Gestroviser: Toward Collaborative Agency in Digital Musical Instruments.

William Marley
Digital Media and Arts Research Centre
Department of Computer Science and Information Systems
University of Limerick
William.Marley@ul.ie

Nicholas Ward
Digital Media and Arts Research Centre
Department of Computer Science and Information Systems
University of Limerick
Nicholas.Ward@ul.ie

ABSTRACT
This paper describes a software extension to the Reactable entitled Gestroviser that was developed to explore musician machine collaboration at the control signal level. The system functions by sampling a performers input, processing or reshaping this sampled input, and then repeatedly replaying it. The degree to which the sampled control signal is processed during replay is adjustable in real-time by the manipulation of a continuous finger slider function. The reshaping algorithm uses stochastic methods commonly used for MIDI note generation from a provided dataset. The reshaped signal therefore varies in an unpredictable manner. In this way the Gestroviser is a device to capture, reshape and replay an instrumental gesture. We describe the result of initial user testing of the system and discuss possible further development.

Author Keywords
Reactable; Machine Collaboration; Stochastic Methods; Instrumental Gesture

ACM Classification
H.5.2 Information Interfaces and Presentation (e.g. HCI): User Interfaces; H.5.5 Information Interfaces and Presentation (e.g. HCI): Sound and Music Computing; I.2.11 Artificial Intelligence: Distributed Artificial Intelligence.

1. INTRODUCTION
There is a long tradition of work that focuses on the design and implementation of systems for musical machine collaboration [1, 2, 3]. Typically these systems respond to the audio or MIDI representation of a performers sonic output and generate musical accompaniment. In these systems the machine collaborator seeks to imitate the role of a human musician improvising in response to the first musician's playing. The machine generates a second voice or sonic output to that of the human player.

Our work differs somewhat from this in that we wanted to explore a system where the machine can collaborate with the player in the creation of a single sonic output. The aim was to pass control of some of the synthesis parameters to the machine whilst allowing the performer to interact with others. The simplest analogy is perhaps that of two players performing on one instrument. We are motivated by an interest in how the notion of instrumental agency, the situating of agency within the musical instrument, affects the quality of the musical interaction experience. Noting Cook's [4] comments with regard to problems associated with programmable instruments, we accept that a variation in the reaction of the instrument to input can make virtuosic control difficult. However we believe the situation to be not all negative. Influenced by notions of playfulness, playful exploration of sound, and serendipitous musical discovery, we wish to explore the space where a reactive and somewhat unpredictable response to input might be appropriate. At its simplest, the functionality of Gestroviser could be likened to the recording of automation data for a musical parameter and then playing it back. However, we were interested in extending this functionality by endowing the system with agency whereby it modifies the input. We wanted the system to reshape the physical gesture, generating a new but related control signal. This control signal can then be applied back onto the module. Our incentive for this design was to encourage sonic exploration of the instrument. Our thinking here is that such an approach could be useful to explore a sense of playfulness and serendipity. We claim that serendipity and playful chance occurrences may provide for more fruitful and exploratory musical experiences than the linear determinacy of conventional score following.

2. BACKGROUND
This research positions itself amongst two fields of digital music system (DMS) design. Within the New Interfaces for Musical Expression (NIME) community we see a focus on tangible user interfaces and body-centered instruments. Physicality and the role of the body remains a central theme and how this physicality can be inserted within electronic music performance. Here the notion of a physical gesture has guided consideration of how the performer interacts with an interface.

2.1 Musical Metacreation
The objective of metacreation is to equip machines with the capacity to be creative. Musical metacreation (MUME) is concerned with autonomy and agency in composition and performance [6]. MUME is a multidisciplinary approach, utilizing tools from Artificial Intelligence (AI), Artificial Life (AL) and Machine Learning (ML). The practice aims to develop metacreations - artificially creative music systems inspired by human creativity and/or systems that surpass human creativity. It is also the basis for focus on computational modeling of human cognitive creativity, leading to computational creativity. Finally, and more relevant to our work, MUME practitioners aim to explore computational creativity through the development of interactive systems and

interfaces for use by musicians and artists. As proposed by Eigenfeldt et al [6], metacreative systems fall under the categories of improvisational systems (online) and compositional systems (offline). A focus toward online systems takes precedence in this research, as it is more concerned with how the system interacts with a live performer in real time. A taxonomy of metacreative systems has been outlined by Eigenfeldt et al [6] as a means of classifying metacreative applications in terms of human designer/composer control over musical output. Their taxonomy consists of seven levels, ordered from least machine-autonomous to most:

1. **Independence** – any process on a musical gesture that is beyond the control of the composer.
2. **Compositionality** – the use of any process to determine the relationship between pre-defined musical gestures.
3. **Generativity** – the generation of musical gestures.
4. **Proactivity** – system/agents are able to initiate their own musical gestures.
5. **Adaptability** – a) Agents behave in different ways over time due to their own internal evolution; b) agents interact and influence one another.
6. **Versatility** – Agents determine their own content without predefined stylistic limits
7. **Volition** – Agents exhibit volition, deciding when, what, and how to compose/perform.

The following are two examples of the online model, designed to extend the creative reach of computer music systems.

George Lewis' *Voyager* system (1985) is an example of a highly complex, deliberately personal and independent metacreation [1]. It is designed to perform at will with single or multiple performers in the style of free improvisation. This style leans towards multi-rhythmic musical content played by several groups of ensembles - sometimes simultaneously, sometimes not – with its 64 asynchronously operating "players" or agents. *Voyager* functions as "an extreme example of a "player" program, where the computer system does not function as an instrument to be controlled by a performer" [1].

Omax [2] uses on-the-fly statistical learning for virtual improvisation generation and stylistic model archiving. Omax's functional concept is based on sequence modeling and statistical learning in a hybrid-architecture of the softwares OpenMusic and Max. The term 'Stylistic Reinjection' was coined by Assayag et al [2] to describe the process of systematically re-sending mirror images of a performance back to a performer. With this feedback of musical data, the performer must react accordingly. Thus, in turn, the future of the performance is in constant flux. Within Omax, sound-images are memorized, stored as compressed models then reactivated as similar but not identical sequences. When building Omax, a particular guideline was set in regard to speed of learning. Learning must take place quickly and incrementally in order to come in-line with real-time interaction.

It can be said that both systems exhibit features that position them within the MUME taxonomy of metacreative systems. Another commonality among these and most other metacreative systems is an emphasis on score following, note generation and accompaniment. In contrast to this, our research is concerned with characteristics and qualities of a physical gesture. We want to use these high-level features as input. We are not concerned with the design of a system that analyses the notes we play or the score we supply it. Instead, we wish to focus on how we play the instrument and the process of constructing the music. We do not consider the artificial agent as accompaniment to our musicianship, but as the instrument itself that must be navigated and explored to produce a single sonic voice.

Toward these ends, we developed a system in which an artificial agent responds and reacts to the user in different ways. In the Gestroviser we take the classic paradigm whereby the system imitates a second performer improvising or accompanying the main performer. In this case, the human player is performing with an artificial instrument. The Gestroviser was implemented in the Reactable, a popular tabletop DMI (briefly discussed below).

3. THE REACTABLE

The Reactable is a musical instrument based around sound synthesis that employs a tangible interface. The interface consists of a table top upon which are placed pucks. Beneath the tabletop a camera based fiducial tracking system is used to track the location of the pucks and a projector is used to project visual feedback by projecting onto the tabletop. Users interact by placing different pucks onto the tabletop and by adjusting the spatial arrangement of pucks on the table. Several different types of pucks exist with different functions that correspond to traditional synthesizer elements, such as oscillators, filters and envelope generators. The proximity of pucks to one another controls the development of virtual connections between them, which correspond to the signal path of the synthesizer. Rotation and movement of the puck adjust the parameters of the represented synthesis function [5].

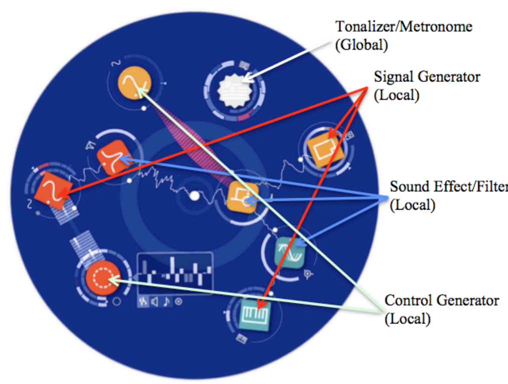

Figure 1. The Reactable with 'Local' and 'Global' objects

4. IMPLEMENTATION

4.1 Playing with Gestroviser

Dave is about to perform on the Reactable. Situated on the table are two signal generators (a sine wave and a sawtooth), an effect module (distortion), a filter module and the Gestroviser. The sinewave oscillator is linked to a band-pass filter. This signal is sent to the output. In a separate signal chain, the sawtooth oscillator is linked to a distortion module. This signal is also sent separately to the output. As Dave begins to play, he rotates each module changing relative frequencies, filter resonances, effect amount etc. He notices that a certain filter sweep, caused by a specific set of gestural movements, sounds especially interesting. Dave links the Gestroviser to the filter. He sees a visual countdown from eight (bars). This is his allotted time to perform the physical gesture. Dave applies this same gestural movement to the filter as he did earlier. When Dave completes the gesture and the countdown finishes, the Gestroviser immediately replays this same gesture movement back to the filter. Dave can now watch the Gestroviser control the filter in exactly the same way as he did moments before. As Dave moves the finger slider of the Gestroviser, the rotation of the filter begins to change slightly. Dave notices the Gestroviser applying subtle changes to his original gesture on the filter. This movement continues to change as he continues to move the finger slider. As he reaches the maximum setting on the Gestroviser, the original gesture is radically altered while still retaining elements of the original.

4.2 Technical Implementation

For reasons of simplicity we initially chose to ignore translation gestures applied to pucks upon the table and instead focused on the rotation data produced when the player rotates a Reactable puck (host). This streamed data represents the reorientation of the tangible blocks situated upon the table. In the standard Reactable software the main function of this rotation data is to communicate audio-signal alterations (effects etc.) to the synthesis engine. The Reactable also has a global timing procedure that is used as a clock to keep all native sampler, sequencer and timing-based effects syncopated. As the Gestroviser would function by means of data capture and playback procedures, it was important to integrate this global timing structure in its development.

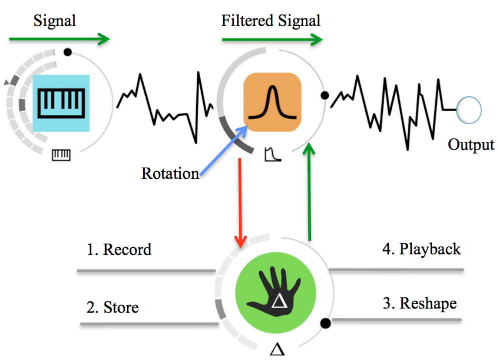

Figure 2: Gestroviser Data Flow

As the host object is rotated the data stream is bifurcated, applying filtering to the original signal while being sent to a ‘sample-record-store’ function within the Gestroviser. This procedure is activated through the virtual ‘linking’ operation between objects (figure 2). This stored data represents a users physical gesture, which is to be then processed for playback to the host object. We sample the input data at 50 smps per second. This low sample-rate is suitable when devising the current method of data capture and regeneration.

We segment the input stream in real-time by feeding the sampled data consecutively into 8 storage buffers. The rationale behind this method is twofold: we wish to minimize the quantity of data points to be processed at one time, while also allowing the re-organization of segments for playback generating new gestures similar to the original. A method of choosing different playback modes was devised. This method is applied to enhance the variability of output, while still resembling the original gesture. Instead of using a discrete choice operation, an alternative method was needed using the finger slider feature of the Gestroviser object. A procedure was applied where the control of the slider would increase and decrease probabilities of the activation of each play mode (figure 3).

Four different play modes are available. At the highest point, marked 1., the original gesture has 100% probability of being chosen for playback. In mode one, each storage buffer is read through consecutively outputting unchanged data back to the host. As the slider moves downward, this probability decreases and the probability of the next mode becoming active increases. As the slider reaches the point marked two on the diagram, the probability of mode two is 100%, with mode 1 at 0%. Mode two is similar to the process involved in mode one, except the data is read from two segments behind mode one. As mode one and two are activated based upon probabilities, the variability of output is evident.

As can be seen from figure 3, the final two modes of playback consist as ‘Processed’ and ‘Processed Shuffle’. For these modes, a state-transition matrix (STM) was created for each segment of samples and a 2nd order Markov chain constructed from the data. A second order STM is created by pairing couples of incoming gesture data for use as indices. These indices represent previous ‘states’, in this case the previous two discrete states of the gesture. The STM is then filled by registering all sampled points that would occur after these states.

The probabilistic process then occurs by choosing randomly from the list of possible points after each index event. However, this is not total randomness. For example, 10 sampled points may be present in a particular list. Out of these 10 points, 5 of these could be the same point eg. 50. Therefore, the point 50 would have a .5 or 50% chance of being chosen from that list. After each new STM is generated, they are then ready for immediate playback. On playback, each STM is read through with newly generated sampled points. Due to the use of probability in choosing output points, a variation on each generated gesture is likely.

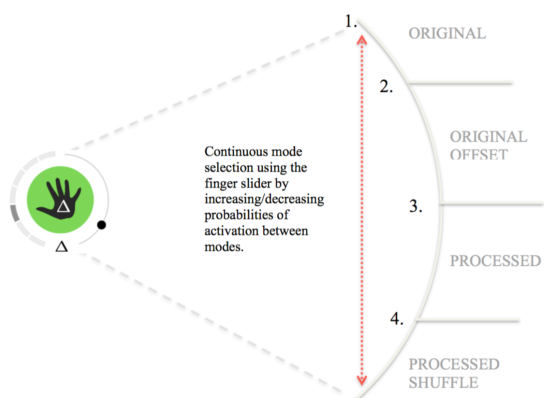

Figure 3: Finger Slider Play Modes

5. USER TESTING

Test users were chosen to play with the Reactable configured to include the Gestroviser addition. Subjects were presented with pre-test and post-test questions. The main criterion for selecting participants in the testing was that they had experience using interactive computer music systems. We withheld information on the functioning of the system from the users. We wished to make judgments on the speed and extent of their understanding of the system. Could they know what was happening straight away? If not, how long would it take them to understand? Is the system intuitively playable without prior knowledge?

It was possible to attribute certain post-test responses to particular users that were observed. One user stated that improvisation was their main form of performance. At times this subject would apply chaotic and speedy movements to all objects, yet their familiarity with the modules and their functions suggested experience in Reactable performance. This user tended to supply the host object with small rotation gestures and wait for the Gestroviser to respond. After the recording of the gesture had taken place, the Gestroviser responded in kind with small rotation movements. When asked to comment on this after the test, this user expressed a preference for more "activity" from the system ("more and faster output density"). In our view, this was a natural function of the Gestroviser; if it is provided a gesture with little activity, it responds with little activity. However, this raises interesting questions regarding the level of playfulness and surprise expected from a machine collaborator. Is a user that is more experienced in music improvisation expecting a wider boundary for indeterminacy? Are they hoping to be surprised by more dramatic interpretations of their musical input? Perhaps this is the case, and the design of a system like Gestroviser needs to accommodate users from such musical practices.

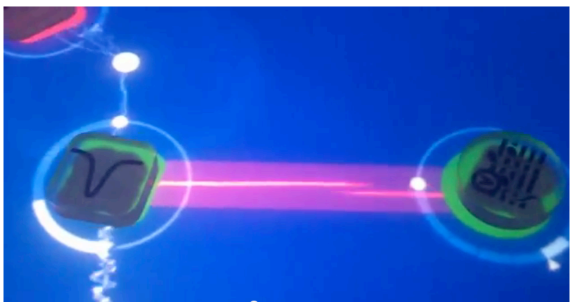

Figure 4: Gestroviser in action – signal in red representing the reshaped gesture applied back to a filter.

We observed another user who admitted having less experience using interactive music systems than listening to music made on them. They experienced light confusion during the test, requiring an explanation of the system when the test was completed. When an explanation was supplied, the subject asked to use it for longer without being tested. With the knowledge of the process involved, they expressed more confidence when re-using the system. When asked about their level of engagement they stated that they were mostly engaged in trying to understand the purpose of the system rather than exploring their creativity.

The third user was an experienced electronic musician, who regularly used DMI's as controllers for composing and performing music. We observed yet another different approach to using the Gestroviser and a quicker understanding of the systems functionality. It is interesting to consider their own thoughts on the Gestroviser: "[I] really liked the idea of the computer being able to sensibly improvise. [I] liked how some of the characteristics are retained in successive loops, but at the same time there are small changes, which give a sense of surprise. This also makes the listener focus more on the musical nuances that keep changing throughout the performance. The ring modulator wasn't as obvious as the filter… [It] was interesting to see the same idea with different control objects/tone generators, even though the sonic output didn't necessarily make total sense to me when applying it to these other objects. [I] would like to spend more time using this to properly get used to performing with it".

A feature noted by all users was the Gestroviser's graphical feedback. As stated earlier, the system starts a graphical countdown as soon as the recording function begins. All test users stated that more graphical labeling and text instructions would be useful in their understanding of the systems functionality. These features are already present for other objects on the Reactable, therefore an absence of visual feedback on the Gestroviser becomes more evident.

6. DISCUSSION

We feel that these varied responses from users with different musical backgrounds provide useful and interesting insight into the potential for a system that reacts and collaborates with a human player. We have learned that a greater understanding of the user by the machine is necessary. The integration of machine learning methods of physical gestures is an obvious course of action in future implementations of this particular design. We purposely bypassed this area in the design of Gestroviser, preferring instead to consider methods of data generation. We look to the important ongoing research in 'on-the-fly' machine learning of user inputs for generating new mappings by example [7]. These methods and applications are encouraging as we look to implement real-time learning of performer behavior and gestures on DMI's.

Our future work involves the implementation of videogame AI routines in a DMI. In this work, we move on to explore the notion of performing **on** a system with artificial intelligence as opposed to performing **with** the Gestroviser. Again, like the Gestroviser, we develop a system that takes physical gesture data as input. Here we want to consider what happens when this physical input, the quality of the input gesture, guides the configuration and reaction of the system to this physical input data resulting in only one sonic voice. Here we make the distinction between the two systems by extending the level of cooperation involved to produce a single output. Our claim is that this cooperation between human and artificial agents, with our physical expressivity as input material, will encourage a greater sense of musical exploration and enrich our experience on the instrument.

REFERENCES

[1] GE. Lewis. Too Many Notes: Computers, Complexity and Culture in Voyager. MIT Press (2000), 33–39.

[2] G. Assayag, G. Bloch and M. Chemillier. Omax Brothers: a Dynamic Topology of Agents for Improvization Learning. ACM 2006

[3] W. Hsu. Using Timbre in a Computer-based Improvisation System. *Proceedings of International Computer Music Conference* (5-9), 2005.

[4] P. Cook. Principles for designing computer music controllers In Proc. New interfaces for musical expression (NIME '01) pp. 1-4(2001)

[5] S. Jordà. The Reactable: Tangible and Tabletop Music Performance. *CHI '10,* ACM (2010), 2991.

[6] A. Eigenfeldt, O. Brown, P. Pasquier and A. Martin Towards a Taxonomy of Musical Metacreation: Reflections on the First Musical Metacreation Weekend. AAAI, 2013.

[7] R. Fiebrink, D. Trueman, and P. R. Cook. A meta-instrument for interactive, on-the-fly machine learning. Proceedings of New Interfaces for Musical Expression (NIME), Pittsburgh, June 4–6, 2009. (paper)