each section of the composition and details of the routing strategies to map performer data streams to graphic processes.

The mapping of data from the performer also varies throughout the composition. There are no fixed mappings of a musical process to an action which are constant throughout the piece, with the exception of one signalling operation. This allows a flexible use of the limited number of data streams which were mapped to a range of computer processes. There is a possibility that a variable mapping system of this sort could generate confusion about the result of certain actions. This was addressed in two ways. Firstly, the performer's actions and their effect are carefully notated in the performance score and, secondly, each action is categorised into generic definitions describing the roles of certain controllers – 'more or less' of a particular effect; 'brighter or darker' timbre or graphic; 'faster or slower' time-based processing, etc. The actual result of any action depends upon the current processes in the composition and the location in the score.

The time flow of the composition is controlled by a number of different methods selected, again, for their potential to solve timing and synchronisation problems in the context of the musical material at any instant of the piece (i.e. no single mechanism is defined for the entire work). Score following (with *explode* in *jMax*) is used extensively but not continuously.

Communication signals between the performer and computer are important. For the most part, communication between the computer and performer is based on aural signals and musical events (although there is also a short section of the work which employs a tempo click in the performer's audio monitors, to synchronise a complex rhythmic counterpoint).

The primary method of communication is based on the transmission of information from the performer to the computer. Having the computer 'look' and 'wait' for certain incoming information at a specific point in the work (say a pitch) worked well for local synchronisation during performance, i.e. employing a mechanism to 'look for a G within the next few seconds'. Many of the score-followed sections of the composition are initiated in this way, but rehearsal requires sections of the work to be performed out of context of the piece and can require quick access to sections in the composition. To realise this asynchronous approach, a series of electronic 'rehearsal cues' were implemented to signal the start of each section of the work. These 'rehearsal cues' provide immediate access to particular points in the composition during rehearsal and were also used to confirm progress through the piece during performance. One of the toe switches was used to provide these section boundary signals. This signal was rarely used to change processes or switch score following on or off directly, but signalled the performer's location and initiated a process which would 'listen' for aural cues or 'look' for trigger actions.

4.1. Processing data streams

Data from the performer sensing system was processed to derive the signals which could be mapped to sound and graphics processes. The methods used to process data varied according to the type of signal and data stream. Source data is categorised in three sets, according to its source: audio data, parameter control data and communications data.

The audio signal was used to derive pitch and amplitude data streams. The latter was achieved by a straightforward calculation of the time-varying rms amplitude from the audio stream. This was calculated at 10 ms

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

```
ptparams coefs 1 0.7 0.7 0.7 0.7 0.6 0.6 0.6,
quality-range 0.3 0.6,
amp-range 0.0001 0.001, pitch-range 50 100,
gliss-time 50, stick 0.1, reattack 0.5 100,
debounce-time 150
```

Figure 4. pt~ parameters used to pitch track the soprano saxophone.

intervals and averaged over five values to produce a continuous, smoothed data stream (*ampstream*).

Following considerable experimentation with the *jMax pitch* object (which was unreliable in our patch configurations), *pt*~[5] was applied with the parameters of figure 4 for pitch extraction. *pt* generated continuous and discrete pitch data streams (*pitchstream*) and note onset data (*onsetstream*, which was also calculated from an audio amplitude threshold). To maintain correct intonation, a floating-point MIDI value was used to represent pitch. There can be difficulties in generating correct pitches from pitch tracking processes. Time to convert pitches particularly at low frequencies, the use of prediction algorithms, etc., are well documented elsewhere (Puckette 1998), but the pitch range of the soprano saxophone largely avoids difficulties of this type. Reliable and accurate pitch detection is problematic. One of our solutions was to use the *onsetstream* as a gate to output a value from a continuous pitch stream. We found that errors could be caused by affecting a pitch detection during the early onset of a note. Pitch is rarely stable during the first few milliseconds of a note attack particularly on a tongued reed instrument. Detecting a note during this time results in a significant detection-error rate. This error can be significantly reduced by delaying the detection by a few milliseconds (1–4 ms), in our configuration by delaying the gating action of the *onsetstream*. This allows the pitch to stabilise after its initial attack. The *pitchstream* was applied to a score-following routine (*explode*) to generate pitch material for synthesizer objects and additional event triggers.

The PIC was used to generate data from the accelerometers at the serial input of the main computer system, representing arm positions. The electronic compass and video-tracking systems provided information about the position and movement of the instrument (some averaging was necessary to smooth quantisation steps which occurred in video-tracking and where the accelerometers were used over a small range). *armstream* and *instrumentstream* were used independently and in combination to derive signals for direct control of sound and graphics parameters and to convey information about performer gestures and actions.

instrumentstream contained some data about the movement of the instrument and the performer, including gestural communication signals. A neural network object was designed for *jMax* which operated on two-dimensional time-varying data. The network was trained to categorise certain gestures made by the performer, in the first instance the circle gestures used to signal the end of sound events. In addition to the pattern recognition process, *instrumentstream* was used to derive data about the direction of movement and the speed of the movement of the instrument.

The remaining foot switch data, *footstream*, provided an event trigger stream to actuate sound and control parameters. Continuous data, e.g. *instrumentstream* or *ampstream*, was normalised before transmission to the graphic generation program and separated into multiple data streams (from vector data).

4.2.1. Audio processes

The realisation of *construction 3* called for the combination of computer-generated sound materials, electroacoustic sources, the soprano saxophone and processed and manipulated soprano saxophone. The piece was written to include relatively simple computer 'instruments', where there were clear opportunities for performer control and interaction. This maintained the focus of the *RIMM* project on the holistic combination of technologies and processes rather than the derivation or implementation of sophisticated signal processing algorithms. A multi-voice synthesizer patch was constructed using the *FMO* objects from the Additive Synthesis library of *jMax* to provide the means of synthetic sound production for the piece. Sound files of electroacoustic materials (including synthesised sounds and recordings) were reproduced from computer disk or memory. All other source sounds were produced by the saxophone or by processes which sampled, processed or manipulated the live saxophone sound.

The performer interaction with each sound generation or processing algorithm varies according to the algorithm being used, e.g. the performer might control the playback parameters of electroacoustic materials, the control parameters of signal processing algorithms or the timbre and pace of computer-generated sounds and musical materials. Control of the spatialisation and movement of sounds within the three-dimensional sound field is also possible. Control of certain algorithms passed between the saxophonist and the computer over the duration of the composition. A summary of the audio processes and their mapping can be found in tables 1a and 1b.

4.2.2. Graphics processes

The concept for the graphic component of *construction 3* was to establish movement and the evolution of colour, light and forms over a large canvas. Two attributes of graphics projection were challenged: the 'viewport' window concept, associated with television and many

[5]http://www.ircam.fr/equip/temp-reel/jmax/doc/index.html (checked 22 February 2002)

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

computer graphics – seeing into some other space through a rectangular hole – and the perspective-based graphic 'universe' concept of objects placed, linked and moved around some imaginary space. Peter Fluck devised a series of sculpted screens arranged in three planes which simultaneously created a projection surface and their own static shapes and patterns of forms and shadows which interacted with the projected images. He developed a series of graphic images from works on paper which were taken into the computer, processed and developed to form the basic materials for the work. The images and computer-generated forms were carefully designed to map performance/performer parameters to graphic objects. The images and forms were designed to investigate how parameter streams might evolve shapes, forms and colours interactively. Ironically, given the two projection attributes described above, the graphics generation program written for this application used *OpenGL* (a graphics system based firmly in the concept of a viewport and graphic 'universe' filled with objects) to take advantage of real-time hardware-based rendering.

The mapping strategy I devised for the control and evolution of colour and forms, initially as a demonstration model of how the graphics, sound and performer interaction might function, achieved many of the types of motion and graphic parameter control envisaged in the graphics specification. The concept was developed to become the basis of the graphics approach for the piece and additional graphic materials were generated to exploit the technique.

A graphics system was required to move colour over the sculpted screens and modify or distort forms – e.g. flexing and bowing geometric shapes or lines, making streams of coloured particles swirl and move quickly across the screen while changing colour, density and size. These types of process require definition within an *OpenGL* paradigm and the simultaneous control of many parameters to enable some form of performer interaction.

Texture mapping an image onto an *OpenGL* object provided a simple vehicle to achieve some of the required graphics motion and interactive control. A series of 'three- dimensional' *OpenGL* objects, spheres, rectangles, etc., were defined. The parameters of these types of objects – size, position, proximity to the viewport, colour, transparency, etc. – are simple to access and control and are computationally efficient. The *OpenGL* objects were used as 'projection surfaces' to distort, transform or move images. The 'projected' images can be viewed close to the surface of the object, or from any position which fills the field of view by transforming the size of the object, to eliminate visual cues about the nature of the entire projection object (e.g. a sphere cannot by perceived as a sphere by viewing a small area of its inside face). The data mapping processes applied to these objects are described in more detail below. During the realisation of this graphics scheme, Peter Fluck also devised a method for generating complex patterns which created coloured and intricate Moiré effects, economically generated by two or more texture-mapped *OpenGL* rectangles.

One of the control mechanisms, which was applied to a number of graphics elements, establishes a continuous movement on an *x* and/or *y* and/or *z* axis (usually with different rates for each axis). The mapping of *onsetstream* data to switch on or off one or more axis of rotation results in a note attack that initialises a continuous movement or graphic transformation until the next onset is applied. This takes the current graphic image through a series of smooth transformations. Continuous controllers, such as *ampstream*, can be applied to modify the colour parameters of the projection object, on which an image is texture mapped. This can be used to change how, and if, certain elements of an image are displayed.

Details of graphics mappings used in *construction 3* are presented in table 2. The score uses all audio and graphics processes in different combinations throughout the work.

5. PERFORMANCE EXPERIENCE

The practical experience of performing the composition was an important element of the *RIMM* project trial and its evaluation of hardware and software technologies. It gave some valuable insights into the practical uses and implications of this approach to interactive technology for multiple media generation, and into the strategies adopted for graphics and audio control. Some of these findings are interesting in the context of this paper but are documented more fully elsewhere.

Seven instances of *jMax* were executed simultaneously to realise the composition. In addition to the sound generation and manipulation algorithms of tables 1a, 1b and 2, the patches included nine instances of *Spatialisateur* (*Spat.*) which placed each sound source within simulated spaces in a three-dimensional soundfield surrounding the audience. The implementation of this spatialisation strategy and its configuration on a high-performance computer was one of the key aspects of the RIMM trial. Shared memory devices, and the high inter-processor bandwidth of the HPC, were used to pass multi-channel audio streams between processors and to combine signals which were to occupy the same Ambisonic soundfield. The resulting audio soundfield was decoded using *Spat.* to produce audio signals for a cubic array of loudspeakers (figure 5).

The components of each patch were distributed across seven processors according to their computational load (one processor was used for *root* access during execution). Control signals were passed between instances of *jMax* using *User Datagram Protocol UNIX* sockets.

Adjusting the parameters of the many algorithms used

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

Table 1a. Audio Mapping in *construction 3*.

Process	Performer data stream	Mapping	Mapped to	Resultant
Frequency modulation of acoustic instrument	1. *footstream* 2. *armstream*	1. binary triggers for control messages 2. exponential	1. modulation frequencies data 2. continuous control of modulation frequency	Sax. player can continuously vary multiphonics from specified starting parameters
Amplitude modulation	*armstream*	exponential	frequency of amplitude modulation (with a fixed depth of modulation)	Variable frequency and extreme vibrato of acoustic instrument from 0 to ~28Hz.
'Cut-off'	*instrumentstream up-down-left-right,*	time-delay neural network analysis of data stream to identify gesture, generates binary trigger	amplitude envelope truncation of computer generated sounds or part	The termination of computer and acoustic sounds can be synchronised using small 'circular' gestures
Mixing	*instrumentstream up-down-left-right*	up-down and left-right vectors mapped to exponential and inverse exponential simultaneously	amplitude of four simultaneous soundfiles (selected to create a complex, intricate texture)	Electroacoustic textures consisting of four audio streams can be balanced and shaped by movement of the sax
Soundfile triggers	*footstream*	binary	soundfile data messages	Some soundfiles can be synchronised by foot trigger
Filter/wha wha	*instrumentstream up-down*	exponential	centre frequency control	Wha effect on the acoustic sax. from a bright 'nasal' to a dark 'hollow' timbre by up-down movement of the instrument.
Multi-brassage	audio data	none	three sample tables	Complex, fragmented textures created from brassage of live sax. material
Score follower	*pitchstream*	none	*explode* score follower	Synchronisation of computer generated events to specific sax. notes/note sequences

in this complex series of patches was problematic. Users are, inevitably, required to make changes to the patch configurations during preparations and rehearsals for a performance, to modify the ranges of controller mappings, scaling, mixing, etc. Using a standard computer window, access to this number of patches and parameters is time-consuming and difficult. A number of trials were carried out to integrate MIDI controller units and control surfaces. The integration of multiple instances of *jMax* (or similar programs) in complex configurations of this type may require future information processing and representation strategies. The computer provision for the first performance of *construction 3* also required that certain patches were removed or modified to achieve stable computer performance. Making these changes prior to a performance increased our awareness of the information handling and patch access problem.

Real-time generation of a soundfield, with the ability to control the placement and movement of discrete sound sources and their interaction with the simulated acoustic space, is essential for interactive work that embraces sound diffusion. We were extremely grateful to Olivier Warusfel of the *IRCAM* Room Acoustics Team for alerting us to the potential interaction of *Spat*. room simulations with the physical acoustic space of the concert hall. Early trials showed that this interaction of real and simulated spaces produced a significant effect when moving between large and small spaces. A variety of constructive and destructive acoustic effects can occur according to the *Spat*. parameters used, the distance

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

Table 1b. Audio Mapping in *construction 3*.

Process	Performer data stream	Mapping	Mapped to	Resultant
Sampler/looper	*footstream*	binary	start/stop control of 6 samplers and loopers	Performer can capture notes and motives which are then repeated and overlayed by the computer
Filter/convolver	*instrumentstream up-down*	exponential	centre frequency of bandpass filter/ convolution amount	Saxophonist can emphasise harmonics or control the degree of soundfile convolution in acoustic sax. sound by up-down movement of the instrument
Spatialisation	*instrumentstream up-down-left-right*	time-delay neural networks categorise data into small, medium and large left or right gestures. Generates binary triggers	triggers for sequences of azimuth and elevation data for sound localisation	Performer is able to make 'pushing' gestures to 'throw' sound around the loudspeaker array from the stage at different rates.
Event trigger	*footstream pitchstream onsetstream*	binary	parameter changes for processes and/ or event trigger from score follower trigger data	Performer can modify process parameters and control timing of soundfile playback events by playing sax.
Specific pitch triggers	*pitchstream (onsetstream)*	logical tests, binary output triggers	triggers for control and computer sound event generation	Allows computer to follow performer when no score following process is active. synchronised start/end of sections of the work, process start, computer generated sound events etc.

between the loudspeakers and the walls of the space, etc. These effects can completely alter balance levels and the perceived effect of spatialisation parameters. Any composition generated using room simulation techniques such as those used in *Spat*. and with-height Ambisonic reproduction has to be mixed in the performance space. Mixes are not transferable to other spaces.

Some of our previous experiences with Ambisonic soundfields has shown them to be highly tolerant to acoustic feedback. We had some expectation that an increased error rate in pitch detection, and consequently score following, might result from the sound pressure levels used in performance, but this was not the case. Although we were not able to gather evidence to attribute this tolerance to the use of an Ambisonic soundfield, we suspect that it made a significant contribution. We experienced very reliable pitch detection and score following during rehearsals and performance.

The transmission of control data from the performer's sensors was subject to considerable radio frequency interference, potentially exaggerated by the building structure of the concert hall. We were able to solve this problem, but it took considerable time and effort.

The majority of sound and graphics processes performed as expected and the performance controls allowed the saxophonist to produce a unique interpretation of the work, different in many respects from any earlier versions we had seen and heard. One of the most significant factors in the extent of this control seemed to be that all timing cues were taken from the performer or saxophone sound. Also, the pace of the work, different dynamic levels, accents and phrasing were quite closely reflected by corresponding sound and graphics processes.

Some audience members reported that they had no concept of how the performer was able to synchronise the saxophone part with the sound and graphics over such a large timescale. This *chicken-and-egg* situation was hard to comprehend until I reviewed video recordings of the performance which were made at the rear of the auditorium. Musical events which synchronised live saxophone attacks and computer-generated material had a high potential for the acoustic transmission path of the reproduced sound to be shorter than that of the live instrument for a significant proportion of the audience. Where score-following techniques are used with computer-generated sounds localised to the rear of the

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

Table 2. Graphic mapping in *construction 3*.

Process	Performer data stream	Mapping	Mapped to	Resultant
OpenGL object map	*pitchstream* (section synchronisation signal)	binary, generated from score follower and/or specific pitch triggers/event triggers (audio processes)	render of new OpenGL texture mapped objects, triggers for application of parameter data messages	Maps images onto GL objects for each section of the work, synchronised to the audio progression of the work
Object colour	*ampstream* (event triggers)	exponential to linear	red and/or blue and/or green channel of OpenGL object	Intensity of acoustic instrument reveals/hides elements of the evolving graphic sychronised to sax. note events
Moiré	1. *pitchstream*, triggered from specified score follower events 2. *onsetstream* every nth event 3. rhythmic evolution of moiré patterns	3 binary signals, random selection of x or y or z	1 and 2. change in x or y or z motion of texture mapped rectangle	1. Key note events affect the beginning of a graphic transformation.
Rotate	*onsetstream*, every nth note or on selected note events	binary signals, random selection of x and/or y and/or z	on/off controls for rotation generating algorithms (with fixed predetermined rotation rates)	Note events trigger new graphic evolutions and transformations
Move (rects)	*onsetstream*, every nth event *pitchstream*, via score follower	binary signals, random selection of x and/or y and/or z	on/off controls for translation algorithms (with predetermined rates of movement); triggers also select direction (or pattern of movement e.g. circle)	Causes moiré patterns to initiate new behaviour and colour evolutions
Colour	*ampstream*	exponential to linear	green value of OpenGL rectangle	A photographic image of the screen sculpture is re-projected onto the screens. Sax. loudness appears to introduce green lights behind screens (shadows from a photographed back light are projected onto screens)
Note count	*pitchstream*			Introduces new objects and/or motion parameter data sets

soundfield, accompanying material can be heard before its causal effect. Score following can appear to be ahead of the performer.

There was little evidence that the performer was connected in any way to the computer system, save for a microphone on the instrument. Additionally, some of the gestures used for control had little obvious theatrical mapping to their resultant effects. Whilst this was not a problem for the audience's perception of the piece, it may also cause perceptual confusion about the method of realisation.

6. OUTCOMES

The multiprocessor approach adopted in this project applied a 'coarse' parallelisation strategy – allocating

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

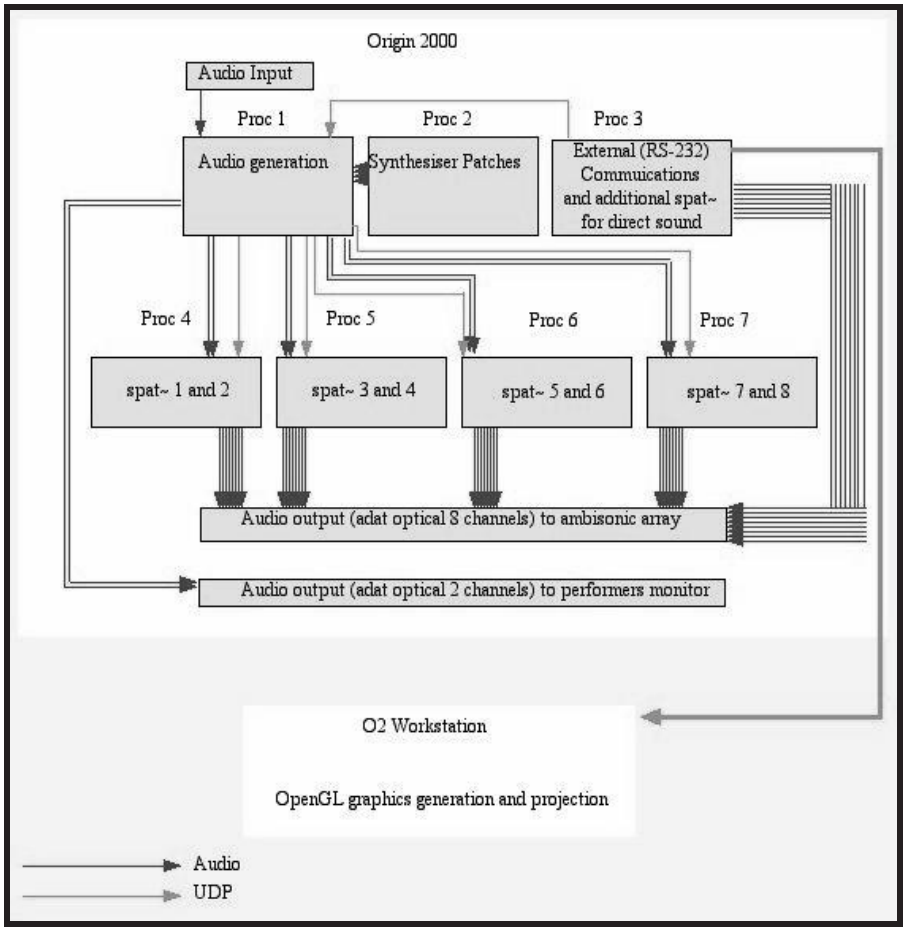


Figure 5. System configuration on HPC, showing processor allocation and Spatialisateur configuration.

generic tasks to certain processors rather than the systematic parallelisation of every algorithm. It would take considerable time to parallelise the huge volume of legacy code associated with applications like *jMax* (*Max*, *pd*, etc.), so this is the only possible approach at this time. Single-processor computers with high-bandwidth networks could provide similar computational resources to those of a High Performance Computer platform in this configuration, but not if any single application used 'fine' parallelised processing routines. Our experiments showed that around 25–28% of the computing resource was used to run the base system of *jMax* (JAVA, FTS, DTD, etc.). Multiple instances of *jMax* result in multiple instances of the base system, so in any configuration that runs or imitates a 'coarse' parallelisation, 25–28% of the total system will be occupied before processing. With seven processors (as in this case) or seven computers, almost two will be devoted to running the base system before any patches or objects are implemented. This is clearly inefficient on one level, but perhaps not if the time and effort to re-engineer the software systems to scale across multiple processors and the potential communication overheads associated with synchronisation, timing, etc., are considered.

We found that it was beneficial to allocate processes such as control and data input and output to one processor, sound generation to another, spatialisation to another, etc., although this was not exclusively the case and depended upon processor loads throughout the work. Inter-process communication and data flow (including multiple multi-channel audio files) did not present bandwidth problems on the HPC platform. The HPC allowed us a freedom to experiment with the software resources in a unified, stable computing environment and was fundamental to our being able to adopt a holistic approach to the integration of the technologies and mapping strategies we developed.

The success of the audio and graphics mapping strategies can be assessed in two areas: the practical use and application of the motion or gesture to the control of an audio or graphics parameter, and the artistic or creative result of the mapping.

Practically, the analysis of the performer and assessment of appropriate sensing hardware developed a good base to establish ease-of-use for the performer. By identifying broad criteria for continuous 'more or less', 'faster or slower', 'left or right' controls of signal processing, 'actuation' switching, the implementation of single interactive processes occupying short sections of the work (such as the cross-fading of four sounds by

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

movement of the instrument), it appeared to be straightforward for the performer to work with motion and positional mappings.

Sensor systems were calibrated for maximum and minimum values and scaling before each performance or rehearsal session. Routines were adopted that made this an easy and quick process. The calibration was done under the control of the performer according to the sound and 'feel' of the response with an aim of increasing repeatability of action/reaction.

The performer was able to control mixing well in the sections of the work which called for the movement of the instrument to shape and develop sound textures while playing the instrument. Cross-fading in this context appeared to work well because it was possible to make appropriate gestures with the instrument related to sound morphologies and nature of sound materials. The mapping between linear distances and exponential amplitude appeared natural and presented no problems for the performer. All the modulation effects applied to the instrument yielded the desired results and appeared to be well controlled by the mappings described in table 1a.

The sensor system and mappings did allow a good and continuous variation of timbre and modulation of the acoustic instrument. No difficulty in the operation of these sensing systems or their techniques of operation were experienced by the performer (except for occasional double strikes on switches which were almost completely eliminated with practice, and the difficulties in audio monitoring of modulation effects while playing the acoustic instrument, caused by masking and proximity to the acoustic sound source).

It was not possible to make full use of the neural network gesture recognition system in the first performance of *construction 3* because of difficulties with the available computer system (instabilities in the configuration of the HPC caused the computer to run less efficiently than the identical machine used to develop the patches, so the networks were sacrificed to allow the remaining patches to run effectively). The use of the networks has only been possible in rehearsal and experimental situations, but it appears to be a valuable technique. Gestures which contain meanings need to be identified before this technique can be applied. The training process requires the preparation of a training data set that illustrates potential variations in the gestures to be classified and other data not related to the gesture. We incorporated the tools of the *Stuttgart Neural Network Simulator* (Zell 1995) to assist in network training and definition. The success of the technique relies upon a clear and appropriate attribution of meaning to each gesture and to generate information and an appropriate mapping of this generative data to musical parameters. We only used the most simple communication gestures to establish the concept, based either upon existing and well understood communication gestures or an extremely direct correlation of action to result (e.g. 'pushing' sounds at different rate to the left or right to control their movement in the sound field).

Future benefits may result from the application of multiple neural networks to categorise different elements of motions and compound gestures, or to identify communication processes or attribute meaning. The implementation of existing performance communication gestures via neural network recognition showed great benefits. This allowed the performer to communicate with the computer using techniques developed for other musical interactions.

Surprisingly, one of the most significant aspects of this project was the freedom and intimate acoustic relationship which was established by the use of wireless technology. This appeared to remove the reliance on, for example, switching actions or movement actions *per se*, in favour of aural relationships between physical cause and aural effect.

The ability to re-interpret the composition in each performance was successful largely due to good synchronisation between performer and computer. This resulted from the flexible approach to synchronisation problems and the implementation of a number of solutions according to the musical circumstances at any instant in the work. There was never pressure for the performer to 'keep up' with computer-generated material. At almost any point in the performance it was possible for the saxophonist to stop, or pause, to begin again or to change the speed and interpretation of a passage in response to the musical context of the performance without the computer losing its place in the work. The combination of logical tests and score following used here will, of course, only function correctly if the performance followed the approximate progression of the score.

Using information about gesture and specific data streams to trigger the evolution of time-based processes, such as spatialisation control and graphics motion/transformation controls (tables 1a, 1b and 2) established a good correlation between performer and audio/visual result.

The colour mapping mechanisms had mixed success. Simple amplitude mapping to the red, green, blue (rgb) parameters of texture mapped graphic objects showed some promise. The process produced good continuous evolutions and transformations of the graphic image which were well controlled. The discovery of the extraordinary effect of re-projecting carefully lit photographs of the screen sculpture back on to the screens perhaps proved to be too seductive for the final audio mappings. When pure red, green and blue lights are used to light the sculpture in a photograph, the rgb controls of the GL object texture mapped with the photograph effectively reproduce the ability to control each photographed lights. Because shadows from the photograph are also projected back on to the screens, the illusion of, for example, back lighting is extraordinary. In a determined

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

effort to find an application for this effect in *construction 3*, we produced a simple mapping of saxophone amplitude to colour intensity variation, and succeeded in re-creating a primitive sound-to-light effect where the direct correlation of audio activity to light variation was too arbitrary to make artistic sense in the context of the piece.

7. CONCLUSION

The creation of a new work as a way of exploring both the technical and artistic potentials of interaction simultaneously, allowing technical developments to lead artistic practices and *vice versa* was a significant aspect of this work. Applying new techniques for human–computer interaction to the generation of new artistic works and processes was extremely important. This generated an artistic rhetoric for new interactive techniques; new music for new and enhanced instruments.

The integration of computer control parameters to instrumental performance technique is complex. The number of different mapping possibilities that will inevitably be applied by composers in the future has some potential to cause problems for performers who will be required to approach a new instrument at each performance. Considering the complexities of established instrumental techniques and natural inter-performer communication may have benefits to reducing the learning curves for enhanced instruments in this context. This may also encourage instrumentalists to apply the motors skills generated over many years, and evolved over many centuries as appropriate mechanisms for musical expression, to the integration and control of computer systems.

The context of performance, sound generation or manipulation is extremely important to work in this field (Orio *et al.* 2001). Reducing investigations of mapping strategies to the consideration of single or a small number of parameter controls/mappings may not reveal musical communication systems that are meaningful to musicians, because the actions are removed from musical contexts.

The entire context of a musical act has a significant impact upon the use and potential of controllers and mapping strategies, upon learning, expressiveness and upon the quality of the musical result. All aspects of interaction, from the interaction with the controller device, the 'feel' of it, its haptic feedback, the relationship of these to the resulting audio, the quality of the audio (or other) result and its potential to inspire exploration, the context of the act in which the musical work is being performed, and the musical context all affect how successful or otherwise mapping strategies can be. A holistic view of interaction, not only in the context of multi-parametric sensing, mapping techniques and control, but also within a musical context of new musical materials, rhetoric and motivation is a vital part of the development and exploration of the interaction between humans and computers.

REFERENCES

Cook, P. 2001. Principles for designing computer music controllers. *NIME workshop – CHI 2001*. New Interfaces for Musical Expression, Seattle. http://www.csl.sony.co.jp/person/poup/research/chi2000wshp/papers/cook.pdf

Déchelle F. *et al.* 1999. jMax: An environment for real-time musical applications. *Computer Music Journal* **23**(3): 50–8. Cambridge, MA: MIT press.

Déchelle F. *et al.* 2000. The jMax environment: an overview of new features. *Proc. Int. Computer Music Conf.*, p. 253. International Computer Music Association, Berlin.

Jot, J. M., and Warusfel, O. 1995. Spat~: A spatial processor for musicians and sound engineers. In *Proc. 2nd Int. Conf. of Acoustics and Music Research.* Ferrara, Italy.

Malham, D. G., and Myatt, A. 1995. 3D sound spatialisation using Ambisonic techniques. *Computer Music Journal* **19**(4): 58–70. Cambridge, MA: MIT press.

Orio, N., Schnell, N., and Wanderley, M. 2001. Input devices for musical expression: Borrowing tool from HCI. *NIME Workshop – CHI 2001*. Seattle. http://www.csl.sony.co.jp/person/poup/research/chi2000wshp/papers/orio.pdf

Puckette, M. 1998. Real-time audio analysis tools for Pd and MSP. *Proc. Int. Computer Music Conf.*, pp. 109–12. International Computer Music Association, San Francisco.

Winkler, T. 1999. *Composing Interactive Music: Technologies and Ideas Using Max*, pp. 23–7. Cambridge, MA: MIT press.

Zell, A. *et al.* 1995. *SNNS, Stuttgart Neural Network Simulator.* http://www-ra.informatik.uni-tuebingen.de/SNNS/UserManual/UserManual.html

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press

https://doi.org/10.1017/S1355771802002078 Published online by Cambridge University Press