

Groove Pizzeria
*Designing a Circular Step Sequencer for Polyrhythm and
Polymeter*

Tyler J. Bisson

Submitted in partial fulfillment of the requirements for the
Master of Music in Music Technology
in the Department of Music and Performing Arts Professions
Steinhardt School
New York University

Advisor: Dr. Alex Ruthmann

November 26, 2018

ABSTRACT

Despite the incredible affordances of DAWs and other hardware step sequencers, these tools present physical or perceptual limitations for the composition and comprehension of rhythm. Circular step sequencers address some of these shortcomings, however, few of these instruments emphasize or even allow the potential for polyrhythmic and polymetrical rhythm composition. In this thesis, research is presented exploring methods for visual representation of polyrhythm and polymeter on a circular interface through the design of Groove Pizzeria, a musical instrument prototype that builds on the strengths of the NYU MusEDLab's Groove Pizza. Through the software design as research (SoDaR) process, an interface was developed that was shown to allow a user to compose and comprehend polyrhythmic and polymetrical drum patterns in a way that would be prohibitive with most existing DAWs or hardware step sequencers. Evaluation of this prototype also concluded that the elements of the interface involved in creating polyrhythmic and polymetrical drum patterns were at least partially successful at three levels of emotional design (visceral, behavioral, and reflective) for the users of product tests. The design of Groove Pizzeria is ongoing and will involve additions and extensions of rhythmic features as well as optimization for usability as both an instrument and general software tool.

ACKNOWLEDGEMENTS

I should start by thanking my mom, dad, sisters, and old friends who believed in me when I said I wanted to pursue a master's degree they'd never heard of. Extra thanks to my mom and dad for continuing to believe in me when things must have seemed a little wacky in NYC. Thanks to Ariana for making endless trips back and forth on Metro North, spending weekends in libraries all over Connecticut and New York, camping out with me in my unheated, unsecured, unstable, and unbelievably hip apartment, and being unwaveringly ride or die the last two and a half years from my first moments studying for the music theory placement exam to this last week finishing my thesis. Thanks to Brad and Dave for helping me develop confidence in my abilities as a novice programmer, letting me keep that prime first letter real estate in the TBD2, and basically being some great buds. Thank you to Andre and Willie for giving me the opportunity to learn what it means, at least in a small way, to be an entrepreneur. Thank you to Andre for understanding when my academics had to come before the startup and being a loyal friend throughout the process. Thank you to my roommates Adam and Corrine for being such kind and calming friends and getting along so improbably well with my sub-leasers. Thanks to Marley and Annie for giving me a place to live when I moved to NYC and for being such kind and creatively energizing friends. Thank you to Dr. Alex Ruthmann for running the MusEDLab with such an inviting and inclusive spirit and for being an indispensable advisor in and outside the purview of this thesis. Thank you to Ethan Hein for letting me base my thesis off of Groove Pizza and for being an inspiring, insightful, and welcoming presence in and around the MusEDLab. Thank you to Dr. Agnieszka Roginska for supporting the research Brad, Dave, and I undertook that grew into all of our individual theses, being a voice of reason and assurance, and recommending me for tutor of DAP, a job I enjoyed immensely. To any other friends that didn't get direct mentions, you know I'm thankful for you too and there's only so much gratitude I can fit on a page in 12-point Times New Roman.

TABLE OF CONTENTS

1. INTRODUCTION	9
2. LITERATURE REVIEW	11
2.1 POLYRHYTHM, POLYMER, AND CONTEMPORARY ELECTRONIC/DANCE MUSIC	11
2.2 RHYTHMIC “PRESETS” IN ELECTRONIC/DANCE MUSIC	14
2.3 POLYRHYTHM, POLYMER, AND MIDI PIANO ROLL	17
2.3 NOTABLE CIRCULAR INTERFACES FOR RHYTHM COMPOSITION	19
2.3.1 <i>The NYU Music Experience Design Lab’s Groove Pizza</i>	19
2.3.2 <i>Meara O'Reilly and Sam Tarakajianz' Rhythm Necklace & Jeff Holtzkenner's Concentric Rhythm</i>	21
2.3.3 <i>Olympia Noise Co.'s Patterning</i>	23
2.4 LITERATURE REVIEW SUMMATION	24
3. METHODS	26
3.1 APPLICATION OF “SOFTWARE DEVELOPMENT AS RESEARCH” APPROACH	26
3.2 APPLICATION OF “EMOTIONAL DESIGN” THEORY	29
3.3 SUMMATION OF METHODS	30
4. INITIAL PROTOTYPE	32
4.1 INITIAL PROTOTYPE OVERVIEW	32
4.2 DESIGN CHARACTERISTICS DERIVED FROM GROOVE PIZZA	33
4.2.1 “Slices”	34
4.2.2 <i>Semitransparent Shapes on Three Concentric Rings</i>	34
4.2.3 <i>Ability to Rotate Shapes on Loop</i>	36
4.2.4 <i>Features Widespread Across Circular Rhythm Sequencers</i>	36
4.3 DESIGN CHARACTERISTICS DIVERGENT FROM GROOVE PIZZA	37
4.3.1 <i>The Second Loop</i>	37
4.3.2 <i>Loop Duration Control and Visualization</i>	37
4.3.3 <i>Wide BPM Range</i>	41
4.4 DEVELOPMENT PROCESS	41
4.4.1 <i>Fundamental Prototype Decisions</i>	43
4.4.2 <i>Basic GUI Design Choices</i>	43
4.4.3 <i>Loop Duration Decisions</i>	44
4.4.4 <i>Decision to Schedule Audio with High Accuracy</i>	48
4.4.5 <i>Terminology Decisions</i>	48
4.5 PRODUCT TEST 1 METHOD	49
4.6 PRODUCT TEST 1 RESULTS	51
4.6.1 <i>User Demographics</i>	51
4.6.2 <i>Visceral Level</i>	51
4.6.3 <i>Behavioral Level</i>	52
4.6.3 <i>Reflective Level</i>	54
5. SECOND PROTOTYPE	57

5.1 PROTOTYPE OVERVIEW	57
5.2 FEATURE ADDITIONS AND REVISIONS	58
5.2.1 <i>Revisions to Step-Count and Loop Duration Controls and Displays</i>	59
5.2.2 <i>Step-Duration-Ratio Displays</i>	62
5.2.3 <i>Timeline</i>	62
5.2.4 <i>Paper Tutorial</i>	65
5.3 PRODUCT TEST 2 METHOD	65
6. CONCLUSIONS	66
6.1 FUTURE ADDITIONS AND EXTENSIONS OF RHYTHMIC FEATURES	66
6.2 FUTURE USABILITY OPTIMIZATION	67
6.2.1 <i>Usability as a General Software Interface</i>	67
6.2.2 <i>Usability as Step Sequencer and a Musical Instrument for the Browser</i>	70
6.3 FINAL REMARKS	70
REFERENCES	72
APPENDIX A	77
ENDNOTES	86

LIST OF FIGURES

<i>Figure 1.</i> Groove Pizza interface	20
<i>Figure 2.</i> Rhythm Necklace interface	22
<i>Figure 3.</i> Concentric Rhythm interface	22
<i>Figure 4.</i> Patterning interface	23
<i>Figure 5.</i> first Groove Pizzeria prototype interface	32
<i>Figure 6.</i> interlocking reflection rhythm on first prototype	35
<i>Figure 7.</i> toggle rhythm on first prototype	35
<i>Figure 8.</i> polyrhythm on first prototype	39
<i>Figure 9.</i> polymeter on first prototype	39
<i>Figure 10.</i> complex pattern on first prototype	40
<i>Figure 11.</i> deceptively complex pattern on first prototype	40
<i>Figure 12.</i> Groove Pizzeria sketch	42
<i>Figure 13.</i> Groove Pizzeria & Groove Pizza loops side-by-side	44
<i>Figure 14.</i> first prototype loop synchronization visualization	47
<i>Figure 15.</i> second Groove Pizzeria prototype interface	57
<i>Figure 16.</i> time unit display & control on the second prototype interface	59
<i>Figure 17.</i> second prototype step-duration-ratio display	61
<i>Figure 18.</i> second prototype timeline display	63
<i>Figure 19.</i> second prototype timeline “double counter” bug	64

1. INTRODUCTION

In recent years, a number of circular step sequencers have become available. These cyclical representations of rhythm evoke modern electronic music's roots in tape manipulation and turntablism, providing screen-based¹ musicians with a visual metaphor for the ever-relevant "loop." The composition of electronic percussion has especially benefited by escaping the flatland of the MIDI piano roll through circular step sequencing. Today, visual representation of what famed polymath Godfried Toussaint has dubbed "the geometry of musical rhythm," make available difficult to play, write, and conceptualize percussion patterns to musicians of all experience levels. No longer merely a concern of ethnomusicologists and music technologists (but thanks to their work), these circular interfaces have increasingly been adopted as learning tools in music classrooms. They have also gained acceptance as bonafide musical instruments as rising stars of the electronic/dance music² scene such as Florian Meyer, AKA Don't DJ, and Joe Baker, AKA Forest Drive West, have become widely recognized for playing circular step sequencers in the club as well as the studio (Don't DJ, 2016; Lynch, 2018).

To this author, an avid fan and composer of electronic/dance music for over a decade, the advent of circular step sequencers gave rise to a hope that through the democratization of difficult to perform and comprehend rhythmic techniques, new modes of expression could be unlocked for this music of mind and body. More specifically, the freedom of these circular interfaces from software and hardware grids seemed to announce a weakening of common time's grip on the otherwise prodigiously evolving metagenre. These circular rhythm interfaces have many affordances, but unfortunately, like most traditional step sequencers, few allow or are well-suited for polyrhythmic or polymetric sequencing. This thesis is the product of research exploring methods for visually representing polyrhythm and polymeter on a circular interface through the design of Groove Pizzeria: a musical instrument prototype that builds on the strengths of the NYU MusEDLab's Groove Pizza.

Chapter 2 is an overview of literature and software development that supports and provides justification for this thesis. Chapter 3 explains the method of “research through design” that tethers the development of a Groove Pizzeria prototype to the academic document that is this thesis. Chapter 4 and 5, provide evaluation and discussion through sections focused on the design elements, design process, product test³ method, and product test results of the first and second Groove Pizzeria prototypes. In Chapter 6, future research directions are discussed and the research presented in this thesis is reviewed.

2. LITERATURE REVIEW

This literature review will begin by exploring the contemporary electronic/dance music community's fascination with and common misunderstanding of polyrhythm and polymeter. Next, a survey of evidence for rhythmic “presets” in electronic/dance music will be presented including a deeper loop at the MIDI piano roll. This chapter will end with a section focusing on four notable circular step sequencers and the way they break from restrictions introduced by digital audio workstation (DAW) presets while enforcing restrictions of their own.

2.1 Polyrhythm, Polymeter, and Contemporary Electronic/Dance Music

Due to the rise in popularity and influence of regional afrodiasporic electronic/dance music genres like Chicago's Footwork or Lisbon's Batida, readers of music journalism are likely to have encountered the term “polyrhythm” at least a couple of times in the last few years. In a 2017 Pitchfork review of Migos single “Stir Fry,” the production is described as being “funkier and more rhythmically complex than any Migos have previously tackled. The song's interlocking polyrhythms—bongos, cricket-like whistles, clicking trap hi-hats, and slowed-down snares... are a beauty to behold (Jayasuriya, 2018).” Listening to “Stir Fry,” it's easy to appreciate the song's intricate drum programming; it's also clear that the author of the aforementioned review is taking polyrhythm to simply mean “many rhythms.”

The hype around polyrhythm has extended into the world of electronic/dance music production along with a haziness of definition. In 2018, Arturia released a firmware update for its Beatstep Pro sequencer touting a new feature for sequencing polyrhythms. Yet, a press release demonstrates that the update actually allows forms of polymeter to be composed:

The new polyrhythm feature for the drum sequencer of the BeatStep Pro now allows you to define different sequence lengths for each of its 16 tracks. A 9-step hit-hat fill aside a 7-step tom lick, a 11-step snare march laying over a straightforward four-on-the-floor kick, all mixing together? It's up to you! (2018)

Similarly, Denis DeSantis' Ableton published book, *Making Music: 74 Creative Strategies for Electronic Music Producers*, describes a clever way of creating polymetric patterns in Ableton but incorrectly categorizes them as polyrhythm. These inaccuracies are not the fault of journalists, product managers, and software documentation specialists; in truth, the definitions of polyrhythm and polymeter have likely been vague for as long as they have been in use. In Jacob Rundall's 2011 doctoral dissertation *Polymeter: Disambiguation, Classification, and Analytical Techniques*, the author writes:

The usage of the word polyrhythm is almost hopelessly confused. Some authors offer definitions that are far too narrow, using polyrhythm strictly to refer to X:Y cross-rhythms. Many other definitions are far too broad and/or vague. In some cases, authors equate polyrhythm with polymeter. (p. 52)

Thus, it is worth providing a clear definition of polyrhythm and polymeter in this thesis. Rundall defines polymeter as "music that generates two or more conflicting rhythmic grouping structures simultaneously (p. 53)," where rhythmic grouping is defined as "a hierarchical structural phenomenon in music that involves the partitioning of consecutive elements into collections that are "close," often in a temporal sense, or in some other musical sense involving factors like pitch, dynamics, articulation and slurring, etc. (p. 21)"

On the other hand, Rundall defines polymeter as: "music that "gives rise to" two or more compelling metrical structures simultaneously that are not fully coincident," where "two metrical structures are fully coincident when they are identical in structure and coincide

completely in time, level-by-level and element-by-element” and “two metrical structures are non-fully coincident whenever they are not fully coincident—in other words, when they have the same structure but disagree temporally, or have different structures altogether (p. 40).”

These definitions, while discrete, are somewhat flexible and wide reaching for utility in academic analysis of all music. For the sake of relevance to step sequencing both linear and circular, these definitions can be greatly reduced and translated into a set of requirements using the terminology of electronic/dance music production:

Polyrhythm

- two or more concurrent loops of equal duration
- each loop has a set of evenly spaced notes
- the amount of notes in each are not equal
- the result of dividing the count of the larger set of notes by the count of the smaller set of notes results in a non-integer number

Polymeter

- two or more concurrent loops of different duration in total steps
- the duration of the steps dividing each loop is identical
- the sequence of notes written in each loop is different in different rotations around the loop's origin (e.g. not 16 identical and equally spaced notes in a 16 step loop)
- the loops begin and end at different times while each step of each loop plays at the same time

2.2 Rhythmic “Presets” in Electronic/Dance Music

From the squelching TB - 303 baselines of acid house to the ultra-bright “Super Saw” leads of the JP-8000 in trance music, the history of electronic/dance music has been shaped by the built-in sounds of various iconic instruments. In his book, *Presets - Digital Shortcuts to Sound*, writer and electronic music producer Stefan Goldman discusses the use of these music production “readymades” with a range of artists, music software designers, and audio engineers. In an interview with Robert Henke, another electronic/dance music producer who is notably a co-developer of Ableton Live, Goldman questions Henke about what he perceives to be the contemporary electronic/dance music community’s obsession with novel sound design:

Essentially, it is not tolerated to use preset sounds or re-use samples, but it is perfectly tolerated to end up with the same beat, in the same tempo with the same type of shuffle, which all happens to be presets too. (p. 34)

The best example of Goldman’s assertion might be found in the meticulous musical practices of Matthew Herbert, an acclaimed composer of electronic/dance music who has gone to war with the preset, releasing a number of records that rely on an extremely restrictive set of sound design rules described in his 2005 manifesto, “PERSONAL CONTRACT FOR THE COMPOSITION OF MUSIC.” The manifesto mentions nothing about rhythm, and true to Goldman’s assertion in *Presets*, much of Herbert’s music uses digital music software or hardware to produce rhythmic form that is common across electronic/dance music sub-genres like house and techno.

Of course, the repetitive rhythms of electronic/dance music can be thought of as another artistic constraint. Neill argues this in his 2002 article, “Pleasure Beats: Rhythm and the Aesthetic of Current Electronic Music,” which compares the metagenre to “academic” computer music:

The two worlds of high art and popular electronic music may use slightly different tools, their aesthetic approaches are most clearly defined in terms of the presence or absence of repetitive beats... Just as composers in earlier historical periods worked within a given set of large-scale formal parameters (sonata form, dance forms, tone poems, etc.), innovative pop electronic composers use steady pulse, loop-based structures and 4/4 time as a vehicle for a wide range of compositional ideas and innovations. Shifts in tempo, subdivision, sonic manipulation and complex quantization structures are making beat science the new jazz of the 21st century. (p. 3 - 4)

Rather than differentiating electronic/dance music's rhythmic sensibility from 20th century art music, as Neill does, Philip Sherburne draws parallels between the two in his essay, "Draw a Straight Line and Follow It: Minimalism in Contemporary Electronic Dance Music":

In the popular imagination, electronic music is often synonymous with spectacle and excess... But the driving impulse in most electronic dance music is far more focused: it comes from minimalism. This is particularly true in house, techno, trance, and other forms built around a four-to-the-floor beat; but it is also the case for genres like jungle and drum 'n' bass, where breakbeats are looped into a heavily repetitive, almost trance inducing churn... What are some of the characteristics of electronic music that could be classified as minimalist? Many of them are related to what the critic Kyle Gann has identified as certain fundamentals of the minimalism of the 1960's and 1970's, particularly repetition, static harmony, steady beats, additive processes, static instrumentation, linear transformation, metamusic (overtone phenomena, psychoacoustics, and other aspects related to a focus on timbre and texture over melody), and, finally, the influence of non-Western Cultures. (2017, p. 465-466)

Although Sherburne argues that certain electronic/dance music producers took direct inspiration from minimalist art music, thereby influencing the metagenre at large, its minimalist qualities are also due to the limited supply and ability of technology available to its originators:

Listen back to virtually any early house or techno from Chicago or Detroit, and you're likely to be struck by its spaciousness, its leanness, its economy. Partly, that has to do with the way the music was produced. Many early dance musicians worked with a modest kit: no more than a handful of synthesizers accompanied by a lonely drum machine on a four-track recorder... doing more with less remains an inspiration and an ethos. (2017, p. 469-470)

At a deeper level, it would seem that techno and house, two “4-on-the-floor” centric dance genres, have come to be defined by the rhythmic affordances, essentially the presets, of their drum machines of choice. The straight, swingless timing of the TR-808 dominates the cold and futuristic sound of Detroit techno tracks⁴ such as the seminal “Clear” (1983) by Cybotron. Later, the “shuffle” settings pioneered by Roger Linn on his Linn LM-1 drum machine, which was accomplished by delaying the playback of alternate sixteenth notes, were adopted on other classic drum machines such as the TR-909 (Linn, 2015). This “shuffle” setting added a distinctive swinging quality to 90’s house tracks by groups such as the aptly named Mood II Swing which went on to influence UK garage around the turn of the millennium (Sherburne, 2016). Anecdotal evidence suggests that more recently emergent genres of electronic/dance music have developed rhythmic traits from software presets. Grime, for instance, an electronic/dance sub-genre that began in early 2000’s London and draws on hip-hop and UK club music, typically features songs at 140 BPM, the default tempo of Fruity Loops.⁵

2.3 Polyrhythm, Polymeter, and MIDI Piano Roll

More influential than any default tempo setting, the standard MIDI piano roll is perhaps the defining rhythmic “preset” in digital music technology. Hein characterizes the MIDI piano roll as a kind of “super” Time-Unit Box System (TUBS)⁶, that “gives little indication as to the function of ... musical events” and does not allow a user to easily “grasp the broader metrical scheme without doing a great deal of meticulous counting (38).” If the MIDI piano roll makes it difficult to understand the function of musical events, one might wonder what repercussions this has for the electronic/dance music made with the tool. Brian Eno has expressed a belief that staleness in rhythmic content may be an unintended consequence:

Eno is disenchanted with almost all the music currently made using computer technology. Speaking to *Wired* magazine, he complained that digital, sequenced music had merely resurrected many of the inherent limitations of classical orchestral music, with its hierarchical ranking of instruments in the mix, its rigid sense of pitch, and its locked rhythms tied to the conductor/timekeeper. "Classical music is music without Africa," he complained, later adding, "the problem with computers is that there is not enough Africa in them..." Eno's model of musical liberation is African percussion ensembles, which are based not just on polyrhythm, as is much rock and dance music, but on multi-tiered meters that coexist in a sort of loose interdependence, locking together at certain dramatic moments and then drifting apart again. Africa's polymetric perversity has a utopian, democratic charge for Eno. By comparison, house and techno producers are "slaves to their machines, just as most of us are slaves to [the machines we use] This is a music that's particularly enslaved. (Reynolds, 1999, p. 47-48)

In his article, “Roots and Wires Remix: Polyrhythmic Tricks and the Black Electronic,” Erik Davis bemoans the genre jungle’s course in the latter half of the 1990’s and echoes Eno’s concerns about music software’s aversion to polyrhythm and polymeter:

Like many listeners, I found that jungle did not explore the most interesting territories it opened up. The polyrhythmic play hinted at in its earlier iterations gave way to, on the one hand, pleasant cocktails of carbonated fusion, and on the other, a nihilistic tedium that reduced the music’s syncopated aggressivity into a jack-hammer monotony better deployed by death metal. Though edgier dancehall carries on the torch of drum’n’bass, the polyrhythmic potential of electronic dance music still remains in many ways untapped. The club remains dominated by the easy sell of four-to-the-floor, while most self-consciously avant-garde electronic music continues to keep the dance – mental or not—at arm’s length. The adherence to a central organizing beat is partly the fault of music software: genuine apart- playing is difficult to arrange, since most programs insist upon a single meter and even automate beat-matching (Davis, 2008).

The notion that the design of DAWs discourages the composition of polyrhythm and polymeter is supported in the “Problems of Progressing” section of the Ableton published book, *Making Music: 74 Creative Strategies for Electronic Music Producers* by electronic/dance music producer and head of documentation at Ableton, Dennis DeSantis. DeSantis introduces the following problem:

Many electronic genres make extensive use of loops. When you listen to your favorite examples of loop-based music, there’s always some sense of motion that keeps the music fresh and interesting, but your own attempts to work this way usually strike you as static and boring. (2015)

DeSantis provides a solution by introducing a method for creating polymetric (mentioned earlier in this thesis for being mislabeled as polyrhythmic) loops in Ableton by triggering an instrument with multiple clips of different durations requiring inter-track routing. Although this solution provides a working strategy, its inclusion in *Making Music* suggests that creating polymetric loops in a DAW is an inherently non-intuitive process (2015).

2.3 Notable Circular Interfaces for Rhythm Composition

Since as early as 2013, a steady stream of circular rhythm interfaces have become available independently or as features of music technology (Hein, 2013). Often described as being designed to help electronic/dance music producers break free of the modern DAW's archetypal grid metaphor, these interfaces afford producers with novel ways of creating and considering rhythm; they also have their own set of creatively limiting rhythmic presets which will be explored throughout 2.3. This section includes descriptions of four notable circular rhythm interfaces and their applicability for electronic/dance music producers from the standpoints of both rhythmic comprehension and composition.

2.3.1 The NYU Music Experience Design Lab's Groove Pizza

In 2013, Ethan Hein completed his master's thesis titled, "Designing the Drum Loop: A constructivist iOS rhythm tutorial system for beginners." The iOS app Hein describes in that thesis became the NYU MusEDLab's Groove Pizza: a browser-based "circular rhythm app for creative music making and learning (<http://musedlab.org/groovepizza>)" (*Figure 1*). In his thesis, Hein covered three notable circular rhythm interfaces that were available at the time: Propellerhead's Figure, O-Music's O-Generator, and Casual Underground's Loopseque. There have been a number of more

fully-featured circular step sequencers released since 2013, including Groove Pizza itself, which will be covered in this literature review.

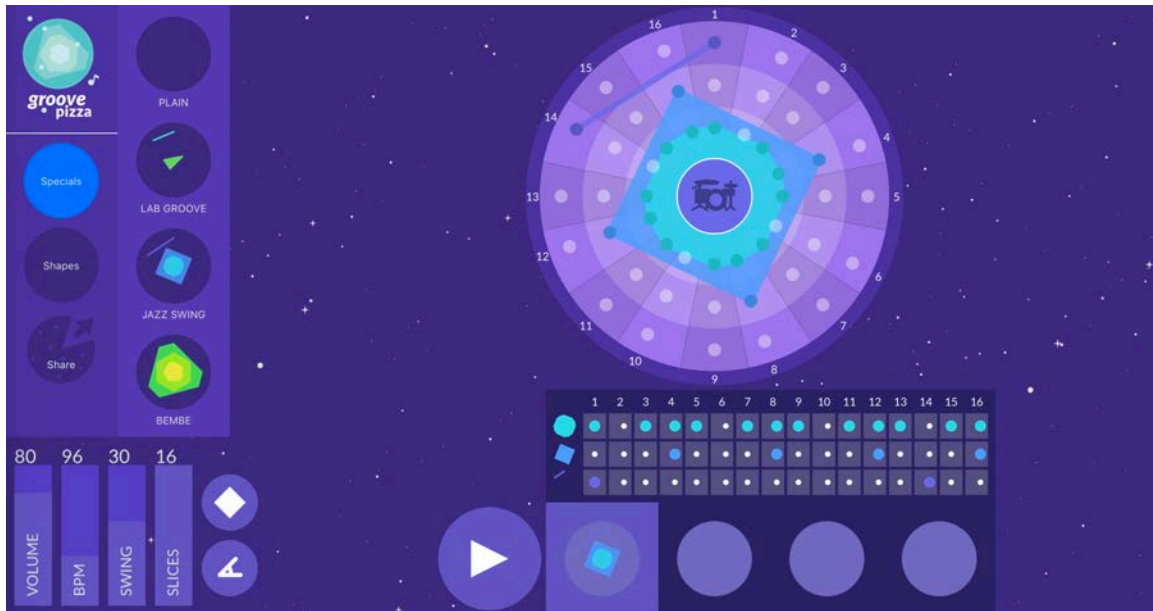


Figure 1: Groove Pizza preset a.k.a. “special” Jazz Swing drum pattern. Notice that evenly numbered steps are shifted forward visualizing swing (Groove Pizza, n.d.)

Groove Pizza is a browser-based drum machine with three concentric circles that map to kick, snare and hi-hat samples in a circular time unit box system (TUBS) diagram, a simple system for notating musical rhythms using a horizontal 1 by N grid denoting a fixed unit of time and a mark within a box representing a musical event (Koetting, 1970), that can be described as a rhythm necklace.⁷ Arriving on Groove Pizza’s home page, a user is treated to a series of “specials”: editable preset rhythms that avoid the common DAW “metaphor of a blank canvas or void (Ruthmann, 2012).” Comparing a drum pattern from Michael Jackson’s “Billie Jean” or Herbie Hancock’s “Chameleon,” one can easily make visual distinctions between the pop-wise symmetry of the former beat or the lopsided funk of the latter. Groove Pizza includes an option for dragging “shapes” onto the pizza. These shapes, equilateral polygons with 3, 4, 5, 6, and 8 beats, automatically become euclidean rhythms when applied to the pizza (Toussaint, 2005); an unthreatening and inspiring way to begin making music. Another interesting feature of

Groove Pizza is that rhythms can be swung and visualized utilizing a novel graphical representation of the technique (Hein, 2013).

Although the MusEDLab's tool is geared towards beginners, electronic/dance producers who have learned their craft in a specific environment, say Ableton Live or Logic, could learn new ways to create and perceive rhythm from Groove Pizza as they work within its limitations. In Hein's words:

Music education has tended to treat composition and audio engineering as separate practices, but in the case of the solo bedroom producer, the distinction is no longer meaningful. There is precious little in the way of formal education for such musicians. In this context, software presets and default sounds become a critical educational resource. Bedroom producers may learn everything they know about EQ or reverb simply by scrolling through the built-in settings in their plugins. (2013, p. 59)

2.3.2 Meara O'Reilly and Sam Tarakajianz' Rhythm Necklace & Jeff Holtzkenner's Concentric Rhythm

Meara O'Reilly and Sam Tarakajianz's Rhythm Necklace (*Figure 2*) and Jeff Holtzkenner's Concentric Rhythm (*Figure 3*) are two iOS applications that take direct influence from Godfried Toussaint's research and offer an intuitive and exploratory way to create polyrhythms. In an interview with the creators of Rhythm Necklace on Vice's former "The Creators Project" website, the app is described as giving users an "instinctive method to make music, while also letting exploration be a key factor." This quality is reflected in the Rhythm Necklace's auto-population of beats in Euclidean patterns if one drags their finger from one step to another. Concentric Rhythm also allows a user to quickly create a rhythm without meticulously entering individual steps into the sequencer by auto-populating Euclidean rhythms and "ghost notes." Concentric Rhythm includes other satisfying features such as an ability to map "random walk" melodic patterns onto

rhythms. The wisdom in implementing these types of quick pattern generating features is supported in findings suggesting that electronic music producers typically enjoy letting their musical machines “have a say in the outcome of their work (Gelineck & Serafin, 2009).”

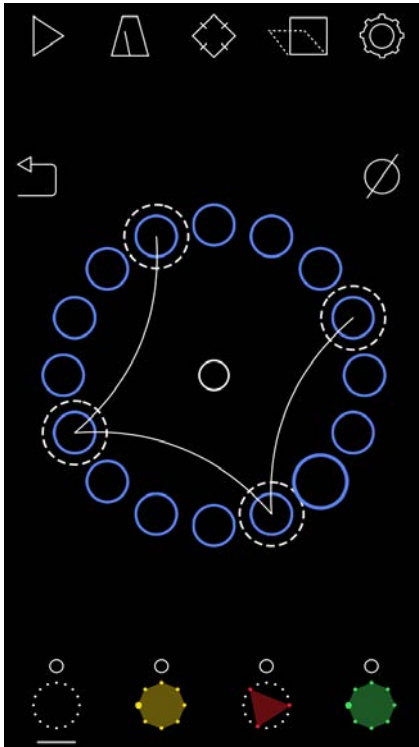


Figure 2: With Rhythm Necklace, dragging a finger between two steps instantly creates a euclidian rhythm. Its sequencers can have different amounts of steps, but are constrained to the same duration, permitting polyrhythm and disallowing polymeter. (O' Reilly et al., n.d.)



Figure 3: Concentric Rhythm offers a wide array of presets that manipulate a given rhythmic pattern, however, because each concentric rhythm exists in a loop of equal duration, polymeter cannot be created. (Holtzkener, n.d.)

Despite the creative opportunities afforded by Rhythm Necklace and Concentric Rhythm, polymetric sequencing is not possible on either application. As a result, the complete rhythmic pattern generated in each tool repeats very quickly and can become monotonous

without continued active engagement. It is not entirely fair to criticize either app; they are both musically successful in unique ways, however, one could imagine that the introduction of new rhythmic affordances might make each more engaging.

2.3.3 Olympia Noise Co.'s *Patterning*

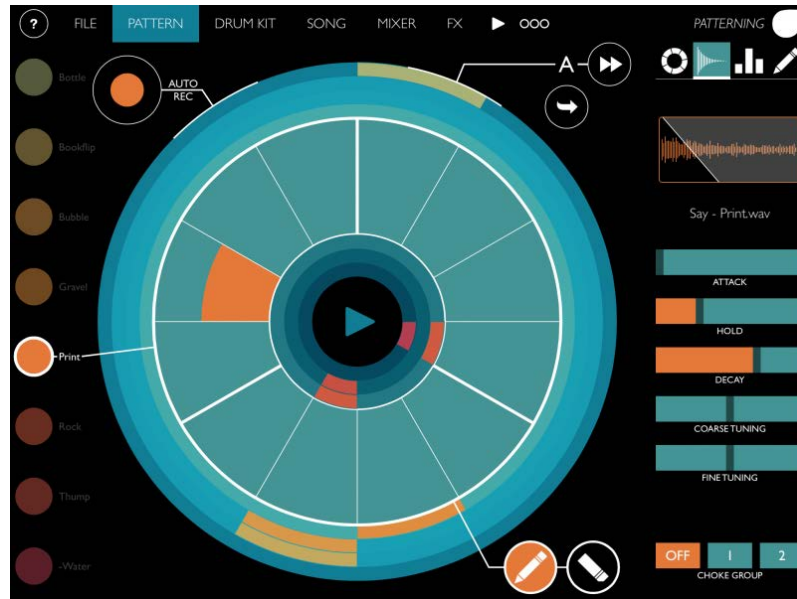


Figure 4: Patterning is deeply featured and playable but lacks the perceptual affordances of circular rhythm interfaces that utilize points for steps and connective polygons.

In 2015, Olympia Noise Co. released *Patterning* (Figure 4): an iOS app that may be the most fully featured circular step sequencer on the market. *Patterning* features both polyrhythmic and polymetric sequencing with independently adjustable rhythm necklaces set in concentric rings. The independence of both step-count and length in each ring grants *Patterning* more rhythmic leg room than a circular rhythm interface with less available steps. To illustrate this concept, consider the 3-4-5 “special” on *Groove Pizza*. Because 5 notes cannot be equilaterally spaced across 16 steps, *Groove Pizza* is forced to distribute the notes with the following durational pattern: [2-3-2-3-2].⁸ One could form the 3-4-5 “special” in *Patterning* with even timing between each note by simply setting

three concentric rings to the same length, dividing each loop into 3, 4, and 5 steps respectively, and setting a note on each step.

However, when it comes to allowing comprehension of polyrhythm and polymeter, Patterning has some shortcomings due to its visual design. For one, the “slider” metaphor, which allows the velocity of each step in a necklace to be adjusted like a volume bar, makes visualizing the geometry of rhythm difficult; thereby neutering one of the strongest perceptual affordances of its circular rhythmic interface. Another perceptual limitation of Patterning is its nesting of each rhythm in a concentric ring. Although perceiving polyrhythm in this way is easy due to the fact that each loop begins and ends at the same time, observing the asynchronous loops of polymeric patterns become quite confusing. Perhaps Patterning’s most egregious omission, from from both an educational and ease-of-use perspective, is its lack of step numbering or gridlines. Hein correctly points out in a blog post, “You can have up to 64 cells in each ring, but without numbering, how are you supposed to figure out where to put your drum hits except by trial and error (2015)?”

2.4 Literature Review Summation

Music journalism and marketing materials for electronic/dance music production tools demonstrate that there is consumer interest in polyrhythm and polymeter. Despite this enthusiasm, the terms are commonly confused due to a long standing scholarly negligence in rigorously defining them. Even armed with an accurate definition of polyrhythm and polymeter, most electronic/dance music hardware or software (really any music composition hardware or software) doesn’t make it easy to utilize the techniques in composition; a fact that has been instrumental in complaints about the rhythmic qualities of electronic/dance music being overly attached to a 4/4 grid.

This grid, which is imposed physically, perceptually, or otherwise spiritually by many hardware and software music creation tools can be thought of as kind of rhythmic

“preset”. Although the word preset might be read to have a negative connotation in the context of electronic/dance music production, they have proven to be cherished creative limitations in the metagenre’s evolution. The grid of the MIDI piano roll is no exception and has certainly influenced the rhythmic content of many well-loved genres under and outside of what would normally be described as the electronic/dance music umbrella.

However, presets are desired or disavowed situation to situation and circular step sequencers have increasingly granted musicians the ability to avoid some of the rhythmic presets of DAWs and other sequencing tools by letting the “geometry of musical rhythm” shine through their circular form. However, few of these instruments fully address the electronic/dance music community’s interest in (as well as its detractors’ claimed absence of) polyrhythm and polymeter.

Research through the design of a Groove Pizzeria circular sequencer prototype is for the purpose of understanding the requirements of a user interface that would allow polyrhythm and polymeter to be default presets. An interface achieving this goal would, ideally and in theory, shape the music it facilitates according to new principles; nudging electronic/dance music off of the common time grid and into the future.

3. METHODS

Documentation and analysis of the design process leading to the Groove Pizzeria prototype, a circular step sequencer for the composition and comprehension of polyrhythm and polymeter, was the core activity of this thesis and can be broadly categorized as a research through design process:

An approach to interaction design [that] integrates models and theories with technical knowledge in the design process. Through a process of ideation, experimentation, and critique, designers then reframe the problem to arrive at the “right” solution. Of the artifacts that emerge from the design process, including sketches, drawings, models, and prototypes, the most critical is documentation, which contextualizes and communicates design action. (Hanington & Martin, 2012, p. 146)

At a macro level, the design process taken in this research corresponds to the five step (empathize, define, ideate, prototype, and test) design thinking process popularized by Stanford’s d.school. However, a more specific research through design approach called Software Development as Research (SoDaR) was utilized as a framework for this thesis because of its pertinence to designing novel musical interfaces (Brown, 2007).

3.1 Application of “Software Development as Research” Approach

Introduced in his essay, “Software Development as Music Education Research,” Brown defines SoDaR as having three stages in which “the researcher is encouraged to describe the objectives for that stage, collect data resulting from work undertaken at that stage, and to reflect on the outcomes, problems, and progress of the research to that point (2007, p. 5)”:

1. Define the activity
2. Software design and production
3. Usage and refinement

In the “define the activity” step of a SoDaR research project, the researcher(s) begin by clearly identifying a situation “in which a new software application is likely to encourage interaction leading to learning (2007, p. 5).” In the case of this thesis, an activity where users compose drum patterns reflecting an understanding of polyrhythm and polymeter on a novel software interface was selected. Because this activity is ultimately one of musical composition, “learning,” does not only apply to the pedagogical goal of teaching polymeter and polyrhythm, but, also the process of familiarizing oneself with a digital music instrument that counts the the ability to compose with these techniques as prominent affordances. Semantically inverted, but equally true, is that this learning process can be described as building an “embodied tacit knowledge” of a digital instrument by “getting a feeling” for an instruments constraints (Magnusson, 2010, p. 65). Although Magnusson characterizes the mapping of constraints as ultimately more important than the designing of affordances in digital music systems, at least in the case of Groove Pizzeria, constraints can be thought of as the boundaries of the sequencer’s feature set rather than the features excluded from the instrument. Therefore, by learning where the Groove Pizzeria’s limits lie, a user has experienced a wide range of polyrhythmic and polymetric possibilities afforded by the instrument.

Following completion of the “define the activity” stage, the “software design and production” stage began. This stage roughly corresponds to the “ideation” and “prototyping” stages of the five part design thinking process. As was done in the process of this thesis project, ideation and prototyping are commonly fused: “Another ideation technique is building – that is, prototyping itself can be an ideation technique. In physically making something you come to points where decisions need to be made; this encourages new ideas to come forward (An Introduction, 2010).”

Design and production of the initial Groove Pizzeria prototype was informed by the concept of the minimum viable product (MVP):

a version of a new product with just those features (and no more) that allow the product to be deployed, thereby allowing a team to collect the maximum amount of validated learning about customers with the least effort. It is an iterative process of idea generation, prototyping, presentation, data collection, analysis and learning (Lenarduzzi & Taibi, 2016)⁹.

While the wording of this definition suggests that the MVP is purchased by a customer, in the case of case of this thesis, “deployed” refers to a user’s testing of a prototype. As such, the Groove Pizzeria prototype is minimal in terms of features relevant to the composition and comprehension of polyrhythm and polymeter as opposed to an MVP with the minimum set of features that would allow it to succeed commercially. Electing these relevant features in order to induce a focused response on polyrhythm and polymeter in product tests was one of the most significant challenges of this project. These decisions were recorded and analyzed throughout the development process.

In the third SoDaR stage, “usage and refinement,” multiple iterations of product tests with individual users and subsequent revisions to the software are undergone. Research in product testing suggests that 100% of usability problems can be discovered through 15 user tests given the equation: number of usability problems = $N (1 - (1 - L)^n)$. Here, “N is the total number of usability problems in the design and L (typically 31%) is the proportion of usability problems discovered while testing a single user (Nielsen & Landauer, 1993). When conducting iterative design for web development projects Nielsen recommends conducting product tests with 5 users per design iteration to identify 85% of usability problems in a design (2000). Subsequent test iterations will unveil the remaining 15% as well as new usability issues created by added and revised features. In the product

tests for the first Groove Pizzeria prototype, a user was verbally guided through a “lesson plan” derived from learning objectives drafted in stage one. Brown notes that during product tests the researcher(s) should record the behavior of users including “comments, observable successes and failures, bug reports, and feature requests (2007).” Throughout each test, supporting data was collected by means of interview and think-aloud protocol as well as casual conversation and observation. Audio and video footage of the user’s engagement with the prototype was recorded for subsequent analysis of both the prototype and the user test itself. No footage or direct transcription from these tests is published in this thesis, rather, a simple form of content analysis was used to track emergent themes and evaluated using Donald Norman’s theory of emotional design which will be discussed in 3.2.

The testing of the second Groove Pizzeria prototype was made more consistent through feature additions and revisions determined by the results of the first set of product tests. The most notable of these additions was a paper prototype of a Groove Pizzeria tutorial. Therefore, rather than verbally guide a user through a “lesson” plan, think-aloud protocol was utilized more heavily as a user was asked to complete the tutorial while narrating their inner monologue. This process was also filmed, the content of which will be subjected to analysis according the emotional design principles to determine third iteration feature additions and revisions.

3.2 Application of “Emotional Design” Theory

Donald Norman’s theory of emotional design defined in his 2003 book, *Emotional Design*, claims that three levels of psychological processing grant humans the ability to achieve complex tasks such as creating music: “the automatic, prewired layer, called the visceral level; the part that contains the brain processes that control everyday behavior, known as the behavioral level; and the contemplative part of the brain, or the reflective level.” These three levels of processing have three corresponding levels of design, the

achievement of which can result in not only greater enjoyment of a given product, but also a greater utility, and indeed for creative tools, a more satisfying end result. Because musical instruments are objects of pleasure, the emotional design framework was deemed an appropriate lens through which to evaluate the successes of product tests, which informing subsequent revisions of the prototype.

Content analysis of video footage from the product tests was used to identify common themes that emerged across the product tests. This is discussed in a deductive format, that is, results are organized according to the three preordained areas of focus of the tests: the visceral, behavioral, and reflective levels of emotional design. However, within each level of design that serves as a category, subcategories of interest to Groove Pizzeria's design were developed inductively, that is, based on emergent themes across the product tests (Hanington, 2012).

3.3 Summation of Methods

This thesis utilizes a research through design process well suited for the development of music education software (and digital musical instruments that can more generally facilitate learning) known as software design as research (SoDaR). In the first step of the SoDaR process, "define the activity," a learning activity was proposed: a user will compose drum patterns utilizing polyrhythmic and polymetric techniques on a novel user interface. In the second stage of the SoDaR process, "software design and production," Groove Pizzeria, a prototype meant to fulfill the proposed learning objective was designed and developed. In the third, and ongoing iterative stage of the SoDaR process, "usage and refinement," the Groove Pizzeria prototype was tested with users. Video footage from these tests was then analyzed according to the theory of emotional design which claims that products, especially those that facilitate creativity such as musical instruments, engage with a user at three levels of psychological processing: the visceral, the behavioral, and the reflective. A second Groove Pizzeria prototype was developed based on the results of the first set of product tests. Additionally, a more streamlined

product testing process was adopted for the second prototype, the results of which are being analyzed. Through this SoDaR process, the author took careful notes of his own design process for documentation, analysis, and discussion that make up the bulk of this thesis.

4. Initial Prototype

This chapter begins with a high-level description of the initial Groove Pizzeria prototype. Following this is a discussion about the prototype’s design considerations broken into sections involving features carried over from MusEDLab’s Groove Pizza and features divergent from Groove Pizza. After this is a description of the design process that lead up to the Groove Pizzeria prototype and the design challenges that arose throughout this process. The final section of this chapter is a discussion regarding the product test and its results.

4.1 Initial Prototype Overview



Figure 5: Interface of the initial Groove Pizzeria prototype

The initial Groove Pizzeria prototype is a locally hosted JavaScript program that runs in Mozilla Firefox (Figure 5). It makes use of the web audio API and the p5.js javascript library.

Groove Pizzeria's visual interface prominently features two circular step sequencers (loops) with a teal and orange color scheme corresponding to each loop. Each loop can be independently divided into 2 - 16 steps and set to an overall duration of 2 - 16 sixteenth note durations. Both of these parameters are controlled by a slider with an accompanied numerical display. Changing the amount of steps in a loop results in an updated set of radial divisions. Changing the duration of a loop results in a proportionally updated loop circumference with the space between every two gray ovular shapes (teeth) along the circumference denoting a sixteenth note duration. Steps on both loops feature three concentric rings of buttons which can be switched off and on. A single "on" step is marked black. Two "on" steps in a concentric ring form a line and multiple form semi-transparent polygons. The six total concentric rings represent six preset drum voices: on the left, 909 kick, clap, hi-hat radiate outward from the center; on the right high, mid, and low 808 toms radiate from the center. A darker teal and orange ovular shape (playhead) travels around the circumference of each loop clockwise at the rate of the set BPM, which is controlled by a slider with a displayed numerical value. As the playhead passes over "on" steps, the notes are played. Another set of sliders corresponding to each loop allows a user to rotate the given set of "on" steps (the sequence) clockwise.

The initial Groove Pizzeria prototype can be controlled with a mouse or trackpad. Clicking on or clicking and dragging over step buttons will invert their state from "off" to "on" or "on" to "off." Sliders are controlled by clicking and dragging or clicking to highlight, then controlling with the keyboard arrow keys.

4.2 Design Characteristics Derived from Groove Pizza

MusEDLab's Groove Pizza is commonly used as a learning tool for elementary school aged music students. The aim of developing Groove Pizzeria in homage of Groove Pizza was to take the design elements that make the latter a successful learning tool for kids and adopt them for use in a tool that would allow electronic/dance music producers to more

readily comprehend and compose polymeter and polyrhythm. Section 4.2 discusses these features derived from Groove Pizza in detail.

4.2.1 “Slices”

One of the most significant features Groove Pizzeria inherits from Groove Pizza is the use of “slices”: radial divisions of the loop into shapes resembling pizza slices. The use of slices provides an advantage over other circular rhythm sequencers by granting the user a greater sense of the rhythmic sequence’s geometry. For example, a user of Groove Pizza does not need to guess which step is exactly opposite or quarter way from the downbeat in an evenly divided loop; a problem that a user of Rhythm Necklace or Patterning may encounter especially at step-counts greater than Groove Pizza and Groove Pizzeria’s maximum of sixteen.

4.2.2 Semitransparent Shapes on Three Concentric Rings

Another trait inherited from Groove Pizza is the overlapping of semitransparent shapes in three concentric rings. These shapes, derived from Godfried Toussaint’s rhythm necklaces, allow a user to visually comprehend the metrical function of a given rhythm in way that is difficult to achieve through the necessary counting of grid lines when using standard western notation (Hein, 2013). Concentric rings extend the perceptual affordance of shapes making by allowing a user to intuit rhythmic concepts that require multiple voices (or hands in the physical world). Thus, a user without music theory expertise can create complex and compelling rhythms by engaging with simple geometric concepts; ideally intuiting the relationship between shape and rhythm. For instance, by creating two mirror symmetrical shapes (*Figure 6*), a user might happen upon an interlocking reflection rhythm; one in the category Toussaint simply calls “good rhythms”:

While it is difficult to define precisely the necessary and sufficient conditions that make a “good” rhythm good, it is not too hard to list a variety of properties that contribute to a rhythm’s goodness. One such property is that the mirror-symmetric transformation of a rhythm about some axis of the rhythm cycle, represented as a circle, be equal to its complementary rhythm. Rhythms that have this property will be called *interlocking reflection* rhythms. (2010, p. 1)

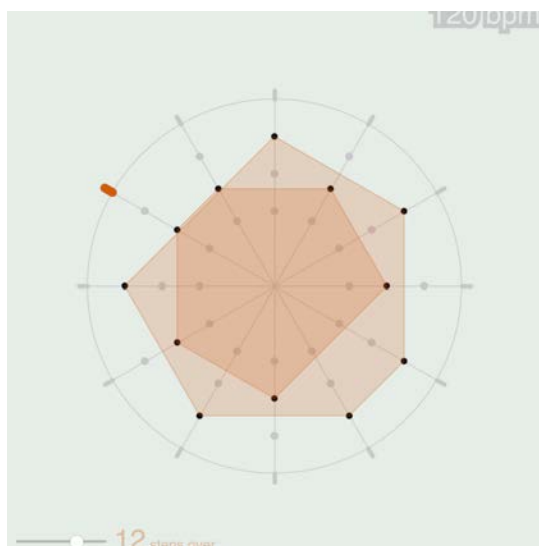


Figure 6: Here is the *guataca* (metal hoe blade) timeline used in Cuban *batá* drumming (Toussaint, 2010). Observe how this is defined as an *interlocking reflection* rhythm because the two polygons are mirror transformations of one another (besides their size difference).

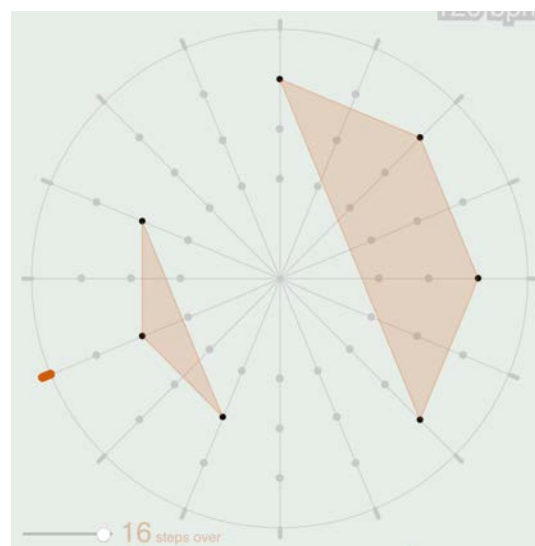


Figure 7: Here is a hand-clapping timeline pattern from Ghana (Toussaint, 2010). Observe how this is defined as a *toggle rhythm* because the two polygons can be separated by a straight line.

Another easily perceived possibility is a toggle rhythm (Figure 7); a type of rhythm “that when played using the alternating hands method, have their onsets divided into two consecutive sets.” According to Toussaint:

The representation of cyclic rhythms on a circle permits an alternate definition of toggle rhythms based on the notion of linear separability in geometry. A rhythm (set of integer points on the circle) is a toggle rhythm if there exists a straight line that separates the left-hand onsets (on odd-numbered pulses) from the right-hand onsets (on even-numbered pulses). (2010, p. 14)

The interplay of these shapes is both simple and rich in possibility, reflecting the “freedom to explore” and “perceived stability” requirements for successful new musical instruments proposed in “Evaluating the Experiential Quality of Musical Instruments” (Schmid, 2016).

4.2.3 Ability to Rotate Shapes on Loop

Like its forbearer, Groove Pizzeria allows sequences to be rotated clockwise or counterclockwise around the origin of each circular interface. According to Hein, this feature effortlessly enables “a rhythmic modularity characteristic of the most sophisticated Afro-Latin and jazz rhythms... [that] typically requires very sophisticated music-reading and performance skills to understand and execute.” Additionally, a user can rotate a sequence such that the perceived downbeat aligns with step 1 of the sequence if it is situated elsewhere on the loop.

4.2.4 Features Widespread Across Circular Rhythm Sequencers

Although Groove Pizzeria looks to Groove Pizza as its primary source of inspiration, certain aspects of its design inevitably recall features and designs of other existing circular step sequencers known and unknown to the author. Some of these features, such as clockwise sequencer motion and BPM control are widespread across circular rhythm sequencers and cannot be directly attributed to MusEDLab’s Groove Pizza. Other similarities are incidental and do not relate to the primary affordances of Groove Pizzeria for composing and comprehending polyrhythm and polymeter.

4.3 Design Characteristics Divergent from Groove Pizza

Groove Pizzeria inherits many of Groove Pizza's features while sporting a number of novel features and designs which section 4.3 provides a detailed discussion of.

4.3.1 The Second Loop

Perhaps the most obvious of Groove Pizzeria's deviations from Groove Pizza's design is the inclusion of a second circular sequencer interface. Although this may seem to be a simple addition, it is at the root of Groove Pizzeria's perceptual affordances for creating and perceiving polymeter and polyrhythm. With Groove Pizza, if a user wants to create a 4:3 polyrhythm, the only available option is to begin working with a loop of steps equal to a common multiple of 4 and 3; the practical and only option given Groove Pizza's 16 step ceiling being the lowest common multiple, 12. The user then has to divide the steps by both 4 and 3 for each loop, placing steps equidistantly in their respective rings. This process is simple enough, but what happens when the user would like to create a 5:4 polyrhythm? Given that the lowest common multiple of 5 and 4 is 20, this would be impossible to create on Groove Pizza due to its 16 step ceiling. Although the Groove Pizzeria prototype also has a loop maximum of 16 steps, because these loops are independent, one can set each to 5 and 4 steps respectively letting them run in parallel. This configuration produces the 5:4 polyrhythm (*Figure 8*); notably with less fussiness than would be required had the single loop contained the requisite 20 steps.

4.3.2 Loop Duration Control and Visualization

Returning to the previous example regarding composition of a 5:4 polyrhythm using Groove Pizzeria, one may notice that there was not explanation for how the respective 5 note and 4 note loops unfold over the same duration as required to produce the 5:4 polyrhythm. With Groove Pizza, adjusting the amount of steps in a loop does not change the duration between each step. Given the single circular sequencer interface of Groove

Pizza, this is intuitive. BPM directly corresponds to the duration between each step in the loop, therefore, a loop with a duration of 4 steps takes one step duration shorter to complete than a loop with a duration of 5 steps. In a hypothetical situation where Groove Pizza had two loops that play synchronously, if one loop had 4 steps and the other 5, the resulting output would be an example of polymeter rather than polyrhythm i.e. each step for each loop triggers at the same time, while the loops themselves are of different durations measured in units equal to the duration of each step.

The Groove Pizzeria prototype addresses the need for independently adjusting the duration of a given loop and the amount of steps within said loop with the introduction of two novel metaphors: loop circumference, and one the author has named “teeth” after the cut teeth of a gear¹⁰. Given a BPM, the distance between two teeth represents the duration of one sixteenth note. On Groove Pizzeria’s interface, the arc length between two teeth is always equal (110 pixels). Therefore, a user can adjust the overall duration of a loop in terms of sixteenth note durations, or teeth. As a user adjusts the amount of teeth in a given loop, the circumference of the loop’s circular interface proportionately grows or shrinks to accommodate the cumulative distances between each tooth of the loop. Returning to the 5:4 polyrhythm example, a user can create a set of 5 equidistant steps over the duration of 4 equidistant steps by setting the amount of slices on each loop to 5 and 4 respectively while setting the amount of time (in sixteenth note durations) of each loop to an equal number (*Figure 8*).

Rather than a 5:4 polyrhythm, say a user desires to create a polymetric pattern utilizing two loop durations of 5 and 4 sixteenth note steps each, for a combined pattern 20 sixteenth note long steps in duration; 20 being the lowest common multiple of 5 and 4. This is easily created with Groove Pizzeria by setting one loop to 5 steps and 5 sixteenth notes in duration and the other loop to 4 steps and 4 sixteenth notes in duration (*Figure 9*).

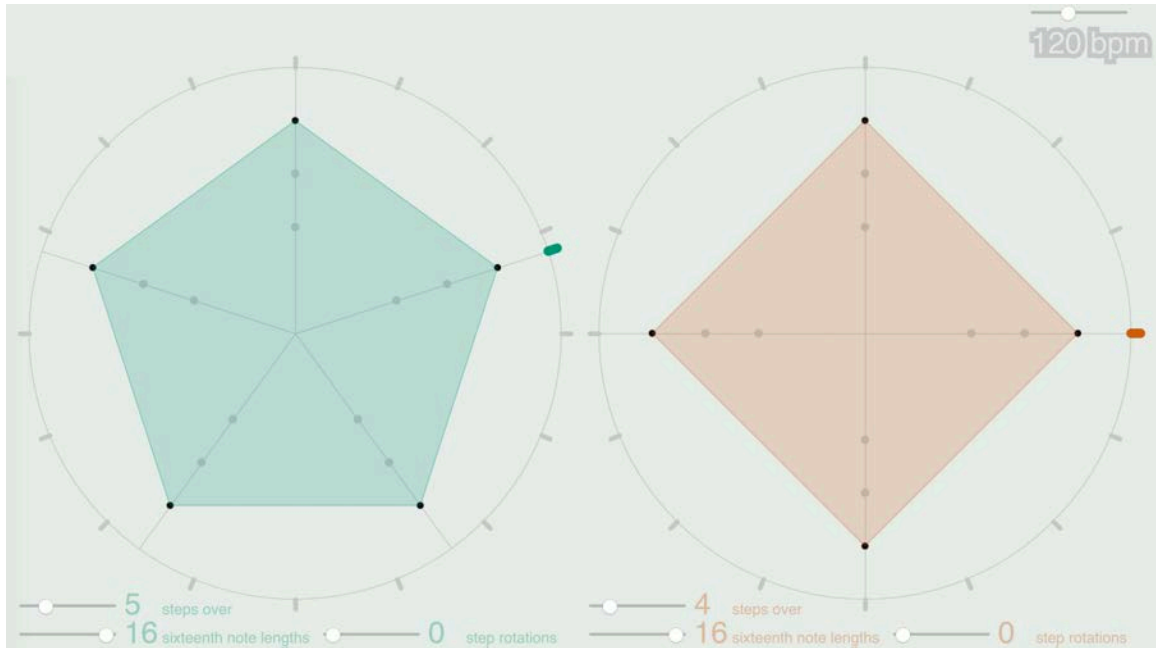


Figure 8: A 5:4 polyrhythm is created on Groove Pizzeria by setting the duration of each loop to the same number of sixteenth note lengths, here, 16 sixteenth note lengths, and dividing the loops by 5 and 4 respectively.

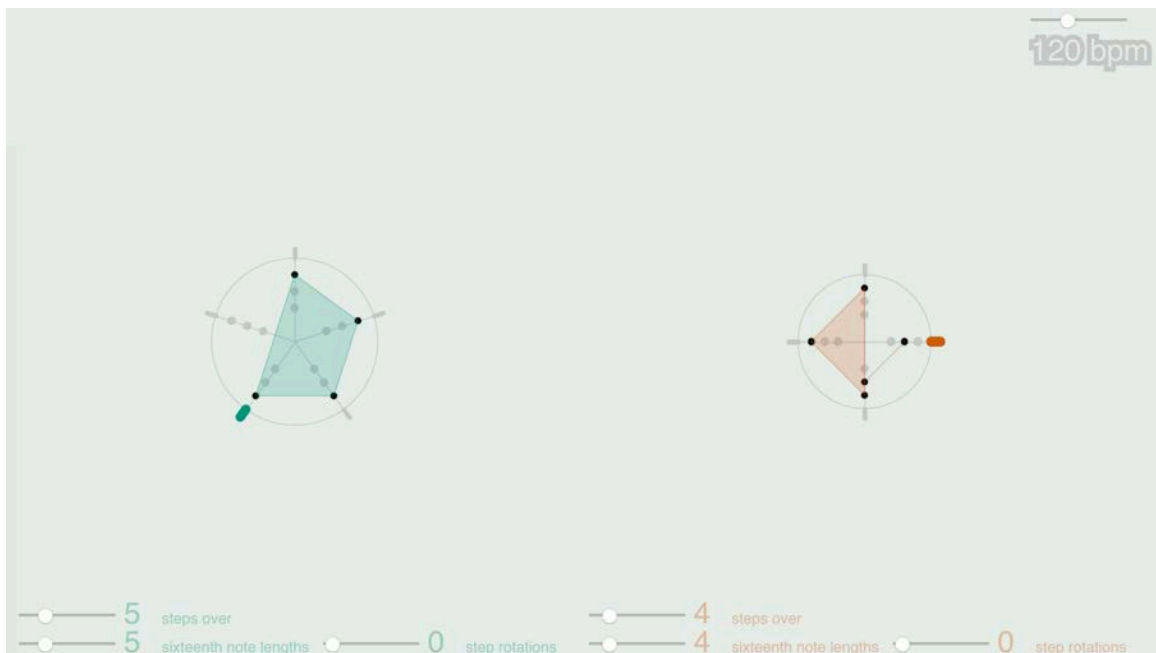


Figure 9: A polymetric pattern is created on Groove Pizzeria by creating one loop that is 5 sixteenth notes in duration and another loop that is 4 sixteenth notes in duration.

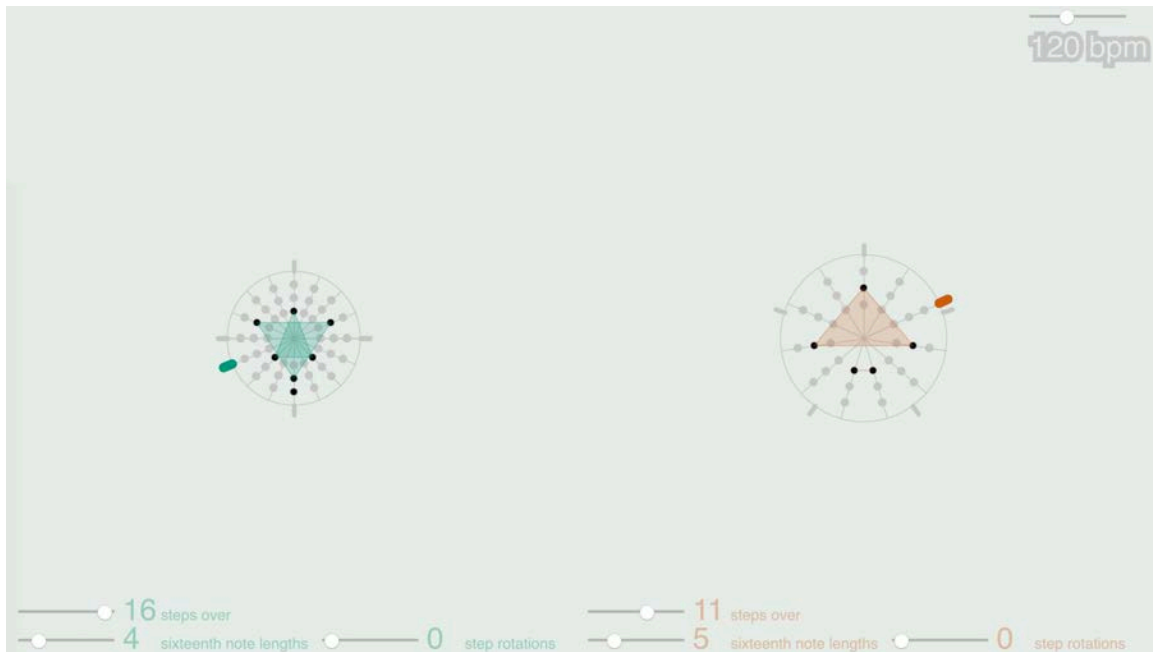


Figure 10: A complex pattern is created on Groove Pizzeria by setting loop durations to different amounts of sixteenth note units and the divisions of these loops such that they do not play concurrently or lock to an invisible grid.

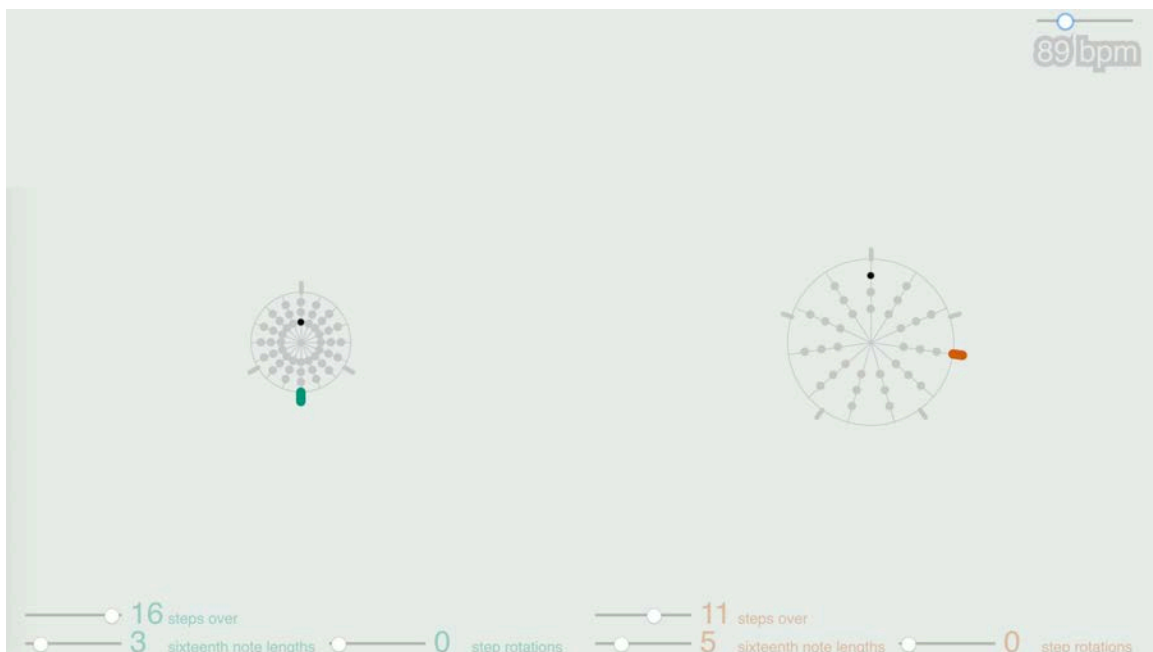


Figure 11: Although the loop durations and step-counts are set to complex values, because only one note is entered on the downbeat of each loop, the resulting pattern is simply perceived as a 5:3 polyrhythm.

Interesting rhythmic combinations that fall outside the reduced definitions of polyrhythm and polymeter (provided on p. 12) occur when a user composes patterns comprised of different loop durations where individual steps do not occur in unison. These combinations can be staggeringly intricate (*Figure 10*), or, depending on the actual steps selected in a given situation, can ultimately be a relatively simple pattern in a complex frame (*Figure 11*). The Groove Pizzeria prototype was designed such that once a user learns the basic methods required to create polyrhythm and polymeter, they can begin to explore beyond these techniques, creating rhythms that would be impossible to create in most circular rhythm sequencers and borderline impossible (due to issues of tediousness) to create in a DAW.

4.3.3 Wide BPM Range

A less significant, but nonetheless novel, feature of Groove Pizzeria is the ability to set the instrument to a wider range of BPM than most circular step sequencers. The BPM slider, which controls the tempo of both loops, can be set to any integer value between 20 and 300 BPM. This was designed to allow a user to allow a user to make very small divisions (such as 2 sixteenth note long loops divided by 16 steps) sound audibly like a drum pattern at very low tempos. A user can also increase the tempo to high values resulting in fusing between drum sounds that can be used to explore harmonic polyrhythms as they change from drum patterns into pitched harmonies at very high tempos; a technique useful for music education settings and experimental electronic/dance genres such as the bizarrely high tempo music of the “extratone” genre (An Introduction to Extratone, 2018).

4.4 Development Process

After conducting research corresponding to the “empathize” stage of the design thinking process (reflected in the introduction and literature review of this thesis), it was time to move onto the “define” stage which corresponds to the “define the activity” stage of the SoDaR process. In this stage, it was determined that requiring a user to compose drum

patterns reflecting an understanding of polyrhythm and polymeter on a novel software interface was a sufficient learning objective for the SoDaR process framing this thesis.

With this learning objective clearly defined, the “ideate” and “prototype” stages of the design thinking process, corresponding to the “software design and production” stage of the SoDaR process, began. A paper sketch (*Figure 12*) for the Groove Pizzeria software was drafted and presented to music technology experts including Ethan Hein, inventor of Groove Pizza and Alex Ruthmann, Director & Lead Research of MusEDLab (also, the advisor of the thesis). Following initial positive feedback from the experts, a hybrid ideation and prototyping process was commenced. Throughout the development of Groove Pizzeria, numerous challenges emerged and encouraged productive design decisions, the most significant of which will be described in the subsequent paragraphs of section 4.4.

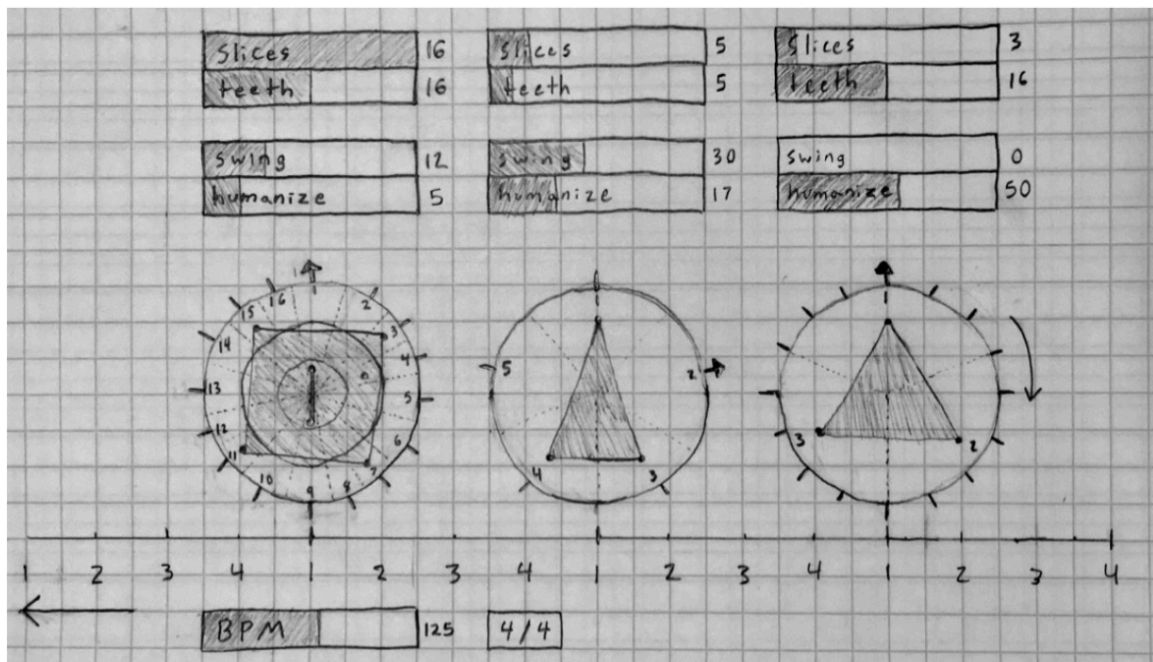


Figure 12: Groove Pizzeria “blueprint” sketch demonstrating core design features such as multiple circular loops and “teeth.” The sketch also features multiple design elements that were scrapped in Groove Pizzeria prototypes including the “conveyor belt.”

4.4.1 Fundamental Prototype Decisions

Development of Groove Pizzeria began in the visual programming language Max with the intention of creating a prototype that could interface directly with Ableton Live. However, it quickly became clear that using Max would limit the possibilities of the interface's visual design; an unacceptable situation given the goal of iteratively designing a user interface with perceptual affordances for composing and perceiving polyrhythm and polymeter. As a result, development of the user interface switched to JavaScript and made heavy use of the Web Audio API and p5.js which was selected due to its open ended visual capabilities, the quality of its documentation, and the large amount of learning resources available for its users. Because most common browsers implement the Web Audio API differently and these implementations are constantly in flux, for the sake of focusing on interface design, the Groove Pizzeria prototype was only optimized to run in Mozilla Firefox.

4.4.2 Basic GUI Design Choices

The first challenge in developing Groove Pizzeria was translating the paper sketch of the software into a static graphical user interface (GUI); a process that required trial and error as decisions were made to maximize the legibility of the interface. At this stage, much inspiration was taken from Edward R. Tufte's *Envisioning Information*, an intricate rumination on articulating complex information through visualizations. A fundamental decision to make was the use of color in the interface. Tufte recommends a strategy of using "colors found in nature, especially those on the lighter side, such as blues, yellows, and grays of sky and shadow... Local emphasis is then given by means of spot highlights of strong color woven through the serene background (2001, p. 90)." Following this advice, a light grey with a hint of green was selected as a background color. Early static interfaces with green chart elements were replaced by those with a darker, but low contrast shade of grey to allow the dark black circles of "on notes" and transparent shapes of light teal and orange to stand out vividly without contributing to visual clutter.

Additionally, the perimeter of the circular interface and its radial divisions are drawn in delicate grey lines to promote accurate reading: “dark grid lines are chartjunk (2001, p. 59).” Tufte’s discussion regarding the avoidance of net-like gridlines lead to a reconsidering of Groove Pizza’s enclosure of steps between radial slices (2001, p. 55). Shifting steps onto the radial lines themselves significantly reduces visual clutter and aids legibility. It has the additional consequence of clearly denoting steps that are directly opposite one another when the amount of steps is set to an even number. Altogether, these graphic design choices enhance the core perceptual affordances of the Groove Pizzeria design (*Figure 13*). However, it is worth noting that Groove Pizzeria and Groove Pizza have different target audiences and the bright blocks of color in the MusED design are likely more appealing to children than the dulled colors of Groove Pizzeria (Pancare, 2018).

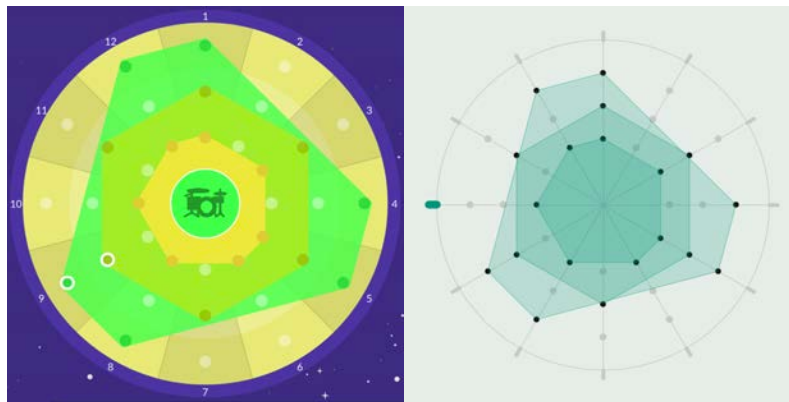


Figure 13: A bembe rhythm on both Groove Pizza and Groove Pizzeria for side-by-side comparison.

4.4.3 Loop Duration Decisions

In the paper sketches for Groove Pizzeria, it can be observed that each circular interface is situated on a repeating musical timeline (*Figure 12*). This timeline was supposed to move sideways to the left, like a conveyor belt, rotating the teeth surrounding each loop in a clockwise direction. In theory, each tooth surrounding a circular interface would rotate in time to meet the current 16th note represented on the timeline, with each circular

loop situated within the same measure, represented by serial repetitions of the measure on the conveyor belt. The timeline would move to the left, with each 16th note notch on the belt progressing at the rate of a set BPM. One of the teeth surrounding each circular loop, denoted with an arrow shape, was to represent a “playhead,” and as this tooth moved over an “on” note, it would play. Overall, this design was meant to provide an interface metaphor that allowed a user to understand how each loop (and the polyrhythm and polymeter they would create in interaction with each other) are subject to the same underlying tempo, represented by the conveyor belt. While developing these features, it became clear without the need for testing that they were unusable because of the high rate at which the teeth of each loop had to move around the circumference of a loop at most musical BPMs. Smooth motion of teeth around each loop came at the expense of blurring due to the limitations of the screen’s frame rate. Worse, this design defeated its own purpose by making it very difficult to perceive the amount of teeth, and therefore, the duration, of a given loop.

Because of these issues, the “conveyor belt” metaphor was scrapped in favor of a static set of teeth accompanied by rotating playheads indicated by a thicker width and a vivid opaque teal and orange. This design has an advantage over the rotating teeth by clearly displaying the amount of teeth, and therefore the duration, of a given loop to a user. However, because of the difficulty humans have subitizing groups of objects over 6 in count (Kaufman et al., 1949) (i.e. immediately judging the count of a set of objects), having the teeth remain static did not do enough to legibly denote the duration of a given loop to a user. To facilitate this process, it was decided that the arc length between teeth remain constant, adding up to a circumference that represents the duration of a given circular loop. This allows a user to judge both the number of teeth and the circumference of a given loop in tandem. Additionally, a number representing the duration in sixteenth note lengths of each loop was added next to the slider controlling this value.

Causing the circumference of each loop interface to shrink as the overall duration of the loop decreased lead to a design compromise. When a loop is set to the lowest possible duration (2 sixteenth note lengths), the buttons of the three concentric rings slightly

overlap. Additionally, when at this loop duration, if there are ten or more steps in a loop, the buttons of adjacent steps overlap. Attempts to reduce button size or increase the distance between teeth to alleviate this problem resulted in either unusably small buttons or an inability to fit two whole circular sequencer interfaces on the screen at one time. Also, perhaps fittingly, the triggering of drum samples at this rate generates an output where the individual drum samples are fused in a process that can be described as granular synthesis. Because of these reasons, it was decided that overlapping at this level was acceptable due to the extremely rapid succession of steps in this use case, a situation uncommon in most musical settings.

Initial designs of the playhead allowed it to move seamlessly around the circumference of each loop. But, because of blurring, juddering, and lagging behind audio output, a compromise was made where the playhead rests on each step of a loop as it is triggered rather than moving fluidly. A negative consequence of this compromise is that loops with equal durations and different step-counts may appear to be out of sync when they are in fact perfectly in sync (*Figure 14*). However, when two loops of equal duration reach their respective first steps (90 degrees or “12 o’clock” on the loop), the playheads of each loop will be in the same position regardless of the amount of steps in each loop. Additionally, when two loops of unequal duration both reach step 1, for example when a loop of 4 steps and a loop of 5 steps reach step 1 after twenty overall sixteenth note durations, the playhead will appear at step 1 for both loops. This is reinforced by a “blinking” of both the first slice line in each loop and the text denoting the BPM (*Figure 14*).

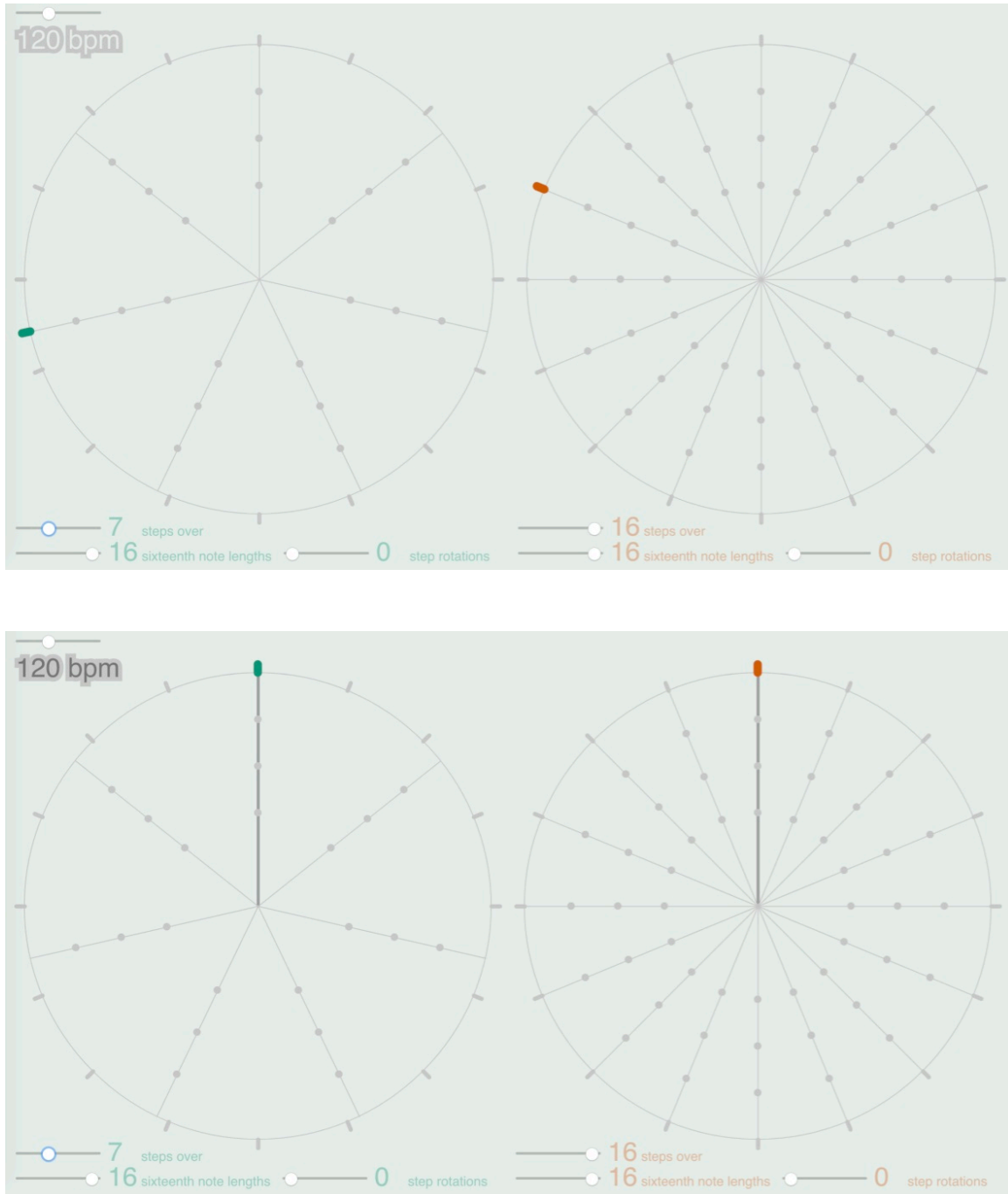


Figure 14: This top image shows two equal duration loops that appear to be out of sync when they are in fact perfectly synchronized. The bottom image demonstrates how each loop is visually synchronized when the loops return to step 1.

4.4.4 Decision to Schedule Audio with High Accuracy

When undergoing the SoDaR process, Brown (2007) recommends only proceeding with software development as far as necessary to enable the learning activity defined in step one. This keeps time and emotional investment in the prototype low enough to discard or overhaul the software design given the results from product tests. Other product design methodologies, such as Lean UX, call for spending the least amount of effort possible on a prototype; making use of paper prototypes or wire-framing software to save time and money (Gothelf & Seiden, 2016). The complexity of the first Groove Pizzeria prototype was kept low by focusing only on features and design salient to composing and comprehending polyrhythm and polymeter. Another strategy that was attempted was utilizing somewhat inefficient code that simply “got the job done.” However, the unfinished first prototype suffered from such distractingly inaccurate timing due to a reliance on the JavaScript “setInterval()” method for drum sample playback that it was clear a user would not be able to hear polyrhythm or polymeter through the distracting variability of the prototype's timing. Because of this, Chris Wilson’s strategy for achieving sample accurate timing by creating a partnership between the Web Audio and JavaScript clocks, famously described in his article, “A Tale of Two Clocks - Scheduling Web Audio with Precision (2013),” was implemented in the Groove Pizzeria prototype.

4.4.5 Terminology Decisions

Finally, a seemingly trivial but challenging design problem was the naming of specific design metaphors on the Groove Pizzeria interface. On Groove Pizza, the circular interface is divided into “slices.” Although this choice cleverly and lucidly refers to the visual metaphor of the circular interface as a pizza, it could easily be replaced with the word “steps,” referencing the Groove Pizza’s linear step sequencing cousins. Jakob Nielsen’s influential “10 Usability Heuristics for User Interface Design”¹¹ suggest that there should be a match between the system and the real world, “[speaking] the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural

and logical order (1994).” Based on this heuristic, the choice was made to change the name “slices” to “steps” on the Groove Pizzeria prototype to make the feature familiar to producers of electronic/dance music. Even thornier was the naming of units denoting the overall duration of loop. Although the verbose “sixteenth note lengths” is confusing, especially when juxtaposed with a count e.g. “15 sixteenth note lengths,” it was chosen over the name of its real-world inspiration “teeth” because of its descriptiveness. Additionally, if each loop is thought of as a pizza, depending on the mindset of the individual using the software, a set of teeth lining the outside of a pizza is either hunger or horror inducing. So, much like the change from “slices” to “steps,” the “pizza” became referred to as the “loop;” harkening back to Groove Pizza’s original name: “the Drum Loop” (Hein, 2013).

4.5 Product Test 1 Method

Testing the first Groove Pizzeria prototype with each user was a multistep process that typically took between 30 minutes to an hour. To begin, a user was asked a set of questions relating to their experiences producing electronic music and their understanding of polyrhythm and polymeter. Next, each user was asked to play with the prototype and compose drum patterns utilizing polyrhythm and polymeter given a verbal definition of the terms. During this stage, they were asked to “think-aloud” and their reactions and conversation were used to answer questions relating to the prototype’s fulfillment of the three aspects of emotional design (visceral, behavioral, reflective). If any questions were unanswered, they were addressed in a post-test conversation. The following is an outline of the test procedure:

1. Setup:

- Prepare screen recording and microphone to record conversation for subsequent content analysis.

2. Initial Interview:

- Briefly introduce Groove Pizzeria and intention of study (IRB).
- Ask the following questions:

- Do you ever use software or physical drum machines or sequencers?
- How familiar are you with the term polyrhythm?
- How familiar are you with the term polymeter?
- Do you ever deliberately use these techniques in your music?

3. Guided Play

- Throughout guided play step, ask the user to think-aloud.
- Ask the user to create a drum pattern they find interesting.
- After this, verbally provide a definition of polyrhythm and ask the user to create a drum pattern utilizing the technique.
- After this, verbally provide a definition of polymeter and ask the user to create a drum pattern utilizing the technique.
- Throughout this process, listen to the user's reactions and attempt to answer the following questions. If any questions are not answered through the user's conversation, ask them those questions following the test:

Visceral:

- Do you find the visual design appealing?
 - What about it? What would you change?
- Do you find the sound output appealing?
 - What about it? What would you change?

Behavioral:

- Do you feel that this software allows you to create rhythms that would be difficult or impossible to make with your current work flow?
- Do you find the application to be intuitive and familiar to use?
- What aspects of the interface do you wish would be improved or explained more clearly?

Reflective:

- Did you enjoy the process of testing the prototype?
- Would you deliberately attempt to use polyrhythm or polymeter in your music after this experience or listen for it in other people's music?
- Were there moments playing with the sequencer that you would be interesting utilizing in a piece of music or writing music around?

4.6 Product Test 1 Results

Section 4.6 provides a content analysis of media resulting from the first set of product tests. The analysis is provided in a deductive format, organized according to the visceral, behavioral, and reflective levels of emotional design. Within each level, subcategories of interest to Groove Pizzeria's design are organized inductively. A section relating to the demographics of users chosen for the tests is also included.

4.6.1 User Demographics

This set of product tests was conducted with six users. Each user could be described as an amateur musician, having spent many years learning, playing, and writing music. Five of the six users actively compose electronic music using a combination of hardware and software instruments. Every user has created music utilizing a DAW.

Out of the six users, five frequently use some form of step sequencing. Half the users were able to accurately define polyrhythm and one third were able to accurately define polymeter. After giving a cursory definition of polyrhythm and polymeter, half of the users reported deliberately using both techniques in their music despite their familiarity with the term or lack thereof. Choosing a set of users including both those who were familiar and those who were unfamiliar with these terms was essential for verifying that Groove Pizzeria prototype is usable for electronic/dance music producers with a range of music theory expertise.

4.6.2 Visceral Level

User reactions to Groove Pizzeria at the visceral level were generally positive. Nearly every user verbally expressed appreciation for the visual aesthetic of the prototype, making comments about its "clean," "legible," or "functional" appearance. One user commented that a "3D" interface like that of popular soft-synth Massive would make it appear more "finished." Reactions to the auditory output of Groove Pizzeria were also

generally positive at the visceral level with users making comments about its output sounding “cool and weird,” “interesting,” or “like a footwork track.” Almost every user asked if it was possible to swap out drum samples used. Although this may initially seem to be an irrelevant reaction to the lack of a common feature in contemporary drum machines and samplers, it may also suggest that users were interested in hearing Groove Pizzeria’s auditory output utilizing samples other than the ones provided in the prototype. Overall, the data collected from the first user test supports that the initial Groove Pizzeria prototype appealed to users at the visceral level.

4.6.3 Behavioral Level

Results at the behavioral level, where usability testing and iterative design are typically focused, were mixed. This, of course, is a good thing considering the purpose of product testing is to unveil usability issues with a given design for subsequent improvement.

One clear area of success at the behavioral level was establishing the functional affordances of Groove Pizzeria for composition of polyrhythm and polymeter compared to other electronic music production tools. Every user reported that the tool allowed them to create rhythms that would be very cumbersome or impossible for them to create on the existing hardware and software they possessed or were aware of. Despite success of this fundamental value proposition to Groove Pizzeria, it was quickly apparent that the initial prototype was not immediately intuitive.

Perhaps unsurprisingly, the greatest area of confusion in product testing related to loop duration; represented through the novel interface metaphors of loop circumference and number of teeth (as in a gear); controlled by a slider labeled with a numerical value followed by “sixteenth note lengths.” A number of reactions to this feature demonstrated that users did not intuitively grasp its function. Two users confused the terms “steps” and “sixteenth note lengths” with one another at multiple points during the product test. Because the number of steps and the number of sixteenth note lengths are situated on top of one another in the initial Groove Pizzeria prototype, one user revealed he was

incorrectly understanding the two numbers to represent a type of time signature. One user said they found it unintuitive that the “playhead” of a loop speeds up when the duration of a loop is shortened, demonstrating that it was not entirely clear that the playheads of both loops move at the same tempo but complete shorter loops more quickly. Although it is not proven, this reaction can perhaps be explained as a violation of conceptual metaphor i.e. a user may subconsciously associate leftward motion of a slider with a slowing of tempo, the resultant increase in the rate of loop completion causing disagreement with user expectation (Holland, Mulholland, & Wilkie, 2013).

Throughout product tests, users began to understand the interface metaphors of Groove Pizzeria, but often had questions regarding the music theory underpinning the software’s design. This experience led to users making statements such as, “your design makes sense, the concept itself is just confusing.” Generally speaking, a user’s self-blame is not a good sign for a product’s usability (Hanington & Martin, 2012). However, a designer of digital musical instruments must consider the implications of usability testing for software that is not “used” as much as it is played. According to Bouwer, Gifford, McDermott, & Wagdy (2013), “engaging, consuming, flow-like activities such as music are characterized by being at an appropriate level of difficulty: not too difficult, and not too easy.” Bouwer et. al, recognize traits of music interaction that must be difficult (open-endedness and long-term engagement; expressivity, creativity and flexibility; demonstrating virtuosity; and communicating effort and emotion) and traits of music interaction that could be made easier (transient and frivolous music, peripheral and technical tasks, learning humps and layered affordance, and instruction and meta-cognition). When discussing the usability of the Groove Pizzeria prototype, it makes sense to consider which traits of music interaction are made easier and which difficult aspects are emphasized. The behavioral component of this product test suggests that peripheral and technical tasks associated with the composition of polyrhythm and polymeter are made easier with Groove Pizzeria. Conceptually difficult aspects of Groove Pizzeria’s design may be associated with open-endedness and long-term engagement or expressivity, creativity and flexibility. This may result in a Groove

Pizzeria user without prior knowledge of polyrhythm and polymeter not be alienated by the interface but instead finding its controls to be intriguingly mysterious, promoting an open ended and long-term exploration of its possibilities. This same category of user, and even users with an understanding of polyrhythm and polymeter, may not find it easy to utilize the techniques creatively. However, this challenge presents an opportunity for a user willing to take the time to learn the interface to develop a unique “voice” through the instrument.

4.6.3 Reflective Level

At the reflective level of emotional design, a user internalizes what they have experienced at the behavioral and visceral level and makes determinations about their relationship with a given product. To use a musical metaphor, the reflective level of emotional design goes beyond aesthetics and usability and is concerned with the greater resonance between a product and user. Because musical instruments are used in a variety of ways, success at the reflective level can come in many forms. In the case of Groove Pizzeria, it was determined that success at this level could be considered as user acceptance or enjoyment of the prototype as a frivolous musical experience, a learning experience, or a musical instrument/production tool.

Determining whether the initial Groove Pizzeria prototype was successful as a frivolous music experience involved testing whether or not a user had fun playing with it. In tests, most users took charge and experimented with the prototype past what was asked of them, seeming to indicate that they enjoyed themselves. However, all but one user predominantly interacted with the prototype quite deliberately; adding in single notes to a sequence at a given time and waiting to hear their effect, later setting out to create specific rhythms. This goes against the author’s expectation that users would experiment more fluidly, dragging their cursor to enter drum hits haphazardly. Only one user interacted with the prototype this way (interestingly the user with the least DAW experience) and seemed to enjoy their experience perhaps more than any other user, even earnestly asking at one point to take a screen shot to remember the configuration of

rhythms on the prototype so the author “could make a beat with it later.” As a whole, determining whether the prototype was successful at the reflective level in terms of being a frivolous music experience is something that could use subsequent testing due to the fact that the test methods themselves and perhaps the researcher’s physical presence may have somewhat discouraged free play.

Another category that the Groove Pizzeria prototype was tested for at the reflective level was success as a learning tool. Testing for this is related to but different than testing whether or not a user was able to learn about polyrhythm or polymeter by interacting with the prototype; a behavioral level concern. Rather, Groove Pizzeria’s success as a learning tool at the reflective level can be evaluated by determining whether or not the user found worth in a learning experience they had using the prototype or considered their learning experience to be fun in its own right, perhaps being described as “edutainment.” Most users verbally reported that they had learned something new about polyrhythm and polymeter and had enjoyed themselves, saying phrases like, “that was an interesting lesson.” Three of the users reported that this experience made them reconsider using polyrhythm and polymeter in their own musical compositions noting that it helped them to think about rhythm beyond the visual restrictions of their DAWs of choice.

Lastly, the Groove Pizzeria prototype was tested for its success at the reflective level as a production tool/musical instrument. This test primarily involved asking the user whether or not they would enjoy using the tool in this capacity. Perhaps because of the relative seriousness music performance and production has in the lives of the users selected for the first round of testing, this question spurred on many a “yes” followed by suggestions regarding usability concerns. For the most part, these suggestions were for additions of standard features in any software sequencer. As was mentioned in the visceral design section, these usability concerns, indicated by common questions such as “would this be available as a VST,” seem to indicate that the core functionality of the interface was of interest to users. One user commented that Groove Pizzeria would be “cool” for someone who makes “music that’s more interesting than mine.” Because Groove Pizzeria is meant to help electronic/dance producers of all levels escape from the rhythmic presets of

DAWs, it is presented with the inherent challenge of proving to a user that this escape is worth doing in the first place.

5. Second Prototype

Following testing of the initial prototype and performing content analysis on the resultant video footage, a second prototype was created. This chapter begins by providing an overview of the prototype. This will lead into detailed discussion regarding salient feature additions and revisions. Unlike Chapter 5, discussion of the second development process will be woven into the aforementioned section because of the relatively lower complexity level developing the second prototype. This chapter will end with an overview of the second prototype product test method.

5.1 Prototype Overview



Figure 15: Interface of the second Groove Pizzeria prototype

The second Groove Pizzeria prototype (Figure 15) revises the loop duration display from “sixteenth note lengths” to “time units” followed by the time unit’s value: the duration of one sixteenth note at the given BPM. Display of step-count is now followed by the duration of each step as a fraction of a whole note (e.g. 1/16 note). The loop duration slider and display is now positioned above the step-count slider and display; the two

spatially divided by a small obelus. To the right of the step-count slider and displays are two new displays of the ratios between the duration of each loop's individual steps. At the top left of the screen is a new data visualization (the time line) demonstrating the amount of cycles it takes each loop to reach synchronization at step 1 (the combined pattern). Each row of the timeline is composed of small ovular shapes representing the time units of each loop. The first time unit of each loop is displayed in a slightly darker hue. As each loop completes a cycle, a black ovular shape (the counter) appears on the timeline, tracking the progress of the loops as they move through the combined pattern. Beneath each row of the timeline, a display reads the number of cycles it takes for each loop to synchronize at step 1 followed by the duration of each loop's cycle in seconds. To the right of the timeline, a display reads the duration of the combined pattern in total time units and seconds. Small changes were made to font and color scheme to make the prototype more legible and relaxing to the eyes.

The second major component of the second Groove Pizzeria prototype is a paper introductory tutorial composed of the following sections:

1. Intro to Circular Rhythm Sequencing
2. Polyrhythm
3. Polymeter
4. Free Play

5.2 Feature Additions and Revisions

Product testing of the first Groove Pizzeria prototype unveiled usability problems at the behavioral level of emotional design in terms of a user's ability to understand how loop duration and step-count interact to allow the composition of polyrhythm and polymeter. These usability problems seemed to have reflective level consequences, resulting in the the first Groove Pizzeria prototype testing well as a trivial musical activity, but not as

strongly as a standalone learning tool or musical instrument. Because of these shortcomings, it was determined that the second Groove Pizzeria prototype should present more information in its interface while including a guide to help users interpret this information.

5.2.1 Revisions to Step-Count and Loop Duration Controls and Displays

To alleviate confusion related to the controls and displays for step-count and loop duration, a number of revisions were made. The slider for loop duration previously labeled, “sixteenth note lengths,” was changed to “time units” followed by the duration of one sixteenth note at the given BPM. The wording of “sixteenth note lengths” in the first prototype seemed to be a red herring, causing users to believe the control related to step-count or other musical parameters unincorporated in the prototypes controls. The new labeling draws attention to the fact that the control directly relates to the duration of each loop, even providing the exact unit of time to clarify that changing the BPM is the only way to manipulate this parameter of the interface (*Figure 16*).

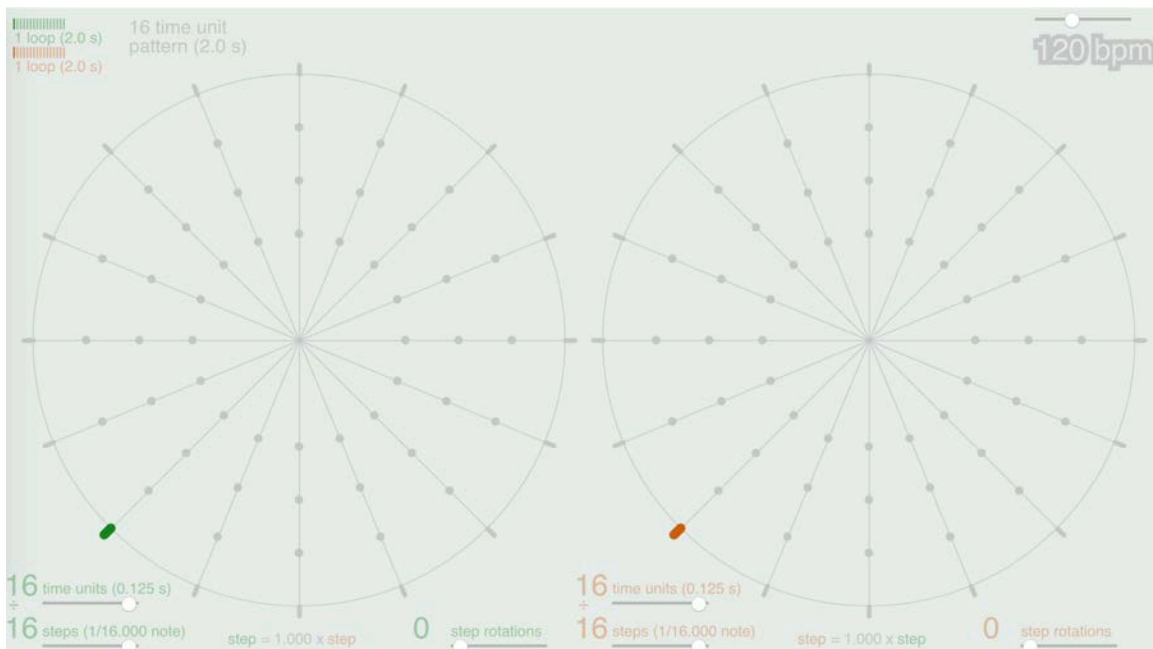


Figure 16: At 120 BPM, the value of a time unit (one sixteenth note duration) is 0.125 seconds. Adjusting BPM is the only way to manipulate the time unit value.

The control and display of step-count for each loop was also revised. Rather than simply stating the amount of steps in each loop, the display includes the duration of each step as a fraction of a common time (4/4) whole note e.g. “5 steps (1/16 note)”. Determining how or whether or not to label the step-count control and display was a decision that required compromise. The first prototype’s lack of detailed labeling lead to many users guessing or calculating the type of notes they were hearing during tests i.e. “is this loop divided into 128th notes or is it something more confusing than that.” Initial revisions to the first prototype included the duration of each step in seconds, mirroring that of the loop duration control’s time unit system. User testing was not required to grasp that this design choice would exacerbate confusion between the controls and was only informative in that it allows a user to discern if the duration of a time unit and a step are the same or different; redundant information due to the spacial comparisons afforded by the “teeth” and “slices” metaphors of the loop display. Therefore, it was decided to display the duration of each step as a fraction of a whole note in common time. Early redesigns that included a fraction with no labeling (e.g. “12 steps (1/16)”) were uninformative and opaque. Other attempts to clearly and accurately display the information were overly verbose (e.g. “6 steps (1/128 of a whole note)”). Ultimately, it was decided to include the word “note” after the fraction for its economy and its accordence with the diction of users in product tests who were attempting to comprehend the duration of steps. The use of “note” is a compromise because although it accurately identifies numerous step durations between Groove Pizzeria’s 128th note to half note range, it also describes many in a style contrary to music theory convention such the “eighth note triplet,” a “1/12 note” on Groove Pizzeria. Even more alarming, step durations of non-integer whole note division are possible with Groove Pizzeria; take the “1/9.143 note” for example. Although this labeling is probably against most conventions, sidestepping the vocabulary of music theory allows the Groove Pizzeria to communicate atypical note durations with lucidity.

In the second Groove Pizzeria prototype, the loop duration control is now situated atop the step-count control with an obelus positioned between the numerical display for the

controls. This is meant to mitigate the impression that the numerical displays for these controls represent time signature and more clearly articulate that the prototype allows a user to divide a length of time into a number of steps.

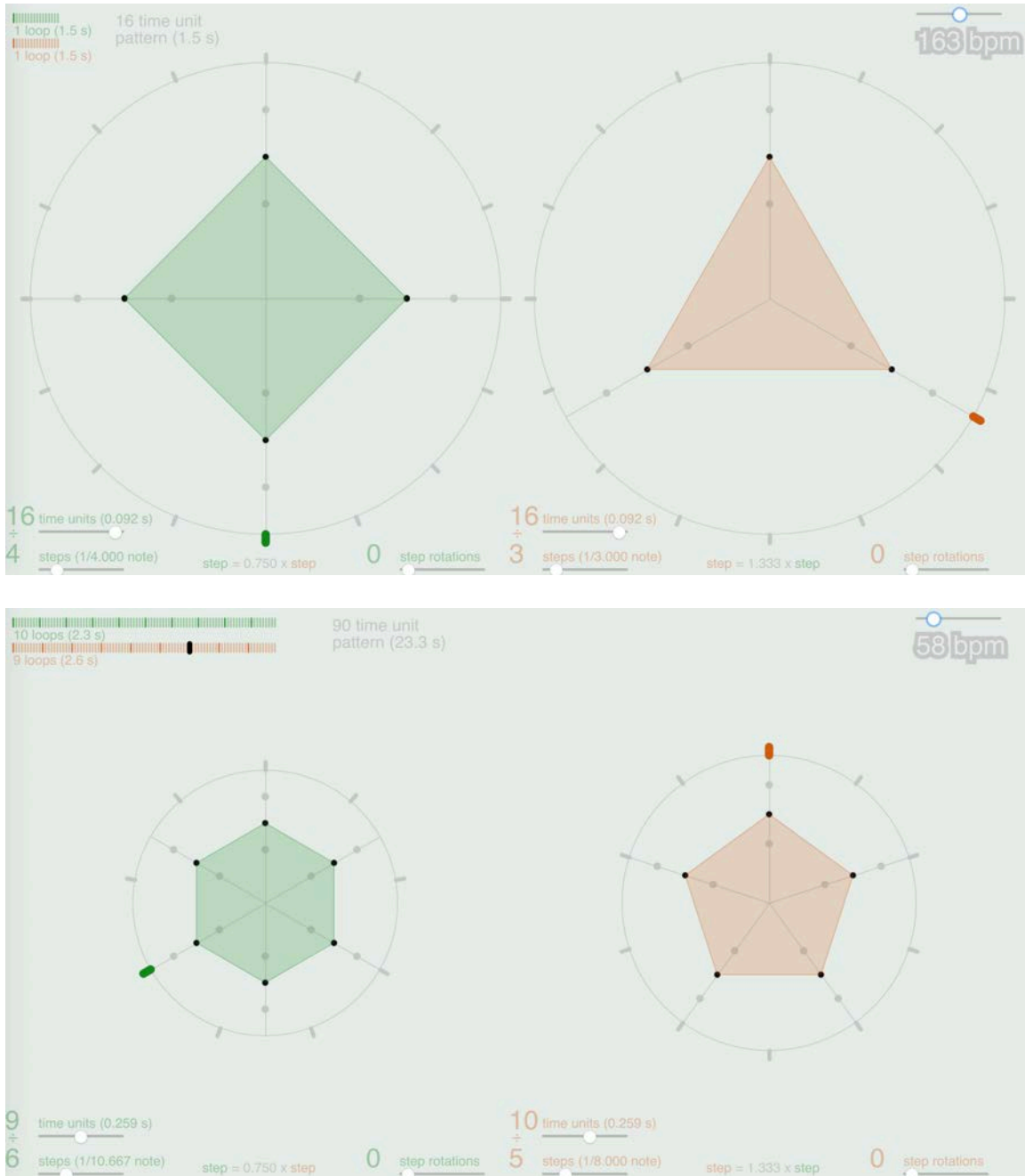


Figure 17: On the top is a 4:3 polyrhythm that is easy to comprehend due to the fact that each loop has the same duration and is divided by 4 and 3 steps respectively. On the bottom, a more difficult to comprehend 4:3 polymeter is made clear through two step-duration-ratio displays.

5.2.2 Step-Duration-Ratio Displays

To the right of each step-count display and control is a new display that represents the ratio between the step durations of each loop to one another. This feature is meant to clarify the relationship between step durations of each loop when the overall duration of each loop itself is different. For example, perhaps the easiest way to create a 4:3 polyrhythm using Groove Pizzeria is to set the loop duration to an equal number of time units, 16 for example, set the amount of steps dividing each loop by 4 and 3, and finally engaging an entire ring of steps of a particular voice for each loop (*Figure 17*). However, a user may come to this same polyrhythm accidentally, for example through a configuration of a loop 9 time units in duration divided by 6 steps and another loop 10 time units in duration and divided by 5 steps (*Figure 17*). Without using mental math, the step-duration-ratio display demonstrates the same relationship between the amount of steps on both the simple and complex configurations: the step durations of the green loop are .75 times the duration of steps on the orange loop, and reciprocally, the step durations of the orange loop are 1.333 times the duration of steps on the green loop. With a bit of background information about polyrhythms, a user can quickly understand that these values represent the ratio between the note lengths of a 4:3 polyrhythm.

5.2.3 Timeline

At the top left of the second Groove Pizzeria prototype's interface is a new feature named the "timeline." The timeline represents the complete rhythmic pattern length by demonstrating how the different loop durations interact over time, both eventually returning to step 1. A black ovular counter follows the progress of the loops as they progress through a complete pattern. This feature is meant to aid users in understanding how loops of different durations interact and the possibilities of polymeter, both points of confusion in product tests. Take for example the polymetric pattern shown below (*Figure 18*). With a green loop 14 time units in duration and an orange loop 16 time units in duration, it takes 112 time units in total for each loop to return to step 1 at the same time;

112 being the lowest common multiple of 14 and 16. The timeline is also labeled to show the duration of each loop in seconds (e.g. 1.8 and 2 seconds respectively) and the amount of cycles it takes for each loop to return to synchronicity at step 1 (e.g. 8 and 7 respectively).

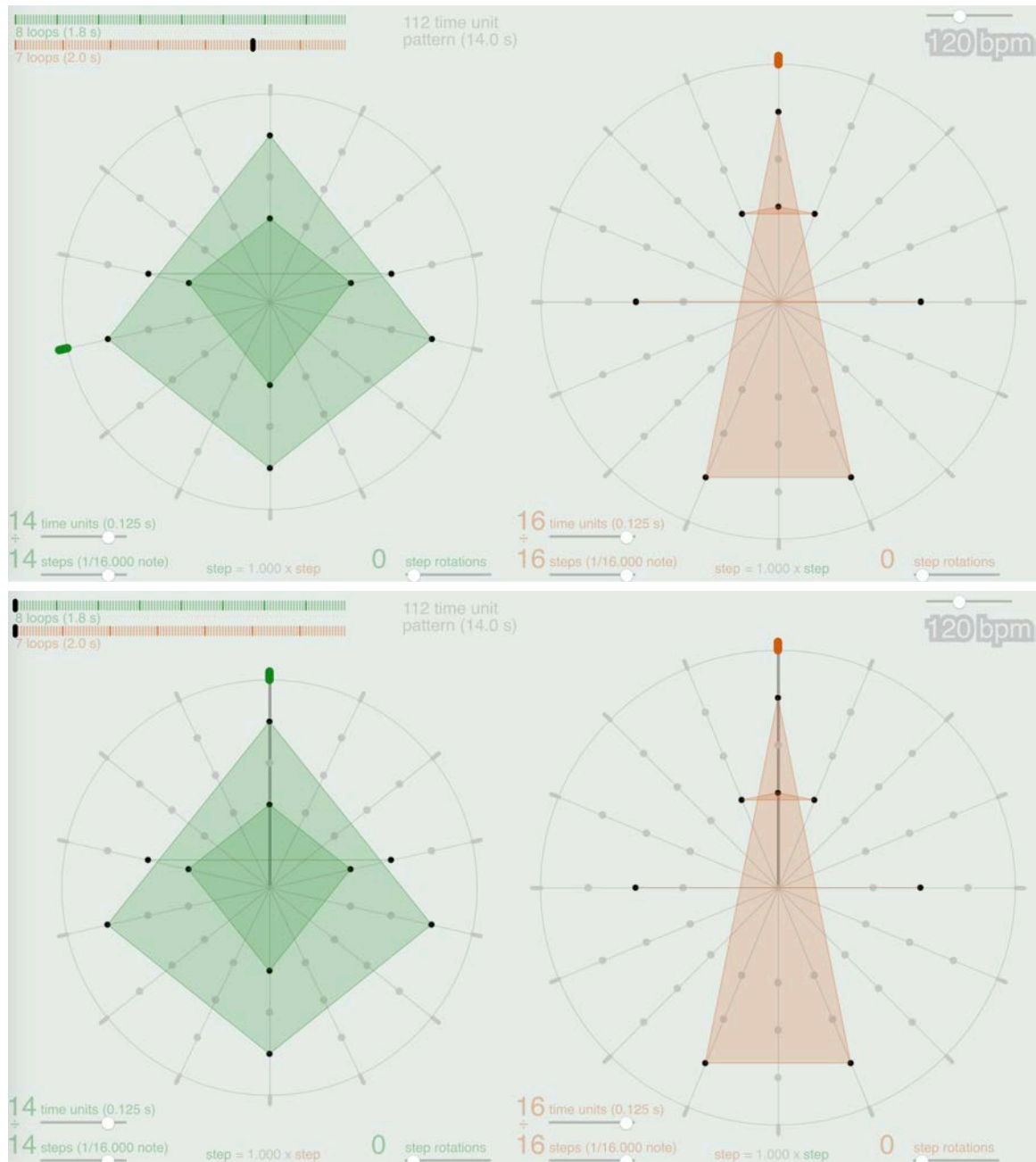


Figure 18: In the top image, the timeline counter shows that the orange loop of this polyrhythmic pattern is at step 1 on the sixth loop of the overall pattern created by the green and orange loop. In the bottom image, the timeline shows that the loops are synchronized at the beginning of their complete pattern on step 1.

A compromise in the development of the timeline was made in its stepwise behavior at the first step of each loop represented on the timeline as repeating darker green or orange ovular shapes. Ideally, the counter would progress through the time line one time unit at a time, however, much like the development of the playhead, this continuous motion was plagued with with juddering, blurring, and frequent lags. Another issue with the counter's progress is that in certain configurations, two counters confusingly appear on the timeline at once (*Figure 19*). This is caused by the fact that the counter moves across the timeline as the playhead moves on and off step 1 of each loop, meaning, if one loop completes a full cycle before the playhead has moved off of step 1 of the other loop, two counters will appear at once. This bug will be resolved in future prototype iterations but was not prioritized for resolution before the completion of prototype two.



Figure 19: A bug where two timeline counters appear at the same time

5.2.4 Paper Tutorial

It became clear during product testing that the function of certain interface metaphors and controls were not self-evident to users. This lack of awareness extended beyond the possibilities of polymeter and polyrhythm to a general lack of understanding about what could be accomplished on a circular rhythm interface. Because of these reasons, perhaps the most urgent addition to the second Groove Pizzeria prototype is the inclusion of a paper tutorial. This tutorial is meant to give the user a sense of what can be generally accomplished with a circular step sequencer as well as educate the user about rhythmic geometry, polyrhythm, polymeter, and how they can be utilized musically. The tutorial can be read in full in Appendix A.

5.3 Product Test 2 Method

The testing of the second prototype was made more consistent with the addition of a paper tutorial into the prototype and a greater reliance on think-aloud protocol. Rather than filming a user as they were asked a number of questions and given a set of tasks, each user was filmed as they completed a tutorial and asked to think-aloud as they progressed. The user was encouraged to ask questions throughout their process as this could provide relevant data for subsequent content analysis, however, not all questions were answered in full, allowing users to progress through the tutorial as naturally and independently as possible. The video content from these tests is undergoing analysis and will inform the design of a third Groove Pizzeria prototype.

6. CONCLUSIONS

Analysis of media from product testing brought to light usability problems at the behavioral level of emotional design related to Groove Pizzeria's strength communicating how loop duration and step-count interact to allow the composition of polyrhythm and polymeter. Users' reflective level reactions to the prototype were affected by these issues resulting in Groove Pizzeria testing more strongly as a trivial musical activity than a standalone learning tool or musical instrument. As a result of these findings, changes were made to Groove Pizzeria's interface to provide more and clearer information to the user. Additionally, a paper tutorial was written to onboard users. The documentation of Groove Pizzeria's development in this thesis ends at the beginning of product testing of the second prototype, however, many directions that can be taken for continued development of the prototype can be predicted and discussed in this chapter. Broadly speaking, future research and development of Groove Pizzeria could focus on two major categories: additions and extensions of rhythmic features and product usability optimization.

6.1 Future Additions and Extensions of Rhythmic Features

The first two Groove Pizzeria prototypes focused on the development of features for the comprehension and composition of polyrhythm and polymeter, however, many other rhythmic concepts related to these ideas are worth exploring in the future. In Hein's 2003 master's thesis he writes, "most music is organized into repeated rhythmic cycles, and cycles of cycles, and very often cycles of cycles of cycles (39)." Groove Pizzeria can be thought of as extending the Groove Pizza design to delve into the ways that multiple repeating rhythmic cycles interact. Subsequent iterations of the Groove Pizzeria prototype could investigate the way that "cycles of cycles" of drum patterns interact. This has some precedent in the world of circular step sequencing with Buchla's 250e Dual Arbitrary Function Generator hardware module that lets you "nest" sequences within steps of the sequence itself (Trickdigital, 2015), but has not been explored with screen-

based circular step sequencers to the author's knowledge. This concept would require significant interface additions to include storage of multiple patterns for each loop. It will also require another circular visualization where whole loops are represented by single points on a "higher-level" sequencer.

Additionally, extending Groove Pizzeria's features for the comprehension and composition of rhythm can come in the form of adoption of features inspired by other circular rhythm interfaces. Two must-implement features originating from Groove Pizza are the ability to add and visualize swing and the ability to automatically populate a loop with a Euclidian rhythm.

6.2 Future Usability Optimization

Content analysis of the second set of user tests will undoubtedly unveil usability issues including those that have been with Groove Pizzeria since the first prototype and well as those that are caused by the second set of feature additions and revisions. That said, this test focuses on usability narrowly related to the ability to compose and comprehend polyrhythm on the prototype's interface. If Groove Pizzeria is to be a commercially viable (or at least fully-featured) learning tool or musical instrument, many usability features will need to be added that can safely be designed into a prototype iteration before they are subject to an eventual product test.

6.2.1 Usability as a General Software Interface

When considering the usability of Groove Pizzeria's interface simply as a piece of software, rather than a musical instrument, Jakob Nielsen's usability heuristics for interface design¹¹ provide an unfussy roadmap towards a higher definition prototype (1994). Some of these heuristics such as "consistency and standards," "recognition rather than recall," "aesthetic and minimalist design," and "help and documentation" have

already been fulfilled by the second Groove Pizzeria prototype and only need to carry over to newly added features. Other heuristics such as “error prevention,” and “help users recognize, diagnose, and recover from errors” are not relevant to Groove Pizzerias design because a user cannot break the software or create an error with any of its controls. The remaining four heuristics that point towards new features are listed and discussed below.

1. Visibility of system status: “The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.”

To some extent, this heuristic encapsulates the goal of this research, i.e. allowing a user to visibly understand polyrhythm and polymeter. Its fulfillment is reflected in features such as the timeline, that is designed to allow a user to visibly observe where playback is in the complete rhythmic pattern formed by the combined playback of the two loops. However, addition of new rhythmic features such as additional loops, “higher-level” sequencing loops, and swing controls present a formidable design challenge because they require a large amount of screen real estate to effectively visually communicate their relationship to one another. A solution to this may be the ability to incorporate the screens of multiple devices in a single user session (e.g. a tablet and a laptop or two tablets).

2. Match between system and the real world: “The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.”

This is another heuristic that was important to the design of the existing Groove Pizza prototype i.e. the design of novel interfaces that reflect the user’s understanding of music theory. The use of “steps” as opposed to “slices” on the

prototypes interface is an example of this. However, many features are missing from Groove Pizzeria that create a gap between the system and a user's real world understanding of digital music instruments. Inclusion of aspects from the visual language of digital music instruments on future Groove Pizzeria prototypes (e.g. play, pause, and stop buttons) will make the design more usable as a musical instrument as well as graphically clarify its use as a musical instrument.

3. User control and freedom: "Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo."

The lack of undo, redo, clear was a major usability problem in prototype testing to the point of distracting users. The addition of these features is an essential before future prototype tests regardless of whether the prototype is being tested for academic reasons related to the composition and comprehension of rhythmic attributes or for entrepreneurial purposes.

7. Flexibility and efficiency of use: "Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions."

Although this heuristic applies to keyboard shortcuts common across many types of software (e.g. "undo" in mac programs: ⌘Z), flexibility and efficiency of use is important to digital musical instruments, where quick edits must be made in musical contexts. This heuristic can also be applied to interaction design for mapping external controllers or touch screens to future Groove Pizzeria prototypes.

6.2.2 Usability as Step Sequencer and a Musical Instrument for the Browser

Although the Groove Pizzeria prototype was created with the intention of becoming a musical instrument, its design excludes many features that are essential for the musicality of step sequencers, most importantly pitch and velocity control. Addition of these features would require subsequent user tests and careful mapping of feature constraints (Magnusson, 2010).

Additionally, although the choice to develop Groove Pizzeria with JavaScript was largely due to the convenience of prototyping with p5.js and the Web Audio API (as opposed to writing a prototype VST with C++ for example) it is worth considering what it means for Groove Pizzeria to be a browser based musical instrument. Bradford Gowland and David Van Dusen's 2018 NYU Music Technology master's theses involved the creation of a remotely collaborative browser based step sequencer named the TB2 (the author had a hand in designing and developing an earlier version of TBD2 called the TBD). Their research though design theses both provide deep discussion and insight into the reasons for creating a web instrument, noting that the web's networking strengths can be utilized in conjunction with desktop or hardware music software, the DSP of which is generally more stable than that of the browser.

6.3 Final Remarks

The terms polyrhythm and polymeter are commonly used in music journalism and marketing materials. Although they are frequently misrepresented, their use suggests a general interest in rhythmic complexity exists in the community of electronic/dance music listeners and producers. Despite the incredible affordances of DAWs and other electronic musical instruments, these tools present physical or perceptual constraints that can be thought of as rhythmic presents. A number of circular step sequencers are available to electronic/dance music producers. Some of these instruments, most notably

MusEDLab's Groove Pizza, grant users a greater command of their rhythmic composition by offering intuitive perception and control of rhythmic geometry. However, few of these instruments emphasize or even allow for the potential of polyrhythmic and polymetrical rhythm composition.

Groove Pizzeria builds on the foundation of Groove Pizza, providing a set of two distinct sequencer interfaces that can be controlled in terms of overall loop duration in sixteenth notes, and the number of equally spaced steps within the overall loop duration. Through the SoDaR process, an interface was developed that allowed a user to compose and comprehend polyrhythmic and polymetrical drum patterns in a way that would be prohibitive with most existing DAWs and other electronic musical instruments.

Evaluation of this prototype concluded that the elements of the interface involved in creating polyrhythmic and polymetrical drum patterns were successful at three levels of emotional design (visceral, behavioral, and reflective). Growing Groove Pizzeria into a full fledged musical instrument would require many subsequent rounds of development and product testing, that will continue, at least in part, beyond the purview of this thesis.

REFERENCES

- An Introduction to Design Thinking PROCESS GUIDE [Pamphlet]. (2010). Stanford, California: The Hasso Plattner Institute of Design at Stanford (d.school).
- An Introduction to Extratone: The World's Fastest Music Genre. (2018, April 30). Retrieved from <https://daily.bandcamp.com/2018/04/27/an-introduction-to-extratone-the-worlds-fastest-music-genre/>
- BEATSTEP PRO Firmware Update. (n.d.). Retrieved May 2, 2018, from [https://www.arturia.com/beatstep-pro-firmware-update-2\](https://www.arturia.com/beatstep-pro-firmware-update-2/)
- Bouwer, A., Gifford, T., McDermott, J., Wagdy, M. (2013). Should Music Interaction Be Easy? In *Music and Human-Computer Interaction* (pp. 259-270). Springer.
- Brown, A. R. (2007). Software Development as Music Education Research. *International Journal of Education & the Arts*, 8(6).
- Butler, M. J. (2006). *Unlocking the groove: Rhythm, meter, and musical design in electronic dance music*. Bloomington: Indiana Univ. Press.
- Dam, R., & Siang, T. (n.d.). What is Design Thinking and Why Is It So Popular? Retrieved from <https://www.interaction-design.org/literature/article/what-is-design-thinking-and-why-is-it-so-popular>
- Davis, E. (2008). *Roots and Wires Remix: Polyrhythmic Tricks and the Black Electronic*. In *Sound Unbound: Sampling Digital Music and Culture*. Cambridge, Massachusetts: MIT Press.
- DeSantis, D. (2015). *Making Music: 74 creative strategies for electronic music producers*. Berlin: Ableton.
- Don't DJ on multimettric, polyrhythmic, circular music. (2016, October 18). Retrieved from <https://www.ableton.com/en/blog/dont-dj-moving-in-circles/>
- Gelineck, S., & Serafin, S. (2009). From Idea to Realization - Understanding the Compositional Processes of Electronic Musicians. In *Audio Mostly Conference*. Denmark: Aalborg Universitet.
- Gothelf, J., & Seiden, J. (2016). *Lean UX: Applying lean principles to improve user experience*. Beijing: O'Reilly.

- Goldmann, S., & Henke, R. (2015). Presets - digital shortcuts to sound: Stefan Goldmann in conversation with Cory Arcangel, Mike Daliot, Robert Henke, Junichi Ikeuchi, Tomoko Itoh, Dimitar Kotev, Dinis Schemann, Tony Stevanovski and Michael Wagener. London ; Berlin: The Bookworm.
- Gowland, B. (2018). A Multi-user MIDI Sequencer on the Web for Distributed DAW Collaboration(Unpublished master's thesis). NYU Steinhardt Music Technology Department.
- Groove Pizza. (n.d.). Retrieved from <https://apps.musedlab.org/groovepizza/?source=pub&museid=Sk954a-p7&show-grid=true&multi-lock=&brainpop=false&midimap=&>
- Hanington, B. M., & Martin, B. (2012). Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions. Gloucester, MA: Rockport.
- Holland, S., Mulholland, P. & Wilkie, K. (2013). Towards a Participatory Approach for Interaction Design Based on Conceptual Metaphor Theory: A Case Study from Music Interaction. In Music and Human-Computer Interaction (pp. 259-270). Springer.
- Hein, E. (2013). Designing the Drum Loop (Unpublished master's thesis). New York University.
- Hein, E. (2015, September 28). User interface case study: Patterning. Retrieved March 22, 2018, from <http://www.ethanhein.com/wp/2015/user-interface-case-study-patterning/>
- Herbert, M. (2005). PERSONAL CONTRACT FOR THE COMPOSITION OF MUSIC. Retrieved March 21, 2018, from <https://matthewherbert.com/about-contact/manifesto/>
- Holtzkener, J. (n.d.). Concentric Rhythm (Version 2.0). Retrieved November 8, 2018.
- Jayasuriya, M. (2017, December 20). Listen to "Stir Fry" by Migos. Retrieved from <https://pitchfork.com/reviews/tracks/migos-stir-fry/>
- Kaufman, E. L., Lord, M. W., Reese, T. W., & Volkman, J. (1949). The Discrimination of Visual Number. *The American Journal of Psychology*, 62(4), 498-525.
- Koetting, J. (1970). Analysis and Notation of West African Drum Ensemble Music. *Selected Reports in Ethnomusicology*, Volume 1, No. 3

- Lenarduzzi, V., & Taibi, D. (2016). MVP Explained: A Systematic Mapping Study on the Definitions of Minimal Viable Product. 2016 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA). doi:10.1109/seaa.2016.56
- Linn, R. (2015, December 03). Roger Linn On Swing, Groove & The Magic Of The MPC's Timing. Retrieved from <https://www.attackmagazine.com/features/interview/roger-linn-swing-groove-magic-mpc-timing/>
- Lynch, W. (2018, October 11). Breaking Through: Forest Drive West. Retrieved from <https://www.residentadvisor.net/features/3349>
- Magnusson, T. (2010). Composing and Performing with Digital Musical Systems. *Computer Music Journal*, 34(4), 62-73.
- McLeod, K. (2001). Genres, Subgenres, Sub-Subgenres and More: Musical and Social Differentiation Within Electronic/Dance Music Communities. *Journal of Popular Music Studies*, (13), 59-75.
- Millard, D. (2013, May 06). Is "EDM" a Real Genre? Retrieved March 22, 2018, from https://noisy.vice.com/en_ca/article/rdd5zr/is-edm-a-real-genre
- Neill, B. (2002). Pleasure Beats: Rhythm and the Aesthetic of Current Electronic Music. *Leonardo Music Journal*, 12, 3-6.
- Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI 93*. doi:10.1145/169059.169166
- Nielsen, J. (1994, April). Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 152-158). ACM.
- Nielsen, J. (2000, March 19). Why You Only Need to Test with 5 Users. Retrieved from <https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>
- Norman, D. A. (2003). *Emotional design: Why we love (or hate) everyday things*. New York: Basic Books.
- Nylander, L., Collins, H., & Rose, O. (2016, November 07). Grime I rep to the T, 140 I kept to the beat. Retrieved March 21, 2018, from https://i-d.vice.com/en_uk/article/papegz/grime-i-rep-to-the-t-140-i-kept-to-the-beat

- O' Reilly, M., & Tarakajian, S. (n.d.). Rhythm Necklace (Version 1.0.5) [Computer software]. Retrieved November 8, 2018, from <http://rhythmnecklace.com>
- Pancare, R. (2018, April 23). How Do Bright Colors Appeal to Kids? Retrieved from <https://sciencing.com/do-bright-colors-appeal-kids-5476948.html>
- Reynolds, S. (1999). *Generation ecstasy: Into the world of techno and rave culture*. New York: Routledge.
- Rundall, J. D. (2011). *Polymeter: Disambiguation, classification, and analytical techniques* (Unpublished doctoral dissertation). University of Illinois at Urbana-Champaign.
- Ruthmann (2012). *Engaging Adolescents with Music and Technology*. In Burton, S. (Ed.). *Engaging Musical Practices: A Sourcebook for Middle School General Music*. Lanham, MD: R&L Education.
- Schmid, G. M. (2016). *Evaluating the Experiential Quality of Musical Instruments*. Retrieved from <https://link.springer.com/book/10.1007%2F978-3-658-18420-9>
- Sherburne, P. (2016, January 11). *Mood II Swing: Strictly Mood II Swing*. Retrieved from <https://pitchfork.com/reviews/albums/21405-strictly-mood-ii-swing/>
- Sherburne, P. (2017). *Draw a Straight Line and Follow It: Minimalism in Contemporary Electronic Dance Music*. In *Audio Culture: Readings in Modern Music* (Revised Edition ed., pp. 465-478). Bloomsbury Academic.
- Stop Calling EDM EDM - Here Is A Proper Definition. (2015, October 07). Retrieved March 22, 2018, from <https://www.magneticmag.com/2015/10/stop-calling-edm-edm-here-is-a-proper-definition/>
- Toussaint, G. T. (2005). *The Euclidean algorithm generates traditional musical rhythms*. *Proceedings of BRIDGES: Mathematical Connections in Art, Music, and Science*, Banff, Alberta, Canada, July 31 to August 3, 2005, pp. 47–56.
- Toussaint, G. T. (2010). *Generating “Good” Musical Rhythms Algorithmically*. In *Proceedings of the 8th International Conference on Arts and Humanities* (pp. 774-791). Honolulu, Hawaii.
- TOUSSAINT, G. T. (2017). *GEOMETRY OF MUSICAL RHYTHM: What makes a "good" rhythm good?* S.I.: CRC PRESS.

- Trickdigital. (2015, October 01). Buchla 250e Dual Arbitrary Function Generator Tutorial & Demo. Retrieved from <https://www.youtube.com/watch?v=DrGqUHIzVU>
- Tufte, E. R. (2001). *Envisioning information* (2nd ed.). Cheshire, Conn: Graphics Press.
- Van Dusen, D. J. (2018). *Development Of An Interactive Web MIDI Sequencer For Remote DAW Collaboration* (Unpublished master's thesis). NYU Steinhardt Music Technology Department.
- Wilson, C. (2013, January 9). A Tale of Two Clocks - Scheduling Web Audio with Precision - HTML5 Rocks. Retrieved from <https://www.html5rocks.com/en/tutorials/audio/scheduling/>
- Yates, K., & Sama, L. (2014, November 05). Logan Sama on how to be a grime DJ. Retrieved March 21, 2018, from <http://www.dazeddigital.com/music/article/22440/1/logan-sama-on-how-to-be-a-grime-dj>

APPENDIX A

GROOVE PIZZERIA TUTORIAL

1. Circular Step Sequencing

Welcome to Groove Pizzeria, a circular step sequencer that lets you unlock the power of rhythmic geometry, polyrhythm, and polymeter. If you don't know what that all means, don't worry, Groove Pizzeria will help you understand. If you *do* know what that all means, you may still want to skim this tutorial to make sure you know how to use Groove Pizzeria.

In either case, let's open Groove Pizzeria, take a look, and get a quick overview of the controls. The first thing you will probably notice is two circular step sequencer loops; one with a green and one with an orange rotating playhead. These are each divided by 16 radial lines with three circles on each line. In this default state, you can think of each loop as a three voice, 16 step drum machine wrapped into a loop. Like any step sequencer, Groove Pizzeria allows you to enter notes on a... well... sequence of steps. Also like most step sequencers, these steps each represent the duration of one 16th note at a given BPM. Beneath each loop are three controls. The slider marked with a number and "time units" controls and displays the duration of a loop. The slider marked with a number and "steps" controls and displays the amount of steps that divide a loop. The slider marked with a number and "step rotations" allows you to rotate a sequence of notes within each loop. Between the step-count and step rotation sliders of each loop is a display demonstrating the duration of a step on one loop versus the other. At the top left a timeline demonstrates the overall resulting drum pattern when the loops are of different durations.

1.1 Making a Beat

After that little feature overview, you are probably wondering how everything works together, but before you dive down an endless well of polyrhythmic and polymetric possibility, let's make a simple drum beat. By clicking or dragging over steps on the green loop, try recreating the pattern below, the drum beat of Detroit proto-techno classic "Shari Vari" by A Number of Names.



Sound cool? Try embellishing this beat by adding in notes on the orange loop on the right. A fun way to do this is to just click and drag your cursor over the loop filling in random notes. Nice, you've got the basics down.

1.2 Geometry of Musical Rhythm

Alright, let's try something that your other step sequencers probably can't do. Below is a screen shot of the "guataca" timeline (a percussion pattern that repeats throughout a piece of music without variation) from Cuba. Take a look at how the sequence of notes on the outer ring forms a six sided polygon. On the inner two rings, the same polygon is displayed as a mirror image. This visual configuration is just one example of the geometry of musical rhythm: a rich concept that seeks to understand global rhythmic attributes by studying their geometry when mapped onto a circular graph. With Groove Pizzeria, you don't need to be an ethnomusicologist to utilize rhythmic geometry. Try recreating the guataca timeline, first by setting the step-count of each loop to 12 and filling in the notes shown on the screenshot below.



Can you hear and see the rhythms of each drum voice locking together? Now try adding notes to the green loop. See if you can utilize geometry to generate rhythmic ideas. Think about geometric concepts like symmetry, maximally distant spacing of corners in a polygon, mirror reflections, or really any use of geometry that looks and sounds

interesting to you. Also, play around with the step rotation slider and observe how this changes the resulting drum pattern.

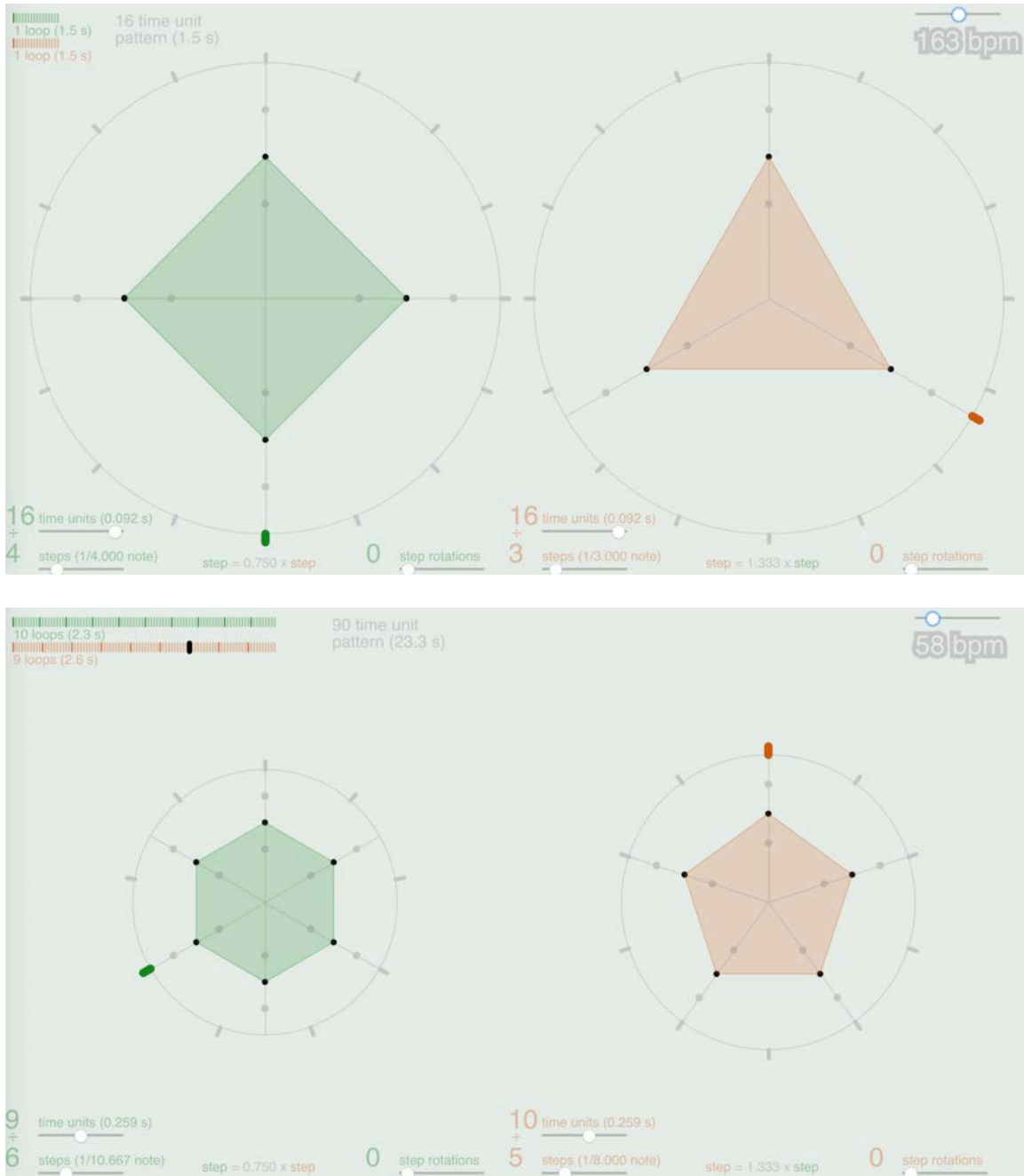
2. Polyrhythm

Now that you have a sense for what the geometry of musical rhythm is about, let's try experimenting with polyrhythm. Explaining why polyrhythm is interesting is a complicated task, so for the sake of this tutorial it suffices to say that polyrhythm is technique at the core what makes many percussion traditions across the world exciting to listen to and is relatively difficult, and sometimes impossible, to implement with most step sequencers. The simplest way to define polyrhythm for use with Groove Pizzeria is a resulting drum pattern of two loops with equal durations, different amounts of steps that are not multiples of one another, and equal spacing of notes. To help you play and understand this concept, Groove Pizzeria lets you control the duration of a loop in total time units (each one 16th note length) and the number of steps that divide a loop. Try recreating the 3:2 polyrhythm shown in the screenshot below.



Do you hear how these two rhythms that divide a time duration by three and two create a new pattern when superimposed? Try changing the amount of steps in each sequence and

generating polyrhythms like 4:3, 5:4, or the wonky 11:7 or 9:5. For now, focus on making sequences of equilateral polygons where a note is on every step in a voice. Experiment with how these rhythms sound at different tempos by playing with the BPM slider. If you start playing around with loop duration, you may wonder if you are still making polyrhythms. That's what the step-duration-ratio displays are for. These display the duration of one loop's steps against the steps of the other loop allowing you to observe if you have a step duration ratio between the loops that could allow polyrhythm even if your loop durations are different. Below is an example of how this can work.



On the top is a 4:3 polyrhythm that is easy to comprehend due to the fact that each loop has the same duration and is divided by 4 and 3 steps respectively. On the bottom, a more difficult to comprehend 4:3 polymeter is made clear through two step-duration-ratio displays.

3. Polymeter

For a long time, polymeter was defined as a type of polyrhythm, but nowadays the terms are usually considered to be distinct. On Groove Pizzeria, you can think about polymeter as being the resulting drum pattern of two loops with steps of the same duration, but different loops lengths that aren't simple multiples of one another. The technique creates a sense of constantly shifting loops and is used many kinds of music including some of the most interesting electronic/dance music. Take a look at the drum pattern in the screen shot below. Notice how the duration of each loop is a different amount of total time units. The amount of steps in each loop is also different, but you can see that the duration of each step is a 1/16 note. Also, take a look at the timeline on the top left of the screen. You will notice that as the playheads move around each loop, a counter moves across the timeline. This is showing you the overall pattern length resulting from the simultaneous playback of different loop lengths. The fact that two short loops can result in a much longer and more complex pattern is at the root of what makes polymeter so interesting. See if you can recreate this pattern and observe how everything is interacting.



Do you see and hear how steps on each loop are triggering at the same time? Play around with the loop duration and step-count sliders. You will figure out that anytime the step-count slider and loop duration sliders match, your loop will have steps with a 1/16 note duration. Setting both your loops to two different sets of matching loop duration and step-count is an easy way to make a polymetric pattern. However, there many ways you can make polymeter. Just take a look at the step durations of each loop and see if you can get them to match with different loop durations.

3. Free Play

Wow, thanks for making it this far through this tutorial. If you learned something about the geometry of musical rhythm, polyrhythm, and polymeter thats exciting for me. But, hopefully, more than learning a couple definitions, you've started to think about some new ways you can add rhythmic variety to your music. More than anything, Groove Pizzeria is meant to give you the tools to be experimental when creating drum patterns while allowing you to understand what you're doing. After you leave this tutorial, see if you can make some drum patterns that are interesting to you; preferably getting a little lost and letting Groove Pizzeria be your guide. Safe travels.

ENDNOTES

¹ Although most available circular rhythm interfaces are screen based there are are hardware examples such as the 252e Buchla Polyphonic Rhythm Generator (<https://buchla.com/252e-buchla-polyphonic-rhythm-generator/>)

² For this thesis, the author will adopt Kembrew McLeod's term "electronic/dance music." McLeod argues for the validity of the umbrella term as "a metagenre that is constantly breaking apart, recombining, and making obsolete numerous subgenres on a yearly basis," "a heterogeneous group of musics made with computers and electronic instruments—often for the purpose of dancing... vague enough to describe the broad variety of musical styles consumed by a loosely connected network of producers and consumers." Kembrew makes a point to state that his term is in no way "an attempt to ignore the very concrete differences between the way these subgenres are consumed and produced (as the term "electronica"—an oft-cited word used by the American music press—does)."

McLeod's article was published in 2001, when the term "electronica" was in popular use. More common in 2018 is the term "EDM" (Electronic Dance Music) which brings with it genre signifiers for a specific genre of pop music despite the encompassing nature of the term itself. Examples of griping over this can be found in the articles, "Is "EDM" a Real Genre? (2013)" published on Noisey.com or "Stop Calling it EDM - Here is a Proper Definition (2015)" published on Magneticmag.com. The author's hope is that the slash in the term "electronic/dance music" is enough return the phrase back to umbrella state.

³ "Product testing" is a preferable term over the more common "user testing" because the later could seem to suggest that the user themselves is being tested (Hanington, 2012).

⁴ In the electronic/dance music community, it is most common to refer to a single composition as a “track.” In *Unlocking the Groove: Rhythm, Meter, and Musical Design in Electronic Dance*, Butler chooses to avoid confusing the term “track” as a single composition with the kind of “track” referenced in a “multi-track recording” by referring to the latter as a “textural layer.” Butler’s choice of terminology will be used in this thesis.

⁵ The importance of Fruity Loops in grime’s development is mentioned in a number of interviews with veteran grime producers, DJs, and MCs. In a 2014 article, “How to be a Grime DJ” longtime Grime DJ Logan Sama states:

“There is no rule of playing songs in 140bpm. 140bpm was just the tempo many Grime tunes were made in. It was just some arbitrary bpm which came around simply because Fruity Loops was set to 140bpm by default. Before that Garage was anything from 132 to 145. The important thing is to play all your tracks at the saaaaaame BPM otherwise you are gonna clang down the house. And obviously music sounds best around the tempo it was actually made at.”

In the article “grime i rep to the t, 140 i kept to the beat” which reflects on the book, *This is Grime*, an oral and visual history of grime, author Hattie Collins states: “Grime really is a human story at its heart, it stretched from 10 kids messing around on Fruity Loops, to grow into and stand for something much bigger.”

⁶ Developed by Philip Harland, Time Unit Box System (TUBS) is a simple system for notating musical rhythms using a horizontal 1 by N grid denoting a fixed unit of time and a mark within a box representing a musical event (Koetting, 1970)

⁷ In “The Euclidean Algorithm Generates Traditional Musical Rhythms” Toussaint describes a rhythm necklace as follows:

“In the remainder of this section we list some of the most common Euclidean rhythms found in world music. In some cases the Euclidean rhythm is a rotated version of a commonly used rhythm. If a rhythm is a rotated version of another we say that both belong to the same necklace. Thus a rhythm necklace is the inter-onset duration interval pattern that disregards the starting point in the cycle (4).”

In this paper, circularly mapped TUBS rhythm, be it euclidean or not, will be referred to as a “sequence” for the sake of relevance to Groove Pizzeria’s interface.

⁸ In the Geometry of Musical Rhythm: What Makes a “Good” Rhythm Good?, Toussaint defines his hyphenated variation of the durational pattern as, “a sequence of adjacent inter-onset intervals present in [a] rhythm as a list (14).”

⁹ There is no one accepted definition of a minimum viable product and disagreement is common in writing and in real world practice in this author’s experience. Lenarduzzi et al. trace a pattern through the various definitions discovering that Eric Ries’ is the most influential. I take the Lenarduzzi et al’s suggestion to fuse Ries’ and Bank’s influential definitions in this thesis.

¹⁰ The word “teeth” has been chosen over “cogs” due to the semantic confusion between the definition of “cog” referring to both the gear’s teeth and the gear itself.

1. *Visibility of system status*: The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
2. *Match between system and the real world*: The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
3. *User control and freedom*: "Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo."
4. *Consistency and standards*: "Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions."
5. *Error prevention*: "Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action."
6. *Recognition rather than recall*: "Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate."
7. *Flexibility and efficiency of use*: "Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions."
8. *Aesthetic and minimalist design*: "Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility."
9. *Help users recognize, diagnose, and recover from errors*: Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
10. *Help and documentation*: Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large. (Nielsen, 1994)