1-1-2011

Musical Parameter Manipulation Possibilities of a Homemade Reactable

James Herrington
Edith Cowan University

Lindsay Vickery
Edith Cowan University

Herrington, J., & Vickery, L. (2011). Musical parameter manipulation possibilities of a homemade reacTable. Paper presented at the Organic Sounds in Live Electroacoustic Music: 2011 Australasian Computer Music Conference, School of Music, University of Auckland, Auckland, New Zealand.

MUSICAL PARAMETER MANIPULATION POSSIBILITIES OF A HOMEMADE REACTABLE

James Herrington
Masters by Research Student

Lindsay Vickery
Head of Composition

Western Australian Academy of Performing Arts
Edith Cowan University
2 Bradford St.
Mt. Lawley WA 6050

ABSTRACT

Musical parameter control is an important part of live interactive electronic computer music. Due to the increasing availability and affordability of music technology, including powerful computer software, advances in this area are being made to enable easier and more effective parameter control.

The purpose of this paper is to investigate and discuss the musical parameter manipulation possibilities of a homemade instrument with a tangible tabletop interface based on the technology of the reacTable. The design and construction of the instrument is documented, including the physical build as well as the software component of the system, which incorporates the computer software *ReacTIVision*, *Max/MSP* and *Reason*. The core of the paper discusses parameter manipulation abilities by way of a comparison between two controllers: the homemade instrument and the *Korg nanoKONTROL*. Mapping strategies – in an interactive music sense – are explored in detail, while the execution and capabilities of parameter control by use of the physical interface devices of the two controllers are assessed.

1. HOMEMADE REACTABLE

Using instructional information found primarily on the *reacTIVision* (*reacTIVision 1.4* nd*)* website and in the paper entitled *ReacTIVision: A computer-vision frameworks for table-based tangible interaction* (Kaltenbrunner et al. 2007), and in conjunction with original creative ideas, a homemade reacTable was designed and constructed by Masters by Research student James Herrington (see Figure 1). The instrument, with a tabletop tangible user interface, incorporates multi-touch technology, and is based on the technology of the original reacTable (Jorda et al. 2005). It can be played by a single performer, or by multiple performers.

Like the *reacTable*, this instrument incorporates a clear tabletop with a camera placed beneath, which constantly examines the table surface, tracking the nature, position and orientation of the tangibles, or objects, that are placed, and moved around, on it. The tangibles display visual symbols, called *fiducials* (see Figure 2), which are recognised by the software. Each tangible is dedicated a function for generating or manipulating/controlling a sound.

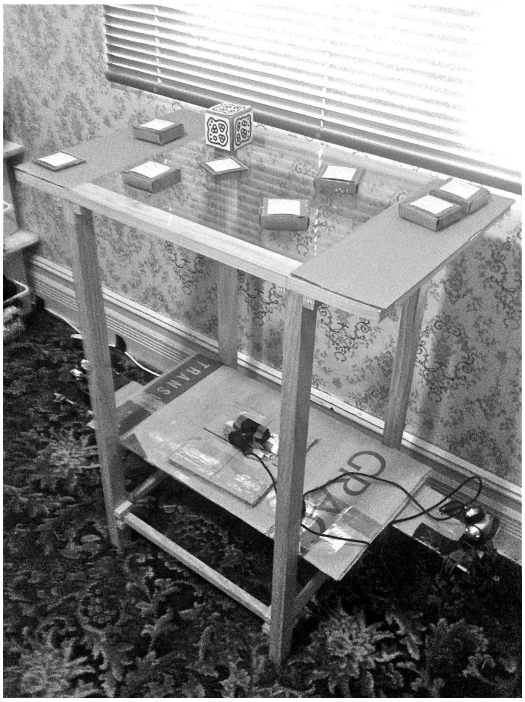

Figure 1: Homemade reacTable

Users interact by moving them around the tabletop, changing their position, their orientations, or their faces (in the case of, say, a cube object) (Jorda et al. 2005; Jorda et al. 2006).

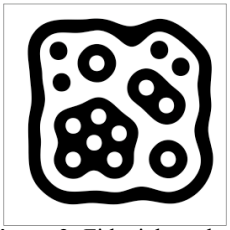

Figure 2: Fiducial symbol

This is where this instrument differs from the original *reacTable*. The vision captured by the camera is sent to the open source software *ReacTIVision*, and then to *Max/MSP*, which allows the instrument to work as a MIDI controller. This information is then sent to *Proppellerhead*'s *Reason*, where the final mapping is completed to allow note on/off events (determined by a tangible being placed and displaced in the camera's vision), along with the x-position, y-position, and orientation of each tangible assigned to manipulate different parameters of music[1].

In recent years, the availability of the previously mentioned software, and growing information on the subject, have resulted in a number of computer musicians and artists creating their own reacTables (or adaptations of them)[2]. At a tertiary level, researchers, such as the Computational Arts Research Group at the Queensland University of Technology, who developed the "Morph Table"[3] (Brown et al. 2007), are also working with the technology. While this use of the technology is utilized to develop an instrument with a set purpose of expanding certain performance and compositional techniques (i.e., morphing between musical patterns), my own homemade reacTable is set up to act as a basic MIDI controller, where the placement/displacement of any object on the table can be assigned any available note on/off function and the movement assigned to control any available parameter.

1.1 Basic physical design and build

The wooden frame structure of the homemade reacTable is based (as the name might suggest) on the shape and design of a table. The table stands 92cm high, at perfect mid-waist height. As it is intended to be performed while standing up, this gives the performer a "birds-eye" view of the tabletop, while relieving them from having to bend or sit down to move the objects around. The dimensions of the tabletop interface – clear Perspex – are 46cm (length) x 37.6cm (width). This provides the performer with quite a large area (1729.6cm^2) to move the objects around. As part of the design, on either side of the interface are two shelves (15cm x 46cm) intended for unused, or "standby", objects to rest on.

A *Sony PlayStation 3 Eye* webcam (*PlayStation Eye*. 2011) – with approx. dimensions of 84 x 67 x 57mm, and a video capture of 640 x 480 pixel – is placed 61cm directly beneath the tabletop, facing upwards in order to capture the vision of the objects being moved around. Two LED torches are placed on either side of the table, on the same level plane as the camera, but roughly 25cm to the left and right respectively of being directly underneath the tabletop interface. They are then angled to shine on the bottom side of the Perspex. This was necessary, as the visual fiducial symbols needed ample light in order to be properly recognised and sufficiently tracked when placed on the table surface. The camera is then able to constantly examine the interface, without any distracting light reflection – as the torches were strategically placed to the sides, rather than directly underneath the clear Perspex.

When it comes to the tangible objects of the homemade reacTable, they can be sorted into four categories of size. Objects can be of all shapes and sizes, and will work as long as they have an attached, and recognisable, fiducial. However, the four size groups used are as follows:

- Cube object – 7cm x 7cm x 7cm (Figure 3)
- Rectangular prism object – 7cm x 7cm x 2cm (Figure 4)
- Large flat object - 7cm x 7cm (Figure 5)
- Small flat object – 5cm x 5cm (Figure 6)

The first three listed objects – the larger objects – when placed on the tabletop interface, take up the same amount of surface area (49cm^2 – on which the fiducial is presented). The small flat object, however, takes up a lesser amount of surface area, with a smaller fiducial attached to a surface area of 25cm^2.

Figure 3: Cube object

Figure 4: Rectangular prism object

Figure 5: Large flat object

Figure 6: Small flat object

1.2 Computer software: *ReacTIVision*, *Max/MSP* and *Reason*

Regarding the computer aspect of the instrument, three software programs are used in conjunction with each other in order for vision to be captured, analysed and then interpreted into sound, or in other words, for the instrument to function. The three computer software programs, which act as the "engine room" of the instrument, are *ReacTIVision* (*reacTIVision 1.4* nd), *Max/MSP* (Cycling '74 2011) and *Reason* (Propellerhead 2011).

ReacTIVision, developed by Martin Kaltenbrunner and Ross Bencina, is the fundamental sensor

[1] Alternative software – and software combinations – can be used as opposed to *Max/MSP* and *Reason* in the set up of a homemade reacTable. This information can be found on the *reacTIVision* website.

[2] Several of these projects are documented anonymously online, including (http://www.youtube.com/watch?v=JuQo25KYELg) and (http://www.youtube.com/watch?v=15IE6d4zfME&feature=related).

[3] See (http://www.youtube.com/watch?v=nKXhfApKCms)

component of the homemade reacTable. The software is open source and can be found at the *ReacTIVision* website (*reacTIVision 1.4* nd), along with information about the internal structures and workings of the software, and instructions on how to set it up. As its function is the analysing of visual information captured by the camera placed beneath the tabletop, *ReacTIVision* does not contain any sound components. Instead, Tangible User Interface Object (TUIO) messages are sent to a TUIO-enabled client application: in the case of my instrument, this is *Max/MSP* (*reacTIVision 1.4* nd).

Max/MSP (version 5) acts as the client application in the instrument. Here, the fiducials' recognition, centre point and orientation information is processed and organised into four groups of numbers: note on/off (0 – 1), x-position (0 – 640), y-position (0 – 480) and angle (0 – 360) [The fiducials' recognition/derecognition relating to note on/off; centre point relating to x and y position; and orientation relating to angle, or rotation]. Using various techniques in *Max/MSP*, this information was organised in such a way that the zero point was located at the bottom, left hand corner of the table. For example, moving an object from left to right raises the value of the x-axis number, while moving an object from bottom to top raises the value of the y-axis number. The processing of information was also organised so that the value of the angle, or orientation, number rises when an object is rotated clockwise. These sets of numbers are then scaled to 0 – 127 in order to be sent as MIDI information to the computer software program *Reason*.

Reason (version 4) completes the process of interpreting object recognition and movement into sound generation and control. To sum up, *ReacTIVision* has analysed vision of objects and their placements, and sent this information to *Max/MSP* where it has been organised into sets of note on/off, x-position, y-position and orientation values and finally sent to *Reason*. *Reason* is where the mapping of these values to parameters of music occurs. An example would be if the y-position value of an object were assigned to the pitch shift parameter, therefore enabling the movement of this object from bottom to top of the table interface to raise the pitch of the sound produced.

2. MUSICAL PARAMETER MANIPULATION

One of the main advantages of the homemade reacTable is the number of musical parameter manipulation possibilities that can be achieved through the use of various mapping strategies. Mapping, in terms of interactive music systems, is the connection between the outputs of a gestural controller and the inputs of a sound generator. The method is typically used to link performer actions to the generation and control of musical sounds and parameters (Drummond 2009; Wanderley 2001; Winkler 1998).

2.1 Homemade reacTable mapping strategies

When it comes to the homemade reacTable, the mapping is the relationship between the movement of the tangible objects and the sounds produced. The relationships can be set up in an obvious, or not so obvious way, and ideally a balance between the two makes for a more interesting instrument. As discussed above, parameters can be mapped (or assigned) to the x-axis, y-axis and rotation of each object, while note on/off functions can be mapped to the recognition/ derecognition of objects on the table interface.

Throughout this paper, specific parameter mapping assignments of the homemade reacTable have not been discussed, other than in examples, because, discussing them in detail here (i.e., each object's function) is not important. It is only important to know that the placement of objects on and off the table surface can generate any accessible sound and activate any accessible effect, and that the x-axis, y-axis and angle movement of the object can manipulate any accessible parameter of the music.

There are four main mapping strategies that can be used in interactive music systems: *one-to-one*, which is the direct connection of an output to an input; *one-to-many*, which is the connection of a single output to multiple inputs; *many-to-one*, which is the connection of two or more outputs to control one input; and *many-to-many*, which is a combination of the different mapping types (Drummond 2009; Miranda and Wanderley 2006). This is certainly apparent in the homemade reacTable, where the output relates to one type of an object's movement (i.e., x-axis or y-axis or angle movement), and the input relates to any desired parameter of music to be manipulated.

A noteworthy feature of the objects is that, because of the way the instrument is set up – specifically the *Max/MSP* element – objects can continually rotate. That is so that the value assigned to the orientation of the object resets to 0 after a full rotation. This enables control of the parameter so that the value can switch straight from MIDI CC 127 to MIDI CC 0, or vice versa. The x-position, y-position and orientation of each object can be used to manipulate parameters of the same "parent" effect (e.g., x = reverb dry/wet amount, y = reverb decay, and angle = HF damp), or alternatively be used to manipulate completely different parameters (e.g., x = reverb dry/wet amount, y = pitch shift, and angle = panning).

2.2 Comparison

The parameter manipulation abilities of the homemade reacTable can be contrasted with the manipulation abilities of a commercial MIDI control device, the *Korg nanoKONTROL* (*Korg* 2011). The choice in comparing the *nanoKontrol* is because, as a general MIDI controller that sends MIDI information to MIDI-enabled devices, it incorporates the basic and universal note on/off functions, and also interface-controls that can access every MIDI CC value. Although new and

experimental MIDI interfaces are being developed all the time, the most common physical MIDI controllers contain pads, or keys, for note on/off functions, and knobs/faders for continuous signal control. This USB bus powered device offers nine faders, nine knobs and 18 switches, with four programmable scenes, along with a full transport section (controlling functions such as *start, stop, loop,* or *record* on the DAW software). It is a small controller with dimensions 320(W) x 82(D) x 29.5(H) mm, and a weight of 29g.

It is important to note that the assessment will only take into account the knobs and faders of the *nanoKONTROL*, which can access every MIDI CC value (0 – 127). Although they can transmit MIDI CC messages, the 18 switches on the device can only access, or trigger, two values. Attack and decay times can be assigned to the switches, and therefore access every value if set up correctly; however, this is a set function and the MIDI CC values cannot be continually controlled. When it comes to the homemade reacTable, the x, y and orientation position of each object (or more specifically, fiducial) can access every MIDI CC value.

In the comparison below, the parameter manipulating devices will be referred to as Physical Interface Controllers (PICs). In the case of the homemade reacTable, this term will refer to the tangible objects with attached fiducial symbols. In the case of the *nanoKONTROL*, this will refer to the knobs and faders of the device. The devices are compared below on four dimensions, in the form of questions relating to the execution and capabilities of parameter control. The two controllers are assessed as if performed as a solo instrument by a solo performer.

How many potential PICs can be used/assigned to manipulate parameters of the music?

Homemade reacTable:

In the setup of the homemade reacTable, the default 'amoeba' fiducial set[4] is used, which includes 216 distinct fiducials. Fiducial IDs 0 – 107 being a black image on white, while fiducial IDs 108 – 216 are the inverse, with the same images reversed (i.e., now a white image on black). Each of these fiducials can be tracked and therefore can be assigned to manipulate different parameters of music. This means that 216 potential PICs can be set up for use in the one instrument (*reacTIVision 1.4* nd).

nanoKONTROL:

The *nanoKONTROL*'s interface consists of nine knobs and nine faders with four programmable scenes. As each scenes memory allows the same settings to be retained, this results in 36 knobs and 36 faders that can be assigned to manipulate different parameters of music. Therefore, 72 potential PICs can be set up for use with the controller.

[4]Alternative sets of fiducials are also available to be used and are available from the *reacTIVision* website.

How many PICs can be used/played/controlled simultaneously?

This question requires two distinct responses. The first relates to the idea that multiple PICs can be accessed and easily moved between when interacting with the instrument, and also without the parameter assignment being changed or replaced mid-performance. The second response relates to the amount of PICs that can be controlled simultaneously under the physical limitation of the human performer.

Homemade reacTable:

Although 216 fiducials can be assigned to manipulate different parameters of music, the objects they are attached to cannot possibly fit on the tabletop interface of the homemade reacTable all at the one time.

In a practical experiment, multiple performances of the instrument using a different amount of objects were trialled, in ways utilizing alternative mapping strategies for each set of objects. Only larger objects were used, those with a surface area of 49cm^2 – which is placed on the interface. The kind of music being performed is not specified, nor the different parameters of music used in the mapping to the objects' movement, for this information is not relevant, as the experiment is to determine the *number* of objects that can be can be interacted with on the interface in a comfortable manner.

Firstly, parameters were only assigned to the objects' x-axis and orientation (i.e., no parameters were assigned to the y-axis of each object). Set up this way, the objects were lined up from top to bottom so that the full x-axis range of values could be realised (i.e., each object could be moved from the left most point of the table [MIDI CC 0] to the right most point of the table [MIDI CC 127]) without clashing with each other. Using this mapping strategy, four objects appeared to be an adequate number. Taking into account the size of the table and the size of the objects, there was comfortable room between the objects so they could all be rotated without interference while on the same x-axis point.

Secondly, the parameter-to-object mapping was set in the same way as mentioned above; however, only assigning parameters to the y-axes and orientations of the objects (rather than the x-axes and orientations). Once again, four objects was found to also be an adequate number. That is to say, when lining the objects from left to right, each object could access every y-axis value without clashing with other objects, and in fact more room – or empty space – was available between the objects. This is because the width of the tabletop interface is longer than the height.

Next, parameters were only assigned to the objects' orientation. Set up this way, the objects were lined up from bottom to top as well as from left to right in a grid fashion, with enough space between them so that the full rotation range of values could be accessed without clashing with each other. Using this strategy, 16

objects were found to be a satisfactory amount of objects on the interface. That is to say, the interface accommodated four rows of four objects.

Finally, parameters of music were assigned to the objects' x-axis, y-axis and orientation. With this mapping strategy employed, it is hard to give an exact number of how many objects allows for satisfactory performance. In a way, it depends on what type of piece is being performed. For example, an experimental free improvisation piece, where objects are moved around at free will, would accommodate more objects than a piece where one or two objects have to be moved at certain times to certain positions without interrupting the placement of other objects. If the piece is in fact structured in this way, the performer would need to pre-determine which objects need to move along entire x and y-axes, and arrange the objects accordingly when first placing them on the table. In saying this, nine objects (three rows of three) were found to be an adequate number where objects can move around freely enough.

When it comes to the number of PICs that can be used simultaneously (i.e., how many on the interface at the one time), it really depends on a variety of factors, such as the mapping strategies employed. Another factor, not discussed thus far, is the size of the objects. The examples above utilized objects with a surface area of 49cm^2. As mentioned, smaller sized objects can be used, which would enable more simultaneous PICs on the interface at the one time. Combinations of larger and smaller objects can also be used. It is also worth noting that when the performer needs to change or replace certain PICs, and thus the parameters to be manipulated, one object at a time can be replaced on the interface.

The second response to the original question involves the human performer's limitations in the physical controlling of the objects. Traditionally, or rather ideally, two objects can be controlled simultaneously by a solo human performer, that is, one in each hand[5]. We say 'ideally' because more than two objects *can* physically be controlled, however, when doing so, restrictions arise. For example, the performer could use his or her nose or teeth, like a modern day Jimi Hendrix, to control a third object, however – when compared to controlling objects with one's hands – this can hardly be achieved efficiently, as it would be awkward. A second example occurs when the performer moves two or more objects with the one hand. Once again, to avoid being extremely awkward, this can only take place if the two or more objects are to be executing the same control, that is, moving the multiple objects up, down, left or right simultaneously. Rotating the objects with the one hand would be difficult without also altering the x and y positions of the object.

[5]Multiple performers result in more objects being able to be controlled simultaneously. For example, two performers on the one instrument can control four objects, three can control six objects, and four can control eight objects, with each performer controlling the standard two objects. More performers can be added, but because of the size of the instrument, however, the space would become more and more cramped when consisting of more than four players. The square design of the instrument also neatly accommodates four players, with one performer on each side.

nanoKONTROL:

As realised in the previous question, 72 PICs can be set up from nine knobs and nine faders on the physical interface of the *nanoKONTROL*, however, only these nine knobs and nine faders (i.e., 18 PICs) can be used without altering the scene, and thus replacing the assigned parameter set of one scene with a completely different set of another scene. The act of switching between scenes directly opposes the idea of being able to easily access and move between parameters to manipulate. For example, it would not be possible to manipulate the parameter assigned to *knob 1* of the first scene and the parameter assigned to *knob 2* of the second scene simultaneously. This applies to any two separate parameters assigned to PICs on contradicting scenes. Therefore, only the 18 (physical) PICs can be controlled "simultaneously" when relating to the first response of the original question. Unlike the homemade reacTable, where one PIC can be changed or replaced at a time, the *nanoKONTROL* can only move between scenes, and therefore 18 PICs (or rather the parameters assigned to them) can only be changed or replaced by 18 PICs at a time.

As considered with the homemade reacTable, to answer the second part of the question involves discussing the human performance confinements in the physical controlling of the – in this case – knobs and faders. Like the homemade reacTable, the ideal number of faders and/or knobs to be controlled simultaneously is two – one with each hand[6]. The same "slight exceptions" examples also relate. That is to say, a performer *could* use his or her nose or teeth (or any other part of the body) to control a PIC, however, it would be awkward; while two or more faders (not so much knobs) can be controlled simultaneously, with enough amount of efficiency, with the one hand, although, only when performing the same control – in this case, being moved up or down.

How many parameters of music can be independently manipulated using the one PIC?

Homemade reacTable:

When it comes to the homemade reacTable, three parameters can be independently controlled using the one PIC. That is to say, a parameter of music each can

[6]Once again, multiple performers result in more faders and knobs being able to be controlled simultaneously, with each performer controlling two PICs. However, as opposed to the homemade reacTable, at a much smaller physical size, it would be extremely cramped with any more than two performers.

be assigned to the movement of the object on the x-axis of the tabletop interface, the movement on the y-axis, and the angle – or orientation – of the object. Each parameter can be manipulated separately. For example, an object can be moved from left to right on the table, controlling the parameter assigned to the x-axis, while maintaining the values, or settings, of the parameters assigned to the y-axis and angle of the object.

Alternatively, the parameters can be manipulated simultaneously at an independent rate. For example, moving the object in an oval-shaped manner, while continuously rotating the object. Each parameter is being manipulated at a different rate, while changes in the shape of the movement add to the independent parameter manipulation possibilities. Furthermore, changing the rotating speed of the object will affect the manipulation rate of the parameter assigned to the angle of the object independently of (i.e., without affecting) the parameters assigned to the x and y-axes[7].

nanoKONTROL:

Only one parameter of music can be independently controlled using only one PIC of the *nanoKONTROL*. Multiple parameters can be assigned to the one fader (or knob alternatively), for example, however, when the fader is controlled (i.e., raised and lowered), the value of each parameter is manipulated at the same rate.

Using various external software, one can alter the nature of how each of the parameters assigned to the one PIC is manipulated. Using a fader once again as an example, the minimum and maximum values can be reversed, so as the fader is physically raised, the value of the parameter is lowered. The minimum and maximum vales can also be restricted to a certain range, so as a fader is raised from the minimum position to maximum position on the physical instrument, the MIDI CC value of the parameter would raise, for example, from 10 – 80 (or whatever range the user has assigned) as opposed to 0 – 127[8]. However, even if both of these examples were allocated to two different parameters assigned to the one PIC, along with another parameter being manipulated in the traditional sense (minimum position to maximum position on the physical interface equates to 0 – 127 parameter value), the three parameters would all be manipulated at the same exponential rate when controlled by the one fader, that is, not independently of one another[9].

Are there any placement restrictions of the PICs?

Homemade reacTable:

Objects are placed and moved around the tabletop interface of the instrument in order to achieve their assigned parameter manipulation functions. In saying this, it is not physically possible for two objects to be in the same xy position on the table surface. Depending on the parameter assignments of each object, this means that certain combinations of audio effects are unachievable. Because of this limitation, a good mapping strategy technique would be to assign music parameters to the movement of the objects' orientation and only *one* of their axes (x or y). This is because an object can be rotated on the table surface without affecting its xy location, and therefore, without clashing with other objects.

Experimenting with the larger objects ($49cm^2$) on the homemade reacTable, it was found that two objects with the same y-position, and side-by-side as close as possible to the same x-position, could not access any values within MIDI CC 20 of each other on the x-axis. Two objects with the same x-position, as close as possible to the same y-position could not access values within MIDI CC 28 on the y-axis. Once again, this is due to the fact that the width of the interface is longer than the height. Using the smaller objects ($25cm^2$), two with the same y-position, could not access within MIDI CC 15 on the x-axis, while two with the same x-position could not access within MIDI CC 20 on the y-axis[10].

nanoKONTROL:

Unlike the homemade reacTable, where parameters are manipulated by moving each object around the one surface plane, the knobs and faders of the *nanoKONTROL* are allocated their own space. Because of this, there are no placement limitations of the PICs.

An example of the difference between the two controllers under assessment (the homemade reacTable and the *nanoKONTROL*) would be as follows:

[Using only x and y possibilities, and ignoring the rotation parameter control function for now] *Object 1* of the homemade reacTable controls the reverberation of the entire sound through the use of a reverb unit in *Reason*. The parameter assigned to the movement of the x-axis is the amount of decay, and the parameter

[7]Multiple fiducials can be placed on the one object, such as a larger object with four fiducials displayed on the one face, and played in a way where one hand is controlling the object. This raises the amount of multiple parameters controlled by one PIC, however, in doing so disregards the idea of independent parameter manipulation between the parameters assigned to the two or more fiducials on the one object. That is to say, however the object is moved around the tabletop interface, all fiducials on the object move in the same way.

[8]Using the *Korg Kontrol Editor* (*nanoKONTROL* 2011), one can edit settings on the *nanoKONTROL* itself (i.e., without using external software) to modify how each PIC manipulates its assigned parameters (as discussed above), although, three alternative ways cannot be designated to the one PIC using this method. That method can only be achieved by using external software, such as *Ableton Live* (*Ableton* 2010).

[9]*Korg* has released a controller, different to the *nanoKONTROL*, called the *nanoPAD* (*Korg* 2011) with an xy pad (where x and y parameters can be manipulated independently) controlled by finger touch. In this case, the user's finger would act as the PIC.

[10]The larger the interface surface area is, the smaller these number values would become

	Homemade reacTable	Korg nanoKONTROL
No. of potential PICs	216	72
No. of PICs that can be controlled simultaneously	*In the same space without being changed/replaced*: Larger Objects (49cm^2 surface area): X and ANG controlling parameters: 4 Y and ANG controlling parameters: 4 ANG controlling parameters: 16 X, Y and ANG controlling parameters: 9 *Under human limitations*: 2	*In the same space without being changed/replaced*: 18: 9 faders, 9 knobs on 1 scene *Under human limitations*: 2
No. of parameters that can be independently manipulated using one PIC	3	1
Placement restrictions	2 objects can not be in the same xy position on the interface	No restrictions

Table 1: Comparison of homemade reacTable and *Korg nanoKONTROL* across 4 dimensions

assigned to the movement of the y-axis is the dry/wet amount. Meanwhile, *object 2* controls the equalisation of the entire sound through the use of a Parametric EQ unit. The parameter assigned to the movement of the x-axis is the frequency value, and the parameter assigned to the movement of the y-axis is the gain value. If the performer wishes to achieve the audio effect of a reverberation with a decay amount of MIDI CC 50 and a dry/wet amount of MIDI CC 30, simultaneously with the equalisation emphasising the frequency at MIDI CC 50 at a gain of MIDI CC 30, this unfortunately would not be possible. This is because, in the case of both object placements, x = 50 and y = 30, and two objects cannot be in the same xy position. To overcome this problem, various mapping strategies can be employed, as discussed previously. One approach would be to switch the assigned x and y parameters of one of the objects, resulting in one object requiring x = 50 and y = 30, and the second object requiring x = 30 and y = 50 to achieve the desired audio combination effect. This method would not work, however, if all four parameters (the parameters assigned to x and y movement of both objects) required the same MIDI CC value, that is, if both objects required the values x = 50 and y = 50.

In the case of the *nanoKONTROL*, using the same parameters and intended values, the audio effect combination *can* be achieved. Using four faders (or knobs alternatively) *fader 1*, assigned to the reverb decay amount, can achieve the MIDI CC value of 50; *fader 2*, assigned to the reverb dry/wet amount can achieve the MIDI CC value of 30; *fader 3*, assigned to the frequency of the EQ, can achieve the MIDI CC value of 50; while *fader 4*, assigned to the gain of the EQ, can achieve the MIDI CC value of 30. These values can be achieved simultaneously, unlike on the homemade reacTable, however, two additional PICs must be used.

The potential of each controller in relation to the four questions is summarised in Table 1 above.

2.2.1 Other things to consider

The above assessment has compared the parameter manipulation abilities of two controllers – the homemade reacTable and the *Korg nanoKONTROL*. Although, if the two were to be compared as the better overall, or more useful, controller, various other aspects would need to be considered. This may include the lag or delay between the physical movement of a PIC and the assigned parameter value. The CPU usage of each would also need to be assessed, as would the restrictions due to size, and the ease of portability.

3. FUTURE WORK

Future work on parameter manipulation utilizing this type of technology could include setting up the homemade reacTable instrument in a way so that the distance between two objects acts as another value that can be calculated and assigned to an additional musical parameter to be controlled. This way, both objects would have to be present on the interface in order for the parameter to be altered. If three parameters each were additionally mapped to the x-axis, y-axis and orientation of the two objects, this would enable the performer to control seven parameters of music independently of each other using only two objects. For many pieces of music, this would be all the control the performer needs.

Although the lag and delay due to the quality of the webcam was not discussed throughout the paper, future work may also involve the use of a High Definition camera to track the fiducials more effectively.

For his own artistic endeavors, James Herrington intends to further this research to develop alternative ways to perform and compose contemporary electronic music, and use these extensive manipulation

possibilities to advance the homemade reacTable as a DJ instrument. He is also currently working with the instrument as a component in an integrated Dubstep Performance environment for his Masters by Research project.

4. CONCLUSION

Advances in music technology in conjunction with increasing accessibility and affordability have contributed to progress in the area of musical parameter manipulation in live electronic computer music. This is made apparent by comparing a homemade electronic instrument with a commercial controller, and showing that in many ways the homemade instrument possesses superior parameter manipulation abilities. Apart from the computer itself, the sum of components of the homemade instrument – including open source, and relatively inexpensive software – come at a reasonable price. The PICs of the homemade reacTable as tangible, freely moving objects generate the novel characteristics, and enable the excellent range of parameter manipulation, of the instrument. A greater scope of versatility and control of the musical output is produced, that, by comparison, can be restrictive when it comes to other electronic devices and controllers. By no means have all possibilities in parameter manipulation utilizing this technology been explored, which leaves the door open for further investigation and exciting advances.

5. REFERENCES

Ableton. 2010. [cited March 23 2011]. Available from http://www.ableton.com/.

Brown, A. R, R. W. Wooler, and K. Thomas. 2007. "The Morph Table: A collaborative interface for musical interaction." Paper read at *Australian Computer Music Conference* at Canberra, Australia.

Drummond, J. 2009. "Understanding interactive systems." *Organised Sound* 14 (2):124-133.

Jorda, S, M Kaltenbrunner, G Geiger, and M Alonso. 2006. "The reacTable: A tangible tabletop musical instrument and collaborative workbench."

Jorda, S, M Kaltenbrunner, G Geiger, and R Bencina. 2005. "The reactable*." Paper read at *International Computer Music Conference*, at Barcelona, Spain.

Kaltenbrunner, M. 2009. "ReacTIVision and TUIO: A tangible tabletop toolkit." Paper read at *ACM International Confeence on interactive tabletops and surfaces* at Banff, Alberta, Canada.

Kaltenbrunner, M., and R. Bencina. 2007. "ReacTIVision: A computer-vision frameworks for table-based tangible interaction." In *Proceedings of the First International Conference of Tangible and Embedded Interaction (TEI'07)*. Baton Rouge, LA.

Korg: nanoSERIES: Slimline USB MIDI Controllers 2011. [cited March 21 2011]. Available from http://korg.com/Product.aspx?pd=415.

Max MSP 5. Cycling '74. 2011. [cited March 20 2011]. Available from http://cycling74.com/

Miranda, E. R., and M. M. Wanderley. 2006. *New digital musical instruments: Control and interaction beyond the keyboard*. Vol. 21, *The Computer Music and Digital Audio Series*. Middleton, WI: A-R Editions.

nanoKONTROL: Slim-Line USB Controller: Owner's Manual. 2008. [cited March 21 2011]. Available from http://www.korg.co.uk/downloads/nano/support/nanoKONTROL_OM_E2.pdf.

PlayStation Eye. 2011. [cited March 19 2011]. Available from http://au.playstation.com/ps3/peripherals/detail/item78696/PlayStation-Eye/.

reacTIVision 1.4: a toolkit for tangible multi-touch surfaces. nd. [cited 15 March 2011]. Available from http://reactivision.sourceforge.net/.

Reason 4. Propellerhead. 2011. [cited March 20 2011]. Available from http://www.propellerheads.se/

Wanderley, M. 2001. "Gestural control of music." Paper read at *International Workshop - Human Supervision and Control in Enginerring and Music*, at Kassel, Germany.

Winkler, T. 1998. *Composing interactive music: Techniques and ideas using Max*. Cambridge, MA: The MIT Press.