PITCHCIRCLE3D: A CASE STUDY IN LIVE NOTATION FOR INTERACTIVE MUSIC PERFORMANCE

Tom Hall
Anglia Ruskin University
tom.hall@anglia.ac.uk

ABSTRACT

Recent decades have seen the establishment of computer software live notations intended as music scores, affording new modes of interaction between composers, improvisers, performers and audience. This paper presents a live notations project situated within the research domains of algorithmic music composition, improvisation, performance and software interaction design. The software enables the presentation of live animated scores which display 2D and 3D pitch-space representations of note collections including a spiral helix and pitch-class clock. The software has been specifically engineered within an existing sound synthesis environment, SuperCollider, to produce tight integration between sound synthesis and live notation. In a performance context, the live notation is usually presented as both music score and visualisation to the performers and audience respectively. The case study considers the performances of two of the author's contrasting compositions utilising the software. The results thus far from the project demonstrate the ways in which the software can afford different models of algorithmic and improvised interaction between the composer, performers and the music itself. Also included is a summary of feedback from musicians who have used the software in public music performances over a number of years.

Keywords: notation, interaction, algorithms, music performance, improvisation, software, SuperCollider

1. INTRODUCTION

Musical notation can have various functions, including as a mnemonic, an analysis or transcription. The most common function is prescriptive, in the form of 'instructions' for performers [1, p.100]. Such instructions in musical scores are by convention *symbolic representations* of the musical elements to be sounded, which are then interpreted by performers as actions to be undertaken to perform the required sounds. In the early 1960s Cornelius Cardew saw notation in terms of a hierarchy of rules, lamenting that interpretation of Western classical music relied on many of these rules being implicit [2]. This learned implicit (and indeed often embodied) knowledge can be considered as notationally contextual information, for example stylistic and/or historical performance practice and so on.

Live notation and digital scores in most cases are created using computer technologies and offer advantages for some kinds of music over 'analogue' (paper) scores. Reliability is a key advantage of paper scores over digital notation (technology often breaks in concerts). Paper scores are also convenient for annotation (musicians bring pencils to rehearsals), however, once printed, are otherwise notationally fixed. On the other hand, live, dynamic notation can enable a 'just-in-time' notational approach, allowing notations digitally assembled in realtime. This is useful for improvised or algorithmic music, in which the notation is able to reflect algorithmic procedures or composer live mediation, for example. Computer technology also has the advantage in making *sharing* notations with an audience trivial via projection, of which more below.

2. TYPES OF DYNAMIC DIGITAL SCORES

The notation of a 'musical score' is often synonymous with Common Western Notation (*CWN*), a highly evolved and efficient method of indicating musical intention within the Western music tradition [3]. However aside from notations involved in non-Western music and those of early Western music, there is now a century of 'non-standard', often experimental scores usually known as graphic scores. The degree to which non-standard notation employs elements or conventions from CWN varies considerably. Whilst on the one hand graphic scores often require lengthy textural performance directions, on the other they may also rely on a performer's more general ability to read 'iconic depictions' of various sorts [1, p.130]. Mortan Feldman's 'graph paper' scores of the 1950s, for instance, can be understood within a long tradition of human culture around 'grids' [4]. Of course, different approaches to graphic music notation offer design trade-offs in terms of representation of musical parameters, as discussed further below.

Live digital notation for music goes under various names: 'realtime-score' [5], dynamic digital scores, 'screen scores' and others. Vickery correctly draws a historical connection here to traditions within experimental 20th Century scores, including the 'mobile scores' of Earle Brown [6]; in the computer music domain, experiments by Max Mathews at Bell Labs in onscreen musical notation by computer are also relevant here. The number of approaches available to screen scores are clearly numerous. Vickery categorises two main types, distinguishing between scrolling and 'segmented scores' on the one hand, and 'realtime scores' (permutative, generative and transformative) on the other [p.131]. Whilst this is a useful start, it could be argued that these are ultimately overlapping categories and live notations may often as not observe such distinctions in the breach as otherwise. This is implicit in the following discussion, which introduces the software used in this project, before contextualising its use in performance projects.

3. PITCHCIRCLE3D

PitchCircle3D is series of custom classes written in the SuperCollider programming language [7]. SuperCollider is an established and sophisticated interactive programming environment for realtime computer music synthesis. Whilst PitchCircle3D can simply be used to visualise note music, a stronger motivation for its development is to enable the sharing of a form of non-specialist music notation with performers and audience alike. Elsewhere I argue that this is broadly inline with a post-war desire towards transparency of communication in art [8]. PitchCircle3D is implemented within SuperCollider as a 'system within a system' [9]. A motivation for this is to allow, to use Leman's term, 'micro-integration' [10, p.3] with the SuperCollider's audio synthesis engine. This in turn enables responsive electronic music performance with very little latency.

PitchCircle3D uses SuperCollider's cross-platform GUI environment (implemented in Qt) to display live, animated non-standard music notation in the form of notes and chords in 12-tone equal temperament (12-TET), as shown in Fig. 1. The notation view is animated at a customisable frame rate which can be updated in realtime within the PitchCircle3D class via SuperCollider's interactive programming environment. The PitchCircle3D class knows nothing of SuperCollider's audio synthesis, use of the class is usually incorporated into individual SuperCollider code as required for each composition. The allows PitchCircle3D to be used in conjunction with most of the many coding styles available to within the SuperCollider language, sclang. Whilst all the music discussed in this paper has been coded in SuperCollider, it should be noted that PitchCircle3D also can be easily configured within SuperCollider to respond to external control through eg MIDI or OSC messages. In the case of the latter, this is discussed further below.

PitchCircle3D currently has three notational views available. A 3D spiral helix illustrates relative register, shown over three octaves in Fig. 1. A 'pitch clock' shows pitch-classes, omitting registral information (see Fig. 2, which shows a pitch-class set view of the pitches in Fig. 1). A third view flattens the view to a 2D spiral, as seen from above and shown in Fig. 3. These different views are relatively trivial to achieve within the class as they are derived from matrix operations on an identity matrix. [1]

Animation of PitchCircle3D's views includes code methods to smoothly tilt, rotate and zoom views, programmatically or by mouse interaction (the former are achieved using 'easing' functions). In each view, small discs represent potentially sounding pitches, by default connected by a line passing through each. Ordinarily the notation indicates corresponding pitch-classes with a separate colour for each pitch/pitch-class (the colour scheme is customisable). Pitch-class or note names are also indicated in each view, allowing the role of note colours to represent other musical parameters if desired (eg dynamics). An additional small circle around a disc is available (shown in Fig. 1 on the lowest note C) to indicate a point of focus or emphasis, for instance a tonic or area of pitch centricity.

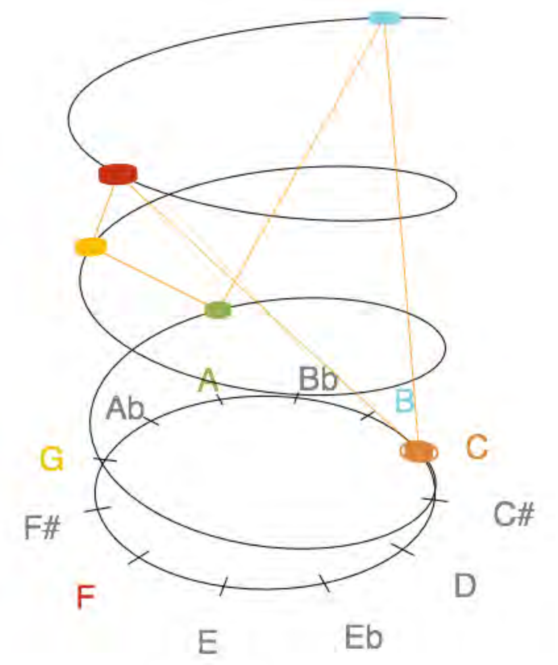

Figure 1. PitchCircle3D spiral notation view.

Notes can be entered and removed individually or in groups, faded in and out at a desired rate (in seconds) and displayed for a specified duration, starting either immediately or at some future point in time. These operations create a series of 'time points' to structure musical progression according to predefined or algorithmic sequences. As a small and simple example, the following SuperCollider code can be used to begin an instance of the PitchCircle3D class within SuperCollider and fade in, over two seconds, a number of notes (as MIDI numbers) to the view. These are connected by default via a line, in the oder listed in the array. The rotate and tilt methods can be used to then generate the view shown in Fig. 1 (these shifts can also be animated over time, something which will be discussed further presently). To clear the view and close it, the methods shown below are used.

```
p = PitchCircle3D.new(numOcts:3);
p.front;
p.addComplex([60, 95, 69, 79, 89], 60, 2);
p.rotateTo(9.5, 1); // rotate over 1 second
p.tiltTo(5.9);
p.clearAll; // remove discs immediately
p.close; // close window
```

[1] This part of the class is based on the SuperCollider Canvas3D quark by Jonatan Liljedahl and Fredrik Olofsson. In the PitchCircle3D classes, however, mathematical transformations are decoupled from SuperCollider's Pen drawing class.

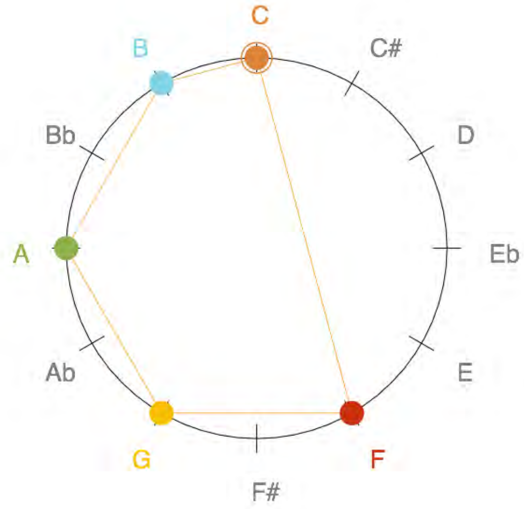

Figure 2. PitchCircle3D clock notation view.

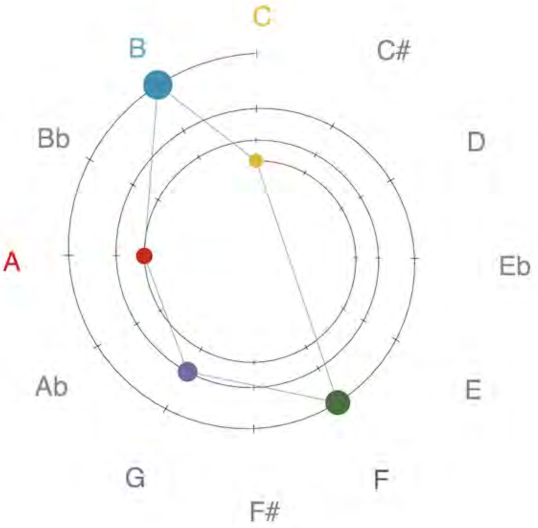

Figure 3. PitchCircle3D 2D spiral notation view.

The notational representations available in PitchCircle3D are in themselves not novel, although their particular implementation and the software's performative context offers affordances. Related software include iPhone apps *Music Set Theory* [11], which presents an interactive pitch-class clock view for the classification of set-class names and the display of their complements according to Alan Forte's naming system. The closest relation to PitchCircle3D is perhaps Chew and François' software MuSA.RT, which also displays pitches around an animated 3D spiral helix, and can do so using live MIDI input [12]. [2] Whilst it shares many of the notational concerns of PitchCircle3D, it focuses on illustrating a specific theory of the analysis of tonal music. It thus appears that MuSA.RT may be categorised more as a visualisation tool for live musical analysis rather than as software for displaying live notation as a digital musical score for performance.

4. CASE STUDIES: PITCHCIRCLE3D IN PERFORMANCE

4.1 *All the Chords*: interaction and improvisation

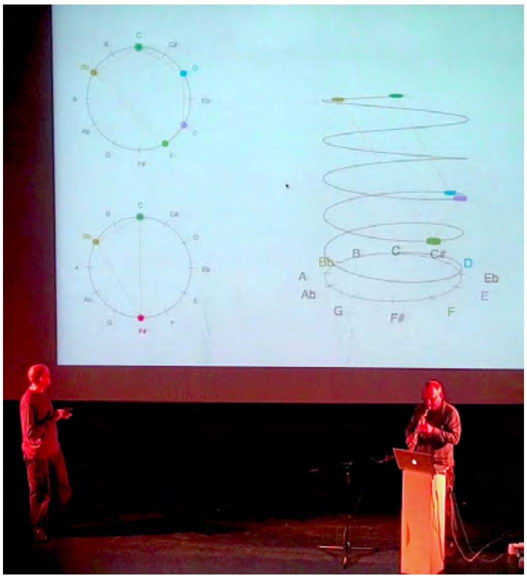

Figure 4. *All the Chords* on-stage configuration of performers, computers and notation.

In 2014 PitchCircle3D was used in a configuration for performance of the author's composition *All the Chords*, involving an instrumental musician, Kevin Flanagan (saxophone), and computer performer (the composer). In performance, PitchCircle3D was used in full-screen mode, and a 'mirrored' screen projected on the rear of the stage. The instrumental performer mediated aspects of the performance, viewing the notation on the screen of the laptop running SuperCollider, whilst the author as computer performer referred to the notation on the rear projection, as shown in Fig. 4. SuperCollider was used to both display the live notation using PitchCircle3D, as well as to synthesise a computer music part. This computer music part comprises a drone, an occasional bassline and pulse emphasised by a synthesised percussive layer. The main musical material in *All the Chords*, however, is a predetermined sequence of collections of notes ('chords'), all the possible subsets of a superimposed major and parallel harmonic minor scale algorithmically ordered according to SuperCollider's powerset method.

To aid sight-reading by the instrumental performer, during the performance, two collections are displayed. The current collection of notes are shown in spiral helix form and also in pitch-clock on the top left of the screen (see Fig. 4). On the bottom left of the screen, the *upcoming* collection of notes is shown, allowing the performer(s) to prepare as necessary for the next note collection of the composition before it arrives.

[2] PitchCircle3D can respond both to external MIDI and Open Sound Control (OSC) messages via core SuperCollider capability.

The computer-performer communicates with the main computer laptop via the Open Sound Control (OSC) protocol. Rather than achieving this via a second laptop, communication with SuperCollider is via a mobile device in order to enable direct on-stage communication with both the instrumental performer *and* appear most present to the audience. To achieve this, the mobile device runs a customised layout of the TouchOSC app [13], and amongst other things allows interaction with the stratified compositional layers of electronic musical elements (chords, drones, arpeggiations, bass notes and overall mix volume). There is also a button to toggle between manual advancement of chord collection frames and a chordal 'autopilot' setting, using a predetermined algorithmic sequence of time intervals specified within the SuperCollider code.

As a representation of musical events, PitchCircle3D's notations combine aspects of both indeterminate and fully determined events. The software's default spiral helix view is fully determinate in terms of its notation of pitch-space, whilst its clock view, as shown in Fig. 2, presents pitch-class space only, necessarily omitting registral information (this figure represents the same pitches as those in Fig. 1). Whilst it might be expected that the choice between displayed views would depend on the level of pitch determinism required, experience in performance has found a friction in reading the spiral helix view quickly and without error (see performer feedback section below). Thus even where pitch is fully determined, in *All the Chords* the spiral helix view is presented simultaneously with the corresponding clock view. This configuration can be seen on the top left hand side of Fig. 4.

The rhythm and tempo of musical events are notationally unspecified in *All the Chords*. This notational constraint within the current PitchCircle3D model of musical representation, however, encourages certain approaches to rhythm and temporality which have been exploited in the music written for the system by the author. In *All the Chords*, musical rhythm (in the sense of sequence) is represented at a higher level in the timing of display transitions ('time points'). Within these time points, rhythms are freely improvised by the instrumental performer around the note collections displayed. In this way, the notes shown are also freely interpreted as material for melodic improvisation, since ordering of each collection is not indicated. Musical continuities are created in the piece by linking these collections via common-tones across time points.

As can be seen in Fig. 4, lines are drawn between notes in each collection, in order to emphasis interval relationships. (This feature works most clearly in PitchCircle3D's pitch-class clock view and is turned off in *Untitled #1*, the composition discussed below, as it exclusively used the spiral helix view.) The pitch-class of the current drone sounding in the electronic part is indicated via the ring around relevant 'note' in the displayed notation. Dynamics for the instrumental performer are notationally unspecified in *Untitled #1*.

In this overall approach to musical temporality, the design of PitchCircle3D sits well with established models of musical improvisation [14]. The following very brief summary introduces the key ideas: Pressing's model divides improvised music into sequential 'event clusters' divided by time points, usually demarcated by 'local musical boundary criteria' including pauses and other phrase junctures [14, p.153]. Musical continuation within and between clusters is determined by 'associative' or 'interrupt' generation across musical parameters [14, p.155].

Overall the project sits between other recent approaches in digital notation that are more indeterminate (graphic notation), or fully determinate—for example employing CWN (such as [15]). Design decisions behind PitchCircle3D offer clear constraints for performance (what notes to play), but leave others relatively open (when and how to play), a mode of performance well documented since at least the 1960s [16]. This notational indeterminacy of PitchCircle3D can be regarded as an affordance, leaving as it does considerable room for collaborative musical improvisation, as noted above.

In the 2014 performance using the software, temporal constraints of the musical improvisation were partially determined by the duration over which each collection was displayed. As discussed above, the duration was in turn determined either algorithmically, or through mediation by the computer musician. Likewise, timings of musically noted material and the relationships between this material thus influenced whether event cluster continuations were associative or interrupt-driven.

These musical decisions and outcomes were the result of the collaborative nature of the musical improvisation, the music of which reflected the musical interests of both parties. In particular Flanagan's improvised jazz vernacular was a clear feature of the performance, constrained as it was by the harmonic material presented to him, yet hopefully still affording the 'intrinsically explorative nature' of improvisation [17, p.53]. Fig. 5 illustrates the feedback between the two performers, the digital notation, and the sounding musical performance (influenced by Nash and Blackwell's approach to diagramming user interaction within music software [18]). Note that the majority of these interactions function as iterative feedback loops which may operate on multiple timescales in relation to the Pressing's model of improvisation presented above.

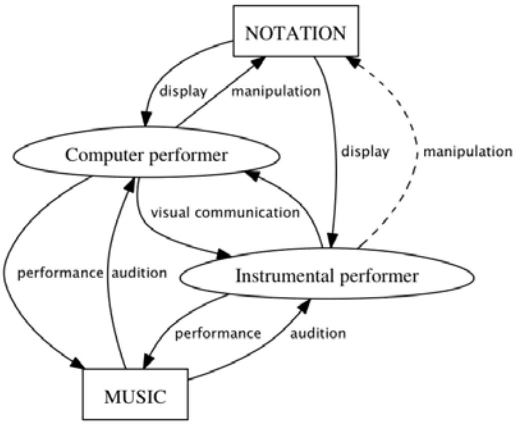

Figure 5. Interactions between performers.

4.2 *Untitled #1*: algorithmic processes

Untitled #1 (2015) is a 15 minute composition for computer performer with computer sound, and two amplified instruments played monophonically. The 2015 first performance at Anglia Ruskin University (2015-04-17) paired electronic guitar with E-bow, performed by Will Crosby, and amplified 'cello sounding only natural and artificial harmonics, performed by Cheryl Frances-Hoad. In this performance the author as composer-computer performer stayed off-stage, using a MIDI controller keyboard attached to a laptop. The electronic music part of the composition was coded in SuperCollider and used PitchCircle3D to display the live notation via a projector. The two instrumental performers were on either side of the stage, forming a triangle with the live score, as is shown in Fig. 6.

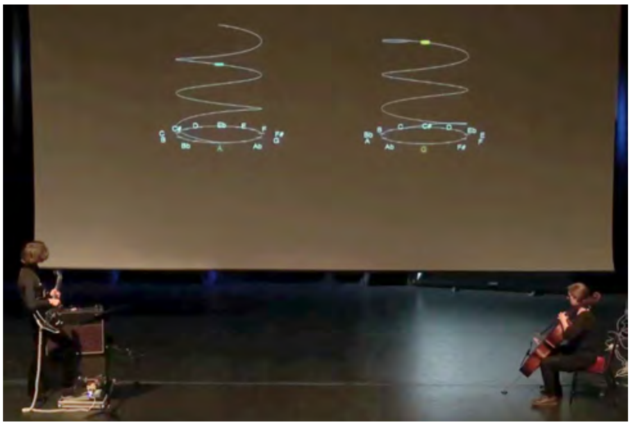

Figure 6. Stage layout for Hall's *Untitled #1 (2015)*.

As should be clear from Fig. 6, the live notation is laid out as two musical parts, one for each performer. This was deemed necessary as overlapping instrumental entrances and exits would have made reading both parts from a single spiral helix overly difficult. Dividing the screen real estate into two in this way made a separate 'look ahead' event cluster frame as used in *All the Chords* impractical. However this constraint was offset by the very slow rate of musical change in this composition, enabling performer cognition of the next note to be played.

Each instrumental part is monophonic, enabling the playing of artificial harmonics in the 'cello part and the use of the E-bow to achieve infinite sustain on the electric guitar (E-bows can only play a single note at a time). The onset and duration of notes is indicated notationally in two connected ways in in *Untitled #1*, which also indicates relative dynamics within notes to the performer. For each part, all notes slowly fade in and then fade out. This is indicated by the changing transparency in real time of the displayed note. The helix is rotated such that when the note displays full colour / amplitude, the horizontal alignment of the helix displays the note at the front of the 3D view.

The musical structure of *Untitled #1* takes particular advantage of the live notation enabled by PitchCircle3D. The algorithmic nature of the composition centres on a relatively simple series of harmonic progression rules which determine in real time the sequence of harmonies within an overall simple modal harmonic framework. These sequences of harmonies occurr either in the live performer parts, the electronic accompaniment, or both.

		Performance	
		Every Chord	*Untitled #1*
Frame	Sequence	determined	algorithmic
	Timing	determined	algorithmic
Intra-frame	Sequence	improvised	determined
	Timing	improvised	determined

Table 1. Algorithmic and improvised musical structures afforded by live notation

Unlike *All the Chords*, in the newer composition a fixed electronic pitched drone establishes a dorian musical centricity throughout. Further differences between the approaches to notation and structure between the two compositions are discussed in the next section.

5. NOTATIONAL STRUCTURES AND INTERACTIONS USING PITCHCIRCLE3D

The foregoing has discussed approaches to using the PitchCircle3D software for the performance of two compositions employing live notation in two different and distinct musical ways. These approaches to musical structures and musical performance has required rethinking the role of the software as live notation for each composition.

Table 1 presents a high level summary of these differences and shows the switch in determinate (fixed) and indeterminate (algorithmic and improved) musical elements in the structural levels of the two compositions under discussion. Here the notion of a 'frame' is one delineated by time points, and 'intra-frame' that of possible event clusters within each animated view (for instance, improvisation on a note collection). At a lower level, relative dynamics ('hairpin' fades) are represented in one of the compositions, but not in the other.

Just as algorithmic music in general challenges the notion of the fixed work, live notation further blurs lines between the composition of a work and its performance, opening up possibilities for structured improvisation between performers coordinated by composer mediation or algorithmic control. Fig. 7 shows the role of live notation in this situation, in which brackets show optional characteristics within compositional and performance processes.

6. PERFORMER EXPERIENCES OF USING PITCHCIRCLE3D IN PERFORMANCE

Four musicians who have performed using PitchCircle3D in public responded to a short written questionnaire about their experience using the software. All had had some previous experience performing using non-standard notation. Each performer had also had some experience with algorithmic and/or semi-improvised music and there were a number of interesting reflections on this. For instance performer A commented on 'serendipitous moments that would have

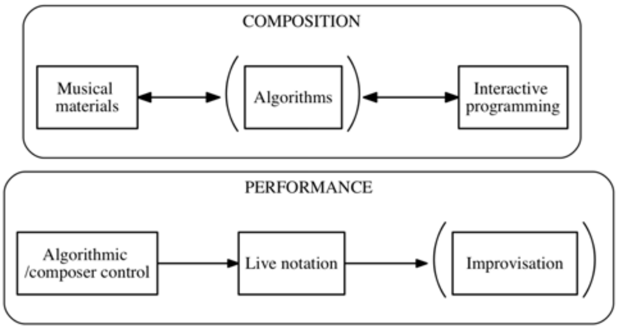

Figure 7. Role of live notation in composition and performance.

never occurred [using other methods]', while performer B claimed that there 'tends to be unspoken stylistic assumptions with each piece' which an improviser must position herself with respect to.

The performers were also asked about the general experience of performing with PitchCircle3D. Performer A stated that, 'once you get your head around it, it feels very intuitive', however commented that tracking note changes required intense concentration at times—being on 'high alert'. Similarly performer B experienced the demands of live notation as enjoyable and viewed this in terms of a challenge around musical (improvisational) inventiveness which 'added to the spontaneity of the performance'. Performer C also considered the novelty of interacting with the live notation 'logical' yet challenging, and experienced this in terms of embodied responses (the physicality and instinct required in performance). Performer D also considered the physicality suggested by the notation, relating the polygons shapes of the 2D form in terms of hand shapes for a keyboard 'guided improvisation'.

Performers were asked about their ability to distinguish between aspects of the notation in each of the two PitchCircle3D views, the 3D spiral helix and the 2D clock. Performers A and C reported that identifying the indicated note correctly was helped by the individual note colours, and the slow rotation which centred the upcoming note to the front of the view. For performers B and D correct identification was reported as more difficult, however it should be noted that the rate of change and number of notes in those performances was much greater. There were mixed responses as to the ease of identifying the correct octave of a note in the spiral helix view.

In some performances, as noted above, performers were presented with both 2D and 3D views of the current note collection for performance, as well as a 2D representation of the upcoming collection. There was a varied response from all four performers as to the usefulness of both of these features as experienced or imagined. Performer D made the observation that the efficacy of these features would depend on on the time available to use them in performance, i.e. the overall level of musical activity. Likewise, performer B commented on this in relation to the potentially different kind of continuities that might be achieved or desired, depending on whether a performer was able to 'read ahead' from the upcoming collection view.

7. CONCLUSION

Writing in 1961, Cornelius Cardew suggested that 'notation should put the player on the right road' [2, p.31]. This metaphor for moving in time in a defined direction sits well with the aims of PitchCircle3D, which has proven to be a flexible tool for displaying and sharing live notation in different musical contexts. PitchCircle3D's implementation in SuperCollider allows tight integration with audio synthesis, and the resulting realtime capabilities have affordances for live algorithmic computer music in tandem with live instrumental performance.

Ongoing software development of PitchCircle3D forms part of a reflective shared practice-led project between the software's author and instrumental performers who use PitchCircle3D in performance. Investigation of the effectiveness of this environment for both flexible and specialist means of communication and sharing between performers and audience forms part of this research context. The results thus far have demonstrated a number of models of performative and compositional interaction as outlined in the case study above. Questionnaires conducted with musicians who have used the software have demonstrated generally positive results, with, however, a common experience being that its use in performance can make cognitively high demands on the musicians.

Future work will aim to reduce some of the friction in the reading of this notation by the performing musicians, as well as explore further options for collaboration between performers. In particular, it might be interesting to involve performers in reciprocal musical interaction with the software—including the ability of the instrumental performer to influence the live notation and possibly the electronic music if desired. Such developments could also be judged against audience reaction analysis as well as further feedback from performers.

The most obvious inbuilt notational constraint within the PitchCircle3D model is that its notations currently provide no rhythmic information except through the realtime temporality of time points structuring a performance. Whilst such indeterminacy might be regarded as an inbuilt limitation of the system as a performance notation, the approach taken maps well onto existing practices of musical improvisation as discussed above. Nevertheless, future research will leverage further possibilities for musical parameter representation within PitchCircle3D. This is intended to enable more varied musical improvisation around flexible live musical structures.

Acknowledgments

Thanks to the performers Will Crosby, Kevin Flanagan, Cheryl Frances-Hoad and Richard Hoadley for permission to include their questionnaire responses in this paper. Thanks also to Alan F. Blackwell, who contributed to an earlier stage of this research, and to the three anonymous reviewers for their helpful suggestions.

8. REFERENCES

[1] S. Davies, *Musical Works and Performances / A Philosophical Exploration*. Oxford: Oxford University Press, 2001.

[2] C. Cardew, "Notation, Interpretation, etc," *Tempo*, vol. 58, pp. 21–33, Summer 1961.

[3] R. Rastall, *The Notation of Western Music / An Introduction*. London: J.M. Dent, 1983.

[4] H. B. Higgins, *The Grid Book*. Cambridge, MA: MIT Press, 2009.

[5] G. E. Winkler, "The Realtime-Score. A Missing-Link in Computer-Music Performance," in *Sound and Music Computing '04*. Paris: IRCAM, October 2004.

[6] L. Vickery, "The Evolution of Notational Innovations from Mobile Score to Screen Score," *Organised Sound*, vol. 17, no. 2, pp. 128–136, 2012.

[7] J. McCartney, "Rethinking the Computer Music Language: SuperCollider," *Compuer Music Journal*, vol. 26, no. 4, pp. 61–68, 2002.

[8] T. Hall, "Sharing Electronic Music Performance," in *Visualise: Making Art in Context. Cambridge*, B. Ferran, Ed. Anglia Ruskin University, 2013, pp. 60–63.

[9] J. Rohrhuber, T. Hall, and A. de Campo, "Dialects, Constraints and Systems within Systems," in *The SuperCollider Book*, N. Collins, D. M. Cottle, and S. Wilson, Eds. Cambridge, Mass.: MIT Press, 2011.

[10] M. Leman, *Embodied Music Cognition and Mediation Technology*. Cambridge, Mass.: MIT Press, 2007.

[11] P. Couprie, "Music set theory," http://logiciels.pierrecouprie.fr/.

[12] Chew, E., and François, A. R. J., "Interactive Multi-Scale Visualizations of Tonal Evolution in MuSA.RT Opus 2," *ACM Computers in Entertainment*, vol. 3, no. 4, p. 16, 2005.

[13] Hexler, "Touchosc," http://hexler.net/software/touchosc.

[14] J. Pressing, "Improvisation: methods and models," in *Generative Processes In Music / The Psychology of Performance, Improvisiation and Compostiion*, J. A. Sloboda, Ed. Oxford: Clarendon Press, 1988.

[15] R. Hoadley, "Calder's Violin: real-time notation and performance through musically expressive algorithms," in *Proceedings of the International Computer Music Conference*, IRZU, Ljubljana, Slovenia, September 9-15 2012, pp. 188–193.

[16] D. Behrman, "What Indeterminate Notation Determines," *Perspectives of New Music*, vol. 3, no. 2, pp. 58–73, 1965.

[17] D. Bailey, *Improvisation / Its Nature and Methods*. Boston, Massachusetts: Da Capo Press, 1992.

[18] Nash, C. and Blackwell, A. F., "Liveness and Flow in Notation Use," in *Proceedings of New Interfaces for Musical Expression NIME*, University of Michigan Ann Arbor, MA, 2012, pp. 28–33.