Performing Sequences: Interaction and Instrumentality in the Design of a Performable Sequencing System

Roberto Chiurazzi
chiurazzirob@gmail.com
Centre for Digital Music
Queen Mary University of London
London, United Kingdom

Anna Xambó
a.xambosedo@qmul.ac.uk
Centre for Digital Music
Queen Mary University of London
London, United Kingdom

Charalampos Saitis
c.saitis@qmul.ac.uk
Centre for Digital Music
Queen Mary University of London
London, United Kingdom

Abstract

Sequencers play a central role in electronic music practice, yet are rarely treated as instruments capable of supporting real-time interaction with sufficient creative flexibility for improvisation. This paper investigates how a sequencing system can be designed as a performable instrument. We present a digital MIDI sequencer implemented in Pure Data, based on Euclidean rhythms and extended through direction-based melodic traversal, rhythmic-melodic coupling, and constrained modulation. Two improvised compositions performed by the first author provide a practice-based evaluation of the system's performability in live use. The findings suggest that low-dimensional, rule-based control and clear gesture-outcome relationships can support interaction-driven improvisation while preserving performer agency and authorship. This work contributes to the NIME community by offering design strategies and practice-based evaluation insights for treating sequencers as performance instruments.

Keywords

Algorithmic Sequencing, Interactive Music Systems, Human-Machine Interaction, Euclidean Rhythms, Direction-based Pitch Traversal

1 Introduction

Sequencing remains a foundational yet underexamined aspect of music production and performance [14]. From early analogue step sequencers embedded in synthesisers to contemporary digital sequencers integrated into DAWs and software instruments, sequencing structures have shaped the rhythmic and textural character of much electronic music [1]. Despite this central role, sequencers tend to sit in an ambiguous position around discussions of instrumentality, as they are typically framed as tools for controlling sound production rather than as instruments in their own right. In performance contexts, this framing obscures the importance of interaction and gesture translation: the ways performers shape rhythm, melody, and musical form through sequencing processes, even when no sound is generated directly [2, 15].

This paper explores the design of sequencers as performable instruments, understood as systems that foreground real-time interaction by translating performer gestures into evolving musical structures. Framing sequencing in this way raises two research questions:

(1) How might sequencers be designed as performable instruments that support interaction-driven improvisation in live contexts?
(2) What interaction affordances emerge when sequencing is approached as a performable system?

To address these questions, we developed a sequencing system designed explicitly for real-time performative interaction. The system is evaluated through two improvised compositions by the first author as a practice-based, first-person reflection on performability in live contexts. The design of the system is informed by research on instrumentality, gesture, and human-machine agency in digital musical devices.

2 Related Work

2.1 Algorithmic Sequencing and Interactive Music Systems

Algorithmic sequencing can be situated within broader traditions of algorithmic composition, in which musical structure is produced through formalisable, rule-based procedures rather than explicit event specification. Nierhaus [12] characterises such systems not in terms of autonomy, but as extensions of human musical intention mediated by computational logic. In performance contexts, this perspective aligns with Rowe's [15] notion of interactive music systems, where musical behaviour emerges through real-time feedback between performer input and system response.

A formative precedent for this design approach is Laurie Spiegel's Music Mouse [17], which frames algorithmic structure as a means of supporting intuitive musical expression. Spiegel argues that well-designed rules can shift attention away from note-level construction toward higher-level musical qualities such as gesture, density, direction, and form [17]. Performance in Music Mouse consists of navigating a constrained space of musically coherent possibilities, establishing a collaborative relationship between human intention and machine-mediated structure. This conception of algorithmic systems as enabling frameworks informs the present work's use of Euclidean rhythm generation [18] as a structured yet flexible approach to temporal organisation, designed to remain legible and controllable under real-time interaction.

2.2 Instrumentality: Gesture, Agency, Authorship

Instrumentality is increasingly understood as a relational and emergent property rather than a fixed attribute of an object [3, 9]. Within this view, instrumentality arises through situated use and through the perceived legibility between performer action and musical outcome, reflecting a broader shift from instruments as entities to instrumental qualities constructed through interaction [3, 7].

NIME '26, June 23–26, 2026, London, UK

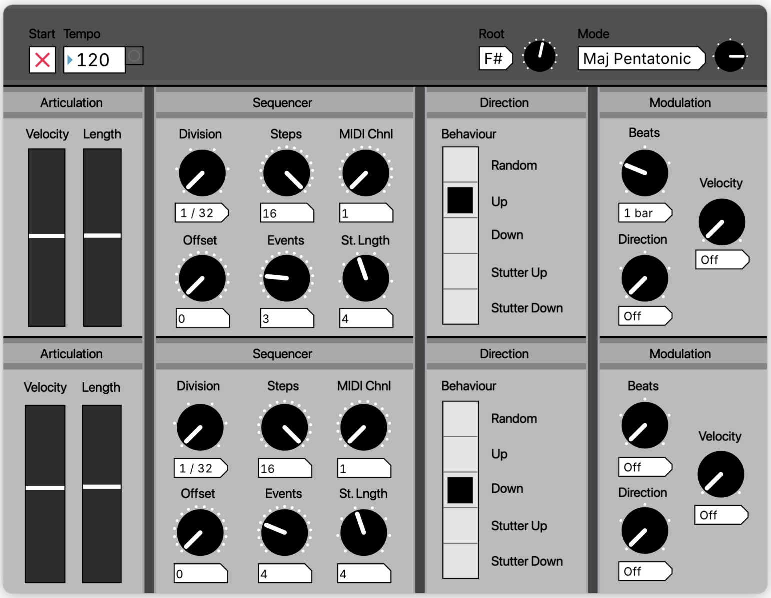

Figure 1: The Sequencer's GUI in Pure Data.

Gesture is central to this process, as instrumental interaction depends on a tight coupling between physical action and perceptual feedback, forming a continuous action-response loop that allows performers and audiences to infer causality in real time [5, 9, 19]. In electronic performance systems, the clarity of this gesture-feedback relationship is often a key condition for perceived liveness and instrumentality [4]. In the present work, gesture operates at two levels: physically, through a MIDI controller, and conceptually, through the sequencing system's low-dimensional parameter abstractions. Low-dimensional here refers to control spaces with few parameters whose effects are concrete and immediately interpretable in performance, such as melodic direction, rhythmic density, or temporal offset, exposing musical behaviour in a form that can be shaped responsively.

This supports an agency model in which performers retain primary control, while algorithmic processes function as constrained, responsive structures that participate in musical decision-making without assuming autonomy [21]. Such a positioning contrasts with many contemporary generative systems, particularly machine-learning and AI based approaches, which often prioritise autonomous musical output and risk displacing performer authorship [6, 8, 13]. By emphasising constrained modulation and low-dimensional control, the proposed sequencing instrument instead supports appropriation, learning, and stylistic differentiation over time, aligning authorship with interaction and performance practice rather than with automated generation [11, 20].

3 Design Goals

Building on the perspectives outlined in Section 2, the system was developed around three design goals:

Gestural legibility. Parameter changes should produce immediate, musically interpretable outcomes, making gesture-outcome causality perceptible in performance. This goal follows gesture as feedback-coupled action [19] and instrumentality as something constructed through perceived causal control [7].

User creative agency and low-dimensional abstraction. The system should support improvisation by allowing rhythmic and melodic behaviour to be shaped through a small set of process-level controls. This draws on low-dimensional control as a basis for appropriation [11] and on interactive responsiveness as a condition for performer-led musical direction [15].

Authorship. Algorithmic processes should function as constrained, rule-based structures whose behaviour remains interpretable in performance, avoiding autonomous generation and preserving the performer's ownership of musical outcomes. This follows accounts of authorship in generative art [20] and agency models in which control is distributed and selectively delegated [6, 21].

4 Development

The proposed system is a fully digital sequencing instrument developed in Pure Data that generates MIDI note events (note number, velocity, and length) in real time (Figure 1).[1] The instrument is designed to be instantiated multiple times, with each instance functioning as a monophonic voice and realised through a modular internal architecture (Figure 2). Musical complexity emerges through the layering of multiple instances rather than through an increase in per-instance control complexity. Each instance consists of:

- A time division selector module, with a master tempo shared between all instances;
- A Euclidean rhythm generator module;
- A melodic traversal engine module operating on a pitch set quantised to a scale and mode;
- A module to set the direction-based melodic behaviour of the sequence;
- Modulation modules for number of events, melodic direction and note velocity.

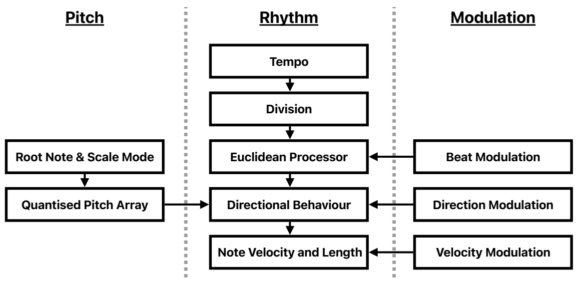

Figure 2: System diagram for a single instance of the sequencer.

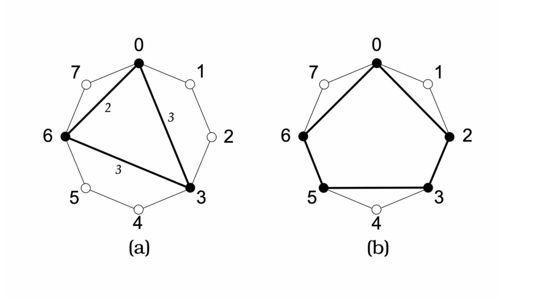

Figure 3: Two examples of Euclidean Rhythms distribution: 3 (a) and 5 (b) onsets over 8 steps.

4.1 Euclidean Rhythmic Core

At the core of each instance is a Euclidean rhythm generator defined by two parameters: steps (n) and events (k). Following Toussaint's formulation [18], events are distributed as evenly as possible across the step sequence, producing maximally even rhythmic patterns (Figure 3). The initial implementation was informed by an existing practical demonstration of Euclidean rhythm generation.[2]

An offset parameter is also implemented to shift the phase of the generated series of beats, affording an intuitive way to create musical variation in a performance context.

4.2 Direction-Based Melodic Traversal and Rhythmic Coupling

Melodic output is generated through direction-based traversal behaviours applied to a quantised pitch collection created by the user selecting scale and mode. Selected modes generate quantised pitch sets that are written into a local array (`$0-pitches`), forming the instrument's scale matrix. This was done through the ELSE library's `[else/scales -pitch]` object.[3] Each traversal behaviour defines how the system navigates the pitch set over time, enabling melodic contour to emerge from abstract movement strategies.

Typical behaviours include ascending and descending motion, randomised traversal, and repeated or stuttering movement, with a specifiable stutter length parameter controlling how many times a note is repeated from the pitch set. These behaviours are treated as performable gestures rather than static pattern selections, allowing the performer to shape melodic evolution through discrete, legible actions.

Melodic traversal is tightly coupled to rhythmic density: the number of events (k) directly determines the way the traversal progresses through the pitch set. From a performance perspective, a single control gesture can therefore affect both timing and melodic motion, with the aim to increase expressive range without increasing cognitive complexity.

4.3 Constrained Modulation and Structural Change

To support longer-term variation while maintaining interpretability, the system includes a constrained modulation framework. Modulation targets include rhythmic density, traversal behaviour, and note velocity, with parameter updates applied at user-defined n-bar intervals so that structural change occurs at predictable temporal boundaries. For example, the number of events (k) can be set to update every two bars, with bar length defined by the number of steps (n). Parameter constraints are applied specifically to event-number modulation, limiting the range of rhythmic densities to prevent abrupt interruptions or excessively dense patterns that could compromise interpretability in performance. These bounds maintain user agency and authorship while introducing controlled variation: modulation remains optional and can be engaged or withdrawn at any moment, allowing the performer to determine when processes are delegated to the system.

4.4 Multi-Instancing Performance

While each instance functions as a monophonic sequencer, the instrument is intended to be used through multiple simultaneous instances. Layering instances enables polyphony, polyrhythms, and emergent complexity allowing performers to construct richer musical textures by shaping relationships between parallel processes.

5 Compositions

Two improvised compositions were performed by the first author to address the research questions through a first-person, practice-based evaluation.[4] Interaction was mediated via a MIDI controller, with faders and knobs mapped to key system parameters to enable continuous, gestural control (Figure 4). In both performances, the sequencer generated MIDI output routed to software instruments hosted in a digital audio workstation. Performance intentions and setup details were documented in accompanying journal entries (see Appendix).

The first improvisation focused on the system's core design goal of gestural legibility. Only two sequencer instances were employed, with all generative features disabled (such as modulation parameters) in order to examine performability through direct gesture–outcome relationships. This setup foregrounded the Euclidean rhythmic core and direction-based melodic traversal, allowing close observation of how rhythmic and melodic structure responds to real-time interaction.

The second improvisation expanded the system configuration to simulate a more realistic live performance scenario. Four sequencer instances were configured in parallel, each controlling a separate software instrument. In contrast to the first improvisation, modulation features and relationships between instances were actively employed, enabling the system to operate as a layered, interactive ensemble. This performance was designed

[1]https://codeberg.org/Anonymus_Submission/Performing_Sequences.git
[2]https://www.youtube.com/watch?v=lCcGeVXHkbE
[3]https://github.com/porres/pd-else
[4]https://youtu.be/b1zX5pjd0qU

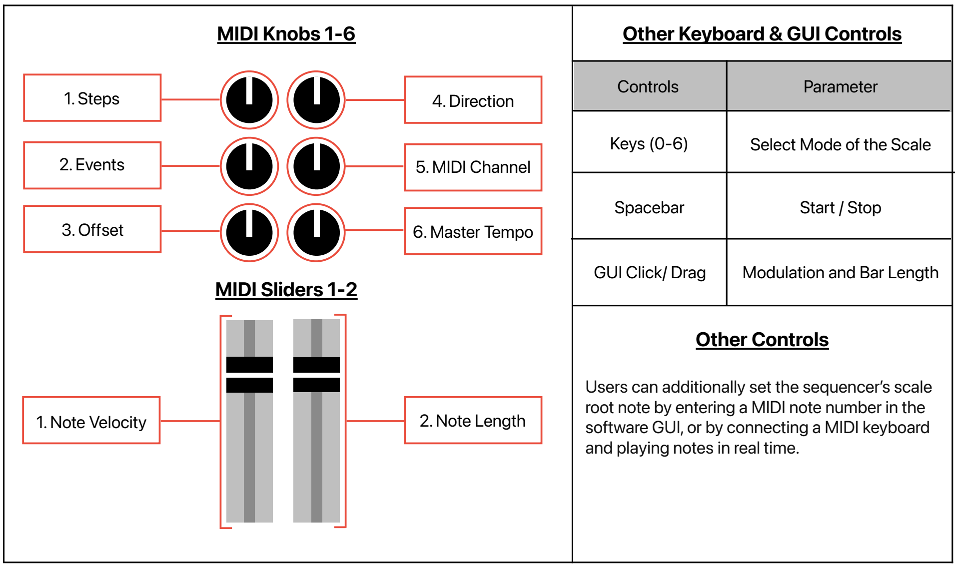

Figure 4: MIDI controller mapping employed for the recording of the improvised compositions.

to investigate the interaction affordances that emerge when sequencing is treated as a performable system, thereby addressing the paper's second research question.

6 Reflections

In the first improvisation, the constrained setup (two instances, no modulation) showed how gestural legibility was consistently achieved: parameter changes produced immediate, musically interpretable outcomes, supporting the design goal of transparent gesture-outcome relationships [4]. This also produced a strong perception of creative agency and authorship, since algorithmic processes primarily structured timing and traversal behaviour rather than generating musical material autonomously, aligning with performer-centred interactive music system models [15]. However, the limited number of instances occasionally reduced the perceived breadth of the musical space, despite remaining sufficient to demonstrate performability.

The second improvisation expanded the system to four instances and enabled modulation. This configuration supported polyphony and more complex arrangements, directly addressing the second research question by revealing interaction affordances at a system level: dynamic layering, polyrhythmic relationships, and distributed control across instances. At times, the density of concurrent rhythmic layers increased perceptual load and made real-time coordination more demanding, consistent with research showing that complex rhythmic structures can challenge temporal entrainment and attentional focus [10]. In these moments, constrained modulation helped manage complexity by allowing the performer to temporarily delegate limited aspects of variation to the system while maintaining overall, human-dominant control, consistent with agency taxonomies in live performance systems [21].

Overall, these findings indicate that the system achieved instrumentality by maintaining a clear relationship between physical gesture and musical outcome, allowing sequencing to function as a site of direct improvisation. When modulation is deactivated, the performer can shift between active manipulation and stable looping by engaging or withdrawing gestural input, with the sustained material corresponding to the last state prior to disengagement. This reveals an additional real-time dimension of agency, exercised not only through active manipulation but also through the intentional absence of gesture.

At the same time, certain limitations emerged. In the melodic domain, the available set of directional behaviours, while designed for legibility and control, occasionally constrained melodic diversity despite the flexibility afforded by parameters such as stutter length and the dynamic interplay with the rhythmic dimension. Expanding traversal strategies or introducing additional low-dimensional contour controls could extend the system's melodic expressive range.

More broadly, these findings support the value of human-centred algorithmic design at a time when many machine-learning-based music systems emphasise autonomous generation with minimal direction from users [8, 16]. While such approaches can be musically effective, their abstraction can reduce perceived authorship and control [13].

7 Conclusions and Future Work

This paper presented a digital MIDI sequencing instrument implemented in Pure Data, designed as a performable system. Built on Euclidean rhythm generation and extended through direction-based melodic traversal, rhythmic–melodic coupling, and constrained modulation, the instrument was developed around goals of gestural legibility, performer-centred agency, and preserved authorship. Two improvised performances provided a practice-based evaluation, suggesting that low-dimensional, rule-based interaction can support improvisation by maintaining clear gesture-outcome relationships while allowing musical structure to emerge through real-time manipulation. At the same time, the study highlighted how perceptual load can increase with the number of concurrent instances, even when constrained modulation already mitigates this by allowing selective delegation of variation to the system.

Future work could address this limitation through two alternative directions. One approach would be distributing instances across multiple performers, reducing individual cognitive demand while opening the system to ensemble-based performance practices. Another would be introducing inter-instance interaction mechanisms governed by moderately higher-level parameters, allowing a single performer to shape relationships between voices through single gestures and strengthening system-level control while preserving performer-centred agency. These directions aim to extend the present design study by further investigating how sequencing instruments can continue to support agency and authorship through performable, human-centred interaction.

8 Ethical Standards

This research involved the first author's own creative practice and did not involve human participants beyond the author. No external funding was received for this work, and there are no conflicts of interest to declare.

References

[1] Raphael Arar and Ajay Kapur. 2013. A history of sequencers: Interfaces for organizing pattern-based music. In *Proceedings of the 10th Sound and Music Computing Conference (SMC 2013)*, Roberto Bresin (Ed.). Stockholm, Sweden, 383–388.

[2] Mark J. Butler. 2014. *Playing with Something That Runs: Technology, Improvisation, and Composition in DJ and Laptop Performance.* Oxford University Press, New York.

[3] Caroline Cance, Hugues Genevois, and Danièle Dubois. 2010. What is instrumentality in new digital musical devices? A contribution from cognitive linguistics and psychology. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2010).*

[4] John Croft. 2007. Theses on liveness. *Organised Sound* 12, 1 (2007), 59–66.

[5] Gina Emerson and Hauke Egermann. 2018. Gesture-sound causality from the audience's perspective: Investigating the influence of mapping perceptibility on the aesthetic perception of new digital musical instruments. *Psychology of Aesthetics, Creativity, and the Arts* 12, 1 (2018), 96–109.

[6] Toby Gifford, Shelly Knotts, Jon McCormack, Stefano Kalonaris, Matthew Yee-King, and Mark d'Inverno. 2018. Computational systems for music improvisation. *Computer Music Journal* 42, 4 (2018), 8–25.

[7] Sarah-Indriyati Hardjowirogo. 2017. Instrumentality: On the construction of instrumental identity. In *Musical Instruments in the 21st Century*, Till Bovermann, Alberto de Campo, Hauke Egermann, Sarah-Indriyati Hardjowirogo, and Stefan Weinzierl (Eds.). Springer, Singapore, 13–27.

[8] Samuel J. Hunt, Thomas J. Mitchell, and Chris Nash. 2020. Composing computer generated music, an observational study using IGME: The Interactive Generative Music Environment. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2020).*

[9] Alexander Refsum Jensenius. 2022. *Sound Actions: Conceptualizing Musical Instruments.* MIT Press, Cambridge, MA.

[10] Justin London. 2012. *Hearing in Time: Psychological Aspects of Musical Meter* (2 ed.). Oxford University Press, New York.

[11] Andrew McPherson and Victor Zappi. 2018. Dimensionality and appropriation in digital musical instrument design. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2018).*

[12] Gerhard Nierhaus. 2009. *Algorithmic Composition: Paradigms of Automated Music Generation.* Springer, Vienna.

[13] Ashley Noel-Hirst, Charalampos Saitis, and Nick Bryan-Kinns. 2025. Sampling the latent space: Exploring the creative potential of generative AI through the lens of sample-based music making. In *Proceedings of the 6th Conference on AI Music Creativity (AIMC 2025).* Brussels, Belgium.

[14] Stefan Püst, Lena Gieseke, and Angela Brennecke. 2021. Interaction taxonomy for sequencer-based music performances. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2021).*

[15] Robert Rowe. 1993. *Interactive Music Systems: Machine Listening and Composing.* MIT Press, Cambridge, MA.

[16] Jordie Shier, Rodrigo Constanzo, Charalampos Saitis, Andrew Robertson, and Andrew McPherson. 2025. Designing percussive timbre remappings: Negotiating audio representations and evolving parameter spaces. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2025).* Canberra, Australia.

[17] Laurie Spiegel. 1993. *Music Mouse: An Intelligent Instrument.* Software and manual. Original Macintosh version 1986; Amiga and Atari ST versions 1986–1988.

[18] Godfried T. Toussaint. 2005. The Euclidean algorithm generates traditional musical rhythms. In *Proceedings of BRIDGES: Mathematical Connections in Art, Music, and Science.* 47–56.

[19] Marcelo M. Wanderley. 2000. Gesture and musical expression. In *Trends in Gestural Control of Music*, Marcelo M. Wanderley and Marc Battier (Eds.). IRCAM / Centre Pompidou, Paris, 15–46.

[20] Adrian Ward and Geoff Cox. 1999. How I drew one of my pictures: or, The authorship of generative art. https://generativeart.com/on/cic/99/0399.htm.

[21] Anna Xambó and Gerard Roma. 2021. Human-machine agencies in live coding for music performance. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2021).*

A Appendix

A.1 Journal: Composition 1

The sequencer was set up to send MIDI from Pure Data to Logic Pro X through a MIDI router, with each of the sequencer's instances sent to a separate software instrument. For this first improvisation, the first author adopted a deliberately minimal configuration, using only two instances with plucked, tonal sounds in order to foreground the instrument's rhythmic and melodic behaviour. This constrained setup allowed the evaluation to focus on the sequencer's compositional and gestural qualities, rather than on timbral design or performance factors external to MIDI generation. During the performance, the generated MIDI output was recorded directly into Logic Pro X. The improvisation was informed aesthetically by the work of composers such as Raymond Scott, Steve Reich, and Carl Stone, particularly in its emphasis on repetitive structures, gradual variation, and emergent rhythmic interplay. Their work provided a stylistic frame against which to test whether the system could sustain that kind of slow additive transformation under real-time control. Working with only two voices foregrounded the relationship between the Euclidean rhythmic core and the melodic traversal behaviours, allowing the musical form to develop through small parameter changes and their immediate consequences in the evolving sequence. Small changes, such as shifting offset on one instance or switching traversal direction on the other, produced clearly perceptible consequences, and the musical form could be felt developing through these incremental gestures, with the system's behaviour shaping the outcome alongside the performer's choices.

A.2 Journal: Composition 2

The second improvisation used the same MIDI routing setup but expanded the system to four parallel sequencer instances, treating the instrument less as a controlled test environment and more as a performable ensemble. While still working primarily with plucked sounds, subtle timbral differences were introduced between instances to support perceptual distinction, allowing each voice to be recognisable within the layered texture. This facilitated a clearer perception of how individual gestures affected each instance in real time. The aesthetic built on the influences explored in the first composition, but adopted a more contemporary and artistically nuanced direction shaped by the first author's own practice as a performer. In this context, modulation and inter-instance relationships were used more actively to sustain longer-form variation, and to explore how the instrument supports improvisation through continuous, real-time interaction. The expanded setup confirmed that the system scales, but it also exposed limits not encountered with two instances. Coordinating four voices in real time was demanding, and there were moments where variation had to be delegated to the modulation system simply to free attention for higher-level decisions. This delegation worked well: it allowed the performer to step back, listen, and re-enter at structurally meaningful moments rather than constantly micro-managing every voice.