Mathematics and Music: Generating Polyrhythms from Cyclic Subgroup Structures

Maryville College

Ian Schomer

May 2019

Mathematics and Music: Generating Polyrhythms from Cyclic Subgroup Structures

A Report of a Senior Study

by

Ian Schomer

Major: Mathematics

Maryville College

May 2019

Date Approved ____________, by ________________________________

Faculty Supervisor

Date Approved ____________, by ________________________________

Division Chair

© by Ian Schomer, 2019
All Rights Reserved.

To my parents, Kathleen and Terry Schomer, for encouraging me to explore music from a young age, and for helping me to develop this passion that is now central to my personal identity.

Acknowledgements

I would like to thank my thesis advisor Dr. Jesse Smith, for his constant dedication to the excellence of this thesis. At every step in the process, he provided valuable insight and vision that helped me to develop a few simple ideas into an exciting yet tangible research goal.

In addition, I would like to thank Dr. Chase Worley, for his continued interest in my resrach and his willingness to volunteer his services for the benefit of my work. Though he had no responsibility to do so, he offered keen observation and experience that allowed me to overcome certain obstacles that would have otherwise delayed the progress of this research.

Lastly, I would like to thank Maryville College for their kindness and consideration in providing a grant that covered the costs of my computer equipment. Working with hardware deeply enriched the quality of my research experience and has opened up the opportunity for more exciting and interactive presentations.

Personal Statement

The system of education in which I was raised seems to have separated many interrelated topics in order to fit the classroom model. The focus of my time in class would jump from literature to science, science to social studies, social studies to the much-anticipated lunch period, then perhaps band, and so on. As a result, I developed a sort of "floating-bubble model" where each subject occupied a distinct space within collective knowledge. As I grew more inquisitive with age and experience, however, I began to notice that some of my biggest passions were intimately related.

I have played the piano since the age of four. The piano bench has become a trusted space for me to pour out countless hours of self-expression, enjoyment, and experimentation. At this same age I started formal schooling, where I quickly and naturally gravitated toward math and science. In the years that followed, two of my greatest passions—mathematics and music—remained disconnected from each other in both mind and space. Not until my college education did I begin to search for connections among the different realms of academia. Quickly I found a wealth of overlap and correlation between the structure of mathematics and the artistry of music. My interest peaked at the thought of creating music using more abstract mathematical concepts such as group theory. This thesis represents the culmination of my research into this deep and perplexing subject, as well as my attempts to utilize certain aspects of group theory to create in the form of rhythm.

May not music be described as mathematics of the sense, mathematics as music of the reason?

— James Joseph Sylvester, 19^{th} century English mathematician

Abstract

We investigate the relationship between mathematics and music, starting with the most basic understanding of musical notes in a mathematical context, and moving into historical applications of mathematics in music. We successfully expand upon two distinct topics of research presented in the introduction. Firstly, we identify an isomorphism for the group of musical operations that was initially defined by Dr. Leon Harkleroad. In this section we include a formal proof that utilizes fundamental concepts from abstract algebra. Secondly, we establish and investigate an application of cyclic subgroup structures in the generation of polyrhythms with "irreducibly periodic perfect balance" (IPPB)—a rhythmic phenomenon first described by Andrew J. Milne and his colleagues. We create an interactive software tool that outputs an audible rhythm and displays a visualization of the corresponding subgroup structure on a 32x32 LED matrix panel. In addition to developing this tool, we provide a theorem that states the necessary qualifications for a group to achieve IPPB. Since the source code is posted online, others will be able to follow the installation instructions and hear for themselves the rhythms made by cyclic groups.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Mathematics and music are often regarded in entirely separate worlds yet are inherently intertwined. Patterns of sound that are naturally pleasing to the ear utilize some of the simplest elements of math such as integers and ratios. As sounds are paired together, aspects of trigonometry and wave functions go hand-in-hand with more subjective responses, such as pleasing or displeasing qualities of the sound. Even aspects of musical composition, ranging from a simple melody to an entire orchestral arrangement, employ elements of group theory operations. Throughout the development of Western European music, mathematicians have played an active role in quantifying musical techniques and supplying a mathematical framework for music theory, as we will discuss later. Ironically, the two subjects were once considered to be part of the sciences, but the bridge has since widened with changes in cultural ideologies. Since music was first defined formally in terms of mathematics, questions have been raised as to what extent music can be contained *within* mathematics. As a response to this discussion, we will attempt to explore movements of thought that have applied mathematical techniques to the synthesis and analysis of music. Moreover, we will attempt to define the music of group theory relative to sound and rhythm by assigning structural elements to various pitches and rhythm patterns, as defined by an assignment algorithm.

1.1 Preliminaries

We begin by providing information on both mathematics and music theory in order to allow for a more fluid discussion later in the study. We will use a keyboard to represent the range of possible notes; this will aid in both simplicity and easy visualization of harmonic theory. A piano contains 88 keys that span just over 7 octaves (defined later), two of which are shown in Figure 1.1. Piano keys are arranged in a repeating visual pattern that distinguishes both octaves and the 12 unique notes within each octave. In order to further distinguish notes that share the same letter label, a number is added to indicate its position on the keyboard; for instance, A0 is the 1^{st} A, starting from the left, and A4 is the 5^{th} A.

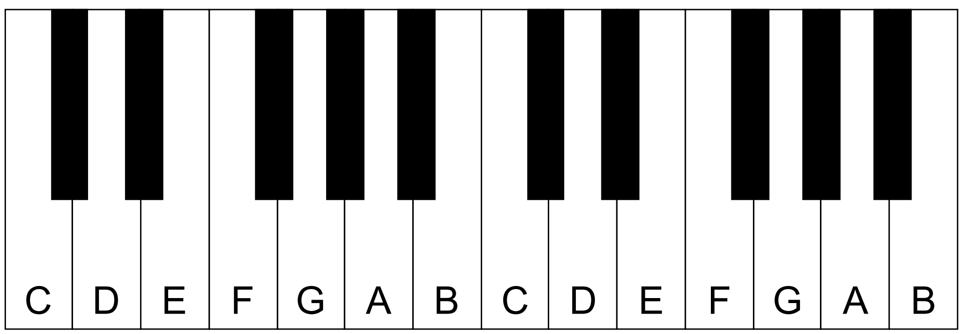


Figure 1.1: Labeled keyboard notes spanning two octaves

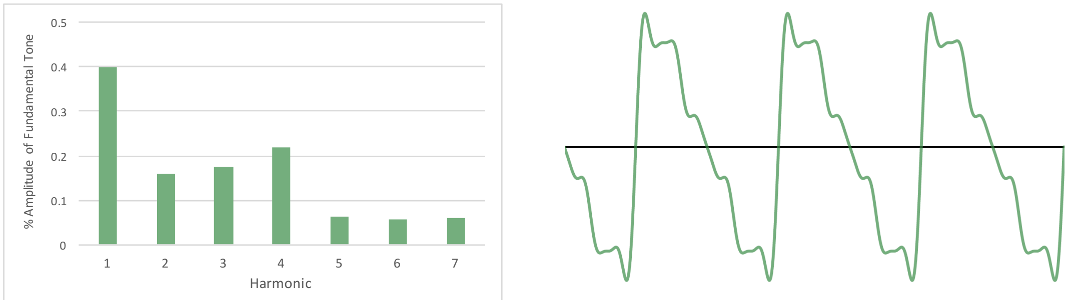
Being that music is inherently audible, we can describe musical notes with physical properties of sound, specifically by the repeating undulations in air pressure that are produced [9, p. 5-6]. This is measured in cycles per second, or Hertz (Hz), which is known as the *pitch* of a note. The frequencies of starting letter notes on the keyboard are presented in Table 1.1, which will serve as a reference for later discussion. Sounds that can be modeled with a simple, sinusoidal wave function are called *pure tones*. However, pure tones cannot be played in isolation by an instrument because the materials vibrate to produce additional tones—referred to as overtones or *harmonics*—that are specifically limited to integer multiples of the frequency

Note Letter	Number (from left of keyboard) 0	1	2	3	4	5	6	7
A	27.50	55.00	110.00	220.00	...			
A♯ / B♭	29.14	...						
B	30.87	...						
C	32.70	...						
C♯ / D♭	34.65	...						
D	36.70	...						
D♯ / E♭	38.89	...						
E	41.20	...						
F	43.65	...						
F♯ / G♭	46.25	...						
G	49.00	...						
G♯ / A♭	51.91	...						

Table 1.1: Note labels and their respective frequencies, in Hertz. Higher octaves of each note are powers of 2 higher than the base pitch.

being produced [8]. More specifically, these integer multiples can be described in the *harmonic series* (1, 2, 3, 4, ...) which looks like the natural numbers. Any reference to harmonics used in this study will refer to this series and not the harmonic series described in mathematics $(1, 1/2, 1/3, 1/4, \ldots)$. The resulting subjective quality of the harmonics produced by an instrument is called its *timbre*. For instance, a clarinet that plays a note with frequency f also produces tones with frequencies $2f, 3f, 4f, \ldots$ that continue past audible frequencies for humans. In this way instruments produce different ranges of harmonics and respective amplitudes that account for a variety of unique sounds while still playing the same note. We can represent this uniqueness visually with a component sine wave, as shown in Figure 1.2b. Here we model an arbitrary instrument's sustained tone by giving amplitudes to the first seven harmonics, which are then combined into a component sine wave to represent its unique timbre.

Harmonics are one of the most obvious bridges between mathematics and music. *Harmony*, derived in name from the harmonic series, refers to any simultaneous combination of notes [11, p. 28]. The term "octave" refers to two notes with frequencies of ratio 1:2 and can also be used to describe the span of pitches between the



(a) Harmonics amplitudes

(b) Component sine wave

Figure 1.2: Representation of a tone and its harmonics produced by an instrument

interval of an octave. To go up an octave of a note with frequency f, simply multiply the frequency by two. Similarly, to play an octave on a piano is to play these two notes simultaneously.[1] Musically, notes separated by octaves sound different yet somehow the same to the human ear (and are even labeled with the same note letter); this is because their wavelengths are "in phase" with one another (Figure 1.3). Of course, there exist intervals both greater and smaller than an octave. The smallest interval found on the piano is that of a *semitone*—often referred to musically as a "half-step"—and can be understood visually as the interval between any two successive notes on the piano. Keep in mind that intervals are not additive quantities but instead ratios that describe the relationship between two pitch frequencies.

Notes are most frequently played within the context of *scales*, which are sets of intervalic distances that form a subset of the 12 unique notes within an octave. We call a piece of music *tonal* if it tends to gravitate toward a certain scale. In contrast, *atonal* music avoids any feeling of key at all times, resulting in what many feel to be random and unmusical [11, p. 33-34]; thus it is important for mathematicians to be aware of tonality when constructing music. Western music uses the interval of a semitone as the unit from which scales are built. For instance, many instrumentalists

[1] As seen in this section, the names of intervals can represent either a leap between two pitches or an instance of harmony.

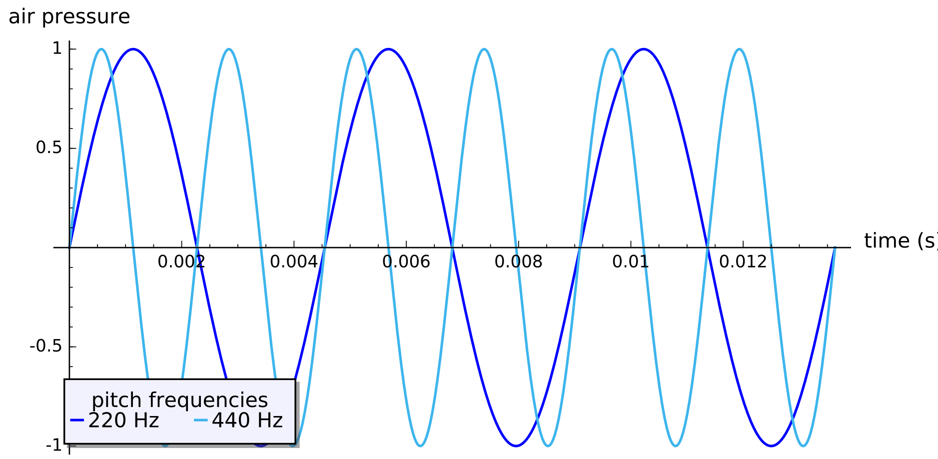


Figure 1.3: An octave in pitch shown as a sine wave

first learn how to play a major scale with the phrase "whole-whole-half, whole-whole-whole-half," where "whole" means two semitones and "half" means one semitone. For instance, the C major scale uses all the white keys of the keyboard (C-D-E-F-G-A-B). Note that, with reference to the above phrase, D and E are a whole-step apart, but E and F are a half-step apart. In this case C is considered to be the "root" of the scale, from which other notes in the scale can be played simultaneously to form intervals of particular musical interest. These intervals are named based on the notes of a major scale, *not* on the number of keys between each note. Figure 1.4 provides the interval names with reference to a C major scale. For example, the interval C-G is called a *fifth* because G is the fifth note in the scale. This naming convention can be confusing because C is actually separated from G by 7 semitones (walked through a major scale by "whole-whole-half-whole"). We emphasize here that, based on popular convention of musicians, intervals are referred to with the names used in Figure 1.4, but they can be determined without a reference scale by counting the number of semitones it contains. A reference (Table 1.2) is provided for visualization of frequently used intervals.

In music theory, the concepts of pitch intervals and scalar tonality come together to form harmonic theory. Simply put, three or more intervals are combined

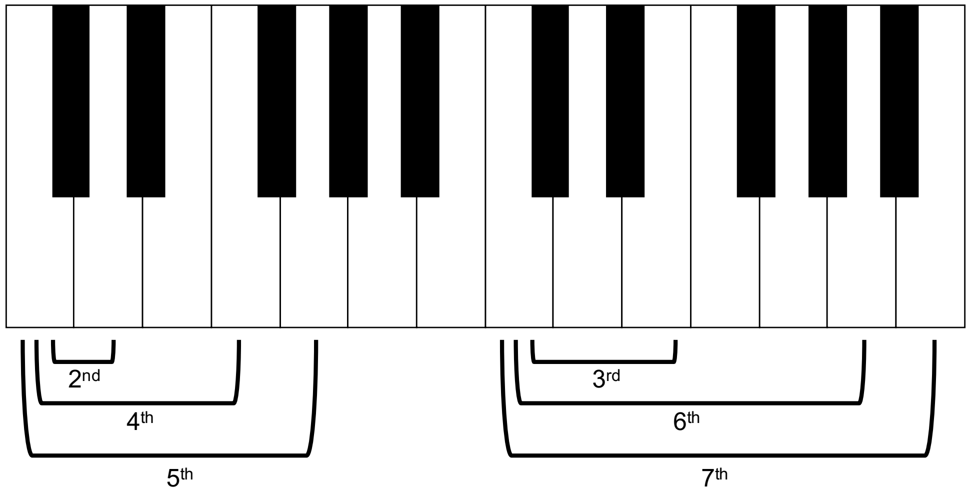


Figure 1.4: Intervals created using the notes of a C major scale

Interval name	# semitones
Minor 2^{nd}	1
Major 2^{nd}	2
Minor 3^{rd}	3
Major 3^{rd}	4
Perfect 4^{th}	5
Tritone	6

Interval name	# semitones
Perfect 5^{th}	7
Major 6^{th}	9
Minor 7^{th}	10
Major 7^{th}	11
Octave	12
(Major) 9^{th}	14

Table 1.2: Common names for particular semitone distances

simultaneously to form chords [11, p. 28]. For example, when considering a major scale, a *triad* is built by taking three notes in a scale, each with a scale note in between them; thus a "C major triad" contains the notes C-E-G, each of which are contained in the C major scale. Notice that the third between C and E differs by the third between E and G by one semitone; this is an example of the difference between a "major third" and a "minor third," respectively. Since most scales, like the major scale, do not have a uniform number of semitones between each successive note, the triads are built with different combinations of major and minor thirds. Interestingly, harmony and chord structure were not incorporated into music until around the ninth century A.D. [11, p. 28]. From then on the Western civilization primarily explored these relationships. With the Boroque period and equal-temperament tuning (discussed later), significant theory was developed on harmony to provide structure to this relatively new element of music. However, we will withhold any further discussion on this topic as it is not pertinent information to this study.

1.2 A Brief History of Mathematics and Music

Although mathematics and music are thought of in largely separate worlds today, the origins of Western European music were built on a foundation of mathematics, starting with the school of Pythagoras in the Sixth Century, BCE. Widely known for his discoveries in geometry, Pythagoras was also believed to be an accomplished lyre player, which might have led him to his experimentation with music. Through a process of scientific experimentation, Pythagoras discovered a beautifully simple relationship of integer ratios between an instrument's physical parameters and the pitch it produces. One such experiment used hanging strings of the same length; each string was a given a different weight in order to determine the resulting effect on pitch produced by the string. Another experiment involved filling vases with different proportions of water and striking the vases simultaneously to hear the resulting intervals created [14]. With each recorded method, Pythagoras quantified the familiar Greek intervals of an octave, a fifth, and a fourth to the integer ratios of $2/1$, $3/2$, and $4/3$, respectively. Pythagoras' discovery highlights the beauty of good theory because it established a link between the familiarity of mathematics and the abstractness of music. These early connections allowed for the foundations of music theory.

After Pythagoras, no significant discoveries in music as a mathematical concept were developed for centuries. The French thinker Marin Mersenne (1588 - 1648) defined in musical terms the first six iterations of the harmonic series for a fundamental tone. Although it likely aided an understanding of musical timbre, Mersenne's research was largely an expansion of the relationships Pythagoras had already described long ago [8]. Jean-Philippe Rameau, the French mathematician that has been praised as rivaling Pythagoreas' influence on music, was the first to link the concept of the harmonic series with the underlying physical phenomena: variations in air pressure at different frequencies. Pythagoras had seen and heard the relationship of ratios to sounds, and Rameau explained this in terms of wave frequencies, which was possible through the research in acoustics by Joseph Sauveur [14]. In addition to these newfound relationships, Rameau applied the harmonic series and other basic rules of integers to develop substantial texts in music theory, in particular his *Treatise on Harmony*. His writing built upon the initial theory of consonant triads (developed largely by Gioseffo Zarlino and Rene Descartes) to produce an all-encompassing theory of both consonant and dissonant chords. Interestingly, Rameau's music theory contained the infinite series of positive integers, which led him to conclude that, since the positive integers also constitute the foundations of mathematical theory, music must be inherently confined within mathematics [14]. Without considering the logic of that claim, most musicians would argue that a certain sense of artistry and subjectivity found in music and its theory simply cannot be quantified through mathematics. Nonetheless, we leave this thought open for the reader to ponder.

As seen above, the influences of the school of Pythagoras on music cannot be overstated. In addition to providing the mathematical link from which Western European music theory would begin to stem, the Pythagoreans also established a tuning system based on their particular fascination with the interval of a fifth. Pythagoras theorized that going up the interval of a fifth repeatedly could span 7 octaves of pitch; however, this problem leads to a mathematical issue [14]. From a base frequency f, going up 7 octaves of pitch would result in a frequency of $(2)^7 f$. If we compare this to going up by fifths, using the multiplier $3/2$, we are left to solve the following equation:

$$(3/2)^n f = 2^7 f$$

Clearly, n cannot be a natural number, so the Pythagorean tuning system stepped around this issue by including true fifths until the final interval, known as the *wolf interval*, which is quite audibly short of a standard fifth.

The presence of an imperfect interval is an inevitable reality in tuning, encountered chiefly when forming scales for musical composition. For instance, consider starting with the note A3 (220 Hz) and going up to B4 (≈ 493.9 Hz) by two intervals of a fifth, which is a distance of 14 semitones; this leads to a frequency of $220 * (3/2)^2 = 495$ Hz. Now go up by 14 semitones by another route, such as a fourth followed by a sixth (an integer multiplier of $5/3$); this frequency is notably dissonant at $220 * 4/3 * 5/3 = 488.9$ Hz [9, p. 21-23].

Which one should be chosen? This question epitomizes the puzzle of tuning systems in that, if simply using integer ratios, it is impossible to match the multiplier of each interval type given a large enough collection of frequencies. This number puzzle—often referred to as the "Division of the Octave"—has caught the interest of mathematicians throughout history, including Pythagoras, Mersenne, Steven, Newton, and Euler [5]. The Pythagoreans offered a scale that maximized pure fourths and fifths, thus incorporating intervals of thirds and sixths that were audibly out of tune.[2] The resulting integer frequencies of the Pythagorean scale are summarized in Table 1.3. Larger integer pairs indicate notes whose interval with the root are approximated to preserve other intervals. Using this method of specific interval prioritization, Greek musicians created a wealth of scales to fit the artistic needs of a composition.

Scale degree	1	2	3	4	5	6	7	8
Ratio	1	$9/8$	$81/64$	$4/3$	$3/2$	$27/16$	$243/128$	2

Table 1.3: The Pythagorean Scale

In contrast, Simon Steven sought a division of the octave into twelve notes separated equally, which equates to the twelfth root of two [5]. With advances in mathematical computation, this calculation was able to be calculated and applied as the equal temperament tuning system. We might conclude equal temperament to

[2] This process is more analogous with selecting variations of string lengths or tautness than of selecting which piano keys to play, since the frequencies of each key are already set.

be an “advance” towards a better tuning system, considering it is the system used globally today. However, this shift occurred only after tumultuous debates in the seventeenth and eighteenth centuries to decide on the better tuning system. Really, this debate can be simplified as a choice between accuracy and precision: do we want our music to include more pure intervals or incorporate perfect yet inaccurate consistency? Johann Sebastian Bach contributed a significant argument in favor of the equal temperament system by composing “Well-Tempered Clavier,” which includes 24 preludes and fugues in both major and minor keys. This composition was made partly as a celebration of equal temperament, as it allowed for songs to be centered in tonalities that would have otherwise sounded peculiar due to the inaccuracies of the tuning system [5]. Without a doubt, Bach’s compositions influenced current perceptions on the ideal tuning system, which has become an absolute norm in Western European music.

1.3 Applications of Mathematics and Music

Despite Western music’s entirely mathematical origins, the continued development of music seems to have separated itself from mathematics, at least through the perspectives of most people. However, mathematical research continues to unveil new connections with music in various facets ranging from the physics of sound to the theory of pleasing melodies and rhythms. We provide a small set of applications below that attest to the fascinating diversity of this field of research.

1.3.1 Fourier Transform and Applications

The Fourier Transform gets its name from Jean Baptiste Fourier (1768-1830), a French mathematician. The work that led to this tool was originally aimed at solving a heat wave equation, headed chiefly by Daniel Bernouli, Lenhard Euler, and Jean-Baptiste D’Alembert [8]. Building off their theories, Fourier proposed that a wave could be equated with a sum of trigonometric components, now known as the *Fourier Series*. Not only did his claim prove true, but his insights have dispersed outward into many applied fields of mathematics.

In the realm of pure mathematics, the Fourier Series can be applied to approximate any continuous time-domain function on some interval such that the mean square error is minimized [3, p. 486]. Within this context, the series can be derived by attempting to find the least squares approximation of some time-domain function $f(x)$ through the use of a trigonometric polynomial.[3] The finite Fourier Series and its coefficients are shown below for an approximation on the interval $[-a, a]$:

$$f(x) \approx \frac{A_0}{2} + \sum_{k=1}^{n} \left(A_k \cos\frac{k\pi x}{a} + B_k \sin(\frac{k\pi x}{a} \right)$$

$$A_k = \frac{1}{a} \int_{-a}^{a} f(x) \cos\frac{k\pi x}{a} dx$$

$$B_k = \frac{1}{a} \int_{-a}^{a} f(x) \sin\frac{k\pi x}{a} dx$$

The coefficients A_k and B_k are obtained through concepts of linear algebra. It can be proven that the least squares approximation is equivalent to calculating the *orthogonal projection of* $\mathbf{f}$ onto a finite-dimensional vector space W ($\mathbf{f}$ is just the vector representation of $f(x)$). The coefficients to the Fourier Series are then based off an inner product between $\mathbf{f}$ and a set of vectors that form the orthonormal basis for W. For a more detailed derivation with theorems, see [3, p. 476-479]. Even without a discussion of the theory, questions are likely to be raised as to how the Fourier Series approximates a function. Figure 1.5 shows a simple example of a function approximated with increasing orders of the Fourier Series.

With reference to both the Fourier equations and Figure 1.5, it is reasonable to assume that the least squared error is further reduced as n gets larger because it incorporates trigonometric terms with higher frequencies. Moreover, the higher

[3] A trigonometric polynomial of order n is of the form $a_k \cos(kx) + b_k \sin(kx)$ where $k = 0, 1, \ldots, n-1$.

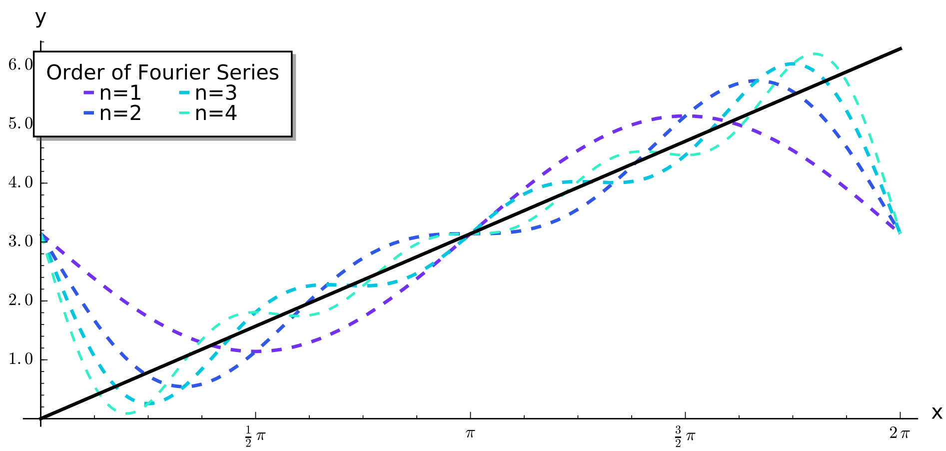


Figure 1.5: A Fourier Series approximation of $y = x$; approximation equations found in [3, p. 478]

frequencies seem to oscillate around the function with increasing nearness. In fact, as $n \to \infty$, the least square error is reduced to zero. Thus we have that

$$f(x) = \frac{A_0}{2} + \sum_{k=1}^{\infty} \left(A_k \cos\left(\frac{k\pi x}{a}\right) + B_k \sin\left(\frac{k\pi x}{a}\right)\right).$$

This equation implies that any continuous function can be approximated with a trigonometric polynomial over some interval. Of course, the Fourier Series alone has limitations in that the function $f(x)$ must be both known and integrable in order to calculate the necessary coefficients. The true beauty of the Fourier Series occurs when an *unknown* time-domain function—like a digital signal or a fluctuating air pressure—can be unraveled into a trigonometric polynomial.

In essence the *Fourier Transform*, derived from the Fourier Series, is used to decompose a single time-domain function into sinusoidal components in order to determine which frequencies contribute most to the formation of that function. The Fourier Transform is analogous to "unmixing" a can of paint into the separate colors that went into creating it [15]. The Fourier Transform works by *transforming* the Fourier Series from a time domain to a frequency domain [8], as shown below:

$$F_T[g(t)](f) = \int_{-\infty}^{\infty} g(t)e^{-2\pi i f t}dt.$$

Here $g(t)$ represents some function over time; the transform maps from frequencies f to the complex plane. An online visual animation (see [15]) describes the components of this equation in great detail, but we will summarize its contents below. This equation utilizes Euler's Formula, which defines an important trigonometric representation for e^{it}:

$$e^{it} = \cos t + i \sin t.$$

On the complex plane, e^{it} wraps around the unit circle at a rate of once every 2π units of t (often seconds). The Fourier Transform incorporates just a few considerations that modify this simple equation. Firstly, the presence of 2π in the exponent shifts the coordinate scaling system out of pi terms and into rectilinear coordinates. Secondly, $g(t)$ forms the amplitude such that, instead of maintaining distance from the origin of 1 as in the unit circle, the function factors in the value of g at that time (and forms undoubtedly beautiful radial graphs). Lastly, the expression is built within a function of frequency f so that we can vary the rate by which $g(t)$ is wrapped around the unit circle. As different frequencies are investigated, the resulting radial graphs change, sometimes lining up nicely and sometimes forming incredibly complex spirals. Essentially, the *integral* within the Fourier Transform works to quantify these spiral graphs into a single number that represents the graph's "center of gravity" relative to the origin. Any symmetry in the graph gets cancelled out within the integral, resulting in values close to zero. After calculating the integral over a broad spectrum of frequencies, however, some will be associated with a peak due to a strong asymmetry in the graph. These are precisely the frequencies that contribute most to $g(t)$—the colors mixed to form the can of paint.

Applications of the Fourier Transform are continually expanding into the realm of music. Currently, the air pressure fluctuation caused by an instrument note can be analyzed with the Fourier Transform to reveal a set of weighted peaks corresponding to the presence of harmonic overtones (imagine using Figure 1.2b to generate Figure 1.2a). More complex sound sources can also be analyzed, particularly to isolate unwanted frequencies. Once an unwanted frequency has been identified with the Fourier Transform, its peak can be "smooshed" and converted back to a signal with an inverse Fourier Transform [15]. In fact, sound manipulation software depends largely on clever applications of the Fourier Transform. A recent IEEE conference on signal processing contained a number of innovations in this field, including the ability to extract a singing voice from a music recording [17]. Although grounded in arguably simple concepts and relationships, the Fourier Transform is truly complex and one of the most important mathematical applications to music in a digitized age.

1.3.2 The Musical Group

Creating a musically acceptable composition is no simple task, even for a musician. Melody, rhythm, volume, and often harmony must be combined to create the familiar sounds of Western music. However, every additional musical element adds a layer of complexity to any algorithm or method that attempts to produce acceptable music. For this reason, we will begin with some mathematical applications in just one aspect of music: melody lines. More specifically, if given any melody line we seek a set of operations that we can perform on it to produce another melody.[4] Dr. Leon Harkleroad defines three such operations in his novel *The Math Behind the Music*. Firstly, a *transposition* is a vertical shift in either a note or a melody by some number of semitones. As an example, we can cause a vertical shift of 7 semitones on C with the following notation: $T_7(C) = G$. Secondly, a *retrograde* is a horizontal flip, meaning that a melody is played backwards. Finally, an *inversion* is a vertical flip about a reference note. Inversions preserve the intervalic distances between each successive note in a melody, but the notes creating the intervals are likely to have changed [9, p.

[4] Here a "melody" can be considered in more mathematical terms as "an ordered set of notes."

33-44]. Moreover, the melody will rise where it had previously fallen, and vice versa. An example of the three operations is presented in Figure 1.6.

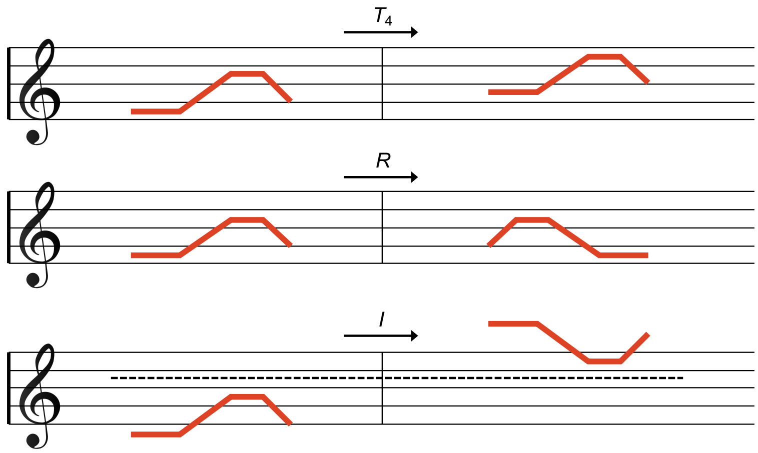


Figure 1.6: Transposition, retrograde, and inversion applied to the same melody line [9, p. 43].

We will now analyze the behaviors of these operations in order to define a "mathematical group" [9, p. 46]. To facilitate this discussion, we define a pitch class to contain all notes of the same letter. Therefore, A3 and A4 both fall within the "A" class, and this process can be repeated to form exactly 12 unique classes. In this way, the transpositions T_{12} and T_{-12} would transpose any note to its nearest octaves, but with pitch classes this would be considered a "do nothing" operation, just like T_0. Moreover, any transposition can be expressed equivalently as T_n, where $0 \leq n < 12$. These transpositions form a group isomorphic to $\mathbb{Z}_{12}$ with function composition defined by $T_a T_b = T_{a+b \pmod{12}}$. To remind the reader, in abstract algebra a set of elements G with a binary operation $*$ is a *group* if and only if the following conditions hold ($\forall$ is read as "for all," $\ni$ as "such that," and $\in$ as "in"):

1. G is "closed" on $*$, meaning that $a * b \in G \; \forall \; a, b \in G$,

2. The binary operation is associative,

3. There exists an identity element $e \ni ae = ea = a \; \forall \; a \in G$, and

4. Every element $a \in G$ has an inverse $a^{-1} \ni aa^{-1} = a^{-1}a = e$ [10, p.40].

In contrast to transpositions, inversions and retrogrades behave with an "on" and "off" nature, like a light switch. If a melody has been either flipped with an inversion or reversed with a retrograde, these actions can be undone by performing the same operation again. To be more precise, these elements are their own mathematical inverses: $II = RR = e$. Although inverses and retrogrades alone do not form particularly interesting groups, they can be combined to the group of transpositions to create a group of musical operations of the forms T_n, T_nI, T_nR, and T_nIR. These operations forms a group of 48 elements; later in the study we will identify a mathematical group to which it is isomorphic. These concepts of modern algebra may seem irrelevant to the world of music, but these patterns arise—perhaps without intention—in compositions ranging from Bach's counterpoint pieces to ballpark trumpet sounds [9, p.36]. When applied alone as a primary tactic for composition, these operations create fairly limited results, considering that an initial melody is required and that rhythm is not varied. Nonetheless, the occurrence of these operations in music suggests that music may be quite algorithmic in nature.

1.3.3 Change-Ringing and Permutation Groups

Change-ringing, first practiced in England in the early Seventeenth Century, is an art form that uses a set of bells within large towers to perform a musical arrangement. In short, these bells were rung repeatedly in a certain sequence (called a *change*) until the conductor would order a "call change," at which point adjacent bells would swap playing order according to their scores. Careful of the terminology here: a "change" refers to a sequence of bell chimes (e.g. 1423), not the difference between one sequence of chimes and the next. The ordered set of changes, known as an *extent*, would be complete only once the bells had cycled back through to the starting change without playing any change more than once [18]. What might seem to be an overly structured process for musical composition actually developed in accommodation to the time requirements and physical limitations of ringing large tower bells [9, p. 58-59]. As a result, we are left with the makings of a wonderful mathematical puzzle; moreover, the process of forming a successful extent is a direct application of the theory of permutation groups. Mathematically, permutation groups are defined as a subgroup of S_n, which is a group of $n!$ distinct elements having the composition of maps as its binary operation [10, p. 73]. Simply put, permutation groups rearrange a set of elements without deleting or repeating any of the elements. Although there exists a mathematical convention for representing permutations, we will define them as follows to avoid confusion: (AB) denotes a swap of slots A and B, as marked from left to right. This system of notation utilizes the fact that any permutation can be listed as a product of *transpositions*, or two-element swaps [10, p. 77]. Figure 1.7 depicts a sample permutation on a set of four bells, labeled by number (as is proper convention).

1 2 3 4
(AB)(CD)
2 1 4 3
(BC)
2 4 1 3

Figure 1.7: A permutation is used to swap adjacent bells.

One can likely see that, if we were to repeat the permutation (BC) again, the operation would undo itself just like with inversions and retrogrades. This observation can be extended to a theorem about transpositions: any even product of identical transpositions yields the identity element [10, p. 77]. So in this case $(BC)(BC) = (BC)^2 = e$. This highlights an important aspect of change-ringing: the composition of all permutations used in an extent should be equivalent to e, the identity element. In order to better illustrate this concept, we will use an extent known as *Plain Bob Minimus* written by Fabian Stedman, a prominent figure in the world of change-ringing [18]. The extent is written for four bells, and the score is provided below in Figure 1.8.

1234	1342	1423
2143	3124	4132
2413	3214	4312
4231	2341	3421
4321	2431	3241
3412	4213	2314
3142	4123	2134
1324	1432	1243
		1234

Figure 1.8: The change-ringing extent known as *Plain Bob Minimus*

Each column of the extent was generated by alternating between two permutations: $a = (AB)(CD)$ and $b = (BC)$, just like the example in Figure 1.7. In order to explicitly define this sequence of permutations, we can represent each permutation uniquely as the product of every permutation that came before it. Thus the set of permutations used to generate column 1 are

$$\{e, a, ab, aba, (ab)^2, (ab)^2a, (ab)^3, (ab)^3a\} = D_4. \quad [18]$$

Notice that, if the pattern was continued once more with the addition of b, then we would find that the resulting change would be identical to the first; thus $(ab)^4 = e$. Though represented in an unconventional way, the permutations listed above actually form a special subgroup of S_4 called the *Dihedral Group*, commonly denoted D_4.[5] In general, the *order*—number of distinct group elements—of the Dihedral subgroup D_n is $2n$, so in this case we can verify that the order is indeed $2(4) = 8$. Moreover, notice that 8 divides the order of S_n ($6! = 24$ elements), which aligns with *LaGrange's Theorem*; this states that the order of any subgroup divides the order of a group [10, p. 80, 92].

Now that we have established that the left column is a subgroup of S_4, we can look to examine the next two columns. As stated earlier, each of the three columns were generated by alternating permutations a and b. After reaching the end of the

[5]The Dihedral Group gets the name because its permutations can be represented by every unique transformation of an n-sided polygon. See [10, p. 80] for a helpful diagram.

first column, however, a new permutation, $c = (CD)$, was applied so that the piece would not resolve to the original change. In essence, the composer introduced a new permutation c, which acted upon every element of the subgroup. Moreover, this new set of permutations was completely *disjoint* from the subgroup comprised of the first column. Though Fabian Stedman likely did not known it, he performed a direct application of *cosets* in his change-ringing extent. Given a finite group G with subgroup H, a left coset[6] with representative $g \in G$ is defined as

$$gH = \{gh : h \in H\}.$$

So if we let $w = (ab)^3ac$ represent the non-trivial permutation that leads to the top of column 2, then the new set of permutations for column 2 can be written as

$$\{w, wa, w(ab), w(aba), w(ab)^2, w(ab)^2a, w(ab)^3, w(ab)^3a\} = wD_4. \quad [18]$$

Similarly, permutation c was applied to the bottom change of column 2 to produce the top change of column 3, and then D_4 was cycled through again. A nice theorem in modern algebra ensures that Stedman played every possible four-bell change: the set of all distinct (left) cosets are mutually disjoint, implying that they form a *partition* of the group [10, p. 92]. In application this means that, in order to create a successful extent where no change is omitted or repeated, simply find a subgroup of permutations—along with all the distinct cosets that it generates—and apply them successively. Clearly, change-ringing is a form of mathematics just as much as it is a form of music. Surely the final change of a successful extent, ringing out its last notes triumphantly from a lofty bell tower, pleased both the mathematician and the musician alike.

[6]right cosets also exist, but only left cosets are relavant to this application. Any further use of the term "coset" will imply a *left* coset.

1.3.4 Stochastic Music: Generation and Relationship

Aside from purely determined methods, a variety of attempts have been made to replicate music through stochasticity. We say "replicate" here because *music* is an inherently subjective term, so any stochastic creation will only be regarded as music if it sounds similar to common types of music. Though the idea of randomness often brings to mind the use of a computer algorithm, these concepts were applied long before the days of digital random number generators. In the mid-Eighteenth Century, a wave of consumer interest developed in Western Europe over musical dice games, "the musical equivalent of paint-by-number kits" [9, p. 71]. The movement began with Johann Philipp Kirnberger's first composition in 1957, translated as "The Ever-Ready Composer of Minuets and Phrases" [p. 72]. Essentially, musical dice games utilize the randomness of rolled dice to select a series of measures that are linked together and played. A composer begins the process of making a musical dice game by formulating a chord progression for some length of measures. For each part of the progression, the composer then writes several musical arrangements that fit well within the harmonic structure. Artfully crafted games will sound seamless to the listeners, as any selected measure will flow well with all possible measures that may precede or follow it. The attractiveness of this novelty stems from the idea that a single play through is likely never to be repeated by another (it seems as if the player really *is* the composer). This claim is well supported by a simple yet powerful aspect of compounded, random generations: the total number of possibilities is the product of all individual sets of options. Thus a game made of m measures each with n options would have a total number of compositions equal to n^m. Kirnberger's game, for instance, consisted of only 14 measures, but each were given 11 different options to be determined via dice roll. To play all possible variations of Kirnberger's game, a player would have to perform over 379 *trillion* different songs!

Another surge of interest in stochastic music popped up in the mid-Twentieth Century in response to the exciting potential of computer algorithms. Instead of building a whole song by selecting from prepared pieces, the research tended to focus more on the algorithmic generation of individual musical elements, all linked together at the end to create music. We will highlight the research of F. P. Brooks and his colleagues at Harvard, published in 1957 (see [6] for further incite on the information we present below). Ironically, Brooks and his colleagues originally determined to establish and test a set of indicators for unsuccessful algorithms; they pursued music composition mainly as an applicable means to accomplish their primary research goal. Nonetheless, the team devised a technique that sampled a small set of only 37 hymnal songs[7] and used them to generate new, acceptable melodies based on m-order *Markov chains* ($1 \leq m \leq 8$). A Markov chain of order m determines the probabilities of possible results based on the previous $(m-1)$ elements. In more visual terms, a Markov chain is analogous to a set of "stacked" decks, meaning that each deck is more likely to draw forth some cards than others. The particular deck from which one would make the next draw depends on the results of the previous draws, and the amount of previous draws one chooses to consider denotes the length of the Markov chain [9, p. 87]. As an illustration, consider the following Markov deck presented by the research team for a three-note sequence, $C - E - G$, shown below in Figure 1.9. Had the sequence instead relied on just $E - G$, then the deck of notes generated would likely contain an entirely different set of notes with different occurrences, all of which are based on the notes of the sample hymns.

The research team applied the sampled note probabilities to an executable algorithm by assigning each note a weighted, non-overlapping interval between 0 and 1 such that all intervals at a given timestep would fill the range from 0 to 1 (see Figure 1.9). A random number was then generated between 0 and 1; the interval within which the number fell would be the next note in the melody. This process of assigning weighted probabilities and randomly selecting a note was repeated for all m-order Markov chains between 1 and 8, thereby altering the interval ranges to

[7] Each hymn was transposed to the key of C for more meaningful probability sequences. As the authors note, however, this generalization does not take into account the fact that hymns are written for a natural vocal range, which affects the melody based on the chosen key.

C—E—G—A
C—E—G—A
C—E—G—C
C—E—G—E
C—E—G—E
C—E—G—G
C—E—G—G
C—E—G—G
C—E—G—G
C—E—G—G

Probability	Weighted Interval
$P(A) = {}^{2}/_{10} = 0.2$	$0 \leq x \leq 0.2$
$P(C) = {}^{1}/_{10} = 0.1$	$0.2 < x \leq 0.3$
$P(E) = {}^{2}/_{10} = 0.2$	$0.3 < x \leq 0.5$
$P(G) = {}^{5}/_{10} = 0.5$	$0.5 < x \leq 1$

Figure 1.9: A fourth-order Markov deck. Respective probabilities are calculated and then assigned a weighted interval from 0 to 1.

account for an increasing sense of "memory" of what was just played. In addition to note pitches, Brooks and his colleagues diversified the rhythm by incorporating a few options for structure. This system broke the tune down into 64 evenly spaced beats and determined whether to "sustain" the note of the previous beat or "strike" a new note, in which case the Markov chain was initiated. Every rhythmic system utilized in the experiment incorporated some degree of constraint in order to conform to the common practices of hymnal tunes.

Metric Constraint	Order of Analysis m							
	1	2	3	4	5	6	7	8
Even Quarters	100	40	32	10.8	9.2	3	2.5	8.4
Quarters with Various Options	/	41	26	15	11	2	/	10
Dotted	100	13.5	5	2	1	0	0	0
Dotted with Various Options	100	17	4	1.5	0	0	0	0
Skeleton	100	/	/	5	4	3	2.4	10

Table 1.4: Yield percentages for acceptable hymns at various chain lengths and metric constraints, adapted from [6]. A / indicates no experiment.

The results of this study were summarized as yield percentages of an acceptable[8] hymn generation for each value of m. Table 1.4 is a recreation of these results. Note that the monogram ($m = 1$) does not base probabilities on previous notes, so the selected notes are chosen from a single deck that considers all the notes played in

[8]We use "acceptable" only as the researchers did; admittedly this term was never fully defined.

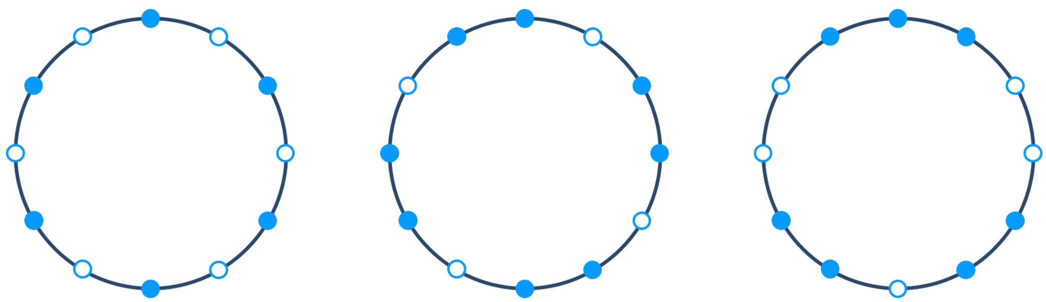
every sample hymn. The monogram results are classified as acceptable, but most contain unnatural intervals that would not typically appear in music. In contrast, the octograms also have a relatively high yield percentage, but only because the Markov chain was too long for the limited diversity of sample pieces. Reportedly, a notable portion of the octogram-generated melodies were exact replicas of sampled hymn melodies, thus indicating that the sample diversity was not great enough to produce acceptable *and* unique melodies.

Despite these shortcomings, the generations for ($m = 4$) and ($m = 5$) seemed to produce the highest rate of new and acceptable melodies. In addition, the incorporation of rhythmic diversity of struck and sustained notes, though less often successful, seemed to add a more realistic quality than the samples of consistently even note duration. Of course, the overall quality of hymns generated are subjectively lower than those composed artistically. Nonetheless, these early attempts at music generation algorithms set a foundation from which research has continually improved. We have presented a relatively rudimentary example for ease of discussion, but a quick internet search will yield examples of generated jazz melodies that to some listeners might sound indistinguishable from live recordings. Whether simulated using a computer or a pair of dice, it seems that randomness must be coupled with structure to produce more acceptable music, although it should be noted that the full potential of this research field has yet to be solidified.

1.3.5 Perfect Balance in Music

As a final application we introduce two geometric principles—perfect evenness and perfect balance—that can be found within the formation of both scales and rhythms. Andrew J. Milne and his research colleagues investigated the nature of perfect evenness and perfect balance[12] and later detailed an algorithm designed to apply them to rhythm[13]. We incorporate this work in the simulation of group music, to be presented later in the study. The remainder of this section attempts to summarize and understand the information presented in [12].

To begin, we must first recall the concept of pitch classes when thinking of "The Musical Group;" there we categorized an infinite set of discrete pitches into a finite set of classes. This structure is similar to the group $\mathbb{Z}_n$ and is indeed isomorphic to a group built from these groups, as we will see later in the study. For smaller values of n, $\mathbb{Z}_n$ is commonly represented with n nodes placed evenly around a circle, as a way of visualizing the modular qualities of the group operation. If pitch and rhythm are viewed in this same way, we can apply principles of geometry to better understand these musical elements.



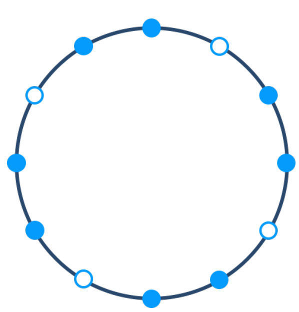
(a) Perfectly balanced, perfectly even, sub-periodic

(b) Perfectly balanced, uneven, sub-periodic

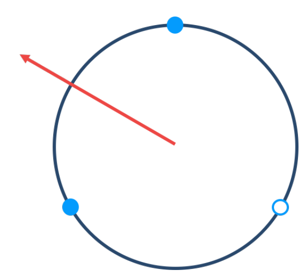
(c) Perfectly balanced, uneven, no sub-periods

Figure 1.10: A demonstration of perfect balance and perfect evenness, in both meter and scale [12].

Figure 1.10 (above) depicts this representation for a group of order 12, based on the chromatic scale—a scale where the interval between any two adjacent notes is a semitone. Each filled node indicates a tone used in the scale. If, for example, Figure 1.10 represented the chromatic notes with C at the top, then the resulting scale would be $C - D - E - F\sharp - G\sharp - A\sharp$. This could also represent a period of time divided into twelve even intervals, where filled nodes indicate a struck beat (of a drum, bell, etc.) and unfilled nodes indicate silence. With a visual now readily accessible, we can proceed to definitions without unnecessary confusion. *Perfect evenness* (PE), a well-established concept in music, arises when a scale or rhythm is composed of a repeating pattern of "on" and "off" segments (Figure 1.10a). For instance, the above scale example is perfectly even because it consists of a repeating pattern of every other chromatic note on the keyboard. In contrast to perfect evenness, Milne and his colleagues define periodic patterns (scales or rhythms) to be *perfectly balanced* (PB) if, when wrapped around a periodic circle as in Figure 1.10, "their 'center of gravity' is precisely at the circle's center." They also provide a helpful analogy: imagine the filled nodes as weights placed around a frictionless, free-spinning wheel. When the wheel is released, any distribution of weight that is not perfectly balanced will cause the wheel to rotate until the resultant weight vector is pointed perfectly downward. An important distinction between these two concepts lies within their relationship with one another. In mathematical terms, $PE \rightarrow PB$. Perfect evenness implies perfect balance, but perfect balance does not guarantee perfect evenness. Equivalently, the absence of perfect balance implies an absence of perfect evenness. Lastly, the non-periodic requirement displayed in Figure 1.10c—also referred to as "irreducibly periodic"—denotes that the circle shown could not be generated by a circle with less nodes. An irreducible periodic system with perfect balance guarantees that the *fundamental period*, found through smallest representation of the system, is perfectly balanced as well as all other possible periods. Figure 1.11 shows an example of a rhythm that is reduced to its fundamental period to reveal an unbalanced arrangement.



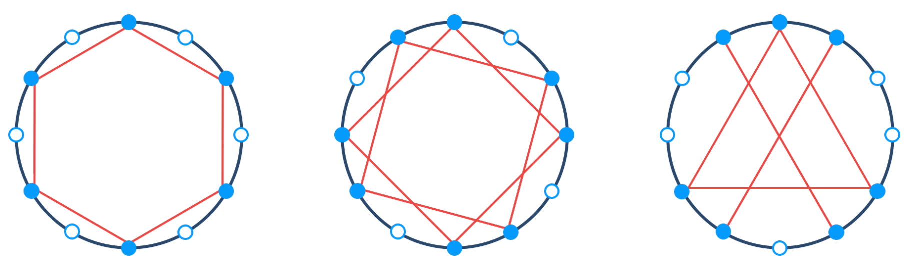
(a) perfectly balanced, sub-periodic



(b) unbalanced fundamental period

Figure 1.11: A perfectly balanced, sub-periodic arrangement that yields an unbalanced fundamental period (based on [12]).

Relating this back to music, one can imagine that a perfectly even rhythm would be subjectively quite boring, yet a non-periodic and unbalanced rhythm is inherently challenging to internalize. Perfectly balanced rhythms that are irreducibly periodic, however, are quite interesting from a musical perspective. As one could guess, meeting these requirements with an increasing number of nodes around the circle makes for longer and more interesting rhythmic structures. In fact, the research of Milne and his colleagues focuses on finding all arrangements that show irreducibly periodic perfectly balance (IPPB) for larger values of n. The researchers present a "heuristic" method of finding IPPB using simple geometry. For an "N-fold universe"—represented by a circle with N nodes—satisfactory arrangements of IPPB can be produced by interposing two regular polygons with non-overlapping vertices such that the number of vertices are relatively prime to one another. Figure 1.12c applies this method to the previously introduced arrangement from Figure 1.10c. Six is the smallest value of N capable of producing IPPB; we can calculate additional values by doubling the product of two relatively prime integers:[9] $2 \times 2 \times 3 = 12, 2 \times 2 \times 5 = 20, 2 \times 3 \times 4 = 24$, and so on. When only one polygon is used (Figure 1.12a) or the same polygon is superimposed (Figure 1.12b), an arrangement is created that is perfectly balanced but *not* irreducibly periodic.



(a) Perfectly balanced, perfectly even, sub-periodic

(b) Perfectly balanced, uneven, sub-periodic

(c) Perfectly balanced, uneven, no sub-periods

Figure 1.12: Using regular polygons to generate the same perfect balance and perfect evenness shown in Figure 1.10.

[9] As Milne and his colleagues explain, the factor of 2 exists so that one of the polygons may be rotated in order to obtain disjoint vertices.

To be plain, the geometric method described here is a fun way to create musically interesting rhythms and scales for small values of N. Of course, as N increases past small rhythm sizes, and as we extend this method into more than two regular polygons, the level of complexity quickly exceeds the capability of hand calculations and visual imagination. Fortunately, Milne and his colleagues also developed a system to represent arrangements in vector notation, which is readily applicable to programming environments. We halt any further discussion of this topic, however, since the method already described is sufficient for our purposes in later sections.

Chapter 2

The Musical Group

2.1 Background

Recall from Section 1.3.2 our discussion of the musical group of operations. Miraculously, a combination of transpositions, inversions, and retrogrades applied to a melody line can come together to obey the necessary properties of a mathematical group. This observation begs a familiar question in the realm of abstract algebra: can we classify this group in a way that tells us more about its properties?

A classification is formally established by proving that there exists an *isomorphism* mapping from the group in question to a more familiar group with well-understood structure. We refer again to our Abstract Algebra text for a formal definition: for two groups $(G, \cdot)$ and $(H, \circ)$, an isomorphism is defined as a one-to-one and onto mapping $\phi : G \to H$ such that group operation is preserved:

$$\phi(a \cdot b) = \phi(a) \circ \phi(b) \quad \forall a, b \in G. \quad [10\text{, p. } 140]$$

If the existence of an isomorphism can be proven, the two groups are said to be *isomorphic*, meaning that they are identical in group structure but perhaps different in how they are represented. Clearly, a blind search for an isomorphic group poses a significant challenge. Using the Groups, Algorithms, Programming (GAP) software package, one can easily determine that there are a total of 52 distinct groups of order 48 [1]. Hence it is not pragmatic to go through the list and successively eliminate groups until we have found an isomorphic pair.

Fortunately, we bypassed this process and arrived at an answer by using a common visualization of Dihedral groups, namely the Dihedral group of order 24. This group, which we will denote D_{12}, can be thought of as all the symmetries of a regular 12-gon. By symmetries we mean any set of moves that would leave a shape looking exactly as it was before being acted upon; for instance, a square rotated 90 degrees is a symmetry, but a square rotated 45 degrees is not. Visually, we can imagine the set of all symmetries for D_{12} as such: rotate the 12-gon through each position (like going through the 12 hours on a clock), flip it over the y-axis, and then rotate it through each position again. Thus we can generate each element of D_{12}, and therefore the whole of D_{12}, through a combination of rotations r and flips s:

$$D_{12} = \{r^n, s^k \mid k, m \in \mathbb{Z}, r^{12} = s^2 = 1, \text{ and } srs = r^{-1}\}.$$

We now further explain this structure with visual reference to Figure 2.1. Each clockwise rotation r can be viewed as a permutation that shifts the verticies of a 12-gon to new positions: $r = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)$. Therefore, the order of r is 12, and we can observe that $r^{12} = e$ and $r^{-1} = r^{11}$. In general, $(r^k)^{-1} = r^{12-k} = r^{-k}$ as we would expect. We can represent the flip s in a similar fashion: $s = (1, 12)(2, 11)(3, 10)(4, 9)(5, 8)(6, 7)$. Since s is the product of mutually exclusive transpositions, $s^2 = e$ and $s = s^{-1}$. Lastly, either through computing permutations or by utilizing the visual aid, we can show that $srs^{-1} = r^{-1}$, or equivalently, $sr = r^{-1}s$.

This final relationship between rotations and flips in D_{12} looks remarkably similar to our earlier relationship between musical transpositions and the musical inversion: $T_nI = IT_{12-n}$. With this key observation in place, an isomorphism seems to fall right out of the structures already present in the musical operations. As we show in the proof below, the important step is to map rotations and flips with musical

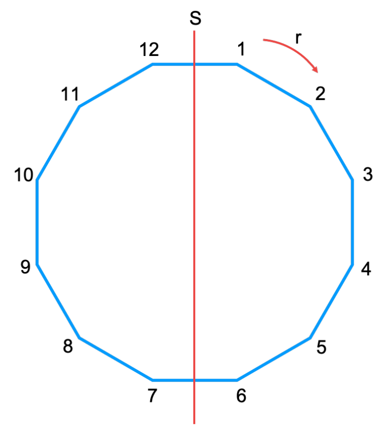


Figure 2.1: The generator elements of D_{12}

transpositions and inversions, respectively. The details of the proof are included in the following section.

2.2 Proof of the Musical Group

As an extension to the Musical Group first introduced by Dr. Harkleroad, we provide an isomorphic relationship between the set of musical operations and a direct product of more familiar groups.

Theorem 2.1. *The Musical Group G—composed of the sets of musical transpositions T, inversions I, and retrogrades R all under function composition—is isomorphic to $TI \times R$, which in turn is isomorphic to $D_{12} \times Z_2$.*

We will verify Theorem 2.1 by proving a series of successively inclusive statements. Firstly we will show that the product group TI is a subgroup of G. Secondly, we will establish an isomorphism between TI and D_{12}. We will then extend this isomorphism to show that the direct products $TI \times R$ and $D_{12} \times \mathbb{Z}_2$ are also isomorphic. Lastly, we need only to affirm that $TI \times R$ is a legitimate representation of G by identifying an isomorphism between the two groups.

Proof. Consider $TI = \{t^n i^k \mid t \in T, i \in I, \text{ and } n, k \in \mathbb{Z}\}$. We will show that this is a subgroup of G. Firstly, note that if k is even for i^k, then $i^k = 1$ as $i^2 = 1$. Also, an important observation is that the interaction between transposition elements and the inversion element commutes but with the "penalty" of leaving the transposition as its inverse:

$$t^n i = it^{12-n} = it^{12}t^{-n} = it^{-n}.$$

With this established, we can now prove that TI is a subgroup by utilizing the one-step subgroup test. Clearly, TI is nonempty. Now, we will show that for some $x, y \in TI, xy^{-1} \in TI$. For $x = t^m i^h$ and $y = t^n i^k$, we must check four cases because the parity of h and k affect the nature of the transposition elements. We provide the case that both h and k are odd below, and claim that the other three cases will yeild similar results that pass the one-step subgroup test.

$$\begin{aligned}
xy^{-1} &= (t^m i^h)(t^n i^k)^{-1} \\
&= t^m i^h i^{-k} t^{-n} \\
&= t^m i^h t^n i^{-k} \\
&= t^m t^{-n} i^h i^{-k} \\
&= t^{m-n} i^{h-k} \in TI
\end{aligned}$$

Therefore, we have concluded through the one-step subgroup test that $TI \leq G$.

We define a homomorphism $\phi : TI \to D_{12}$ as follows:

$$\phi = \begin{cases} t^n & \longmapsto r^n \\ i^k & \longmapsto s^k \\ t^n i^k & \longmapsto r^n s^k \end{cases}$$

Let $m, n, h, k \in \mathbb{Z}$. This time, we need only two cases to account for the parity of h:

Case 1: Assume h is even (thus i^h is the identity element).

$$\phi\left((t^m i^h)(t^n i^k)\right) = \phi\left(t^m(1)t^n i^k\right) = \phi\left(t^{m+n} i^k\right) = r^{m+n} s^k$$

$$\phi\left(t^m i^h\right)\phi\left(t^n i^k\right) = \phi\left(t^m(1)\right)\phi\left(t^n i^k\right) = (r^m)(r^n s^k) = r^{m+n} s^k$$

Case 2: Assume h is odd. Recall that $t^n i = it^{-n}$ and $r^n s = sr^{-n}$.

$$\phi\left((t^m i^h)(t^n i^k)\right) = \phi\left(t^m t^{-n} i^h i^k\right) = \phi\left(t^{m-n} i^{h+k}\right) = r^{m-n} s^{h+k}$$

$$\phi\left(t^m i^h\right)\phi\left(t^n i^k\right) = r^m s^h r^n s^k = r^m r^{-n} s^h s^k = r^{m-n} s^{h+k}$$

Thus in either case, the operation is preserved. Moreover, we can easily see by our definition of ϕ that it is a bijection: each element in TI maps to a distinct element in D_{12} (one-to-one) such that all elements in D_{12} are mapped to (onto). Therefore, $\phi : TI \to D_{12}$ is an isomorphism, and $TI \cong D_{12}$.

We now show that $TI \times R \cong D_{12} \times \mathbb{Z}_2$. However, this is trivial. Since we know $TI \cong D_{12}$ and $R \cong \mathbb{Z}_2$, we can conclude that $TI \times R \cong D_{12} \times \mathbb{Z}_2$ because each isomorphism is preserved within its respective side of the direct product. Specifically, this happens through the isomorphism $\phi : TI \times R \to D_{12} \times \mathbb{Z}_2$ via

$$\phi(t^a i^b, r^c) = (r^a s^b, c_{\text{mod } 2}) \text{ for } a, b, c \in \mathbb{Z}.$$[1]

Lastly, we show that $G \cong TI \times R$. We define the mapping ϕ below and will show it is an isomorphism:

$$\phi = \begin{cases} t^n i^k & \longmapsto (t^n i^k, 1) \\ t^n i^k r & \longmapsto (t^n i^k, r) \end{cases}.$$

[1]Note here that r^c is the *musical* rotation whereas r^a is the *Dihedral* rotation. We allow this ambiguity in order to avoid straying from the previously established naming conventions.

As the order of R is 2, it is clear that this mapping is a bijection. Again, we must break into four cases, this time to account for whether the retrograde operation is present in one or both of the group elements. Fortunately, it is easy to verify that the homomorphic property is preserved because the retrograde commutes with the other musical operations. We depict this by showing the case of retrograde present only in the first element:

$$\begin{aligned}
\phi\left((t^m i^h r)(t^n i^k)\right) = \phi\left(t^m i^h t^n i^k r\right) &= (t^m i^h t^n i^k, r) \\
&= (t^m i^h t^n i^k, 1r) \\
&= (t^m i^h, 1)(t^n i^k, r) = \phi\left(t^m i^h\right)\phi\left(t^n i^k r\right)
\end{aligned}$$

The other three cases are proven with similar arguments. Thus $G \cong TI \times R$. Therefore, we have proven that $G \cong TI \times R \cong D_{12} \times \mathbb{Z}_2$. $\square$

This theorem allows for the potential of a much deeper understanding of the structure of the Musical Group because we have a clear connection to an established and well-known group. From here, the structure of $D_{12} \times \mathbb{Z}_2$ could be examined in the context of the Musical Group in search of interesting applications. For example, what sets of musical variation can be produced from its different subgroups? Questions of this nature remain unanswered as of the time of this thesis.

Chapter 3

Generating Polyrhythms from Cyclic Subgroup Structures

As alluded to in Chapter 1, we dedicate the latter half of this study to examining a particular relationship between mathematics and music: the subgroup structures of cyclic integer groups in the context of rhythm. As a result, we created an interactive software tool that allows its users to freely compare the rhythms generated by cyclic groups of various orders, with additional control over stylistic elements such as tempo and sound packages. The program is accentuated visually by a 32x32 LED matrix panel that helps users to understand the relationship between the rhythm being played and its underlying subgroup structures. In the following sections, we delineate the process of formulating this tool, starting with the logic of the code and moving into the technology utilized to execute it.

3.1 Perfect Balance in Subgroup Structures

Before transitioning into the nature of our research, we list a few important group properties that will aid understanding. Recall that a group is called *cyclic* if it can be represented by a *generator*—a single element that can produce any other group element by operating on itself some number of times [10, p. 56]. So for a cyclic group G with generator g,

$$G = \langle g \rangle = \{g^k \mid k \in \mathbb{Z}\}.$$

A related theorem states that any subgroup of a cyclic group is also cyclic. These properties are substantially useful in coding because they imply that all the elements of a cyclic group or subgroup can be represented by its generator, which can be stored as a single variable. Finally, we have through LaGrange's Theorem a distinct relationship between the structures of subgroups and their heretical group: the order of a subgroup divides the order of the group. These theorems provide an intuitive, programmatic method for calculating and representing cyclic subgroups.

The research presented in this thesis deals specifically with finite subsets of the group of integers under addition; we denote $\mathbb{Z}_n$ to mean "the integer group of order n", or equivalently, "the group of integers under addition mod n" (where mod is the modulus operator). We will occasionally reference $\mathbb{Z}_{12}$ as a simple example to illustrate the relationships presented below. $\mathbb{Z}_{12}$ contains the integers 0 through 11, and its subgroup structure is depicted in Figure 3.1.[1]

This thesis builds upon the research of Andrew J. Milne and his colleagues, as described in Section 1.3.5. We can imagine Milne's "N-fold universe" instead as a cyclic group of order N. Referring back to Figure 1.12, we can think of this ring of 12 nodes as the cyclic group $\mathbb{Z}_{12}$, with the elements $\{0, 1, 2, \ldots, 11\}$ mapped around clockwise starting from the top of the circle. Just as a second hand moves around the numbers on a clock, we can iterate through the elements of $\mathbb{Z}_{12}$ at a constant speed to lay the foundation of a rhythm. We have now established a connection between the elements of cyclic groups and the basis from which a rhythm will be built; next we look to establish a novel approach to the rhythm itself. We assign each

[1] Note, generators are not unique; as one can show, $\langle 2 \rangle = \langle 8 \rangle$ in $\mathbb{Z}_{12}$. However, it is sufficient for the purposes of this study to refer only to a subgroup's smallest generator.

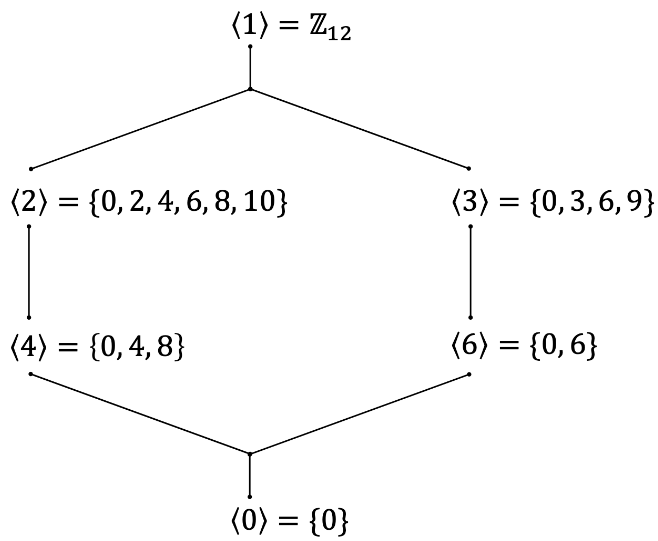


Figure 3.1: The distinct subgroups of $\mathbb{Z}_{12}$, where lines represent set containment.

nontrivial subgroup[2] to an arbitrary sound, such as a snare tap or a drum kick. By definition, the elements of each subgroup are elements of the group in consideration. Thus we develop a method that iterates through the elements of the collective group in order from least to greatest at a constant rate, resetting back to 0 after reaching the final element n. During each iterative step, the sound of a subgroup is triggered whenever the iterator points to a member of that subgroup. Continuing with our earlier example, we visualize the subgroups of $\mathbb{Z}_{12}$ in Figure 3.2. As opposed to Figure 1.12, now each polygon represents a distinct, nontrivial subgroup of $\mathbb{Z}_{12}$.

The above example reveals two important characteristics that are shared with every group investigated in this study. Firstly, as every subgroup contains the identity element 0, every polygon overlaps at this point. For some groups (namely, the most interesting in terms of rhythm) this will be the *only* point where all subgroups overlap. Secondly, each subgroup is represented with a straight line or a *regular* polygon (e.g. a subgroup of order 3 always forms an equilateral triangle). This property follows

[2]In contrast, the trivial subgroups are the set containing the identity element and the set of all elements. As one can imagine, these two subgroups do not add rhythmic complexity, so we do not assign them a sound.

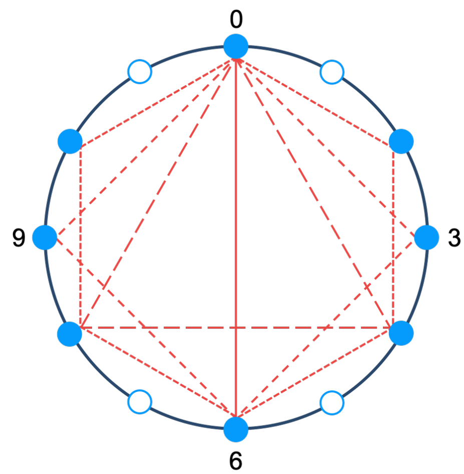


Figure 3.2: The subgroup structure of $\mathbb{Z}_{12}$, mapped with distinct polygons.

directly from the fact that subgroups are generated by a single element. Thus any subgroup on its own generates a perfectly even, sub-periodic rhythm.

As we will see, the application of subgroup structure into rhythm can produce the irreducibly periodic perfect balance that captured the interest of Milne and his colleagues. Moreover, we will expand Milne's "heuristic approach" to identify a particular group property necessary to developing rhythms of interest.

3.2 Development of an Interactive Tool

We execute our software from a Raspberry Pi model B+, a microcomputer equipped with a 1.4GHz 64-bit quad processor. The appeal of the Raspberry Pi for this research is described best through the words of its nonprofit foundation: "We provide low-cost, high-performance computers that people use to learn, solve problems, and have fun."[7] In addition to a great cost-performance ratio, the Pi gives easy access and compatibility to external hardware with a 40-pin GPIO (general purpose input-output) header. The LED matrix (discussed in detail further on) was connected to the Pi with an Adafruit matrix "bonnet"—an adapter piece between the input/output cables of the matrix and the Pi's GPIO header. Clearly, the functionality and portability of this hardware setup made good on the Pi Foundation's promise to "solve problems." Moreover, the process of mounting and configuring these various hardware components contributed a unique educational component that enriched the overall experience of this thesis. With that said, the executable software is capable of being downloaded and run from any computer with access to the internet; the visual aid through the matrix panel is lost, but this does not affect the primary functionality of generating polyrhythms.

The heart of the code was written in Sonic Pi—"The live coding music synth for everyone."[2] A variant of Python, Sonic Pi provides a seamless coding environment for the production of music, equipped with a vast array of sound samples from bass drum kicks to bird calls. Moreover, Sonic Pi integrates communication with remote devices through platforms such as Musical Instrument Digital Interface (MIDI) and Open Sound Control (OSC).

Since a major goal for this thesis was the development of a tool that was interactive and intuitive, much of the research and writing was devoted to the communication between different software packages. Ultimately, we utilize an iPad application named touchOSC (developed by Hexler) to send OSC messages to a running Python script, which then handles the signals and passes appropriate, well-timed commands to both Sonic Pi and the LED matrix panel. So for some group size n, Sonic Pi produces the rhythm, and the matrix panel compliments this audio by displaying a ring of n points (one for each group element) and highlighting the point which has the current beat. All the while the Python script handles the communication between user-input, audio, and visuals. This system allows for easy comprehension of where the rhythm is in its progression and helps to distinguish the various group structures available.

TouchOSC was chosen for its ability to make a custom control panel; Figure 3.3 is a screenshot of the interface that users control. Our Python script relies on the module "pythonosc," written by Github user attwad (see [4]), which utilizes built-in Python objects such as socketservers and threads to match OSC addresses with appropriate handling functions. In order to maintain the best timing between

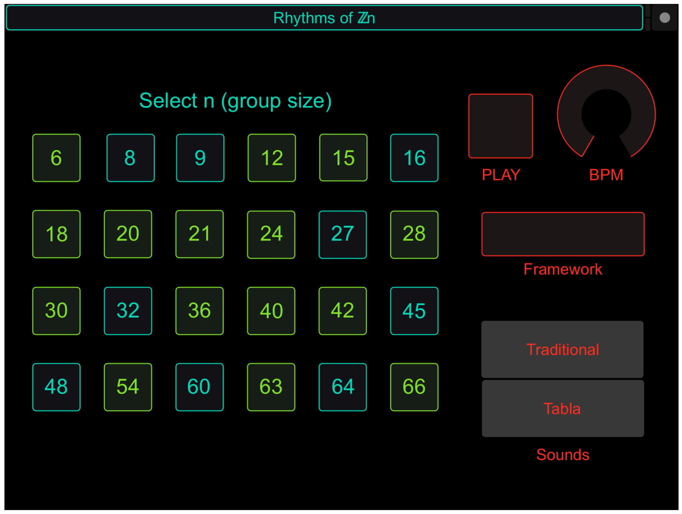


Figure 3.3: A screenshot of our layout design through touchOSC. Users can control group size, beats/minute, and the current sound package all in real time.

programs, outgoing OSC signals are sent from Sonic Pi back to the Python handling script when it is time to cue a shift in display. Hence the handling script has two main functions: receive touchOSC signals from the user and pass on modified signals to Sonic Pi, and receive OSC signals from Sonic Pi to trigger display commands for the matrix panel. We encapsulated the various interactions with the matrix panel in a class called RingRunner. Our class interacts with the matrix through a package designed by Henner Zeller (see [19]), which allows us to set the color and brightness of individual pixels. Working with a class instance also serves to track the state of the matrix among multiple threads.

Great rhythm is inherently dependent on great timing; thus one of the biggest challenges in the development of this tool was maintaining timely interaction between user input, audio, and visuals. Fortunately, Sonic Pi was designed in anticipation to this very issue. Using built-in "get" and "set" functions in the right places allows for code whose reference values can constantly update without corrupting the current rhythm. In this way, the Sonic Pi code interacts perfectly with user input. However, a particular timing issue still persists between Sonic Pi and the LED matrix. Despite frequent—even redundant—checks to ensure that the matrix waits on the sound before updating, the animation appears to be slightly ahead of the sound. At lower tempos this phenomenon is hardly noticeable, but at higher tempos the animation becomes more and more offset until it is at most one full beat ahead of the rhythm. Despite this minor flaw, the matrix panel sufficiently provides a handy visual reference for a better user experience.

Lastly, we elected to 3D print a case to house the Pi and matrix for a more aesthetic presentation. Figure 3.4 shows the end-product design. We chose the Autodesk Tinkercad platform to create the 3D mesh for the case; the software's simple and intuitive interface was more than sufficient for our needs. The case consists of two parts: a box to hold the Raspberry Pi with slots for wiring, and a lid on which the matrix is fastened. When presenting, the lid can be inserted into the side of the box to hold the matrix upright.

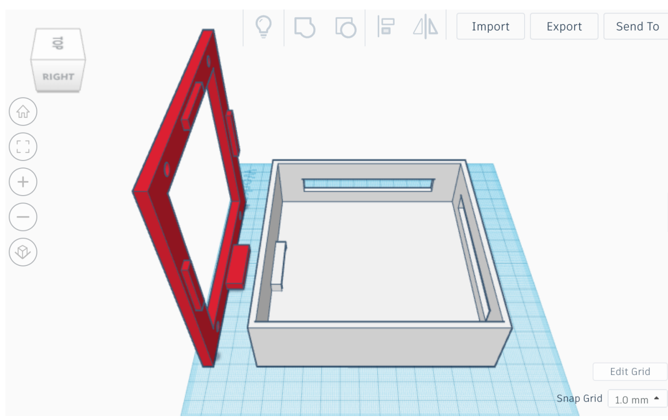


Figure 3.4: The final schematics for the case, created in Autodesk Tinkercad.

We exported two mesh files from Tinkercad—one for each part—sliced them using Ultimaker Cura, and printed them with a Balco Touch 3D printer. The printer handled both jobs with remarkable precision—only minimal sanding was required. Figure 3.5 provides an image of the setup in "presentation mode." When the Pi is shut down for transportation or temporary storage, the matrix can remain on the lid as it fastens down onto the box.

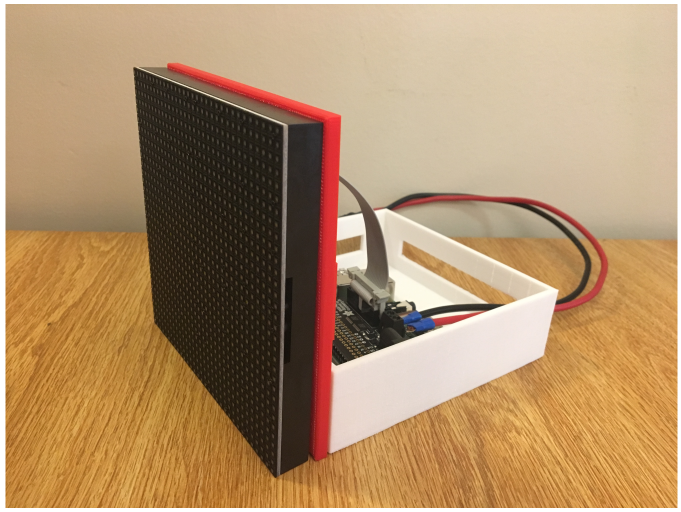


Figure 3.5: A photograph of the LED matrix mounted to the lid and fastened upright for presentation.

3.3 Results

As the primary goal of this chapter of the thesis was the *development* of an interactive tool, the first result is that the tool was, in fact, developed successfully. In addition to a successful process, the tool was also used to better understand rhythmic perfect balance and its relationship to subgroup structures. Firstly, since every subgroup generates a perfectly even rhythm, and since perfect evenness implies perfect balance, every cyclic group creates a perfectly balanced rhythm. However, not all groups sizes lead to IPPB—perfect balance *without* sub-periods. Recall from Section 1.3.5 our description of Milne's method for finding sizes of N that are capable of IPPB: double the product of any two relatively prime integers. It is important to note here that Milne initially defines perfect balance with the criteria that no point has a "weight" of more than one. In other words, our overlapping polygons would violate this criteria because the 0 point would have a weight of N. However, this criteria is not necessary to achieve perfect balance when thinking in terms of weight distribution. Therefore, we expand the definition for our purposes to include points of overlap. We also expand the method to find group sizes capable of producing IPPB:

Theorem 3.1. *Let g_1, g_2 be generators of two distinct subgroups of a group G such that g_1 and g_1 are the smallest representations of their respective subgroups. If the least common multiple $LCM(g_1, g_2) = |G|$, then G forms a rhythm that achieves irreducibly periodic perfect balance (IPPB).*

This theorem follows from the observation that all group polyrhythms start their cycle with a simultaneous playing of all sounds (because the identity element 0 is found in every subgroup). Thus if all subgroups align to play their sounds at any *other* element, then the polyrhythm must include sub-periods, because the cycle has started over again before reaching the identity element. Hence we need only to find two subgroups that share *only* the identity element to ensure that we cannot reduce the rhythm to sub-periods. We sum up this logic formally below.

Proof. Let g_1, g_2 be generators of two distinct subgroups of a group G, and set $N = |G|$. Assume $LCM(g_1, g_2) = N$.

We already know that G will form a perfectly balanced rhythm because G is cyclic. Thus we need only to show that it does not have sub-periods. Based on the nature of subgroup structures as regular polygons/straight lines, we can conclude that a sub-period occurs when all subgroups intersect at a group element other than the identity element.

In other words, we must negate the following statement: at some integer $P \in (0, N)$, all subgroups intersect at P (i.e. a sub-period begins at P). Hence we need to show the following: *for all* integers $P \in (0, N)$, there exist *at least two* subgroups that do *not* intersect at P.

By hypothesis, $LCM(g_1, g_2) = N$ implies that $\langle g_1 \rangle$ and $\langle g_2 \rangle$ only intersect at the identity element. Therefore, $LCM(g_1, g_2) = N$ implies that G forms a perfectly balanced rhythm without sub-periods, so G achieves IPPB. □

The source code for our tool is stored in in the Wiki pages of Maryville College's Mathematics and Computer Science Division (see [16]). This way the code can be continually updated and improved even after the conclusion of this thesis. The page includes instructions for how to install and configure the files so that curious readers can try out the tool for themselves.

3.4 Conclusion

As part of a broad investigation on the relationship between mathematics and music, we introduced examples of mathematical applications to music (and musical applications to mathematics). We looked to continue parts of the research presented in two specific facets. Firstly, we analyzed the musical operations on a melody line that form a mathematical group, and we proved that this group is isomorphic to $D_{12} \times \mathbb{Z}_2$. Secondly, we proposed a relationship between the structure of cyclic subgroups and perfect balance in rhythm. We developed a tool designed to allow users to discover for themselves the wonderful application of group structures in generating rhythms. As a result, we followed the example set by Milne and his colleagues to produce our own method for finding group sizes that generate irreducibly periodic perfect balance.

This thesis leaves opportunity for further research to expand upon these findings. Music has a certain quality of preference and subjectivity that (at least currently) seems to escape the scope of a mathematical theorem. With this in mind, a statistical survey conducted with our tool could provide some interesting findings on which group sizes produce preferable rhythms. In addition, this research could be expanded to include the rhythms of more complex subjects such as direct products of cyclic groups or a simultaneous overlay of two groups. It is in this spirit of looking forward that we close by leaving the door open to encourage more questions and more possibilities for mathematics and music.

Bibliography

Bibliography

[1] Groups, algorithms, programming—a system for computational discrete algebra. https://www.gap-system.org/gap.html. 29

[2] S. Aaron. Sonic pi. https://sonic-pi.net/. 38

[3] H. Anton and C. Rorres. *Elementary Linear Algebra Applications Version*. John Wiley & Sons, Inc., 2005. xi, 11, 12

[4] attwad. Python-osc — github. https://github.com/attwad/python-osc. 38

[5] H. F. Cohen. Simon steven's equal division of the octave. *Annals of Science*, 84(5), 1987. 9, 10

[6] J. F. P. Brooks, J. A. L. Hopkins, P. G. Neumann, and W. V. Wright. An experiment in musical composition. *IRE Transactions on Electrical Computers*, EC-6(3), 1957. x, 21, 22

[7] T. R. P. Foundation. About us. https://www.raspberrypi.org/about/. 37

[8] J. K. Hammond. Mathematics of music. *UW-L Journal of Undergraduate Research*, 14(1), 2011. 3, 7, 10, 12

[9] L. Harkleroad. *The Math Behind the Music*. Cambridge University Press, 2006. xi, 2, 9, 14, 15, 16, 17, 20, 21

[10] T. W. Judson. *Abstract Algebra Theory and Application*. Orthogonal Publishing, 2016. 16, 17, 18, 19, 28, 35

[11] H. M. Miller. *Introduction to Music: A Guide to Good Listening*. Barnes & Noble, Inc., 1968. 3, 4, 6

[12] A. J. Milne, D. Bulger, S. A. Herff, and W. A. Sethares. Perfect balance: a novel principle for the construction of musical scales and meters. In *Mathematics and computation in music*, volume 9110 of *Lecture Notes in Comput. Sci.*, pages 97–108. Springer, Cham, 2015. xi, 23, 24, 25

[13] A. J. Milne and R. T. Dean. Computational creation and morphing of multilevel rhythms by control of evenness. *Computer Music Journal*, 40(1):35–53, 2016. 23

[14] A. Papadopoulus. Mathematics and music theory: From pythagoras to rameau. *Springer-Verlag New York*, 24(1), 2002. 7, 8

[15] G. Sanderson. But what is the fourier transform? a visual introduction. `https://www.youtube.com/watch?v=spUNpyF58BY`. 12, 13, 14

[16] I. Schomer. Mathematics and music: Generating polyrhythms from cyclic subgroup structures - source code. https://cs.maryvillecollege.edu/wiki/Ian_Schomer's_Senior_Study. 42

[17] P. Seetharaman, F. Pishdadian, and B. Pardo. Music/voice separation using the 2d fourier transform. *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, page 36, 2017. 14

[18] A. T. White. Fabian stedman: The first group theorist? *The American Mathematical Monthly*, 103(9):771–778, 1996. 16, 17, 18, 19

[19] H. Zeller. Rpi-rgb-led-matrix — github. `https://github.com/hzeller/rpi-rgb-led-matrix`. 39