INTERVAL CYCLES IN THEORY AND PRACTICE
Designing a Musical Interface for Composition and Performance

By

Daniel Leon Harrell
M.M., University of Louisville (2006)

Proposal of Dissertation for the Doctorate of Musical Arts

School of Music
University of Illinois Urbana-Champaign
Abstract

This study focuses on implementing an interactive musical instrument interface for the Apple iPad™, which constructs and manipulates interval cycles in real-time to facilitate composition or live improvisation. In the first part of this dissertation the interval cycle and its musical function is described. Part two discusses the implementation of the musical interface including developing methods of visual representation, algorithmic data filtering and specific issues of interface design.

Introduction

The impetus for this project came from my desire to understand intuitively-based pitch organization within my own compositions. I began to find the connections I was looking for when I was introduced to interval cycles.

Example 1. Interval cycle - IC(2 3)

Interval cycles are pitch collections that consist of a repeating pattern of intervals. For example IC(2 3) is an interval cycle of an alternating pattern of a major second (2) and a minor third (3). Because there is no standard method of notating interval cycles, for convenience I will label them with the abbreviation IC and the interval pattern notated in the number of half steps of each interval within parentheses.

IC(2 3) is an example of an interval cycle with pitches that do not repeat in each octave. This unique feature of interval cycles makes it difficult to calculate quickly what pitches will be in the cycle beyond a few iterations. To compound this problem, a more
complicated pattern of multiple intervals can quickly become too complex to easily manage without a proper tool.

**Statement of Problem**

Although there is great potential in the organization of pitch with interval cycles, efficiently calculating them restricts composers and performers from using them with ease. Additionally, even if a composer has the pitches for an interval cycle at hand, the act of composing at the piano with them is still difficult due to the layout of the conventional keyboard. Since the piano keys are oriented in repeating patterns of twelve it is difficult to work with interval cycles that do not align with this pattern. A keyboard that is capable of changing its layout dynamically with the interval cycle would be ideal for eliminating this hurdle.

**Background of the Study**

The use of interval cycles can be traced back as early as 1920 in a letter that Alban Berg writes to Arnold Schoenberg in which he encloses his “Master Array of Interval Cycles.” This master array is a chart of the twelve possible single-interval cycles. Berg describes this idea to Schoenberg as a “theoretical trifle” though ultimately used it as a central method of organizing pitch in many of his works. In example 2, Berg uses IC(7) in the bass and IC(1) in four parallel descending chromatic lines in an excerpt from Op. 2, No. 2.

---

Howard Hansen extends the concept of interval cycles by using multiple intervals within a cycle called a projection.\(^2\) Since some cycles, such as IC(3), exclude some pitches, Hansen extends these cycles by inserting an arbitrary interval of the perfect fifth. This allows any cycle to use all twelve chromatic pitches. Example 3 shows an IC(3) with the inserted perfect fifth. This cycle would now be notated as IC(3 3 3 7).

Example 4. IC(7) in projected form 1, and scale form 2.

Related to interval cycles are Olivier Messiaen’s Modes of Limited Transposition; symmetrical scales that have less than twelve possible transpositions. Messiaen created six modes that meet these criteria, however it is possible to create more using interval cycles. Any symmetrical interval cycle with pitch content that repeats in every octave will be a mode of limited transposition. Example 5 shows that IC(1 5) will produce a mode of limited transposition that is a subset of Messiaen’s modes 2 and 7.

Example 5. (1) Mode 2, (2) Mode 7, (3) IC (1 5)

When composing with interval cycles it is also important to understand the degree to which each interval cycle is related. To do this it is necessary to understand them within the context of a musical space. Musical spaces organize all possible elements of the space in order to show the relationships between each of its members. The most common type of

---

musical space is pitch space, commonly associated with Hugo Riemann. Example 6 shows a portion of Reimann’s tonnetz, a pitch space arrangement that shows all relative keys, chromatic mediant, and dominant relationships.

![Diagram of Reimann's Tonnetz](image)

**Example 6. A portion of Reimann’s Tonnetz**

Clifton Callendar, Ian Quinn and Dmitri Tymoczko describe multidimensional pitch spaces. Example 7 shows the voice leading relationships between any two possible tri-chords. Additionally, Tymoczko uses geometric figures to visually represent the chords placed within the full continuum of pitch space.

![Diagram of pitch space showing all tri-chords](image)

**Example 7. Pitch space showing all tri-chords.**

Interval cycles can be placed into an interval cycle space because they are collections of distances between pitches. Since interval cycles can be analyzed for their interval vector they can also be placed into an interval vector space. Since this would need to be a

---

multidimensional space, algorithms would be required to calculate and map these spaces. These spaces can be used to understanding the complex relationships between various interval cycles.

Research Questions

The first task of this dissertation will be to construct a standard notation to represent interval cycles in a logical manner, with both standard musical notation and alternate visual representations. Secondly, the design of an instrument that visually shows the relationship of pitches within the interval cycle that has logical keyboard layouts. Finally, the task of how to manipulate the interval cycle via the interface in real-time with precision will directly influence the usability for the end user. The core algorithms used in this program will be used in the mapping of the interval cycle space, which will be available for the user to compare different interval cycles.

Methods and Procedures

This study focuses on creating an instrument to produce MIDI data, rather than an audio signal. Since the user interface of this program will be the Apple iPad™, the ultimate goal will be a stand-alone application for the iOS platform. However, the interim step of using the iPad as an input surface to output data to MAX/MSP or SuperCollider for processing the pitch algorithms may be necessary due to limitations of the Apple App Store development process. It is yet to be determined how the algorithm for computing this data will function.

Attached to this document is a prototype schematic of the interface with sample data output. Page 1 is a diagram of the interval cycle input interface. Page 2 shows the dynamic
keyboard interface. Page 3 shows the alternate grid keyboard interface, which is design for efficiency on a flat surface. Page 4 demonstrates chord entry on the dynamic keyboard. Pages 5 and 6 show two alternate views of the iPad application.

Limitations

The interface will be limited exclusively to the output of pitch information in the form of MIDI data and is not intended for use to control other parameters of music. Also the interface is not designed to handle any other decision-making requirements of the composition and improvisation processes outside of the realm of pitch.

The instrument interface is designed for formulating and playing interval cycles primarily, thus will require some method of handling pitches that are excluded from the interval cycle. This limitation has not been accounted for yet but will be implemented in the software.

Finally this study and instrument is limited to 12-tone equal temperament for practical purposes, but may be extended in the future for use with alternate tunings.

Summary

The goal of this project is to create an instrument to assist in creating interval cycle-based music. It is designed to increase efficiency in calculating and performing interval cycles. Priority will be placed on ease of use in the interface and robustness of the algorithm to sort interval cycles into a comprehensive interval cycle space.
References


References (cont.)


The pitch C4 is entered using the onscreen keyboard or from the input of an attached MIDI keyboard. Since there is no interval the output is only one pitch.

The pitch E4 is entered, this creates the first interval. Both pitches are side by side because this completes the cycle. The pitch G#4 is entered causing a line to split the cycle between the first and second interval.

The pitch F#4 is entered causing a line to split the cycle again.

The pitch G4 is entered causing another line to split the cycle.

The user continues to enter pitches until the desired interval cycle is complete. As more pitches are added the circle is divided evenly. As additional pitches are added the desired interval is complete. As more pitches are added the circle is split between the first and second interval.

The pitch E4 is entered. The pitch F#4 is entered causing a line to split the cycle again.

The pitch G#4 is entered causing a line to split the cycle.

The pitch B4 is entered causing a line to split the cycle.

The pitch D#5 is entered causing a line to split the cycle.

The pitch C5 is entered causing another line to split the cycle.

The pitch C4 is entered causing a line to split the cycle again.

Interval cycle input interface
This grid keyboard layout is an alternative to the more conventional layout. It has the advantage of being more ergonomic on the flat surface of the iPad.

The example shows IC(4 2) arranged from lowest pitches on the bottom to the highest on the top. Each repetition of the interval cycle is the next row upward. Also the grid is four squares wide to align with the interval cycle pattern. Each grid is four squares wide to align with the interval cycle pattern. Each repetition of the interval cycle is the next row upward.
Example output

The black dots represent the selected chord from within a I(C4 2 1 2) that will result from the root note being pressed.

Chord entry on dynamic keyboard interface