Mutable Mapping: gradual re-routing of OSC control data as a form of artistic performance

Ilias Bergstrom
University College London
Institute of Ophthalmology
11-43 Bath Street, EC1V 9EL
+44 (020) 7608 6800
i.bergstrom@ucl.ac.uk

Anthony Steed
University College London
Department of Computer Science
Gower Street, WC1E 6BT
+44 (020) 7679 4435
a.steed@cs.ucl.ac.uk

Beau Lotto
University College London
Institute of Ophthalmology
11-43 Bath Street, EC1V 9EL
+44 (020) 7608 6800
lotto@ucl.ac.uk

ABSTRACT

In most contemporary forms of musical and other technologically mediated artistic performance, systems are used where digital control information is transmitted from the interfaces used by the performers, to the devices producing the intended output. There is considerable ongoing discussion regarding how this control data is mapped between its source and destination; however in this discourse mappings have always only been treated as remaining fixed during the course of the performance. We describe here a novel performance role, that of the 'mapping engineer', whose task is to manipulate the mapping during a performance, gradually altering and re-routing digital control data. Additionally we introduce a user interface paradigm towards facilitating the mapping engineers' task, an implementation demonstrated with the 'Mediator' software.

Keywords

Mapping, Live, Performance, Audiovisual, Open Sound Control

1. INTRODUCTION

In modern musical performance, the devices serving as interfaces for the performers (controllers) are separated from the devices producing the actual output (sound generators) [1]. The controllers detect the musical gestures of the performers, and encode these using discreet digital data for the sound generators to interpret and produce the final sound. This is in contrast to traditional musical instruments, where the device used to detect the musical gestures (e.g. the piano keyboard) cannot be separated from the sound generator (the pianos hammers and strings). Instruments where the gestural controller can be separated from the sound generator are collectively referred to as Digital Musical Instruments (DMI's) [1]. The industry standard communication protocol for gestural controllers to communicate to sound generators is MIDI, with the much more modern Open Sound Control (OSC) protocol [2] now beginning to gain prominence.

This separation between control and output devices can today be witnessed in many more forms of artistic performance. Stage lighting rigs are remotely controlled using purpose-made control devices, which transmit data following the DMX protocol. Visual performance artists such as VJ's, use off the shelf or custom made controllers to generate MIDI and OSC data, which is to be interpreted by VJ software and other real-time computer graphics applications. In modern dance, it is increasingly common that dancers have sensors attached to their bodies, which transmit data to be interpreted by real time musical and/or visual devices, thus allowing the dancers' movements to influence the music played, as well as the imagery projected on the stage. Finally, in the New Media Arts movement, some or all of the above mentioned conducts are fused, resulting in interactive multimedia performances and installations, oftentimes allowing also the audiences to be directly involved in the piece/performance.

The fact that different data protocols are employed for different contexts of use has purely historical reasons only, as the data transmitted is of a very similar nature in all cases. Due to this very high degree of similarity, today modern systems converge towards using what has emerged as a potential future universal standard: OSC. Despite it having originally been designed as a protocol for musical control data, OSC has in addition proven to be equally suitable for all above application areas. Existing legacy devices can also be addressed using OSC data, since OSC commands can be translated to and from most legacy protocols with relative ease. In addition, modern developments in signal processing have made it possible to derive digital control data even from analogue sources, thus allowing for example that discreet musical notation data is derived from audio of an acoustic live performance [1].

The connection between one or more sources in the controller parameter space, and one or more destination in the parameter space of the controlled devices, is referred to as mapping [3]. A simple mapping is where each controller parameter is connected to precisely one parameter on a single device on the receiving end (one-to-one). Mappings can also be complex, where either a single controller parameter is connected to multiple parameters, possibly on multiple receiving devices (one-to-many), or a single parameter on the receiving end can be the target of more than one parameter from the input parameter space (many-to-one) [3].

Most recently, controllers initially created for one context of use, are often successfully repurposed for another, while a large body of research has been directed towards creating new, novel controllers. Additionally universal controller interfaces are created, agnostic in their design with regards to what context they are to be used in. In all cases, the output of the controllers no longer has a default mapping to the parameters of the devices they control. Instead, when preparing for a performance, artists define their own mappings, based on their personal preferences, artistic and pragmatic.

Ace 2009, Oct 29-Oct 31, 2009, Athens, Greece

All previously mentioned conditions pave the way for a new form of artistic performance: that of gradually creating, destroying and altering mappings between the two parameter spaces of input and output control data.

This novel conduct is inspired by a development which occurred when music began to be electrically amplified and recorded: the emergence of the mixing engineer role. As the audio signal from the performance of each individual musician could be conducted in its entirety through a cable, the signal could also be manipulated to affect how it sounded. The mixing engineer role emerged, along with a new set of tools for him/her to use, the centerpiece of which is the audio mixing desk, usually accompanied by a multitude of different audio processing effects. Though initially mixing engineers were regarded only as technicians, this soon changed, and they increasingly began partaking in the creative process, thus blurring the creative roles. The established role of originating the music is now accompanied by live alteration of the signal, with the end result being a joint creation of both instrument players and live mixing engineer, all indispensable members of the group. Today in many cases this development has gone full circle, with mixing engineers taking centre stage as the main artist, while the instrument players – if these have not been entirely replaced by recordings – hold only a secondary role. A contemporary expanded form of the above performance mode is at present embodied in the widely popular Ableton Live software [7]. In Live, the traditional mixing engineer toolset is augmented with facilities for detailed live triggering of multiple short loops of pre-recorded music. Today many electronic music composers use Live in order to perform their music as solo artists.

Analogous to the mixing engineer, the mapping engineer fulfils a similar role, but instead of manipulating the audio signal produced by the sound generators, he/she manipulates the digital control data produced by the gestural controllers.

2. RELATED WORK

Although there is extensive past and ongoing discussion regarding mapping, primarily with relation to developing new DMI's, no previous publication has been found on gradually altering mappings as a form of performance. The only work found which is somewhat related to the ideas presented here is that of Joseph Malloch et Al [4]. They have developed the application OSCMapper, intended to be used as an aid in the process of designing new DMI's, through allowing the quick prototyping and alteration of mapping setups. Their software application has functionality analogous to that of the application presented here, with the significant difference however that it does not allow the gradual altering of mappings, only switching them on/off and introducing discreet, non-animated changes, thus rendering it unsuitable for use by a performer in the role of mapping engineer.

We very briefly introduced the idea of mutable mapping in a recent previous paper [5], where it was described as part of a wider context. Here however the idea is described in full detail, along with the first concrete solution for how it may be practically facilitated, and integrated in a real performance setting.

3. THE MAPPING ENGINEER

The new role of the 'mapping engineer' allows for novel forms of performance, both within only a single artistic conduct (e.g. music), as well as performances where multiple art forms - for example music, visual art and dance - are fused.

The role of the mapping engineer can vary greatly depending on the specific context it is applied to. Moreover the role is bound to evolve significantly once a culture of use is established, as it adapts to the particular idiosyncrasies of each context. Arguably, new not yet envisioned forms of use are bound to appear. Following is a non-exhaustive list of examples detailing possible use cases.

In musical performance, the control data originating from DMI controllers is subsequently routed through the Mediator system. The mapping engineer then alters the control data and selectively redirects it to the control parameters of one or many sound generators on the receiving end. Examples of what possible alterations may be introduced:

- Arpeggiators/automated accompaniment may be gradually applied to the notes from an individual instrument
- Control data may be gradually rerouted between different sound generators
- Procedural modifications may be applied, so that for example note length or pitch are altered
- Combinations of the above allow for complex "layering" techniques, through combining the sound from multiple sound generators into one intricate evolving instrument resulting from their fusion.

It is also possible that the mapping engineer acts as a solo performer, in which case the control data is either pre-recorded - possibly performed using Ableton Live, or generated algorithmically.

Beyond solely musical performance, in a context where live music is accompanied by visual art, for example VJ-ing, real-time computer graphics or light shows, the parameter space addressable by the mapping engineer grows significantly. Control data originating from DMI's can now also be mapped to the control parameters governing real-time generated computer graphics, VJ software, or the lighting rig [6].

The novel advantages stemming from such interconnection are significant: it facilitates creating a narrative in the visual output that is closely correlated to that of the music, because both music and visual performances are primarily controlled using the same control data [5]. The ability to gradually re-route control data in this context is imperative, enabling performers to construct a non-linear, dynamic and free-form narrative in the correlation between music and visual accompaniment. This is in stark contrast to established current practice in VJ-ing and music visualization, where the mapping between sound and graphics is at best typically reduced to the beat and amplitude of the music. Such mappings quickly become highly predictable, thus limiting the use of suspended expectations [9], as well as tension and release [10], both crucial aspects of artistic/esthetic narratives. It is known that the human perceptual system is apt at detecting correlated stimuli across modalities, and fusing these into a single percept before their interpretation [11]. More detailed correlation may thus further encourage the unified experience of music and image, and a necessary requirement for achieving more detailed correlation is to have a user interface with which the artist/performer may readily define and alter complicated mappings between input and output control data. It is for this use case we have conducted initial testing, with successful enough outcome to justify pursuing future larger scale work.

In all contexts of use, either a single mapping engineer can work alone, or multiple may work collaboratively, each responsible for his own group of destination parameter spaces. Conversely a solo audiovisual performer can map pre-recorded or algorithmically generated control data both to audio synthesizers and to visual synthesizers and/or VJ software.

Note also that a solo performer need not restrict him/herself to the role of mapping engineer. He/she can also simultaneously perform as an audio mixing engineer, as a VJ, as a lighting artist or even as a musician using conducting gesture controllers [1], or software such as Ableton Live [7].

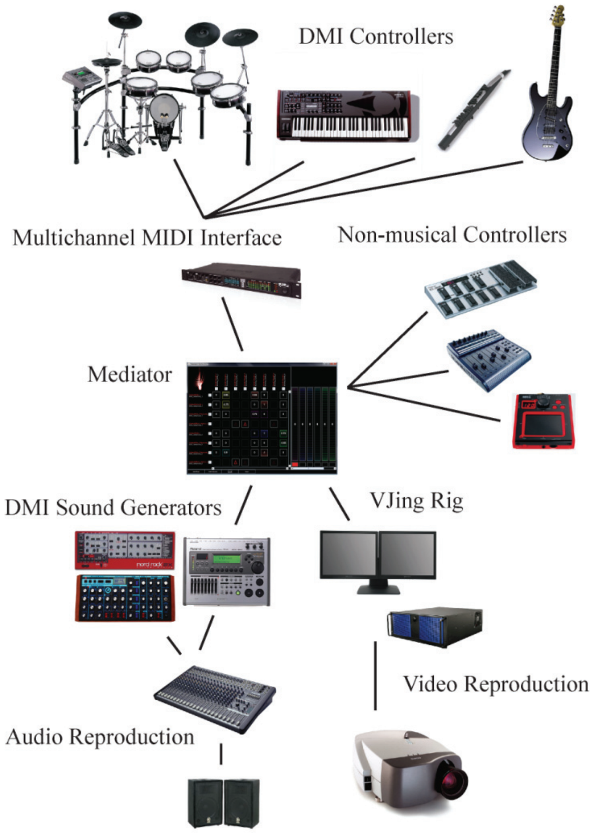

Figure 1: Signal flow example, top to bottom.

Like the role of live audio mixing engineer, the role of mapping engineer is not singularly defined, and varies greatly depending on context. He/she may be an indispensible member of the group, or just a technician, depending on the level of initiative and intervention he/she has in augmenting the performance of the musicians. The mapping engineer may furthermore be partly or wholly a content creator for the performance, such as when performing using pre-recorded material of his/her own creation as a source of control data, or when control data initially generated by musicians concentrating on its' musical use, is used to control live visuals.

At the time of writing, the role has been identified, and initial experimentation has taken place. The conduct was conceived in connection to the concurrent performance of correlated music and real-time visual art, and this field is where mutable mapping has initially shown to be useful in practice. It is however not difficult to imagine the paradigm being usefully applied to many more artistic conducts, providing for intriguing future research.

4. PROPOSED USER INTERFACE: THE MEDIATOR SOFTWARE

To support the mapping engineer role, we propose a user interface paradigm, which we have implemented in a prototype application dubbed the 'Mediator'. The only control data protocol supported is OSC, which has been selected because it is universally compatible with all domains where described program would be useful. It is also with little effort compatible with most legacy protocols, as their vast majority may be translated to OSC.

The ideal implementation is envisioned as a standalone hardware controller with a multi-touch screen, the device for which is not unlike the JazzMutant Lemur controller [8]. The Mediator's current incarnation however is as a standard software application controlled using a single pointing device, presently a touch screen.

The central feature of the Mediator user interface is the 'mapping matrix'. Each of its rows corresponds to the source of a single control parameter, while each of its columns corresponds to a single destination. Each row or column has a type assigned to it: floating point, integer or text value. At the intersection of each row and column appears a cell, thus forming a matrix. Rows and columns can be dynamically added and removed at runtime, and their assigned OSC address can be altered.

Figure 2: Input row & cell detail

The predominant data type handled is numerical values. For these, unlike a patch panel, in which the cell connecting source and destination may only be on or off, the mapping matrix is novel in that it is weighted. Each cell holds a floating point value, which is used as a factor with which incoming numerical values are multiplied before they are forwarded to their destination. A value of 0 means incoming values are not forwarded at all. Changes in connections may thus be introduced gradually, by increasing or decreasing the cells' multiplier values, allowing for much more detailed performance than just the on/off states possible using a normal patch panel. Cell's values are altered through dragging, and thus on a multi-touch screen the paradigm allows for the values of multiple cells to be altered simultaneously.

When a cells corresponding row and column both are assigned to handle text values, the cell changes behavior to normal patch panel-style on/off forwarding of the unaltered data. Finally if the source row's type is numerical and the destination column's type is text or vice-versa, the values are incompatible and thus their corresponding cell is grayed out, supporting no form of forwarding at all.

In addition to the mapping matrix, numerical control values can be further altered using input and output offset values: a number that is added to the incoming value prior to multiplication with the cell values, or added to outgoing values after multiplication with cell values. These offset values are set individually for each row or column, by dragging the numerical field appearing to the right/bottom of the row/column identifier.

Finally, numerical cells can be assigned to groups. Those which appear with a white outline and red centre are not assigned to any group, while those which have the outline and centre coloured with the same colour are assigned to that colour's corresponding group. A numerical cell is assigned to a group through tapping/clicking it.

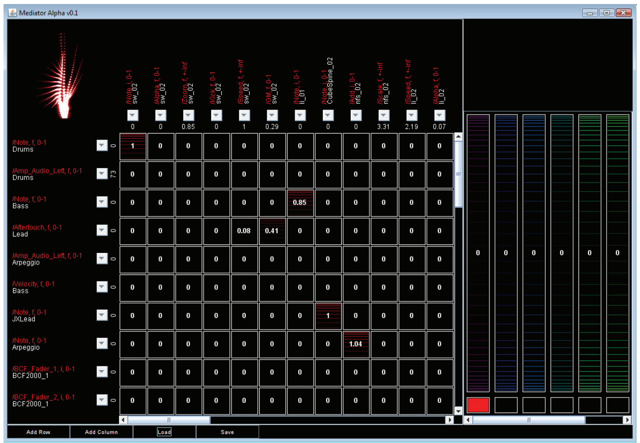

Figure 3: The Mediator user interface.

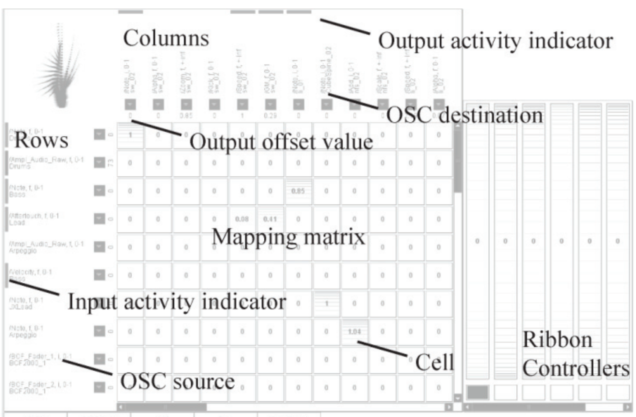

Figure 4: User interface legend

Each group is represented by one out of eight ribbon controllers located to the right of the mapping matrix. By dragging the ribbon controller up or down, the user can increase/decrease the multiplier value of all cells assigned to the group simultaneously, by the amount corresponding to how far the ribbon controller is dragged. Multiple ribbon controllers can be manipulated simultaneously, thus allowing the concurrent modification of a large number of cell multiplier values. Under each ribbon controller appears a button, which is coloured red if it is the currently active controller. When a cell is tapped, it will be assigned to that ribbon controllers. With no ribbon controller active, tapping a cell clears its group assignment.

From the authors' previous experience with programmatically implementing mappings, the above interface supports the creation of a vast majority of possible desirable mappings, through allowing a multiplier and two offsets, one before and one after multiplication. Having said that, it is a simple exercise to describe a mapping that our current implementation does not support; for example, any mapping that requires a non-linear value transformation, such as spline interpolation, is currently not feasible.

Despite its current limitations, we have found from our initial informal experimentation that the presently defined incarnation of the program is a sufficient platform using which the new artistic conduct of mapping as performance can both be practiced and further developed. What the current design has to its advantage is that it achieves quite a powerful set of features with simple and intuitive means, arguably a highly important criterion for a user interface designed to be used in real-time performance.

5. DISCUSSION / FUTURE WORK

As the system is designed and implemented currently, new devices connected to the OSC network are not automatically detected, but need to be declared explicitly in order for them to appear as sources and/or destinations of control data. This could be remedied in the future however, by implemented one or more of the proposed protocols for this purpose [4].

Through future research and user trials, the first prototype of the Mediator program presented here may be employed to further refine the idea of mapping as performance, while additionally also help describe a second incarnation of the program which better facilitates the task at hand.

6. REFERENCES

[1] E.R. Miranda and M.M. Wanderley, *New Digital Musical Instruments: Control And Interaction Beyond the Keyboard*, AR Editions, 2006.

[2] W. Matthew, "OpenSound Control: A New Protocol for Communicating with Sound Synthesizers," *Proceedings of 1997 International Computer Music Conference, ICMA*, 1997, pp. 101-104.

[3] A. Hunt, M.M. Wanderley, and M. Paradis, "The importance of parameter mapping in electronic instrument design," *Proceedings of the 2002 conference on New interfaces for musical expression*, Dublin, Ireland: National University of Singapore, 2002, pp. 1-6.

[4] J. Malloch, S. Sinclair, and M.M. Wanderley, "From controller to sound: Tools for collaborative development of digital musical instruments," *Proceedings of the 2007 International Computer Music Conference, Copenhagen, Denmark*, 2007.

[5] I. Bergstrom and B. Lotto, "Harnessing the Enactive Knowledge of Musicians to Allow the Real-Time Performance of Correlated Music and Computer Graphics," *Leonardo*, 2009.

[6] I. Bergström and B. Lotto, "Mother: Making the Performance of Real-Time Computer Graphics Accessible to Non-programmers," *(re)Actor3: The Third International Conference on Digital Live Art Proceedings*, 2008.

[7] "Ableton Live 7", http://www.ableton.com/live

[8] "JazzMutant Lemur", http://www.jazzmutant.com/lemur_overview.php

[9] N. Steinbeis, S. Koelsch, and J.A. Sloboda, "The Role of Harmonic Expectancy Violations in Musical Emotions: Evidence from Subjective, Physiological, and Neural Responses," *Journal of Cognitive Neuroscience*, vol. 18, 2006.

[10] B. Evans, "Foundations of a Visual Music," *Computer Music Journal*, vol. 29, 2005.

[11] P. Larsson, "Virtually Hearing, Seeing, and Being: Room Acoustics, Presence, and Audiovisual Environments," *Doctoral Thesis,* 2005