

Creating a Web Based 3D Sequencer in order to Investigate its Effects on Composition

Milo Beuzeval
University of Sussex

Chris Kiefer
University of Sussex

ABSTRACT

We present a 3D music sequencer environment¹ within Sema[4]. It allows for sequencing in terms of interacting "cylinders" and "pegs". Sequences are made within the 3D graphical environment and triggers are sent to a live-coding window when pegs collide together. We discuss the motivations behind the design and evaluate its effect on composition through a user study. We found that users enjoyed the environment we created and that its interface provides a novel perspective on sequencing composition of music.

KEYWORDS

Music Sequencer, 3D, Live-Coding

1 INTRODUCTION

The traditional music sequencer is based around a linear sequence of steps. When a track is played to the end it loops back to the start. Users place down notes and sample triggers on steps to compose a sequence of music. Functionality is based on predictability and a fixed path. Famous examples of this design are the Roland TR-808 drum machine[6], which help to inspire many genres particularly dance focused music. The step based design that lends itself towards the creation of repetitive beats, an influence stylistically on early House, Hip-Hop and Techno music.

Much of what makes this traditional step based design great can also be its limitation. Steps are limited by number, usually multiples of four and constrained to 16 or 32 steps. This rigidity in rhythm removes some of the possibilities for dense shuffling polyrhythms that are difficult or impossible to program within a step based system. However the step based design does make it easy to learn through experimentation as it becomes quickly obvious how sounds are programmed through placing notes and hearing them loop.

We implement a sequencer that challenges the standard design and tries to re-imagine a sequencer in 3D space. Instead of a step based timeline with note and sample triggers our system replaces this with 3D shapes (cylinders and pegs), within a sandbox environment.

Instead of sequencing music on a linear timeline limited to a common tempo, there is freedom to create cylindrical timelines of any size and segmentation. One might create a cylinder with 100 faces and another with 5. Speed of rotation is individual to each cylinder and so sequences may playback at different speeds. This creates an opportunity to experiment with time and space in a way that is not possible in a traditional sequencer.

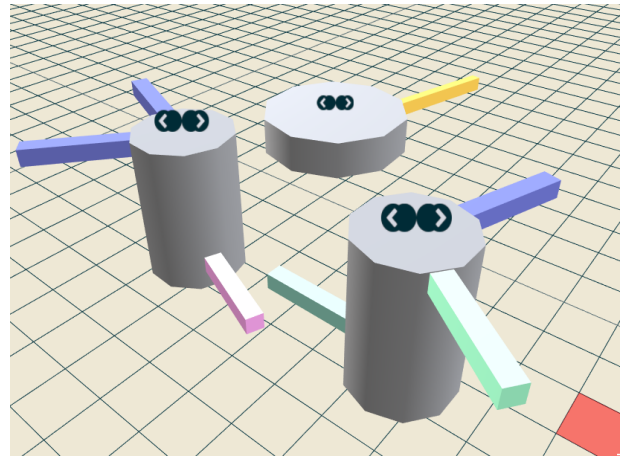


Figure 1: Screenshot from our 3D Sequencing environment

2 BACKGROUND

Within mechanical musical instruments there is many examples of sequencing in a 3D environment, from the virtue that our physical world is three dimensional. However examples of this such as mechanical music box simply use the rotating cylinder as a mechanical way of achieving a loop. Hardware sequencers are not limited in the same way and instead unravel the cylindrical face into a flat timeline of steps, to loop the sequence jumping from the end to the beginning.

Much of digital audio workstations and digital music software in general are re-implementations of popular hardware interfaces, including software music sequencers. Software instruments are not limited in the same way as its acoustic and studio counterparts, so investigating new designs for sequencing in the software space is worthwhile pursuit as there are possibilities in software that have been overlooked by the commercial music software landscapes focus on re-implementations of traditional designs.

The Sinkapater [2], a sequencer allowing for multiple sequences which can be set to loop over different intervals, has been explored by Harriman. It has a 2D interface for programming sequences through three separate channels; each can have different number of beats. A 3D visual of water droplets falling in correspondence to each sequence is generated thereafter in order to better visualise this. However, there is not much in the way of proper evaluation of the system and how it performed.

¹amalgah.github.io

3 DESIGN

Sequences. Sequences are created through a mixture of cylinders and pegs which can be attached to the cylinder surface. Playing the sequence will cause cylinders to rotate based on their user assigned rotation speeds. Through this rotation, pegs may collide with other pegs, sending signals that can be received in the live-coding window.

Pegs. Pegs have three user adjustable settings, accessible by a right click menu. Settings are whether to mute the peg, when muted no signal is sent to its channel. A pegs channel is the audio channel it sends signals on, and can be set between 0 and 15. Finally a pegs signal is the message it transmits when triggered by a collision with another peg. A signal can be any number, and can be received and used as any numerical parameter within the users live-code.

Cylinders. Cylinders can be made to be any size, the height, radius and segments per face are all given by the user. The speed of rotation can also be edited. Interacting cylinders of varying size and speed creates a range of interesting scales and dynamics to be explored by the musician. One might create a larger cylinder to have a sequence repeat over a longer period than a smaller one, creating two scales of time and space for repetition of sounds and interaction between the sequences.

A sequence may start to be made by placing a cylinder and specifying its dimensions on an interactive grid, which you can see in Figure 1. The design encourages users to think spatially about the placement of sequences, similar perhaps to engineering cogs in a mechanical machine. However since the mapping between sound generation and interface is arbitrarily related in digital instruments, unlike sound generation based on acoustic means (such as an actual machine), there is more flexibility for unbounded experimentation.

4 METHODOLOGY

4.1 Implementation

The software was implemented as a widget within the Sema playground. A widget in Sema is a sub program that launches as a movable window within the playground. For 3D graphics three.js a WebGL library was used.

Interaction between the sequencer and live-code functionality is done through a custom command *fromSeq* in the default language[3] in Sema. Its returns values received on its given channel parameter. Figure 2 shows an example of this command being used. Live-coding is the practice of real time programming and performance, in this case for music.

```
+ 1 :a:{{0}fromSeq,{0}fromSeq}\909;
  2 :b:{{1}fromSeq}\909b;
  3 :c:{{2}fromSeq}sin;
  4 >{:a:, :b:, :c:}mix;
```

Figure 2: Screenshot of the live coding window

4.2 User study

4.2.1 Goals. To seek feedback on this early version of the system and evaluate it in terms of the users perceived effect it has on composition. We tested the software with a group of 9 users of mixed programming ability and musical experience.

4.2.2 Approach. Users completed a questionnaire after using the software in their own time. Questions were a mixture of tick box questions and text boxes for longer form responses. Recommended tasks were given in order to make sure everyone tests enough functionality to answer the questionnaire fully. Tasks included making a simple drum beat, a melody and some music that changed over time. However everyone is recommended to explore in their own capacity. Additionally volunteers were asked to use a traditional sequencer if they had not done before, we recommended one available to use freely[1]. This is to act as a baseline of comparison useful in evaluating musical controllers [5].

4.2.3 Audience. The user study was advertised towards musicians and programmers and anyone interested in the project. Promotions were made on social media and forums and various online communities related to live-coding. Due to this we had a range of people sign up with varying abilities and backgrounds. To cater for this testers were provided with resources² to learn the environment through written and video tutorials.

5 RESULTS

Testers were asked their level of programming experience between 0 (Novice) to 10 (Very Experienced). We received a range of results with 4/9 candidates reporting they were above 5 and 5/9 reporting 5 and below.

0/9 people had used Sema before and 2/9 had used another live coding environment before.

Proportion Having Used a Type of Sequencer or not Before the Study

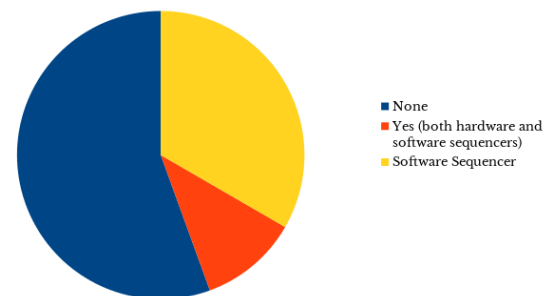


Figure 3: Proportion having tried a sequencer before the study

Figure 6 shows how the majority of those taking part in the study were unfamiliar with any type of sequencer.

5/9 people found making sequences using the sequencer easy. 4/9 who didn't were asked to give their reasons as to why they found the environment difficult to understand. Two of these responses

²github.com/amalgah/rhythmmachines-tutorial/blob/master/README.md

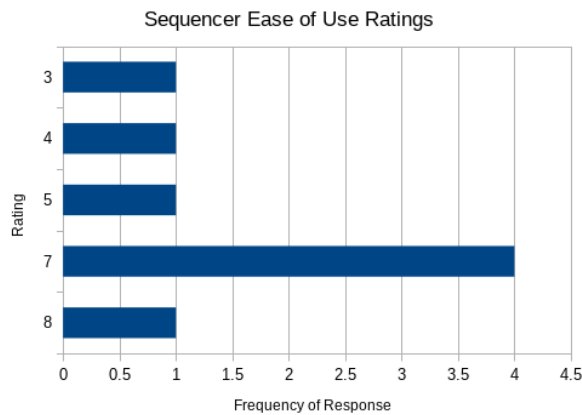


Figure 4: Ease of use rating

centered around difficulty in the programming side of sequencing, asserting they had little prior experience. Another of these responses said that "the environment was pretty intuitive once I had been using it for a short time my issue was a musical one but I did manage to make some short repeating beats". Finally another user mentioned that some information regarding navigation controls was missing from the tutorial and that right click menu's would display oddly in specific edge cases.

When asked the question *Did the sequencer change your thoughts on how you could go about composing music?* 6/9 responded "Yes" and 3/9 responded "A little"

Then out of those who responded (some empty responses) 5/8 responded "Yes" to "Does the sequencer provide any possibilities for composition that would be difficult or hard to do using a traditional music sequencer?" and 3 responded with "I'm not sure".

Users were asked if the music they created with the sequencer as turned out as they initially intended. The majority responded saying they were just experimenting, and so had no prior to the outcome.

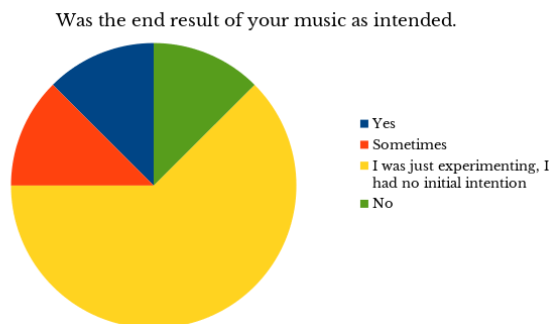


Figure 5: Ease of use rating

Out of those that responded, 83% of people said that the software they made with our sequencer was different to the kind they would make with a sequencer of a traditional design. One user stated that the "3-D space gave me a stronger perspective on how many unique sounds I was using. I often struggle with over-complicating my own compositions in sequencers, so a 3-D representation of each type of sound helped me to limit unnecessary additions.". Others commented on "creating music is a lot more satisfying and rewarding in 3D space".

"With a traditional sequencer design, it perhaps easier to see the different layers of composition. I think the 3D aspect of this sequencer allows the user to think and see the music in an entirely different way than they are used to allowing a different style and approach to music-making."

The comment above from a tester highlights how the 3D interface allowed users to think about music making in a different way allowing for a "different style".

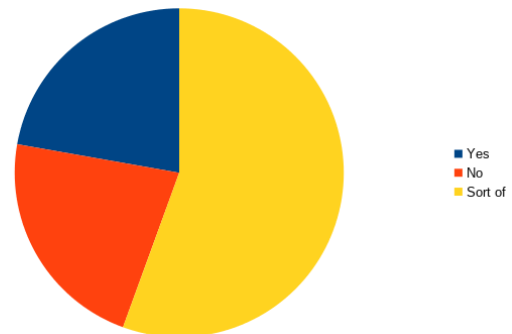


Figure 6: Proportions of Levels of Task Completion

Most people were able to "Sort of" complete the tasks. The tasks set were fairly open ended and so this result is to be expected. Those who responded "Yes" to completing all tasks spent the longest time using the software at 3hr+. It is likely everyone would be able to complete the tasks with more experience using the software.

6 DISCUSSION AND KEY FINDINGS

- Majority of people found sequencing in the environment intuitive and easy to use, particularly the 3D construction environment. However some found the live-coding aspect difficult, but this was reportedly due to lack of experience.
- Some issues with a newly developed piece of software are to be expected. Some users reported usability issues with some of the controls and user interface. More user options for customization was requested and will be added in the future.
- Users who had no programming experience found the environment difficult to use at first, and so much of their time was spent getting to grips with the environment as opposed to exploring it musically. This will have had some impact on results related to the impact on musicality.

- The majority of people expressed that the interface gave possibilities that were different to that of an ordinary sequencer. Some expressed that the 3D element encouraged more careful thinking of additions to a composition, a useful addition as over complication of a composition in electronic music can often be a difficulty. Although the majority of people thought that they made different music than they normally would with the sequencer it was unclear from the responses exactly what in terms of musicality was different about the music they made. Despite this the majority of testers cited the sequencer as rewarding and satisfying to use, and reframed how they could go about composition. This will undoubtedly have some effect on musicality despite peoples responses lacking formulation on exactly how.

7 CONCLUSIONS AND FUTURE WORK

We set out to create a 3D sequencing environment and investigate its effects on the composition of music. We successfully made a novel 3D sequencing design and investigated its effects on composition through a user study. Our results, although preliminary due to time constraints and participant numbers, are positive in regards to our research question. The majority of respondents thought that the sequencer allowed them to create different music than normal, through possibilities not provided by a traditional sequencer design. An instrument with surprising musical qualities unique to its 3D interface. There is great potential to expand the design further and

to fully integrate it within all the capabilities in Sema. Adding the ability to map vertical movement paths for cylinders and adjust size of pegs would allow for more complex and generative music to be made. Furthermore a function allowing for communication between the 3D sequencer and the Machine Learning window would open possibilities for models to interact with Cylinder and Peg parameters in real time, allowing for evolving compositions and visuals.

ACKNOWLEDGMENTS

This work was funded by the University of Sussex as part of the Junior Research Associate (JRA) program³. I would like to thank my supervisor Chris Kiefer for his support and advice throughout the project, as well as the rest of the Sema team for creating a great platform.

REFERENCES

- [1] [n.d.]. Make music online. <https://onlinesequencer.net/>
- [2] Jiffer Harriman. [n.d.]. Sinkapater – An Untethered Beat Sequencer. ([n. d.]), 2.
- [3] Chris Kiefer. [n.d.]. Sema Default Language Documentation. <https://github.com/mimic-sussex/sema/blob/master/docs/default-livecoding-language.md>
- [4] Chris Kiefer, F Bernardo, and Thor Magnusson. 2020. Sema. <https://github.com/mimic-sussex/sema> original-date: 2019-02-09T01:52:32Z.
- [5] Chris Kiefer, Nick Collins, and Geraldine Fitzpatrick. 2008. HCI methodology for evaluating musical controllers: A case study. In *In Proc. NIME*.
- [6] Kurt James Werner. 2014. The TR-808 Drum Machine and its Emulations. <https://pure.qub.ac.uk/en/publications/the-tr-808-drum-machine-and-its-emulations>

³www.sussex.ac.uk/suro/jra