RECENT APPROACHES TO REAL-TIME NOTATION

Seth Shafer, B.M., M.M.

Dissertation Prepared for the Degree of

DOCTOR OF PHILOSOPHY

UNIVERSITY OF NORTH TEXAS

May 2017

APPROVED:

Andrew May, Major Professor
David Stout, Minor Professor
Jon C. Nelson, Committee Member
Joseph Klein, Chair of the Division of Composition
Benjamin Brand, Director of Graduate Studies in the College of Music
John W. Richmond, Dean of the College of Music
Victor Prybutok, Vice Provost of the Toulouse Graduate School

Shafer, Seth. *Recent Approaches to Real-Time Notation*. Doctor of Philosophy (Music), May 2017, 191 pp., 3 tables, 47 figures, bibliography, 148 titles.

This paper discusses several compositions that use the computer screen to present music notation to performers. Three of these compositions, *Law of Fives* (2015), *Polytera II* (2016), and *Terraformation* (2016–17), employ strategies that allow the notation to change during the performance of the work as the product of composer-regulated algorithmic generation and performer interaction. New methodologies, implemented using Cycling74's Max software, facilitate performance of these works by allowing effective control of generation and on-screen display of notation; these include an application called VizScore, which delivers notation and conducts through it in real-time, and a development environment for real-time notation using the Bach extensions and graphical overlays around them. These tools support a concept of cartographic composition, in which a composer maps a range of potential behaviors that are mediated by human or algorithmic systems or some combination of the two. Notational variation in performance relies on computer algorithms that can both generate novel ideas and be subject to formal plans designed by the composer. This requires a broader discussion of the underlying algorithms and control mechanisms in the context of algorithmic art in general. *Terraformation*, for viola and computer, uses a model of the performer's physical actions to constrain the algorithmic generation of musical material displayed in on-screen notation. The resulting action-based on-screen notation system combines common practice notation with fingerboard tablature, color gradients, and abstract graphics. This hybrid model of dynamic notation puts unconventional demands on the performer; implications of this new performance practice are addressed, including behaviors, challenges, and freedoms of real-time notation.

ACKNOWLEDGEMENTS

In the development of my real-time notation compositions, I faced numerous obstacles. The work would not be in its present state without the support and encouragement from professors, colleagues, friends, and family. In particular, I thank Andrew May for his artistic and technical guidance in the development of *Terraformation* and *Polytera II*, David Stout for his advice in the area of algorithmic art, Elizabeth McNutt for her insight into the performer's mindset and the use of her concept of "triage," Daniele Ghisi for his assistance in adapting his BACH: Automated Composer's Helper library for my real-time notation needs, and Michael Capone for generously sharing his knowledge of the viola and being willing to play anything I put in front of him. I thank my family, my parents Dan and Kathy Shafer for their love and support. Finally, I thank my wife Abigail Shafer, whose continual encouragement throughout the entire process made this dissertation possible.

TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS iii

LIST OF TABLES vii

LIST OF FIGURES viii

PART I. CRITICAL ESSAY 1

Chapter 1. Introduction 2

The Variable Nature of All Music 2

On-Screen Notation and Real-Time Notation 5

Dissertation Overview 9

Chapter 2. Integrated Intuitive Performance for Real-Time Electronics 12

Introduction 12

Problems of Synchronization 13

General Design Principles for On-Screen Notation 15

An Overview of VizScore 18

Introduction to VizScore 18

Design Goals of VizScore 19

Creating a Score in VizScore 19

Current Uses 23

Comparing VizScore to Other Software 24

Expressive Scores in Real-Time 24

Solutions for Real-Time Notation 25

The Use and Limitations of bach 26

Better Notation Delivery and Time-Location Tracking in bach 27

New Score Elements for Expressivity in bach 29

Conclusion 33

Chapter 3. Algorithmic Art and Cartographic Composition 35

Introduction 35

Parameter Control in Algorithmic Art 36

Cartographic Composition in Some Recent Real-Time Notation Compositions .. 45
Expansions of Control 47
Control Structures in *Law of Fives* 48
Control Structures in *Polytera II* 49
Control Structures in *Terraformation* 51
Predictable and Variable Elements 52
Conclusion 71
Chapter 4. Performer Action Modeling in Real-Time Notation 75
Introduction 75
Notating Action 76
Resultant Sound Notation 77
Performative Action in Notation 78
Cognitive Attention Balancing 82
Performative Action in Real-Time Notation 83
Purposes of Action-Based Notation in Real-Time Notation 83
Performer Action Modeling in *Terraformation* 86
Overview of the Notation Used in *Terraformation* 87
Efficiency in Hybrid Notation 95
Conclusion 96
Chapter 5. Performance Practice of Real-Time Notation 97
Absence of Performance Protocols for Real-Time Notation 97
New Freedoms for Musical Expression 98
Freedom From Replication 98
Improvisational Freedom 100
Interactive Freedom 102
Ephemerality and Multiplicity 103
Problems in Rehearsal and Performance 104
Traditional Purposes of Practice and Rehearsal 104
New Purposes of and Approaches to Rehearsal with Real-Time Notation 106
Performer-Composer Trust in Performance 115

Comparing Real-Time Notation with the Complex Score 116
Conclusion 119
Chapter 6. Future Directions 121
Introduction 121
Combining Real-Time Notation with the Complex Score 122
The Machine is the Artist–Authorship in Algorithmic Art 124
The Future of Notation? 125
Bibliography 127
PART II. DOCUMENTATION OF *TERRAFORMATION* 138
Performance Notes for *Terraformation* 139
Graphical User Interface for *Terraformation* 140
Representative Score for *Terraformation* 142

LIST OF TABLES

Page

Table 3.1: A table showing the Markov transition probabilities of pitch sets f_0 and f_1 from Iannis Xenakis's *Analogique A et B* (1958). 41

Table 3.2: A formal outline of *Law of Fives* showing tempo, orchestration, and textural changes throughout the piece. 53

Table 3.3: A formal outline of *Polytera II* showing tempo, meter, orchestration, and textural changes throughout the piece. 59

LIST OF FIGURES

Page

Figure 1.1: Measures 12–25 of Chopin's E Minor Prelude Op. 28, No. 4 highlighting an area of increased difficulty and therefore likely performance variation due to error. 3

Figure 1.2: Categories of real-time and non-real-time music notation. On the right of the table, the types of scores commonly encountered in these two categories are listed. 6

Figure 1.3: Attributes of the real-time score. 7

Figure 2.1: Two common on-screen notation paradigms. 16

Figure 2.2: A suggestion for a minimum segmented score layout showing previous, current, and future systems. 20

Figure 2.3: Mapping beats within a bar in VizScore. 21

Figure 2.4: The momentum of the jumping time-location tracker in VizScore allows the performer to anticipate downbeats and construct an accurate sense of tempo. 21

Figure 2.5: A screenshot of the performer interface for *Law of Fives.* 27

Figure 2.6: The conductor patterns outlined by a bouncing ball in *Polyera II*. 28

Figure 2.7: Individual graphics specified by pixel location and size within a larger, single image file. 31

Figure 3.1: Every possible choice for m. 2 in Mozart's *Musikalisches Würfelspiel* (1787) demonstrates a similar harmonic and melodic function. 38

Figure 3.2: A. Michael Noll's *Gaussian-Quadratic* designed using a random movement algorithm. 39

Figure 3.3: An approximation of a normal distribution curve by rolling two six-sided dice. 40

Figure 3.4: Two views of a Mandelbrot function at different scales. 43

Figure 3.5: Herbert Franke's *Konforme Abbildung* (1986). 44

Figure 3.6: A network of live influence in *Law of Fives.* 49

Figure 3.7: The performer interface for *Polytera II* showing a progressive removal of score elements to create proportionally notated music. 50

Figure 3.8: Two central pitches surrounded by chromatic neighbors follow a sequence of fifths to generate all of the pitches in the first section of *Law of Fives*. 55

Figure 3.9: The orchestration of the central pitches in the first section of *Law of Fives* indicating moments of cohesion connected through pitch interpolation. 55

Figure 3.10: A quintuple rhythmic system showing all divisions of a bar. 56

Figure 3.11: An example of a rhythm constructed using rhythms available in the quintuple system in Figure 9. 56

Figure 3.12: The rhythm algorithm for *Law of Fives*. 57

Figure 3.13: The 32-step rhythmic sequence that seeds the rhythm algorithm at the beginning of *Law of Fives*. 57

Figure 3.14: One potential notation resulting from the *pairs* pitch algorithm in *Polytera II*. 61

Figure 3.15: A sample four-bar rhythmic pattern. 62

Figure 3.16: A composite of the sample rhythmic pattern showing all locations of note onsets. 63

Figure 3.17: The original rhythmic pattern constructed from note onsets in the composite rhythm, ignoring any resulting durations. 63

Figure 3.18: An analysis of the probability of each note onset being followed by a note or a rest based on the sample four-bar rhythm. 63

Figure 3.19: The syntax of the rhythmic Bayesian network algorithm. 64

Figure 3.20: Four iterations of the rhythmic Bayesian network algorithm. 65

Figure 3.21: The message from systems 8 through 10 of *Polytera II* with a visual guide to the Markov table and one resulting notation. 66

Figure 3.22: A catalog of gestures used in *Terraformation*. 67

Figure 3.23: The organization of gestures in *Terraformation*. 67

Figure 3.24: The degree of variance of pitch and rhythmic content over the course of *Terraformation* based on parameter data in the qlist. 67

Figure 3.25: The pitch spelling algorithm in *Terraformation* maps given pitches along a continuum of fifths and selects the closest grouping. 70

Figure 4.1: Action notation in Luigi Russolo's *Risveglio di una città* (1913–14). 79

Figure 4.2: Decoupled string tablature in the second violin part from Aaron Cassidy's Second String Quartet (2009–10). 81

Figure 4.3: Action notation in Gerhard E. Winkler's *Hybrid II (NetWorks)* (1996, rev. 2001). 84

Figure 4.4: Radial notation in Ryan Ross Smith's *Study no. 40.3* (2014). 85

Figure 4.5: The aggregate notation and performance interface for *Terraformation*........ 88

Figure 4.6: A block diagram of the algorithm for generating new chords in *Terraformation*...... 89

Figure 4.7: The range of possible "fret" positions and maximum finger stretch in those positions. 90

Figure 4.8: All twenty-four unbarred left hand positions on the viola ranked by difficulty......... 91

Figure 5.1: Computer-generated instructions in Karlheinz Essl's *Champ d'Action*. 101

Figure 5.2: Notational variability from live performer influence in Seth Shafer's *Law of Fives* (2015). 108

Figure 5.3: An excerpt of the formal plan for *Polytera II* showing a system of graphic icons that represent large-scale changes in tempo, texture, orchestration, and tessitura. 109

Figure 5.4: A graphic depiction of the form displayed on-screen in *Terraformation*........ 109

PART I

CRITICAL ESSAY

Chapter 1

Introduction

The Variable Nature of All Music

Music is a dynamic art form, inextricably rooted in time. As such, music is intrinsically non-replicable; as anyone in the habit of attending live concerts can attest, two different performances of the same work can vary markedly from one another. This is self-evident in the realm of free improvisation; in the case of notated common-practice era works, a great deal of variety in interpretation is also possible. Even music heard on recording can vary according to the playback equipment, the acoustic environment, and the listener's psychological state.

Music described by highly detailed notations or including pre-recorded elements seems to offer the tantalizing possibility of determining every sonic parameter, thus ensuring perfect reproduction; inevitable variations in psychological, biological, physical, and acoustic parameters undermine this possibility in practice. The invention, refinement, and contemporary expansions of the common practice notation (CPN) system in Western music has tended historically toward greater and greater specificity, evidently with the goal of perfect representation and reproduction in sound of the composer's intent. On the other hand, compositional practices ranging from ornamentation in Baroque music to aleatory and open-form compositions in the present day explicitly invite variation from one performance to the next. Where variability is not clearly specified in the score, it nevertheless occurs, whether because of the inability of notation to specify all the details of execution, because of notation ambiguity that leaves room for interpretation, or because of performer error.

Performer error accounts for variation in otherwise predictable sonic outcomes, including pitch and rhythm mistakes, dynamic and expressive miscalculations, and mechanical malfunction. In many cases performer error arises from a sudden increase in difficulty level. Given a work of consistent difficulty with an isolated area of increased difficulty, one can predict increased variability in that location. An example can be found in measures 16–18 of Chopin's E Minor Prelude Op. 28, No. 4:

Figure 1.1: Measures 12–25 of Chopin's E Minor Prelude Op. 28, No. 4 highlighting an area of increased difficulty and therefore likely performance variation due to error.

The material that comes before and follows mm. 16–18 is of a consistent character: a slow stepwise melody, a simple eighth note accompaniment with three pitches per chord, harmonic changes every measure or half measure, subdued dynamic. The *gruppetto* ornament in m. 16 signals a dramatic shift that powers the climax of the piece: an active leaping melody, an accompaniment with four-note sonorities that sets triplet against eighth, changes of harmony on every beat, and swelling dynamics. The shift in character and dynamic invites interpretive variation at this moment in the piece; moreover, the sudden increase in difficulty of execution creates opportunities for performer error and additional performance variation.

Complex rhythmic figures and extreme ranges on an instrument can purposefully lead to variations from one performance to the next. Composers associated with the so-called complex score and New Complexity, such as Brian Ferneyhough, Richard Barrett, and Aaron Cassidy (among others), ask players to perform at the limit of what is possible. This is often accomplished by presenting the performer with conflicting instructions or goals represented in a meticulously detailed, high-density fixed score. The result is a collision of actions with a variable sonic outcome in each performance.

Works depending on particular technologies, instruments, or performance environments will vary significantly when these change, either due to degradation over time or adaptation to available resources. Examples of such works include Varèse's *Poème électronique* (1958) written for the 350 speakers installed in the Philips Pavilion, La Monte Young's sound and light environment *Dream House* (1963–93),[1] Jem Finer's 1000-year composition *Longplayer* (1999–2999),[2] and Bill Fontana's use of accelerometers on the London Millennium Bridge in *Harmonic Bridge* (2006).[3] Variation is introduced in each of these pieces through media

[1] La Monte Young and Marian Zazeela, "Dream House: Sound and Light Environment," MELA Foundation, http://www.melafoundation.org/ (accessed February 10, 2017).

[2] Jem Finer, "Longplayer," The Longplayer Trust, http://longplayer.org/ (accessed February 10, 2017).

[3] Bill Fontana, "Harmonic Bridge," http://resoundings.org/Pages/Harmonic_Bridge1.htm (accessed February 10, 2017).

degradation, media obsolescence, or physical destruction (as in the case of the Philips Pavilion). It can be unclear how much of the original concept and experience of a work persist in later versions, as in the case of the *Poème électronique* being presented on 2-channel reproduction equipment over headphones.

The composition *Terraformation* (2016–17)[4] presented in this dissertation explores the issue of variability through the use of real-time notation to create a musical work that specifies a degree of variation from one performance to another in several dimensions.

On-Screen Notation and Real-Time Notation

The evolving world of interactive computer music has, in recent years, witnessed a trend of using an on-screen display for communicating directions to the performer. Not surprisingly, this is mirrored in the larger musical context with a growing number of both performers and conductors preferring to use tablets or displays rather than printed parts or scores. For young musicians, the screen and associated computing power have been present during their entire compositional development, making the integration of such devices into the concert environment a logical development.

After adopting on-screen displays as a notational medium, many performers are willing to explore new domains of notational interactivity. Such interaction evokes a continuum delimited by pre-determined and immutable paper or digital scores at one end and free improvisation at the other. Gerhard E. Winkler suggests that between these two extremes lies a

[4] Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

“Third Way” made possible by recent technologies that support various types of real-time notation (RTN).[5] This emerging practice of using computer screens to display music notation goes by many names: *animated notation*, *automatically generated notation*, *live-generative notation*, *live notation* or *on-screen notation*. These new notational paradigms can be separated into two categories: real-time notation and non-real-time notation:

Real-Time Notation	Live-Generative Live-Animated
	Live-Permutated
Non-Real-Time Notation	Fixed Animated Fixed Non-Animated

Figure 1.2: Categories of real-time and non-real-time music notation. On the right of the table, the types of scores commonly encountered in these two categories are listed.

Real-time notation encompasses scores that contain material that is subject to some degree of change *during the performance* of the piece. Many works fit this definition, from those that use pre-determined musical segments that are reordered in performance to those that are completely notated in the moment of performance. Non-real-time notation accounts for the vast majority of uses of the computer display as a notational medium; both static (fixed notation) and animated scores occupy this category. The boundary between these two primary approaches to notation on the computer screen is not rigid, and examples of the live-

[5] Gerhard E. Winkler, “The Realtime-Score: A Missing-Link in Computer-Music Performance,” *Proceedings of the 2004 Sound and Music Computer Conference* (2004): 1.

permutated score can be found that fit in both categories.

In order to better understand how and where a specific on-screen work fits along this continuum, it will be helpful to further categorize it according to its various attributes. In both real-time and non-real-time scores, attributes commonly encountered include notation style, interpretive paradigm, time synchronization and location tracking management, degree of on-screen movement, whether the performer reads from a part or a score, and if there is non-notational interactivity:

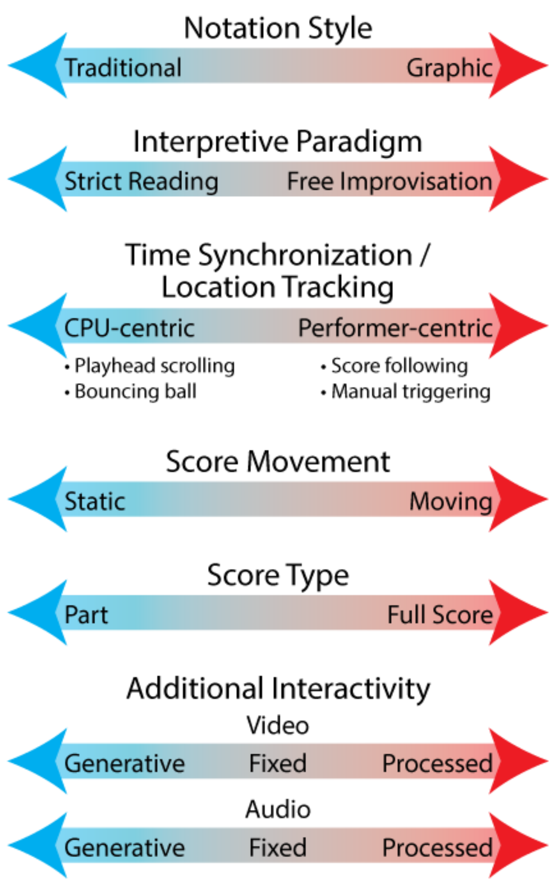

Figure 1.3: Attributes of the real-time score.

Notation style refers to the spectrum between traditional symbol-based notation and graphic notation. Many real-time notation scores use graphic notation or a combination of traditional graphic symbols and abstract graphics. The *interpretive paradigm* of a piece determines whether the performer is expected to perform the music "exactly as written" in the score, or is encouraged or even obliged to incorporate some degree of improvisation to interpret the notation. The method of *time synchronization*, *location tracking*, and the *degree of on-screen movement* can be important in solo and ensemble pieces for which musicians are reading from a computer screen. Relying on eye movement research, Lindsay Vickery[6] (2014) and Richard Picking[7] (1997) conclude that using a play head cursor or a scrolling score discourage rapid eye repositioning, something associated with experts reading fixed notation, and instead promote a detrimental perceptual fixation. Instead, a bouncing-ball-type tracker embodies expressive and anticipatory tempo information by drawing on a performer's skill of following a conductor.

The question of whether the performer reads from a *part or score* has implications for coordination, not to mention the appropriate visual size of the music on the computer screen. Works using real-time notation often incorporate *additional interactivity* incorporated into the audio or video components of the work. Live processed, fixed, or generative video elements may be used as an extension of graphic notation. In addition to various challenges of performing real-time notation, the performer of such works must grapple with the practical and interpretive issues associated with *musique mixte* and interactive electroacoustic music. For example, depending on the programming, the performer may not always be entirely aware of

[6] Lindsay Vickery, "The Limitations of Representing Sound and Notation on Screen," *Organised Sound* 19, no. 3 (2014): 215–27.

[7] Richard Picking, "Reading Music from Screens vs Paper," *Behaviour and Information Technology* 162 (1997): 72–8.

what the live electronic component of the work will process, play back, or generate each time through the piece. An excerpt from the performance instructions from my 2015 work *Law of Fives* illustrates the kinds of demands this situation can put on performers:

> The notation is generated in the moment of performance and requires the performers to sight-read the notation in front of an audience.… The goal of a performance of the piece, therefore, is not about perfect adherence to the demands of the score, but about the interaction between human and artificial intelligence. The performers should attempt to both read the music as accurately as possible and respond to and influence the computer's musical decisions.[8]

Dissertation Overview

The following chapters document my recent approaches to composing using both on-screen notation and real-time notation. The scope of the discussion is limited to the immediate issues surrounding my creative experiments with new modes of interactivity through notation, and the context of these projects in the repertoire of interactive and algorithmic music and art. Chapter 2 addresses the design considerations of on-screen notation. By moving the notation into the same medium as other computer-driven interactive elements, such as live performer processing and sample triggering, the entire work may become more integrated. If designed thoughtfully, the musician's experience can be more intuitive, allowing more room in the performance for expression and creativity.

Many composers from the middle of the 20th century onward have embraced the variable nature of music, creating music that specifically defies replication. The indeterminate works of John Cage, graphic scores of Earl Brown, and stochastic applications of Iannis Xenakis

[8] Seth Shafer, "*Law of Fives* (2015), for viola, bass clarinet, marimba, and computer," http://sethshafer.com/law_of_fives.html (accessed February 12, 2017).

are examples. A parallel pursuit of dynamism, algorithmic invention, and human-computer interaction drives art in the visual, literal, and performative domains to similar ends. Chapter 3 discusses some ways that artists have used algorithms in their art, with particular attention to the control mechanisms that shape the underlying processes. This historical discourse leads into analysis of the mechanisms that drive the real-time notation in three of my recent works, introducing a theory of composition called "cartographic" composition.

My most recent work, *Terraformation* (2016-17) for viola and computer, uses an intricately multidimensional algorithm to generate music based on models of the physical actions required to play the viola.[9] This is communicated via on-screen notation using a combination of CPN and action-based graphics. Chapter 4 traces some precedents for action-based notation and describes the particular mechanisms that model the performer's actions in *Terraformation*.

The musician's experience is a critical issue in any work using a new mode of performance. Works using RTN are potentially unnerving because they appear to defy usual approaches to practice and rehearsal. Chapter 5 elaborates on the performance practice of RTN works, suggesting new freedoms for the performer and new purposes for practice and rehearsal.

The issues faced by composers and performers using new modes of interaction involving the computer display as a notational medium are not as novel as they might first appear. As the discussions about algorithmic art, action-based notation, and the experimental musical

[9] Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

tradition show, many of the issues surrounding on-screen notation and RTN have extensive historical lineages. The final chapter draws comparisons with some of the aesthetics of the so-called complex score, typified by the work of composers such as Brian Ferneyhough, Richard Barrett, and Aaron Cassidy, and raises issues of identity and authorship in algorithmically mediated artwork. As our culture continues to evolve within the context of human-computer interfaces, we can expect such issues to become both more evident and more pressing; by situating my own experiments within a broader aesthetic context, I seek to provide a foundation for both my own future endeavors and for the work of others.

Chapter 2

Integrated Intuitive Performance for Real-Time Electronics

Introduction

Displaying a fixed score on-screen has many potential advantages to paper scores, including synchronizing ensemble performance, overlaying tempo information on the page, and coordinating unmetered music. There are also many challenges associated with on-screen notation relating to design principles that take music reading studies and eye movement research into account. My VizScore software addresses several of these challenges and has been implemented successfully as the primary notation delivery system in several compositions.

Following a discussion of fixed on-screen scores and VizScore, I focus on the growing use of on-screen scores whose content is generated in the moment of performance. The key interest in this so-called real-time notation (RTN) is its idiomatic use of the medium, the screen and computer, as both a reactive and an interactive agent in the music making process. After discussing my first work using RTN, *Law of Fives* (2015),[10] I discuss the opportunities of including expressive score elements such as slurs, dynamics, articulations, textual indications, and other symbols using examples from my pieces *Polytera II* (2016)[11] and *Terraformation* (2016-17).[12]

---

[10] Seth Shafer, "*Law of Fives* (2015), for viola, bass clarinet, marimba, and computer," http://sethshafer.com/law_of_fives.html (accessed February 12, 2017).

[11] Seth Shafer, "*Polytera II* (2016), for flute, piano, and computer," http://sethshafer.com/polytera_2.html (accessed February 12, 2017).

[12] Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

I pursue a similar goal in both fixed on-screen scores and real-time notation: to integrate the notation of interactive electronic compositions into the performance of the work. I propose that by using intuitive and expressive tools such as those described here, the screen can be a powerful tool to share musical information with performers, directing them toward exciting creative territories.

Problems of Synchronization

One of the primary problems in works requiring synchronization with an electronic source is the predominant strategy for synchronization: the in-ear click track. While reasonably reliable, the inherent weakness of the click track is the necessary aural distraction of the click and the lack of location-specific information. Given the importance of the auditory sense to a musician, the click track has the potential to distract the performer from the primary tasks of listening and playing musically.

An alternative to the click track relies on the performer to accomplish the synchronization based on familiarity with the electronic component. This is the case in many works that combine fixed-media electroacoustic music with live performance. Mario Davidovsky solves this issue in his *Synchronisms I* (1963), for flute and electronic sounds, by asking an operator to start and stop electronic playback at several key moments in the work.[13] The score implies that the flutist should attempt to synchronize the notated flute materials with the sparsely notated electronic part. While this strategy is successful for certain kinds of

[13] Mario Davidovsky, *Synchronisms No. I* (New York: McGinnis & Marx, 1966).

compositions where timbral variety or moments of synchronicity are easily identified and located in performance, other works require more precise means of coordination.

Works that use interactive electronics present special issues of coordination and synchronization, especially if the electronic component varies from one performance to another. One common method of synchronization is the model of pitch-tracking and score-following, proposed independently by Roger Dannenberg[14] and Barry Vercoe[15] in the 1980s.[16] Philippe Manoury's *Jupiter* is one of the first works using this method.[17] Score-following involves computer comparison between a list of expected pitches and a real-time analysis of the pitches generated in live performance. One of the primary drawbacks of this technique is that the computer may skip ahead or fall behind the performer due to analysis errors in the computer or performance errors made by the musician. More robust score-followers, such as the mechanism in Andrew May's *Chant/Songe* (2004)[18] and IRCAM's Antescofo software,[19] improve results and provide predictive capabilities by considering not only a list of pitch events but also analysis of variance over time.

A simpler method of synchronization implores the performer to trigger cues by

---

[14] Roger Dannenberg and Hirofumi Mukaino, "New Techniques for Enhanced Quality of Computer Accompaniment," *Proceedings of the 1988 International Computer Music Conference* (1988): 243–49.

[15] Barry Vercoe, "The Synthetic Performer in the Context of Live Performance," *Proceedings of the 1984 International Computer Music Conference* (1984): 199–200.

[16] For more information see Nicola Orio and François Déchelle, "Score Following Using Spectral Analysis and Hidden Markov Models," *Proceedings of the 2001 International Computer Music Conference* (2001): 1–5.

[17] Phillippe Manouray, *Jupiter: Pour Flûte & Électronique En Temps Réel* (1987, Rev. 1992, 2008),(Paris: Amphion, 2011).

[18] Andrew May, "*Chant/Songe* (2004), for clarinet and computer," http://andrewmaymusic.com/Works/Chant_Songe.html (accessed February 10, 2017).

[19] Arshia Cont, "Antescofo," IRCAM, http://repmus.ircam.fr/antescofo (accessed February 10, 2017).

depressing a MIDI foot switch. This is one of the easiest and most reliable solutions for pieces involving either fixed or interactive electronics. For example, the beginning of Russell Pinkston's *Lizamander* (2003),[20] for flute and computer, asks the performer to play freely, changing the computer's behavior with foot switch cues every few measures. The second half of the piece, however, is metrically strict and requires the performer to conform to the steady tempo of the electronics. Even with several opportunities to resynchronize with foot switch cues, the performer is still subservient to the computer's time structure. This problem is compounded in works involving ensemble and interactive components.

General Design Principles for On-Screen Notation

On-screen notation offers an approach to synchronization that addresses some of the concerns associated with fixed media synchronization, click tracks, score-following, and foot switch triggering. On-screen notation can be less distracting, more information-rich, and relatively precise: at best, it provides the performers with visual indications of time, context, and coordination in a piece of music.

Many principals of graphic design, notation layout, and score-progressing mechanisms need to be considered when using software to display a score on-screen. Lindsay Vickery has proposed several general design principles for presenting notation on a computer display.[21] The

[20] Russell Pinkston, "*Lizamander* (2003), for flute and Max/MSP," http://russellpinkston.com/?portfolio=item-three (accessed February 10, 2017).

[21] Lindsay Vickery, "The Limitations of Representing Sound and Notation on Screen," *Organised Sound* 19, no. 3 (2014): 215–27.

two critical features of an on-screen display system are the delivery of the notational content and the time-location tracker.

The most common notational delivery paradigms used in works relying on screen displays are the segmented score and the scrolling score:

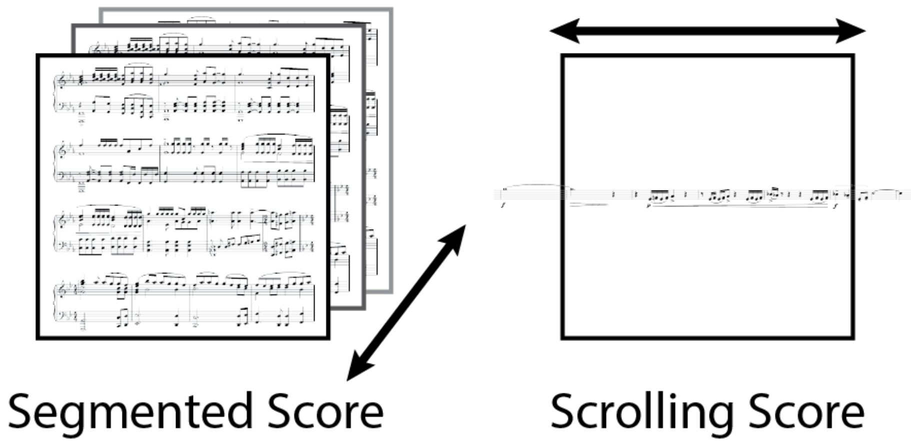

Figure 2.1: Two common on-screen notation paradigms.

According to Vickery, a segmented score comes the closest to mimicking a traditional paper score experience by breaking a musical staff into multiple lined systems much like a printed part. A segmented score allows the performer to look ahead to future musical events and to see their current position in a larger context.

The scrolling score best approximates the linear experience of time as an unbroken continuum. As a single stream of notation smoothly traverses the display, the temporal nature of sound is imbued on the notation itself. The scrolling score can present many challenges to a performer trained in traditional concert practices. Extrapolating from eye-movement research during music reading, one of the primary limitations of the scrolling score is the fixed perceptual frame representing the current "now." This is quite unlike the experience in the segmented score model where the performer's eyes scan a complex pattern across the notation, keeping the perceptual frame in constant motion.[22] The fixed perceptual frame in a scrolling score works against the traditional left-to-right and top-to-bottom reading of music, a deeply ingrained and highly trained skill employed by performers. In addition, Richard Picking shows that animated scrolling movement of the notation above a certain tempo threshold has negative effects on the readability of the score.[23] The ideal layout for on-screen notation appears to be the segmented score as it increases readability by remaining stationary and allowing the performer to control the perceptual frame.

If the notation is immobile on-screen, another crucial element is necessary for time-synchronization: a time-location tracker. Picking describes three types of animated trackers: the smooth tracker, the stepper tracker, and the jumper tracker. The smooth tracker shares many similarities and problems with the scrolling score system described above. As the smooth tracker glides across the display, the location of "now" is clearly visible at the point where the tracker moves over the notation. While a sense of forward motion and anticipation is clearly embedded in the smooth tracker paradigm, it lacks any downbeat preparation or rhythmic integrity. In this way, the smooth tracker is analogous to the play head of an open reel tape machine, continuously feeding musical information to the performer at an even pace regardless of the content.[24] The stepper tracker shares some similarity to the smooth tracker with the exception that the even rate of motion is rhythmically quantized so as to visually snap to every

---

[22] Thomas Goolsby, "Eye-Movement in Music Reading: Effects of Reading Ability, Notational Complexity, and Encounters," *Music Perception* 12 (1994): 77–96.

[23] Richard Picking, "Reading Music from Screens vs Paper," *Behaviour and Information Technology* 16, no. 2 (1997): 72–8.

[24] Jason Freeman, "Extreme Sight-Reading, Mediated Expression, and Audience Participation: Real-Time Music Notation in Live Performance," *Computer Music Journal* 32, no. 3 (2008): 24–41.

bar, beat, or subdivision. While this improves on the rhythmic information communicated to the performer, the stepper tracker's jerky movements caused it to be the least favorable among Picking's study participants. Finally, the jumper tracker provides the performer with a bouncing ball that leaps from beat to beat in an arc motion. This improves upon the previous strategies by imparting both location and tempo in a fluid gesture. In Picking's study, the jumper tracker was the preferred method of tracking time-location.

An ideal on-screen notation delivery system (as advocated by Vickery, Picking, and others) is one that engages the skill and experience performers develop in reading traditional paper notation.[25] While a scrolling score with a smooth tracker might appear to be an idiomatic use of the computer display, it counteracts the benefits of on-screen notation by freezing the location of the performer's perceptual frame and impeding the readability of the score. In contrast, a segmented score with a jumper time-location tracker allows the performer to retain free control of the perceptual frame and reduces unnecessary motion of the notation. The net effect of these designs should therefore improve accuracy in both music reading and time-location tracking in a performance.

An Overview of VizScore

Introduction to VizScore

VizScore is an open-source, on-screen notation delivery system designed to complement the performer's skill and experience. By harnessing a performer's ability to read traditional

[25] Richard Picking, "Reading Music from Screens vs Paper," 75–6.

paper notation and to interpret time from a conductor's gestures, VizScore creates a notation environment that can integrate seamlessly into any performance situation and help musicians play in time with other instruments, live or computer-generated.

VizScore is a suite of abstractions for Cycling '74's Max visual dataflow programming environment, and is compatible with both Windows and Mac. The suite presents a very low processor overhead, allowing it to be combined with a variety of signal-processing and other computer music techniques. VizScore includes a score display, a score editor, and a tempo management system with transport functions. The design and function of each of the components is described in detail below.

Design Goals of VizScore

VizScore is designed with several crucial goals in mind. The first is to create a system that allows for clear notation display and tracking. To this end, VizScore implements a segmented score display and a highly customizable jumper time-location tracker. The second design goal is to create a flexible system that can be implemented across a wide range of notational styles. The final design goal is to provide the user with a simple but powerful interface to create new scores and adapt them to the VizScore framework.

Creating a Score in VizScore

With VizScore, a user can create a segmented score with a jumping location tracker from any graphic notation file. Any style of score can be used with VizScore including computer engraved notation, scanned handwritten music, symbolic graphics, or any other raster image.

Upon loading the notation file, either as a PNG or JPEG, the score editor allows the user to align page and system elements as well as adjust margins and zoom level. For best results, the minimum segmented score display layout should include one previous system of music, the current system, and one or more subsequent systems:

Figure 2.2: A suggestion for a minimum segmented score layout showing previous, current, and future systems.

This configuration is ideal because it allows the performer to place their current location in a larger context and anticipate future musical events. However, because the score alignment settings are completely user-definable, this means that one could choose to let the performer view only one staff at a time, the entire score at once, or any possible configuration in-between.

VizScore uses the best features from both the scrolling score and segmented score paradigms to create a fluid, natural music reading experience. Vickery's model of a segmented score involves turning virtual "pages" of a score, limiting the performer from looking beyond the current segment of notation. Instead of flipping from one "page" to the next, VizScore's tempo management system tracks time-location and quickly slides to the next system when the end of the current system is reached. This brief scrolling motion has a user-definable speed and provides the performer with a sequential stream of staff systems.

In order to track time-location correctly, the jumper tracker in VizScore aligns visually with every beat in a bar. Bar width and beat placement within a bar is notoriously variable due to meter changes and differing degrees of rhythmic complexity. VizScore allows the user to define the span of individual bars of music and the layout of the beats within the bar:

Figure 2.3: Mapping beats within a bar in VizScore.

Options to properly configure a bar include bar size, meter, and a map matching each beat within the bar. These settings determine the placement of the jumping time-location tracker.

The jumping time-location tracker takes advantage of the performer’s sense of anticipation typically placed in a conductor. The tracker moves in an arcing gesture designed to mimic the momentum of a bouncing ball or a conductor’s baton.

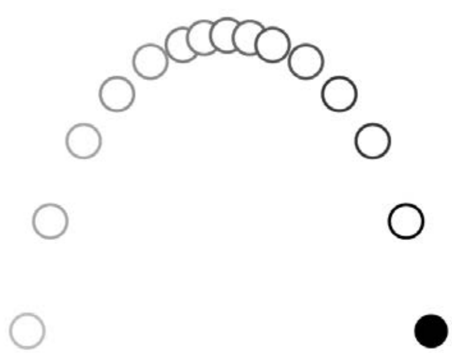
Figure 2.4: The momentum of the jumping time-location tracker in VizScore allows the performer to anticipate downbeats and construct an accurate sense of tempo.

The visual device is borrowed from Max Fleisher's bouncing ball animations used by movie theaters in the 1920s to synchronize the lyrics of audience sing-alongs with pre-recorded accompaniment.[26] In addition to communicating an accurate sense of the tempo, the jumping tracker in VizScore helps the performer anticipate tempo changes, gives preparatory cues, and parses rhythms inside the bar.[27] This is perhaps the most critical feature of VizScore.

The tracker is a raster graphic, meaning that its shape and color are user-definable. In addition, the tracker's arcing path can be altered by changing the amount of gravity in the trajectory algorithm. Altering the tracker's gravity imparts a variety of characteristic styles in much the same way a conductor indicates style by varying the fluidity or rigidity of their physical movements.[28] A high gravity setting, for instance causes the ball to move from one beat location to the next in a type of staccatissimo gesture, while a low gravity setting causes the tracker to move as smoothly as possible from one beat to the next. The gravity setting can also be disabled causing the tracker to cease from bouncing and instead act as either a smooth tracker (useful for proportional notation) or a stepper tracker. In addition, the height of the tracker above the notation can be adjusted over the course of the score, meaning that it can be programmed to move out of the way for notational elements in one system and stay close to the staff lines in the next.

The tempo management system gives users access to tempo and meter maps as well as

---

[26] Richard Fleischer, *Out of the Inkwell: Max Fleischer and the Animation Revolution* (Lexington, Kentucky: The University Press of Kentucky, 2005), 37.

[27] Henkjan Honing, "Structure and Interpretation of Rhythm and Timing." *Dutch Journal of Music Theory* 7, no. 3 (1998): 227–32.

[28] Geoff Luck, Petri Toiviainen, and Marc Thompson, "Perception of Expression in Conductors' Gestures: A Continuous Response Study," *Music Perception* 28, no. 1 (2010): 47–57.

transport controls. The tempo and meter maps dictate changes in both parameters over the course of the piece. Tempo can be set to abruptly change, increase or decrease linearly, or change according to any user-definable path. Support for fermati means that the tempo can arbitrarily stop and start again with visual pick-up beat cues from the time-location tracker. Finally, the transport controls allow the user to fast-forward, rewind, or play from any point in the score.

Current Uses

VizScore is used in two original compositions by the author: *Pulsar [Variant II]* (2014)[29] for trumpet and computer and *Ursa Minor* (2015)[30] for euphonium and computer. Both works use live interactive electronics that involve a combination of real-time processing and synchronized fixed media. The location tracker for *Pulsar [Variant II]* changes tempo multiple times and contains an extended section of proportionally spaced music where the ball glides evenly from left to right across the screen instead of bouncing. The smooth tracker mode is used exclusively in *Ursa Minor* as the notation is entirely proportionally notated.

A number of features have been recently implemented in VizScore in collaboration with Andrew May. These include a more robust clock and tempo system, beat and pulse management to accommodate additive meter, multiple VizScore instances on a single machine, and OSC synchronization of multiple VizScores over a network. A 2017 revision of Andrew

[29] Seth Shafer, "*Pulsar [Variant II]* (2014), for trumpet and computer," http://sethshafer.com/pulsar_variant_2.html (accessed February 12, 2017).

[30] Seth Shafer, "*Ursa Minor* (2015), for euphonium and computer," http://sethshafer.com/ursa_minor.html (accessed February 12, 2017).

May's *Vanishing* (2000, rev. 2017)[31] uses these new features to coordinate a seven-member ensemble and live electronics. Each musician reads from a separate stand-mounted screen displaying an individual instance of VizScore; all seven instances are synchronized to a master clock.

Comparing VizScore to Other Software

While several other software packages resemble the functions of VizScore, including an assortment of tablet apps facilitating the reading of PDFs or MIDI files, INScore is perhaps the closest comparison with VizScore.[32] Developed by the Grame Computer Music Research Lab, INScore supports MusicXML, raster, and vector graphic files as score data. It operates as a standalone application and can be controlled with Open Sound Control (OSC) messages. It also supports a robust time synchronization engine for location tracking. In addition, INScore supports the creation of notation within its scripting language and allows for score interaction. While it surpasses VizScore in its breadth of functions and possible uses, INScore lacks an animated jumper tracker to convey rhythmic anticipation, style, and other attributes described above.

Expressive Scores in Real-Time

Incorporating a fixed on-screen score into a work using live electronics with an

[31] Andrew May, "*Vanishing* (2000, rev. 2017), for chamber ensemble and computer," http://andrewmaymusic.com/Works/Vanishing.html (accessed February 10, 2017).

[32] Dominique Fober, "Time Synchronization in Graphic Domain: A New Paradigm for Augmented Music Scores," *Proceedings of the 2010 International Computer Music Conference* (2010): 458–61; http://inscore.sourceforge.net/ (accessed February 10, 2017).

application like VizScore frees the musician from the aural and physical impediments of alternative approaches like click tracks and MIDI foot pedals. It also suggests deeper integration between the notation and other interactive components involved in the work. This integration goes by many names, but the underlying function is identical: the real-time generation of notation that changes in the moment of performance. By merging notational interaction with other forms of interaction common in live electronic performance, the composer comes closer to making music idiomatic to the medium of computer and screen.

Solutions for Real-Time Notation

Some applications exist for composing a work using real-time notation (RTN). Many of these fall into the category of computer-assisted composition (CAC) tools such as FTM,[33] PWGL,[34] MaxScore,[35] and BACH: Automated Composer's Helper (hereafter referred to simply as "bach").[36] These CAC tools generally have a broad feature set with the expressed purpose of aiding the composer in the composition process outside of real-time. When repurposed for RTN, these composition-oriented software systems do not provide many affordances for the live performer.

---

[33] Norbert Schnell, et al., "FTM: Complex Data Structures for Max," *Proceedings of the 2005 International Computer Music Conference* (2005): 9–12; http://ftm.ircam.fr/index.php/Main_Page (accessed February 10, 2017).

[34] Mikael Laurson, Mika Kuuskankare, and Vesa Norilo, "An Overview of PWGL, a Visual Programming Environment for Music," *Computer Music Journal* 33, no. 1 (Spring, 2009): 19–31; http://www2.siba.fi/PWGL/index.html (accessed February 10, 2017).

[35] Nick Didkovsky and Georg Hajdu, "MaxScore, music notation in Max/MSP." *Proceedings of the 2008 International Computer Music Conference* (2008): 1-5; http://www.computermusicnotation.com/ (accessed February 10, 2017).

[36] Andrea Agostini and Daniele Ghisi, "bach: automated composer's helper," http://www.bachproject.net/ (accessed February 10, 2017).

MaxScore and bach both extend the functionality of Cycling ‘74’s interactive audio and video programming environment Max. MaxScore uses the Java Music Specification Language (JMSL), a Java API developed by Nick Didkovsky, to facilitate computer-assisted composition, notation, and interactive performance. While MaxScore is free to use, it requires purchasing a JMSL license.

The Use and Limitations of bach

The bach project is a free alternative to MaxScore. Like MaxScore, the core use of bach is the processing and display of score data as notation. The score data can be generated using a tree-like structure called Lisp-like linked lists (llll).[37] The bach processing modules range from simple list processing to constraint programming. The resulting data can be displayed and played back in real-time using the notation objects *bach.score* and *bach.roll*. The playback synchronizes the audio output and visual location with a stepper tracker. The score can be forced to continuously scroll horizontally or advance a measure at a time. Both methods are difficult for music reading purposes, as previously described. In addition, beside pitches and rhythms, the bach notation objects only support a limited number of articulations. Other remaining score elements such as dynamics, slurs, and textual indications are not yet implemented.

[37] Andrea Agostini and Daniele Ghisi, “A Max Library for Musical Notation and Computer-Aided Composition,” *Computer Music Journal* 39, no. 2 (2015): 11–27.

Better Notation Delivery and Time-Location Tracking in bach

My piece *Law of Fives* (2015) for viola, bass clarinet, marimba, and computer uses the bach library for generating RTN. One critical feature of the piece is that the musicians' performance effects the upcoming notation. This mechanism is in opposition to the normal affordances of sight-reading where one is able to read ahead of their current playing position. In order to accommodate both of these considerations, the score the *Law of Fives* is split across two, static *bach.score* objects that act as staff systems. This requires that the performers alternately redirect their eyes between an upper and lower staff system. When reading the upper staff system, for example, the lower staff system refreshes. Upon completion of the upper staff system, a performer begins to read the lower staff system, while the upper one meanwhile is refreshed:

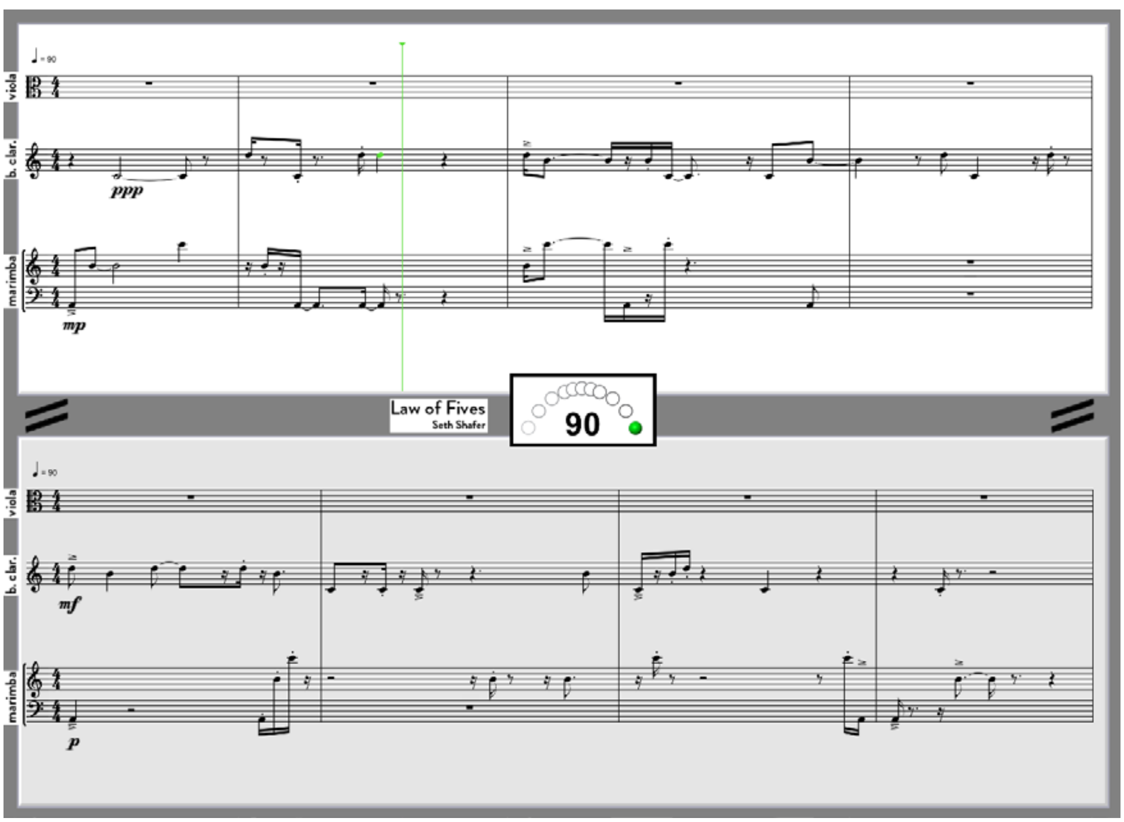

Figure 2.5: A screenshot of the performer interface for *Law of Fives.*

A slight color change from gray to white helps the performers focus on the currently active staff system. This notation requires the performer's eyes to track upward on the screen when returning to the upper staff system. Although this is somewhat counterintuitive for musicians trained to read top-to-bottom through cascading staff systems in CPN, musicians performing *Law of Fives* have not found it to be a serious impediment to playing the piece.

The time-location tracker in *Law of Fives* is the default stepper-tracker built into *bach.score*. However, it is reinforced with a bouncing ball metronome in the center of the screen. This metronome provides a degree of anticipatory information as in VizScore. Additionally, the numeric metronome marking flashes green to warn of impending tempo changes.

The time-location tracking in *Polytera II* (2016) improves on *Law of Fives* by hiding the stepper-tracker in *bach.score* and instead using a custom play head overlay. This play head is a green, vertical line, similar to the default locator in *bach.score*, drawn using Max's *lcd* object. By obtaining horizontal positioning information from *bach.score*, this custom play head only moves once per bar to show the current measure. To supplement the lack of intermediate tempo information, the bouncing ball metronome from *Law of Fives* is replaced with a bouncing ball that emulates conducting patterns:

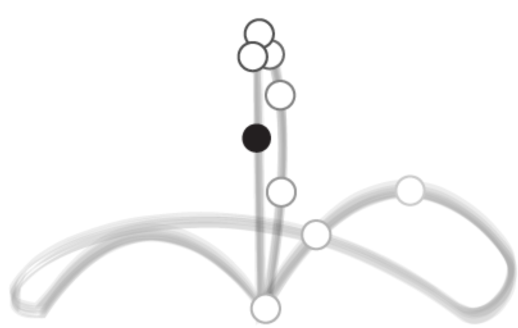

Figure 2.6: The conductor patterns outlined by a bouncing ball in *Polyera II*.

This bouncing-ball conductor, based on a patch by Andrew May, traces patterns for measures of two, three, or four beats, depending on the time signature of each bar.[38]

The final section of *Polytera II* introduces a new time-parsing system that facilitates proportionally spaced music. The strict metrical demands evidenced in the initial sections of the piece dissipate, requiring the removal of bar lines and the replacement of the bouncing-ball conductor with a smooth scrolling time-location tracker.

The score delivery and time-location tracking in *Terraformation* is different from the previous pieces. Instead of alternating one's eyes between upper and lower staff systems, the performer only reads from the upper staff system with the notation from lower, read-ahead staff system being transferred upward by depressing a MIDI foot pedal. This gives the performer a single point of reference for music reading and another for looking ahead at the upcoming music. *Terraformation* does not need a visual time-location tracker because the MIDI foot pedal gives the performer agency and complete control over score advancement.

New Score Elements for Expressivity in bach

The bach library is current limited to a small set of articulations that it can apply to selected notes. As of this writing, it lacks support for dynamics, slurs, textual indications, repeat signs, and any custom musical symbols. The library is still undergoing constant development so this will likely change in the future. Until that time, some measures need to be taken in order to expand the expressive capability of bach's on-screen notation.

---

[38] Andrew May, "Bouncing Ball Conductor Patch," http://andrewmaymusic.com/Software/conductor.zip (accessed February 16, 2017).

In *Law of Fives*, one crucial augmentation of *bach.score* is the use of dynamic marks on each part. This is done by superimposing an image of a dynamic marking on top of the score. Each dynamic is chosen from a bank of pre-made images and displayed with the *fpic* object at the beginning of the first measure. The placement of the marking is always in the same fixed location regardless of whether there is a note above it.

The limitations from the simple dynamic system from *Law of Fives* inspired the development of a universal, arbitrary graphics overlay module called *fpicmarkings*. This module sits transparently on top of *bach.score*, communicates with the score to determine the pixel position of requested elements, and displays user-generated graphics in precise locations. One of the key components is a scripting command that creates a new *fpic* object, loaded with a pre-made image file, centered, cropped, stretched, and positioned as requested:

```
script newdefault [unique_scripting_name] fpic @presentation_rect
      [pixel_location_on_screen] @presentation 1
script send [unique_scripting_name] read [image_file]
script send [unique_scripting_name] offset
      [pixel_location_in_image_file]
script send [unique_scripting_name] destrect [stretch_factor]
```

The first line of the script specifies a new *fpic* object with a unique scripting name that allows it to be addressed remotely after creation. This *fpic* is placed in particular pixel location on-screen and of a specified size with the command "presentation_rect." Lastly, this is added to Max's so-called "presentation mode" where only selected graphic user interface (GUI) elements are displayed. The second script message specifies the pre-made image file, usually in JPEG or PNG format, which the *fpic* object will reference. The third and fourth script messages navigate to particular pixel location in the image file and stretch it to fill the *fpic* accordingly. The newly created *fpic* can be deleted when no longer needed using a scripting command that references the unique scripting name.

The dynamic markings in *Law of Fives* were stored as separate graphic files. In the *fpicmarkings* module, individual graphic elements can be saved as a large, single file. The following example shows how an individual graphic is specified within a single file:

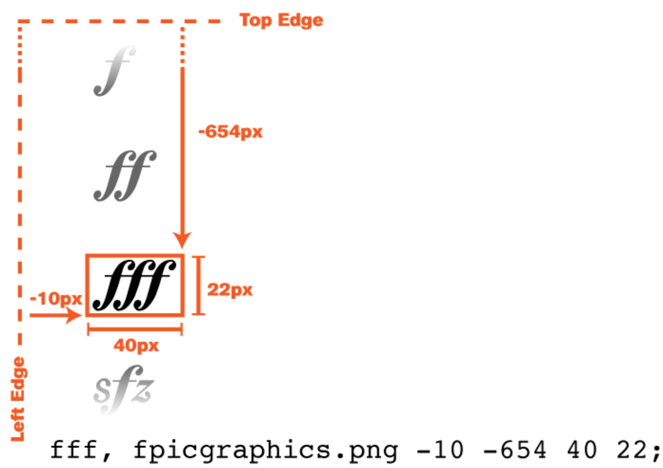

Figure 2.7: Individual graphics specified by pixel location and size within a larger, single image file.

The *fff* dynamic marking is one of many musical symbols inside of an image file called "fpicgraphics.png." Each of the symbols is comfortably padded by white space to avoid overlapping one another. When a *fff* dynamic is called, it is referenced by a variable name, in this case "fff," followed by a set of four integers that provide *fpic* the location and size of the box needed to display the entire dynamic marking.

The *fpicmarkings* module can also produce text. The parameters for creating text are similar to specifying graphics and include the content of the text, font size, font weight, typeface, and screen location. This is used extensively in *Polytera II* to frame the character of each staff system and to direct the performance of multiphonics, flutter-tongue, vibrato, and sustain pedal.

The on-screen pixel location of *fpicmarkings* elements can be defined in several ways. One method is to attach a marking to a note. To facilitate this, the *fpicmarkings* module connects to the *bach.score* object to request pixel location data for certain notes. An example location request includes the marking information, graphic or textual, followed by the instrument, measure, beat, and subdivision to which it should be attached. In addition, the request can prescribe any additional vertical or horizontal offsets from that note location. This is especially useful when defining a marking that should follow a note, such as a glissando line. The *fpicmarkings* module automatically performs vertical compensation for dynamic markings and text expressions so that they remain outside of the staff and avoid overlapping notes, stems, and beams. The location of an element can also be defined by absolute pixel position. This is useful for placing textual information such as tempo and character markings in a fixed location.

Terraformation uses the *fpicmarkings* module in combination with some additional notation elements that supplement the notation provided by *bach.score*. These new elements collectively describe actions that the musician performs that correlate to the common practice notation (CPN) on-screen. One component is a depiction of a viola's fingerboard with indications of finger placement and glissandi direction and span. Another component is a color gradient stretching the length of the CPN corresponding to bow contact position over time. The final new element uses the color parameters inside of *bach.score* to indicate the performer's left-hand finger pressure, from the light touch of a harmonic to full pressure against the fingerboard. Together with the CPN from *bach.score* and the *fpicmarkings* module, the notation for *Terraformation* expands on the expressive possibilities of the bach library to make more detailed and characteristic score in real-time.

Conclusion

By all accounts, on-screen notation is just entering its adolescence.[39] The latest generation of composers and performers are children of the computer age and therefore have complex and meaningful working relationships with the computer display. The performer-display relationship has largely only just begun to be explored. Software like VizScore can help composers integrate their fixed notation into a single, cohesive performance environment where intuitive visual synchronization frees the musician from in-ear click tracks, MIDI foot pedals, or computerized score following systems. Network synchronization and support for multiple simultaneously displayed parts solves problems with conductor-less ensemble performance.

Fixed on-screen notation only begins to scratch the surface of the computer and screen as an interactive artistic medium. A work that uses RTN approaches a more idiomatic use of the screen. Of the software options available for RTN, the bach library provided the most flexibility for generating and processing score data. While the current version is limited primarily to pitches and rhythms, by building a universal tool to generate arbitrary graphics that can be

[39] Lindsay Vickery, "The Evolution of Notational Innovations from the Mobile Score to the Screen Score," *Organised Sound* 17, no. 4 (2012): 128–136.

visually aligned with notes, bach can be augmented to produce expressive and evocative scores in real-time.

On-screen notation, and real-time notation in particular, necessarily raises issues not only concerning the human-computer interface but also algorithmically-based artwork. Elements of graphic design, animation, and real-time decision making create a subtle and dimensional connection between composer, machine, and performer, in parallel with and augmenting the relationships implicit in computer music performance. The following chapter will situate these issues in the historical context of algorithmic art and in the present context of my own notion of “cartographic” composition.

Chapter 3

Algorithmic Art and Cartographic Composition

Introduction

Since the middle of the 20th-century, many composers have embraced the variable nature of music, creating music that specifically defies replication. The indeterminate works of John Cage, graphic scores of Earl Brown, and aleatoric elements of Witold Lutoslawski's music are examples. A number of visual, literary, and musical works have relied on algorithmic generation or human-computer interaction in a parallel pursuit of dynamism and variation. This chapter discusses several approaches to algorithmic art and their implications for variability in musical works. These approaches offer models for music in which the composer no longer needs to specify every detail of a work to ensure a singular sonic outcome. Instead, the composer may map the contours of a work much as a cartographer maps a terrain, indicating significant structures in the environment, areas where categories of vegetation exist, transition boundaries between distinct areas, and so on. In this way, the composer imbues a work with a repeatable, pre-determined sonic identity while allowing the variation inherent in music to manifest itself differently across multiple performances. Throughout this chapter I use the term "cartographic composition" to elaborate on Richard Hoadley's metaphor comparing composing to "mapping a new territory." [40] Following the discussion of algorithmic art, this type of "cartographic" composition is demonstrated in several works by the author.

[40] Richard Hoadley, "Notating Algorithms," *Symposium for Performance of Electronic and Experimental Composition*, University of Oxford, January 6–7, 2012.

Parameter Control in Algorithmic Art

Algorithmic art as such is a 20th-century invention. This art goes by many names—cybernetic art,[41] computer art,[42] telematic art,[43] and generative art[44] to name a few.[45] However, several precursors in chance and permutation can be found in such phenomena as the musical dice games of the 18th century and the aesthetic concern with organicism in the 19th century. Goethe's exhortation to the artist summarizes the desire to mimic the natural world's self-sufficiency and internal coherence: "The highest demand made on an artist is this: that he be true to nature, that he study her, imitate her, and produce something that resembles her phenomena…. Nature is separated from art by an enormous chasm which genius itself cannot bridge without outside assistance."[46] The advancement of computer technology in the mid-20th century provided a kind of "outside assistance" needed for many artists seeking to create art according to a system or a set of rules instead of working directly in a medium. The system's governing rules, algorithms, and control structures reveal important aspects of the resulting art.

One type of system that attracted many early experiments in algorithmic art was permutation, or the rearrangement of materials. The concept is similar to combinatorial musical games like Mozart's *Musikalisches Würfelspiel* (1787) where dice rolls select a series of

---

[41] Norbert Wiener, *Cybernetics: or, Control and Communication in the Animal and Machine*, 2nd ed. (Cambridge, MA: MIT Press, 1965).

[42] Arnold Rockman and Leslie Metzei, "The Electronic Computer as an Artist," *Canadian Art* 94 (1964): 365–7.

[43] Roy Ascott, "Is There Love in the Telematic Embrace?" *Art Journal* 49 (1990): 241–7.

[44] Georg Nees, *Generative Computergrafik* (Berlin: Siemens AG, 1969).

[45] For more terms and their descriptions, see Margaret A. Boden and Ernest A. Edmonds, "What is Generative Art?" *Digital Creativity* (March 2009): 1–31.

[46] Johann Wolfgang von Goethe, "Introduction to the 'Propyläen,'" in *Goethe on Art*, ed. and trans. John Gage (London: Scolar Press, 1980), 6.

musical fragments from a table of pre-composed material. Semantician and computer scientist Christopher Strachey used the principle of permutation to write a series of imaginary correspondences in *Love-letters* (1952). In the project, Strachey populated a computer database with words pre-sorted by their grammatical function and used a computer to randomly select and combine the words into full sentences. In some ways, the work parallels the innovative manual cut-up technique popularized by writers such as William S. Burroughs and Brion Gysin.[47] One of Strachey's resulting letters reads:

> Darling Sweetheart
>
> My sympathetic affection beautifully attracts your affectionate enthusiasm. You are my loving adoration: my breathless adoration. My fellow feeling breathlessly hopes for your dear eagerness. My lovesick adoration cherishes your avid ardor.
>
> Yours beautifully
>
> M.U.C. (Manchester University Computer)[48]

Several categories of control are at work in Strachey's system. The selection of pre-determined words appropriate for the subject of a love letter is a type of manual control, much as the tabulated musical fragments in Mozart's dice game are written in his own style and according to his aesthetic choices. Manual control often imbues in a work the particular character or style of its designer. In addition to manually creating the raw materials, an artist might further choose to select which outputs of the algorithmic process are worthy and which should be ignored; the selection of existent *Love-letters* represents Strachey's judgment about

[47] William S. Burroughs, *Word Virus: The William S. Burroughs Reader* (New York: Grove Press, 1998).

[48] Christopher Strachey, "The 'Thinking' Machine," *Encounter: Literature, Arts, Politics* 13 (October 1954): 26; see also Matt Stephton's implementation of Strachey's algorithm at http://www.gingerbeardman.com/loveletter/ (accessed February 11, 2017).

which letters were the most artistically successful. This intuitive selection process injects personality, taste, and point of view into an algorithmic work. Agostino Di Scipio describes this place as, “the border beyond which compositional [or artistic] decisions and choices are found that evidently could not be dealt with in a systematic and wholly rationalized approach, and that were dealt with by the composer [or artist] in more qualitative and informal – maybe unformalizable – manners.[49]”

Another factor at work in Strachey’s letter is a control mechanism that sorts the database of words by their grammatical function and places them in their appropriate location in a sentence. Theo Lutz similarly transferred chapter titles and subjects from Kafka’s *The Castle* into a database and used a system outfitted with grammatical rules to create his poem *Stochastic Text* (1959).[50] A basic system of musical grammar also guides Mozart’s dice game so that any measure chosen for m. 2, for example, will share a similar harmonic and melodic function:

Figure 3.1: Every possible choice for m. 2 in Mozart’s *Musikalisches Würfelspiel* (1787) demonstrates a similar harmonic and melodic function.

[49] Agostino Di Scipio, “Formalization and Inuition in *Analogique A et B*,” *International Symposium Iannis Xenakis* (2005): 96.

[50] Christopher Thomas Funkhauser, *Prehistoric Digital Poetry: An Archeology of Forms, 1959–1995* (Tuscaloosa, AL: University of Alabama Press, 2007), 37.

By using a grammatical system of control that determines the logical placement of pre-composed materials, the artist is able to impart a sense of logic and syntax in a work.

Finally, Strachey's *Love-letters* uses a random number generator to control the word selection process. The random function is widely used in algorithmic art; by deferring decision-making to chance, it adds an element of variability through repeated executions of the algorithm.[51] In *Love-letters*, one replaced word has the power to change the tone of the entire correspondence. In the visual domain, A. Michael Noll designed a random movement algorithm by which a computer controls an automatic drawing board to create an overlapping mesh of polygons in *Gaussian-Quadratic* (1962-3).[52]

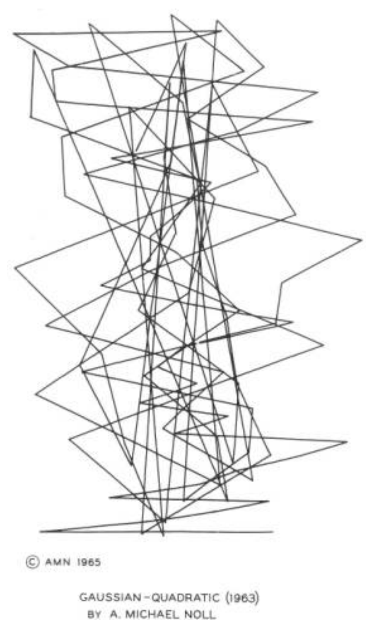

Figure 3.2: A. Michael Noll's *Gaussian-Quadratic* designed using a random movement algorithm.

---

[51] To be completely accurate, no computer or other deterministic device is able to generate truly random numbers. Instead, computers use algorithms called pseudo-random number generators (PRNG) that use linear congruence and a starting seed to create a sequence of seemingly random numbers. There are many different variations of PRNGs that employ more complicated formulae in order to protect everything from files on a home computer, to web commerce, to national security.

[52] Grant D. Taylor, *When the Machine Made Art: The Troubled History of Computer Art* (New York: Bloomsbury, 2014), 32.

Mozart's *Musikalisches Würfelspiel* uses an element of chance, using a pair of six-sided dice to select a measure from the table. The outcome from the roll of two six-sided dice is not entirely random, however. Of the 36 possible combinations of two dice, a few combinations yield the same sum total, meaning that some indices on the table, and their associated musical segments, are more likely to appear in the composition than others. Rolling two six-sided dice results in an approximation of a normal distribution curve:[53]

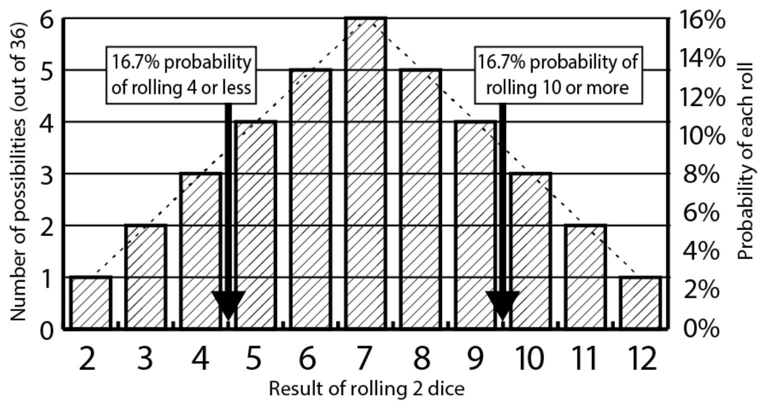

Figure 3.3: An approximation of a normal distribution curve by rolling two six-sided dice.

The implication of this distribution for *Musikalisches Würfelspiel* is that musical segments with indices closer to the number seven have a higher probability of selection. In addition to random and normal distribution control, many other methods of control contribute to animating the results of an algorithm.

Iannis Xenakis's *Analogique A et B* (1958) is an example of "stochastic music," a composition that uses statistical and random procedures to generate material.[54] In the work,

[53] Chris Beales, "Explaining Probability using Dice," http://www.chrisbeales.net/environment/consequences_probability%20dice.html (accessed January 12, 2017).

[54] Iannis Xenakis, *Formalized Music: Thought and Mathematics in Composition*, Harmonologia Series, Rev ed. Vol. 6 (Stuyvesant, NY: Pendragon Press, 1992), 103.

Xenakis employs Markov transition probability chains to determine the selection of pitch sets. In his book *Formalized Music*, Xenakis explains that starting with pitch set f_0 he determined that there would be an 85% chance that the same f_0 would come next, and a 15% chance that pitch set f_1 would come next. If pitch set f_1 was selected, there would be a 60% chance that f_1 again would follow and a 40% chance that f_0 would return:

	f_0	f_1
f_0	0.85	0.4
f_1	0.15	0.6

Table 3.1: A table showing the Markov transition probabilities of pitch sets f_0 and f_1 from Iannis Xenakis's *Analogique A et B* (1958).

While the basic, underlying process is a random number generator, a Markov table describes the probabilities of transitioning from one pitch set to another. Like the probability distribution of a pair of dice in Mozart's game, this guides the outcome of the composition away from a purely random outcome to something more predictable and characteristic. In addition, although Xenakis's work resulted in fixed composition, the processes involved could potentially be run again and yield a work that would exhibit some similarities with the original.

Another area of interest for artists using algorithms came from mathematician John Conway's *Game of Life* (1970).[55] Conway's game popularized the field of cellular automata, originated by John von Neumann in the 1950s, by using a computer to run a simple simulation: a two-dimensional grid of on/off states transitions through an endless series of stages. The outcome of the game depends on a rule set which defines the properties of each grid cell based

[55] Mitchell Whitelaw, *Metacreation: Art and Artificial Life* (Cambridge, MA: MIT Press, 2004), 148-9.

on neighboring cells. In this rudimentary genetic simulation, the result is a constant bloom and decay of cells turning on, spawning life around them, and then dying. Paul Brown was one of the first artists to implement this concept in his 1979 piece *Cellular Automata* where he designed an array of red LEDs to run two variations of Conway's original rules.

More sophisticated genetic art, such as work by Stephen Todd and William Latham, uses a complex set of parameters to define a virtual creature, its survival and mating habits, and a mechanism to combine parameters from mates to create offspring.[56] Inspired by Richard Dawkin's *Biomorphs* simulation,[57] Todd and Latham's *Mutator* (1991) generates abstract forms that progress through a life cycle of growth, reproduction, and death. Two entities can mate, generating an interpolation between their characteristics and passing those traits on to their children.[58] One of the attractions in genetic or evolutionary art is the element of surprise through parameter recombination. Karl Sim's work often invites interaction with the process. His *Genetic Images* (1993), for example, asks participants to select one of sixteen colorful abstract images to proceed to the next round of genetic breeding. The result is a type of selection based on human preference, as is found in the evolution of cultivated plants and domestic animals.

Chaotic, non-linear functions are another source of algorithmic interest to artists. Following the popularity of two texts on the subject, Ilya Prigogine's and Isabelle Stengers'

---

[56] Stephen Todd and William Latham, *Evolutionary Art and Computers* (New York: Academic Press, 1992), 33–4.

[57] Richard Dawkins, *The Blind Watchmaker* (New York: Norton, 1996) 55.

[58] Stephen Todd and William Latham. *Evolutionary Art and Computers,* 75.

Order Out of Chaos: Man's New Dialogue with Nature (1984)[59] and James Gleick's *Chaos: Making of a New Science* (1987),[60] artists began to incorporate chaotic mathematical functions in their work. In general, these algorithms have deterministic behavior in the long-term, but exhibit short-term unpredictability. According to the Central Limit Theorem, this is also true of iterated random processes such as Mozart's multiple throws of the dice; in contrast, however, chaotic functions have particular and non-random tendencies. [61] One tendency of particular interest to artists is self-similarity. Benoît Mandelbrot explored dimensional complexity using point set functions that are self-similar, but not identical across an infinite degree of scale.[62] Mandelbrot coined the term *fractal* to describe his findings and generated images of the resulting point sets:

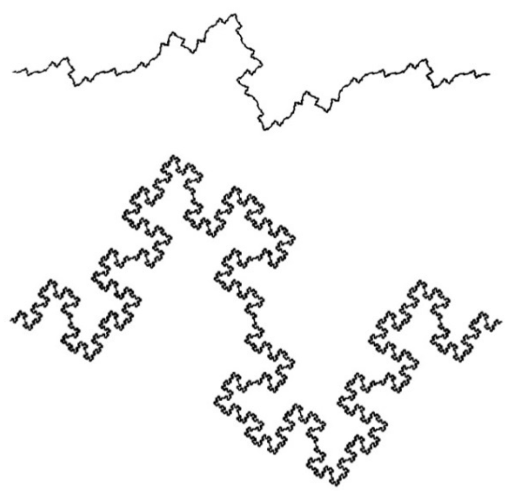

Figure 3.4: Two views of a Mandelbrot function at different scales.[63]

[59] Ilya Prigogine and Isabelle Stengers, *Order Out of Chaos: Man's New Dialogue with Nature* (New York: Bantam Books, 1984).

[60] James Gleick, *Chaos: Making of a New Science* (New York: Penguin, 1987).

[61] Xinxing Wu and Guanrong Chen, "Central Limit Theorum and Chaoticity," *Statistic and Probability Letters* 92 (2014): 137–142.

[62] Benoît Mandelbrot, *Fractals: Form, Chance, and Dimension* (New York: W. H. Freeman, 1977).

[63] Ibid., 37.

Computer artist Herbert Franke's work in the 1980s reflects the early influence of chaotic systems. Franke's *Konforme Abbildung* (1986) uses a combination of random number generators, iterative techniques, and complex mathematics to generate a colorful, abstract image:[64]

Figure 3.5: Herbert Franke's *Konforme Abbildung* (1986).[65]

Scott Draves's *Boom* (1995) and *Electric Sheep* (1999) are more recent examples, both of which use a combination of functions that generate a constantly evolving audio-reactive computer animation. In addition, *Electric Sheep* incorporates an artificial life system that links user's computers across the Internet to allow new algorithms to emerge as the result of genetic crossbreeding. Each animation, or *sheep*, is the product of a distributed network of computers working together, "blending man and machine with code to create an artificial lifeform."[66]

---

[64] Grant D. Taylor, *When the Machine Made Art*, 168-9.

[65] Herbert Franke, *Konforme Abbildung*, compart: Center of Excellence Digital Art, http://dada.compart-bremen.de/item/artwork/341 (accessed February 11, 2017).

[66] Scott Draves, "*Electric Sheep*: Crowdsourced Evolving Art," http://electricsheep.org/ (accessed February 11, 2017).

Artificial neural networks are an advanced application of mathematical algorithms called threshold logic and are used to simulate brain function. Many artists have employed neural networks in an attempt to simulate autonomous behavior that might be perceived as intelligent. Yves Amu Klein's *Octofungi* (1996) is a robotic octopod equipped with locomotion abilities, optical sensors, and a neural network that adjusts its behavior based on external stimulus.[67] Some activity will cause the robot to recoil while other events might invite the robot to approach. The neural network allows *Octofungi* to learn new responses and refine old behavior to stimuli over time. The most recent examples of artificial intelligence, such as Google's DeepMind,[68] demonstrate the ability of complex algorithms to teach themselves, solve problems, and mimic human behavior.

Cartographic Composition in Some Recent Real-Time Notation Compositions

The works described above are cartographic: their designers mapped out territories within which a range of potential behaviors was possible, mediated by human or algorithmic systems or some combination of the two. In a general sense, composition is implicitly cartographic: a composer is a designer of systems that map out possible sonic terrains. Real-time notation (RTN) extends this perspective, allowing the composer to design a system capable of creating several versions of a work that share similarities and yet are distinctly different. In my own works using RTN, critical larger-scale features are consistent from one

---

[67] Yves Amu Klein, "*Octofungi*," Living Sculpture, http://www.livingsculpture.com/works/octofungi (accessed February 11, 2017).

[68] David Silver et al., "Mastering the Game of Go with Deep Neural Networks and Tree Search," *Nature* 529 (January 2016): 484–9.

realization to another, while surface material is subject to variation. In this way, a work can exhibit multiplicity, on the one hand, and distinct compositional characteristics, on the other.

Computer art pioneers Stephen Todd and William Latham are often credited with the analogy of the artist as a gardener in works that display emergent behaviour: "The artist first creates the systems of the virtual world…then becomes a gardener within this world he has created."[69] The image is well suited to musical works that exhibit multiplicity. Gerhard E. Winkler, for instance, uses the analogy, describing the composer as someone, "who plants 'nuclei' or germs, and watches them grow, depending on influences from the environment, in this or that way. All versions are welcome."[70] John Cage made similar remarks about multiplicity regarding his *Concert for Piano and Orchestra* (1957–58), saying that every performance contributes to a holistic understanding of the piece: "I intend never to consider [the work] as in a final state, although I find each performance definitive."[71] Richard Hoadley uses another analogy, asserting that the process of composing a real-time work is similar to mapping the landscape of a geographic territory without describing every rock, tree and bush.[72] In this way, the composer acts as a cartographer, designing a landscape and releasing the performer to explore its details.

Several of my recent works are explicitly cartographic in their design: *Law of Fives*

---

[69] Stephen Todd and William Latham, *Evolutionary Art and Computers*, 12.

[70] Gerhard E. Winkler, "The Realtime-Score: A Missing-Link in Computer-Music Performance," *Proceedings of the 2004 Sound and Music Computer Conference* (2004): 5.

[71] John Cage, *John Cage*, Richard Kostelanetz, ed. (New York: Praeger, 1970) 131.

[72] Richard Hoadley, "Notating Algorithms."

(2015),[73] *Polytera II* (2016),[74] and *Terraformation* (2016–17).[75] *Law of Fives* is written for viola, bass clarinet, marimba, and computer. It was premiered at the Shanghai Conservatory of Music Electronic Music Week in 2015. *Polytera II* is written for flute, piano, and computer. The Calliope Duo, consisting of flutist Elizabeth McNutt and pianist Shannon Wettstein, commissioned and premiered the work in 2017. Finally, *Terraformation*, for viola and computer, resulted from a collaboration with Michael Capone. These works represent a progression and maturation of my cartographic approach, the control structure guiding the works, and the resulting predictable and variable material.

Expansions of Control

In analyzing algorithmic art, it becomes clear that the underlying system—the rules, mathematical functions, and nature of chance operations—reveals meaningful aspects of the resulting art. Each of the three RTN compositions described here uses a multidimensional algorithmic system that controls the output of notation and electroacoustic audio from the computer. Each composition uses a similar paradigm to update the results of the background algorithms and the way they are influenced by the live performer. This paradigm involves two musical staff systems: one that displays notation that the performer is asked to play, and another that shows the performer what is coming next. When the performer begins playing on

[73] Seth Shafer, "*Law of Fives* (2015), for viola, bass clarinet, marimba, and computer," http://sethshafer.com/law_of_fives.html (accessed February 12, 2017).

[74] Seth Shafer, "*Polytera II* (2016), for flute, piano, and computer," http://sethshafer.com/polytera_2.html (accessed February 12, 2017).

[75] Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

one staff system, the other staff system refreshes its contents with the next iteration of the RTN. There is, therefore, a distance of one staff system's worth of music between any real-time influence and algorithmic generation and the presentation of that notation to the performer. This latency is purposely designed to afford the performer the time needed to perform accurate sight-reading by allowing them to read ahead by one full staff system.

Control Structures in *Law of Fives*

Each performance of *Law of Fives* is governed by a web of interconnected algorithms controlled by a scheduled cue list (referred to hereafter as a *qlist*) containing a sequence of pre-determined parameter settings which are further manipulated in real-time by the live performers. The notational behavior of the algorithms behind *Law of Fives*, explained in detail below, applies to all live performers and the computer triggered synthesizer as a group. The pointillistic behavior at the beginning of the piece, for example, must occur in all active voices. In other words, there is no way to have a separate type of behavior occur only in one of the performer's parts.

The performers have some influence over the output of the notation algorithm, however it is not directly discernable. Microphones placed near each musician capture their performance. The computer analyzes the pitch and amplitude envelope of each musician and this interpreted data is routed to control inputs on the various algorithms in the piece. The amplitude envelope of the viola player, for example, controls a randomness factor for the generation of the bass clarinet's rhythms, the probability of rests in the marimba part, and the decay time in the computer's synthesizer. The bass clarinet and marimba player have a similar network of influence over other notational and synthesis parameters:

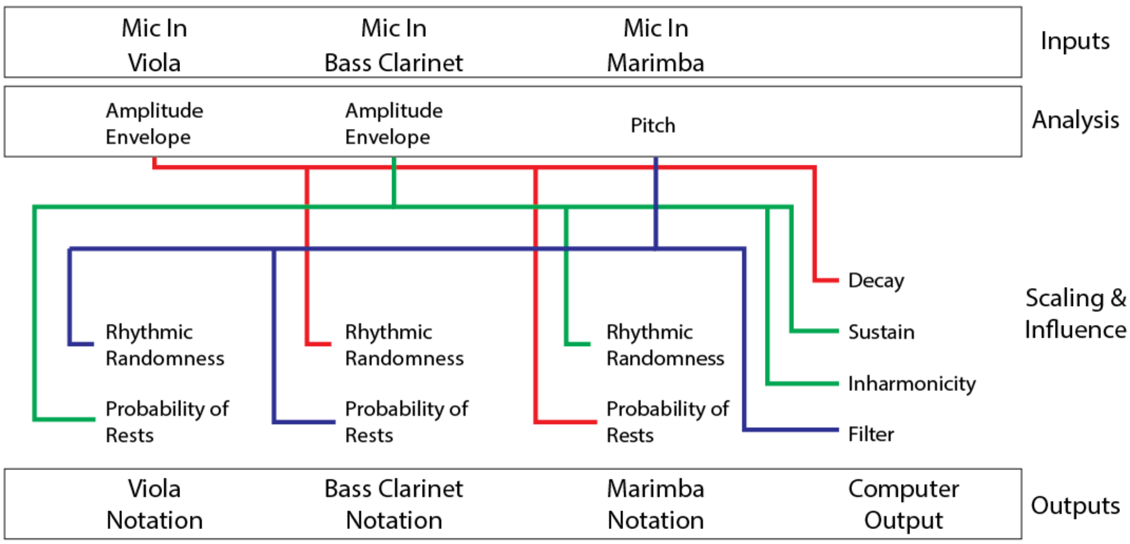

Figure 3.6: A network of live influence in *Law of Fives.*

Although this network shapes the outworking of many parameters in the piece, one performer's individual agency is not readily perceptible.

The musical content of *Law of Fives* is notated using traditional music symbols from common practice notation (CPN) and generally exhibits a strong metrical character. The focus of the piece centers on unusual rhythms and metric modulations. The form, notation style, rhythmic focus, and tempo changes are all pre-determined by the composer.

Control Structures in *Polytera II*

Polytera II expands upon the control structures created in *Law of Fives* and creates more diversity of algorithmic behavior and more opportunities for performer agency. In *Polytera II*, the notation algorithms can be independently assigned to the flute, piano, or computer. Unlike

Law of Fives, each voice can have an independent notational behavior, creating the possibility for more textural variety and idiomatic content. Further, where in *Law of the Fives* the parameters of the algorithms were only assignable to create one behavior per staff system, *Polytera II* allows for multiple algorithmic behaviors to exists in a single staff system. This opens up the possibility to create a transition between behavioral mode A and B, for example, by generating material that is a mix of modes A and B. Again, this opens new doors for more nuanced music composition.

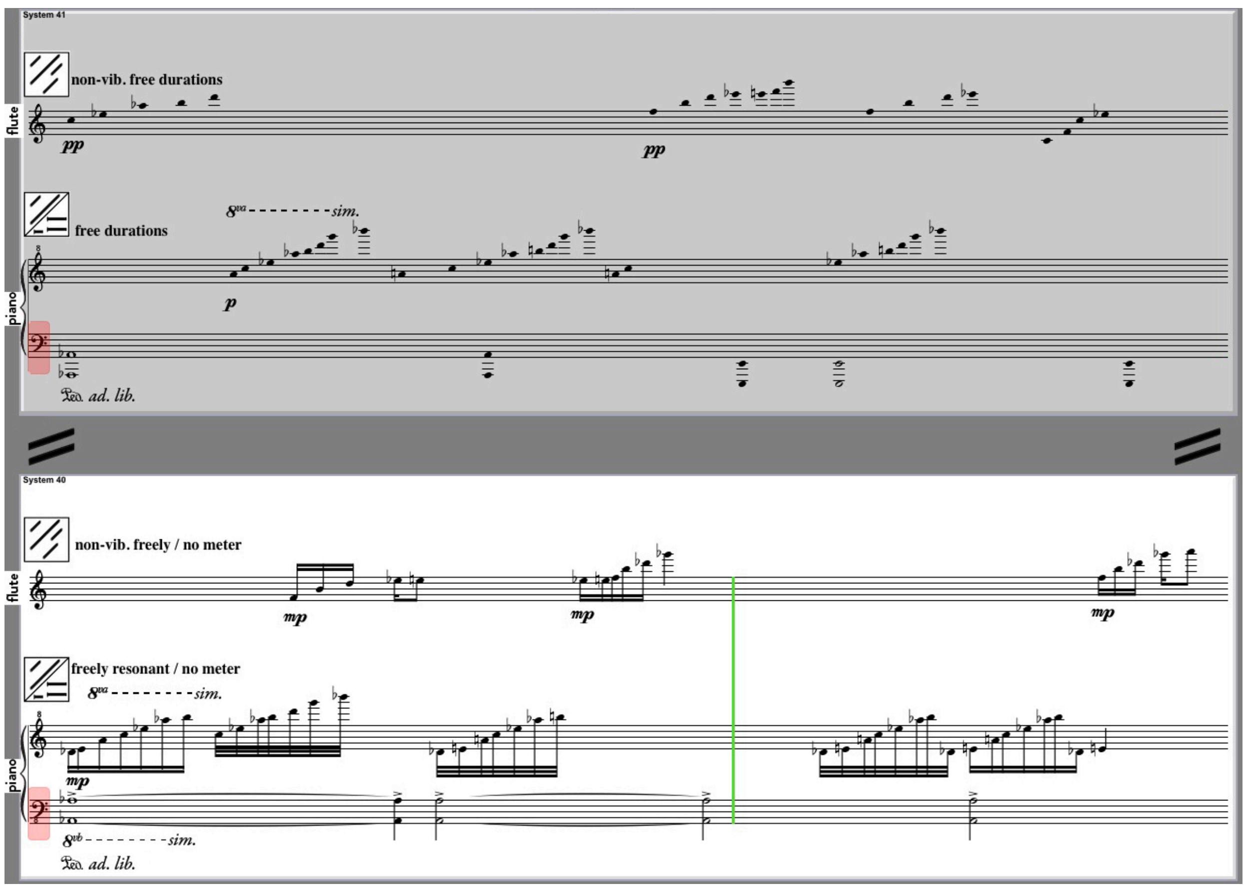

Figure 3.7: The performer interface for *Polytera II* showing a progressive removal of score elements to create proportionally notated music.

As in *Law of Fives*, the performers are able to influence the notation and computer synthesizer in *Polytera II*. The network of influence is similarly indirect in that the musician will not easily connect their performance to an algorithmic response. The work does, however, expand performer agency by loosening the notational specificity in the final section of the work the strict bouncing-ball conductor is replaced by a scrolling play head. Note stems, beams, and all other rhythmic indications disappear over the course of several systems, leaving a proportionally-spaced notation that the performers are instructed to play *ad libitum* (see Figure 3.7).

These additional tools for controlling the composition and freedoms for the performer allow for situations that approach the expressive precision found in my non-RTN works.

Control Structures in *Terraformation*

The underlying algorithmic systems in *Terraformation* are markedly different than the previous two works. One departure from the previous works is the elimination of a visual metronome to dictate the progression through the piece. Instead, *Terraformation* is controlled by the performer with a MIDI foot switch which steps through the pre-determined qlist and triggers each staff system update. This gives the musician the freedom to expressively alter the tempo and choose when to move on to the next staff system. In addition, the performer can choose to cycle through several alternative musical paths at each foot switch. A long press on the foot switch cycles through different options for the next set of musical materials, shown on the lower, read-ahead staff system. The performer can choose any of these outcomes with another short press of the foot switch; this moves the selected material to the upper staff system and generates subsequent notation based on that choice. In this way, the player has great freedom of choice in directing their path through the piece.

The algorithmic control in *Terraformation* is just as flexible as previous works allowing for the combination of behaviors on a single staff system. The notation itself expands on the CPN and proportional notation models by adding a type of action-based notation showing left hand finger placement, left hand finger pressure, and bow contact position. These notation elements are independently displayed and are intended to work together to create the most efficient sight-reading experience possible.

Predictable and Variable Elements

My pursuit of a cartographic approach to composition has led to more flexibility in algorithmic design and a greater amount of freedom given to the performer to contribute to the outcome of the work. As stated before, my desire is to create a work that exhibits a great variability in the specific sonic result without losing an identity that clearly characterizes the piece. Further, the amount of variability over the course of the work is itself a variable element, subject to composer or performer control. The following discussion reveals some specific details of algorithmic design in my RTN works and the procedures taken to ensure the predictability of some musical elements and the variability of others.

Predictable and Variable Elements in Law of Fives

Many elements in *Law of Fives* are repeatable from one performance to another. The formal outline and tempo map are pre-determined in the qlist and remain outside of any algorithmic variation or live influence. The following describes the predictable form of the piece:

System	Part	Tempo	Meter	Orchestration	Pitch Rule	Behavior
1	I	90	4+4+4+4	syn	weighted probability table	Sparse staccato notes
2				mar, syn		
3				bcl, mar, syn		
4						
5				vla, bcl, mar, syn		Note density increases
6						
7		72				
8		90				Medium high note density
9						
10						
11		72				
12		90				
13				vla, bcl, syn		Very sparse note density
14				vla, syn		
15				vla, bcl, mar, syn		
16				bcl, mar, syn		
17		72		vla, syn		
18		90		bcl, syn		
19				syn		
20				vla, bcl, mar, syn		Growing in density
21						
22						
23						
24		72				High note density
25		90				Suddenly sparse with some sustained pitches
26	II	72				Sustained pitches, tremolo sul pont. in vla, timbral trill in bcl, roll in mar
27						
28						
29						
30						
31						
32	III			bcl, mar, syn	random walk	Playful, unison rhythms
33						
34						
35						
36						
37	IV			vla, bcl, mar, syn		Lyrical vla solo, with small interjections from ensemble
38				vla		Lyrical vla solo
39						
40				vla, bcl, mar, syn		Lyrical vla solo, with small interjections from ensemble
41	V	90			weighted probability table	Unison rhythms in ensemble that begin to disintegrate
42						
43						Tutti rhythms disintegrate and grow more sparse
44						
45						
46						Texture decreasing in density
47				bcl, mar, syn		
48				mar, syn		
49				syn		

Table 3.2: A formal outline of *Law of Fives* showing tempo, orchestration, and textural changes throughout the piece.

The form of the piece follows five major behavioral modes that correspond to textural and tempo changes. These modes are the result of the pitch and rhythmic algorithms that create the content of the work.

Two algorithms determine pitch content in *Law of Fives* and introduce some degree of variability in the work. The first produces pitches according to a weighted probability table of possible pitch choices. The table's pitches and associated weighting are pre-determined and therefore somewhat predictable. The viola's pitches in the first five systems of the piece are generated with the following message:

```
vla_pitches      57 80 61 60;
vla_weight       0.15 0.15 0.3 0.4;
```

This message specifies four pitches by their MIDI note number and their associated probability weighting. Every time that a note is requested for the viola, the algorithm will choose a pitch, selecting A♮3 15% of the time, A♭5 15% of the time, D♭4 30% of the time, and C♮4 40% of the time.

The second algorithm chooses pitches based on a random walk, or a succession of random steps, through a table of pitches. The walk is regulated by the size of step, meaning that the composer can impose sequential relationships between the given pitches in the table.

```
mar_pitches      66 67 68 71 72 78 80;
mar_weight       -3 -2 2 3;
```

With the algorithm set to these conditions, the piece will generate a pitch sequence that will randomly step forward or backward by either two or three places in the pitch table. For example, if the current pitch is MIDI note number 71, or B♮4, then the only possible subsequent choices are either forward two or three steps in the table (78 or 80), or backward two or three steps in the table (67 or 66).

The tables of pitches used in the algorithms are based on a dual attractor system where two centricities trace independent paths throughout the piece. In the first section of *Law of Fives*, the two central pitches leapfrog through a sequence of fifths offset by a minor third:

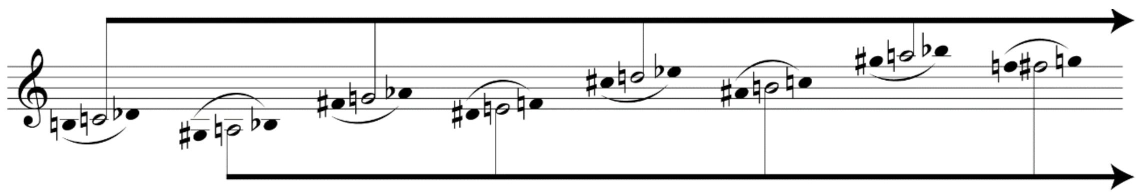

Figure 3.8: Two central pitches surrounded by chromatic neighbors follow a sequence of fifths to generate all of the pitches in the first section of *Law of Fives*.

Each of these central pitches is surrounded by an upper and lower chromatic tone which are orchestrated across the instrumental parts:

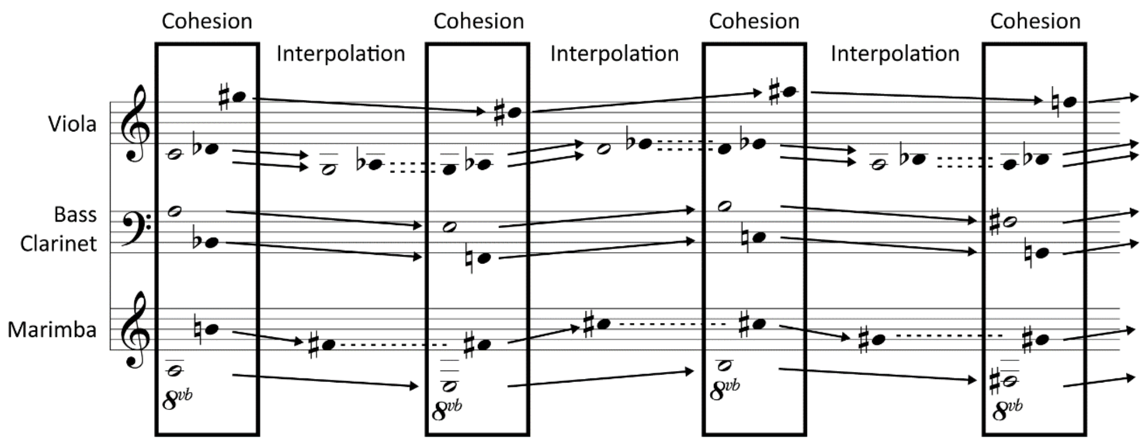

Figure 3.9: The orchestration of the central pitches in the first section of *Law of Fives* indicating moments of cohesion connected through pitch interpolation.

In addition, the transition between one pitch grouping and the next is interpolated so that the overall effect is blurred. In this way, the pitch content of the entire work is pre-determined and it is the order and probability of pitches that is subject to chance.

The permutation algorithm that generates all the rhythmic content of *Law of Fives* is highly variable. The primary rhythmic concern in the work is to create a system of rhythms based on quintuple divisions of the beat. In this proposed system, a given duration can only be divided into five smaller durations, each of which can only be further divided into five divisions. This yields the following rhythmic table showing all possible divisions of the bar:

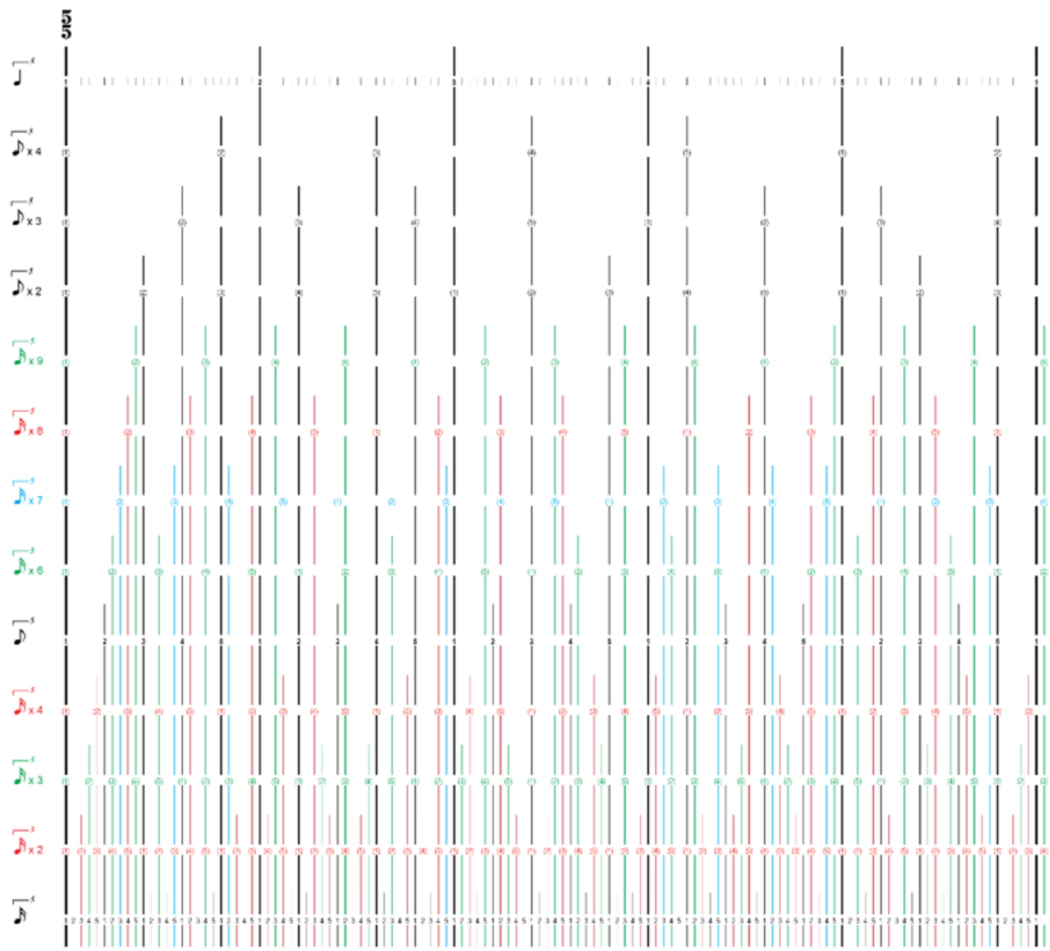

Figure 3.10: A quintuple rhythmic system showing all divisions of a bar.

Many of the available rhythms within this table, when notated in CPN, will require the use of nested quintuplets:

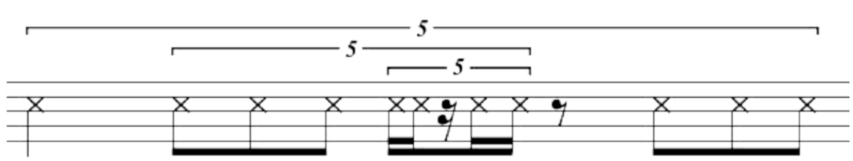

Figure 3.11: An example of a rhythm constructed using rhythms available in the quintuple system in Figure 9.

This notated rhythm will be incredibility difficult to perform accurately and likely impossible to do so when sight-reading. Therefore, the rhythmic algorithm in *Law of Fives* uses durational multiplication and division to eliminate the need for nest tuplets, using instead 16th, 8th, and quarter notes. The full algorithm is illustrated in the following flow chart:

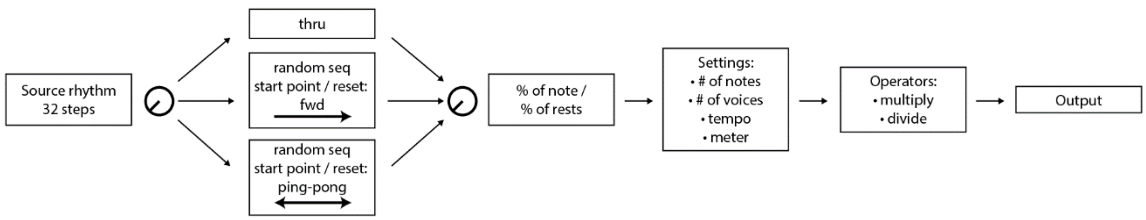

Figure 3.12: The rhythm algorithm for *Law of Fives*.

The first stage of the algorithm reorders a set of pre-determined 32-step sequences of quintuple durations through one of three permutative processes. The first process is analogous to randomly picking up and dropping the needle on a record player; the sequence of durations always moves in a forward direction starting from whatever point is randomly selected. The randomness of resetting the sequence start point is controlled by a value between 0 and 100 where 0 will never reset and 100 will reset after each step in the sequence. Another option is a slight variation on this idea where the direction of the sequence "ping-pongs" forward and backward between the first and last index in the sequence. The remaining option is to simply pass the original 32-step sequence unaltered to the next stage in the algorithm. The following is an example of the 32-step sequence that begins *Law of Fives*:

Figure 3.13: The 32-step rhythmic sequence that seeds the rhythm algorithm at the beginning of *Law of Fives*.

The next process randomly selects some rhythmic values to become notes and some to become rests using a percentage value where 0% is all notes and 100% is all rests. The penultimate stage in the algorithm grooms the durations to fit the number of notes required, the number of voices (in the case of *Law of Fives*, this number is always set to four; one for viola, bass clarinet, marimba, and computer each), the current tempo, and meter. The final step in the algorithm performs the necessary ratio adjustment to make the resulting rhythms readable for sight-reading.

A combination of predictable and variable elements results over the course of the piece by using pre-determined messages to control the algorithms just described. A fixed formal plan with set tempo, meter, orchestration, and texture modulations communicates a reliable character from one performance to the next. The semi-variable pitch content and highly variable rhythmic content create a fluid surface that will be composed differently each time the piece is performed.

Predictable and Variable Elements in Polytera II

Polytera II uses a similar approach to guaranteeing similar large-scale outcomes by employing a set of scheduled parameters to control tempo, meter, orchestration, and texture. Table 3.3 displays the formal scheme for *Polytera II.*

As previously described, the various algorithms controlling the notational output are independently assigned to each instrument. To maximize this feature, several new types of pitch algorithms are introduced that expand on the designs used in *Law of Fives*. Each algorithm uses a pre-composed table of pitches as a pallet of choices.

System	Part	Tempo	Meter	Orchestration	Flute Behavior	Piano Behavior
1	I	77	4+4+4+4	fl, pf, syn, tape	Main Statements, fluttering tuplets	Main Statements, emphatic chords in high register
2					Fluttering tuplets	Emphatic chords in high register
3						
4			4+4+4+3		Mix of fluttering with lyrical lines	Low sustained notes with emphatic chords
5			4+4+4+4		Fluttering only, an increasing number of rests	Emphatic chords only, pitches begin falling in register
6					Fluttering still, lyrical line begins to emerge	
7			4+4+4+3			Emphatic chords still, low sustained notes begin to emerge
8			4+4+4+4			Chords continue to descend, more frequent low notes
9			4+4+4+3			
10			4+4+4+4			
11						
12			4+4+4		Fully lyrical line, no fluttering	Longer durations, sparser texture
13		66			Slowing and losing energy	Slowing and losing energy
14						Pitches continue to descend
15		55		fl, pf, syn		
16		44				
17	II	77	4+4+4+4	fl, pf, syn, tape	Main Statements, fluttering tuplets	Main Statements, emphatic chords in high register
18					Sparse fluttering tuplets	Emphatic chords in high register
19				fl, pf, syn		Emphatic chords mixed with fluttering tuplets
20			4+4+4+3	pf, tape	Resting	
21			4+4+4+4	fl, pf, syn, tape	Isolated impulses	Fluttering tuplets only
22				fl, pf, syn		Emphatic chords mixed with fluttering tuplets
23			4+4+4+3		Sparse fluttering tuplets	Emphatic chords in high register
24			4+4+4+4		Isolated impulses	Emphatic chords mixed with fluttering tuplets
25			4+4+4+3		Multiphonics ad lib	
26			4+4+4+4			Fluttering tuplets mixed with sustained mid-range chords
27						
28			4+4+4		Rising lyrical line, crescendo	Sustained chords rising in register, crescendo
29						
30					Multiphonics ad lib, decrescendo	Fluttering tuplets mixed with sustained mid-range chords
31					Flutter tongue	Same, decrescendo
32						
33			4+2	tape	tape solo	tape solo
34	III	77	4+4+4+4	fl, pf, syn, tape	Main Statements, fluttering tuplets	Main Statements, emphatic chords in high register
35				fl, pf, syn	Fluttering tuplets	Emphatic chords with low sustained octaves
36						Rising lyrical line with low sustained octaves
37			4+4+4+3		Rising lyrical line	
38		66	4+4+4+4			
39		sys ≈ 13 sec	free			
40					Resonant effect on lyrical line (comb filter), sparser	Resonant effect on lyrical line (comb filter) + low octaves
41		sys ≈ 16 sec				Same, sparser
42						
43					Final rising line	Final rising line with low octaves

Table 3.3: A formal outline of *Polytera II* showing tempo, meter, orchestration, and textural changes throughout the piece.

One algorithm selects pitches from the table based on weights applied to each pitch, exactly as before in *Law of Fives*. Pitch choice is further regulated by the size of allowed steps through the table. Essentially, this algorithm combines the features of both pitch algorithms from *Law of Fives*.The following message, excerpted from line 21 of the music, controls the parameters of the algorithm:

```
flu_pitch_mode   weightedpitches;
flu_pitches      94 97 78 81 84 89;
flu_weights      0.25 0.2 0.15 0.15 0.1 0.05;
flu_steps        2 3 -3 -2;
```

The message sets the algorithm mode to *weightedpitches* and then specifies three parameters: the pitch table, the weights associated with each pitch, and the allowed steps through the table.

Another pitch algorithm in *Polytera II* implements a random walk through the table controlled by step size. The allowed steps are further governed by a weighting applied to each allowed step. The following message from system 6 illustrates the how the parameters are applied:

```
flu_pitch_mode   weightedsteps;
flu_pitches      66 69 74 77 80 85 88;
flu_steps        1 2 -2 -1;
flu_weights      0.3 0.15 0.15 0.3;
```

The message first sets the algorithm to the appropriate mode, *weightedsteps*, and populates the pitch table. A list of possible steps through the table follow with weightings that apply to each of the steps. In this case, the algorithm is most likely going to choose adjacent pitches in the table because the weighting for a single step forward or backward is each set to a 30% chance of happening. Moving forward or backward by a step size of two is less like as each is set to a 15% chance of happening.

One pitch algorithm used extensively in the flute part chooses a pair of pitches from the given table and oscillates randomly between them. Each time a rest occurs the algorithm chooses a different pair of pitches. The following is the message to control the pitch output in system 2:

```
flu_pitch_mode   pairs;
flu_pitches      74 77 80 85 88 91 96;
flu_steps        -1 1;
flu_weights      0.45 0.55;
```

Besides declaring the algorithm mode and pitch table, the message also says that only a step size of one is allowed between the pair of pitches and that the upper pitch is weighted more heavily compared to the lower pitch. In one sequence of the algorithm, the following output occurs:

Figure 3.14: One potential notation resulting from the *pairs* pitch algorithm in *Polytera II*.

Again, the rests between the rhythmic groupings trigger a new selection of note pairs. The oscillation between the pairs is random but distributed according to the weighting table, favoring the upper of the two selected pitches.

The final pitch algorithm provides an option to read linearly through the pitch table in ascending, descending, ping-pong, or random order. This allows for the possibility of specifying an exact ordering of notes for predictable, non-variant, moments in the work. This is extremely important in *Polytera II* as a pre-composed staff system occurs at the start of each of the work's three sections. Not only does this provide the piece with a sense of recapitulation, but it also addresses a chief concern in real-time notation; that although the work exhibits a multiplicity across different performances, audiences will likely only hear a single performance of the work. In *Polytera II*, the piece is constructed so that it "restarts" three times with pre-composed "seed" material and proceeds to algorithmically develop in three distinctly different directions. The form of the work, therefore, solves the problem of multiple performances by recapitulating familiar material and nudging the algorithmic development in three contrasting directions.

In designing the rhythmic content of the work, it was decided that the permutative engine used in *Law of Fives* had severe limitations and needed a complete redesign. The new rhythmic algorithm is similar to a Bayesian network where each chosen duration is the result of a Markov table associated with the previous duration.[76] The algorithm is a kind of first-order causal chain in which future choices are based on previous choices. This allows for the construction of rhythms that exhibit a large variation in content while retaining similar characteristics. Take, for example, the following rhythmic pattern:

Figure 3.15: A sample four-bar rhythmic pattern.

[76] Judea Pearl, "Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning," *Proceedings of the 7th Conference of the Cognitive Science Society* (1985): 5–6.

One can construct a message to the rhythm algorithm to generate similar patterns. By first analyzing the location in the bar of each note's onset, we can summarize all rhythmic activity as a single composite measure. Such a composite would look like:

Figure 3.16: A composite of the sample rhythmic pattern showing all locations of note onsets.

Each bar of the original sample rhythm could be reconstructed with the composite rhythm, ignoring the issue of durations:

Figure 3.17: The original rhythmic pattern constructed from note onsets in the composite rhythm, ignoring any resulting durations.

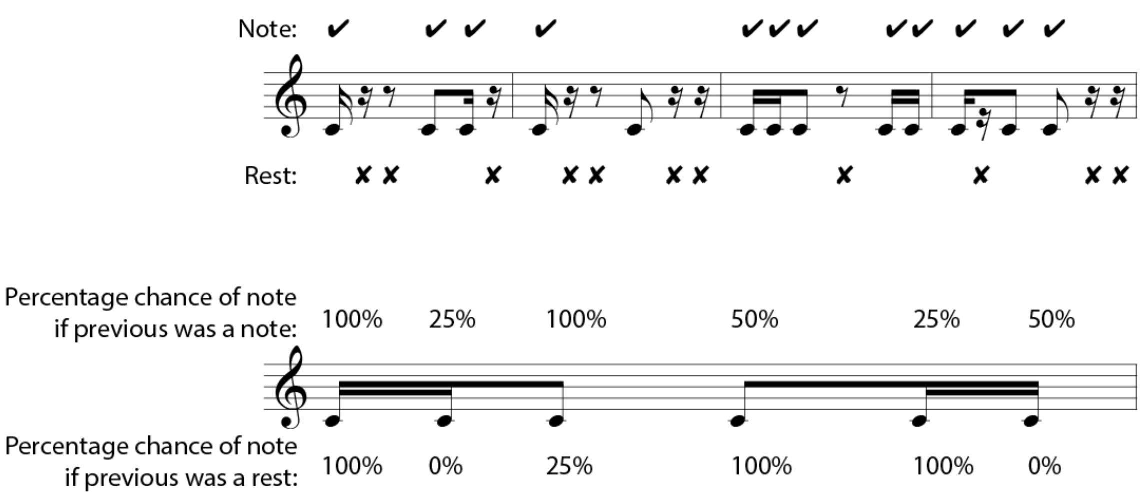

Figure 3.18: An analysis of the probability of each note onset being followed by a note or a rest based on the sample four-bar rhythm.

In order to build a message for the algorithm that will generate more material like the sample rhythm, we must build a Markov table that describes the probability of each note onset being followed by a rest or being followed by a note. An analysis of the original four bar rhythm yields the probabilities displayed in Figure 3.18.

This analysis first counts the number of rests and notes and compares their location to the composite rhythm. Then, the total number of times notes in each onset location were preceded by a note are compared with the number of times notes in the same onset location were preceded by a rest. The second beat of the composite rhythm can be used as an example. In m. 1 beat 2 of the four bar sample rhythm, the note occurs and is preceded by a rest. The same is true of the note in the identical location in m. 2. Therefore, 100% of the time that beat 2 is preceded by a rest, a note occurs. In m. 3 beat 2, the duration is a rest and it is preceded by a note. The final measure has a note on beat 2 and it is preceded by a note. Therefore, 50% of the time that beat 2 is preceded by a note, a note occurs.

Once these values are tallied, their information can be used to build a message that describes the behavior of the four bar sample rhythm:

```
(2 4) (1/16 (100 100)) (1/16 (25 0)) (1/8 (100 25))
(1/8 (50 100)) (1/16 (25 100)) (1/16 (50 0))
```

The syntax of the message is illustrated in Figure 3.19.

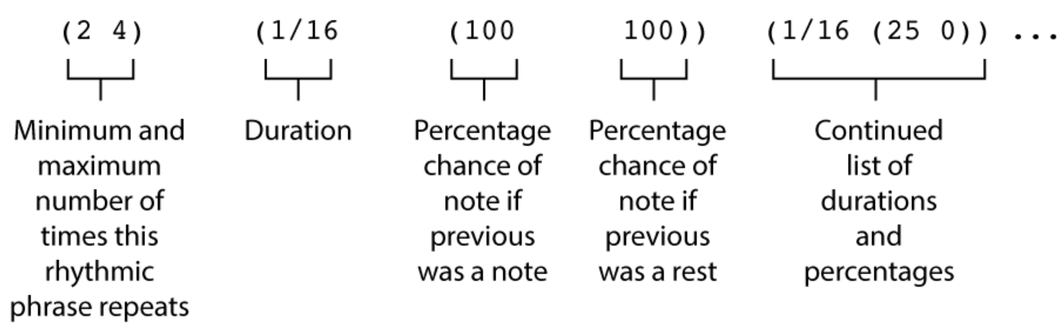

Figure 3.19: The syntax of the rhythmic Bayesian network algorithm.

The message describes every onset location in the composite rhythm along with the Markov table associated with whether the note should be a note or a rest. An additional parameter is appended to the beginning of the message that asks the algorithm to loop the message a random number of times between two given constraints to make rhythms that are somewhere between two and four bars in length. The result of four separate iterations of the algorithm demonstrate a remarkable number of similarities to the sample rhythm:

Figure 3.20: Four iterations of the rhythmic Bayesian network algorithm.

The algorithm allows for many variations while retaining the behavioral characteristics of the original sample. The actual application of this system in *Polytera II* involves much more complex messages with equally complex outputs. The following example message and notational output illustrate its usage:

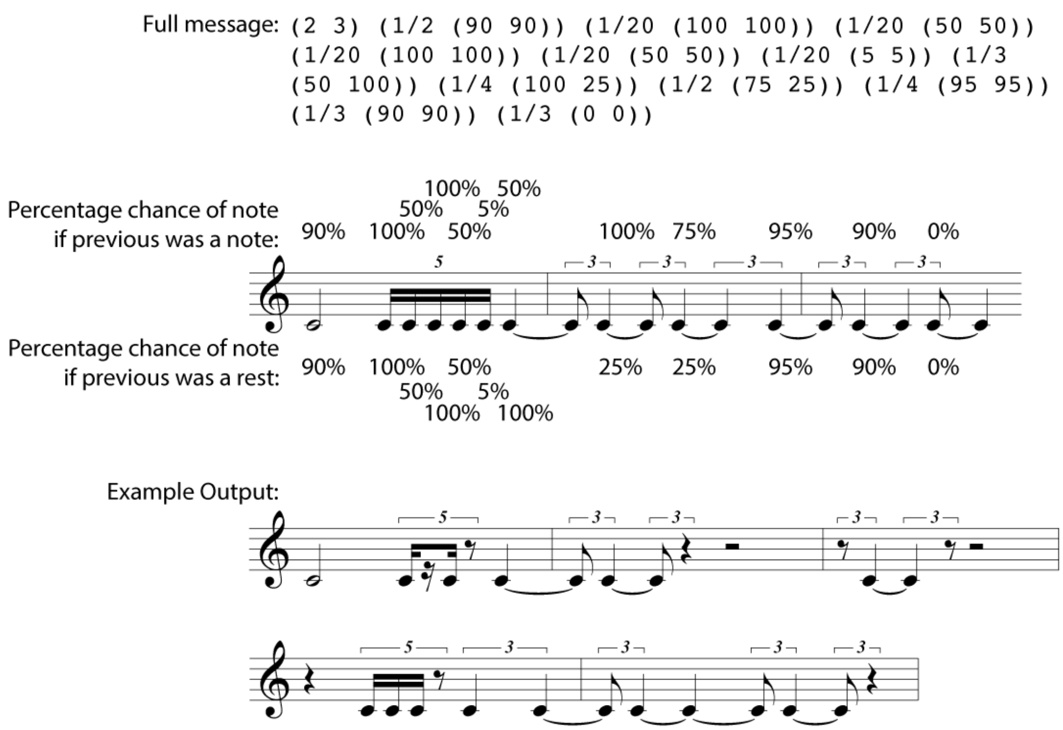

Figure 3.21: The message from systems 8 through 10 of *Polytera II* with a visual guide to the Markov table and one resulting notation.

The content of the message loops between two and three times and contains a complex network of interdependent rhythms that are either chosen as a note or a rest based on past choices. Ultimately, the content of the rhythmic algorithm is joined with the output of the pitch algorithms to make the surface material for *Polytera II*.

Predictable and Variable Elements in Terraformation

As previously described, the performer controls the progress through *Terraformation* by stepping on a MIDI foot switch. Each time the foot switch is pressed, several interconnected processes calculate the notational output and real-time computer audio. The notational content comes from a catalog of pre-composed performative gestures and modes of behavior:

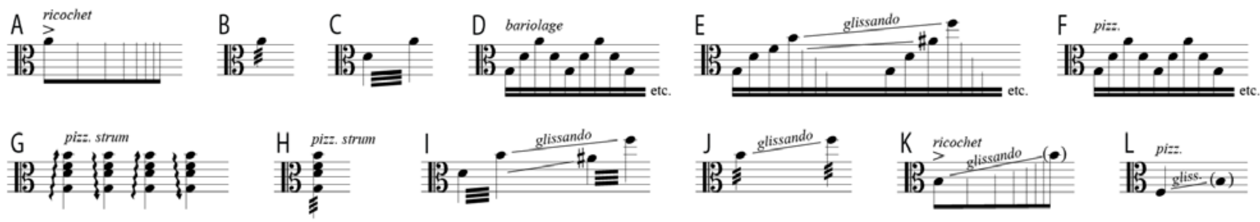

Figure 3.22: A catalog of gestures used in *Terraformation*.

Each of the gestures in the catalog represents a host of similar gestures designed to have a degree of variability and difference. The catalog is organized in the following general form over the course of the work:

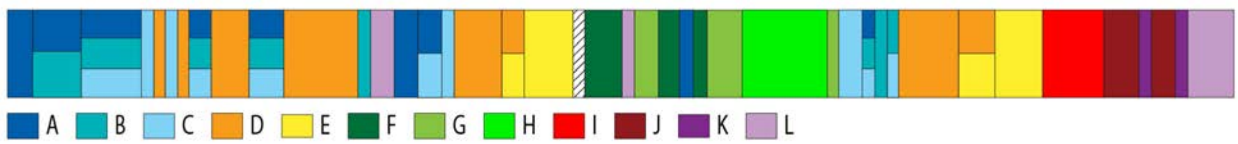

Figure 3.23: The organization of gestures in *Terraformation*.

Each of the gestures is constructed algorithmically so that the resulting notation in performance can be anywhere along a continuum delimited by rigidly fixed and wildly variable. The location of the notational output along this continuum is itself another compositional device. In other words, the variability of pitch and rhythmic content is another parameter in the composition. The following diagram illustrates the degree of variance of the pitch and rhythmic content over the course of the work:

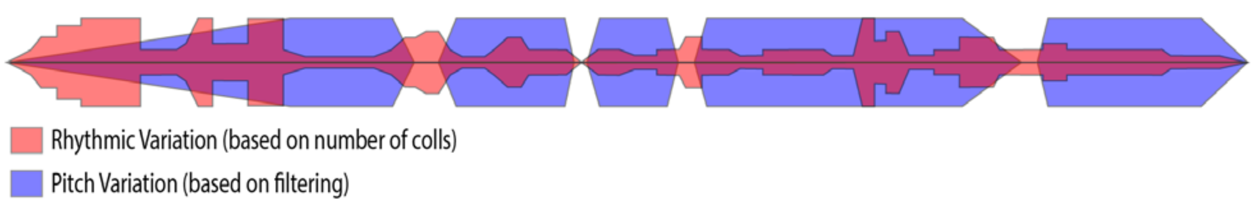

Figure 3.24: The degree of variance of pitch and rhythmic content over the course of *Terraformation* based on parameter data in the qlist.

The fluctuation in rhythmic variance is based on the number of possible rhythm patterns fed from the qlist to the Bayesian network-based algorithm at each foot switch press. There are many points throughout the piece where rhythmic variance is so minimal that certain rhythms are guaranteed to occur. The beginning and end are such places. Other places in the work are highly variable. Similar statement can be made about the pitch variation. The beginning and ending are very controlled while many sections in the work are fluid.

The pitch algorithm is based on a model of the hand and finger movements required to performer a quadruple stop on the viola. The specifics of this action modeling are described later in Chapter 4. The control structure for guiding the resulting pitches is based on a filtering scheme. The algorithm proposes a list of likely candidate pitches, ranked by the difficulty of their performability in context. The filtering options cull the results based on two criteria: pitch class and pitch class set. A request to filter by pitch class will rank pitch candidates based first on their inclusion of the pitch class and then by their difficulty. Filtering by pitch class set more broadly searches for members of a pitch class set and its associated inversions and transpositions. Both pitch filters can run in sequence, first prioritizing pitch class, then pitch class set, and finally difficulty. The candidate at the top of the filtered list is combined with the other notational elements and proposed to the musician.

The performer is invited to participate directly in the selection of material throughout the work. The pitch algorithm will always suggest the highest scoring candidate based on the filtering prerequisites. However, the performer may choose to see the second, third, or fourth highest scoring candidates by depressing and holding the foot switch for approximately 500ms. The “long press” lets the performer cycle through the top four pitch candidates, selecting an option for performance with a "short press." Once selected, the pitch algorithm creates a list of subsequent options based on the current pitch material. In this way, the performer's choice effects the outcome of subsequent material which are also open to performer selection. The pitch material of the piece can be thus directly shaped by the performer's selection process.[77]

The resulting pitch material is potentially very chromatic, involving the use of accidentals in the notation. Without intervention, the chromatic spelling of pitches can result in a dense, unintuitive score where sharps and flats are mixed without any regard to conventional pitch spelling rules. An algorithm roughly based on David Meredith's *ps13* pitch spelling algorithm maps given pitches along a continuum of fifths and finds the closest possible spread (see Figure 3.25).[78]

"White" pitches, or those that do not need accidentals, are used as anchor points from which to measure "black" pitches. In this example, MIDI note number 56 is notated as A♭ rather than G♯ because it is closer in proximity to both C♮ and G♮. Some pitches defy easy solutions. Tritones, for example, are equidistant from each other; their spelling must be determined using more criteria. One criterion is an additional rule that tries to not duplicate letter names (ie. F♮ and G♭ are generally preferable to F♮ and F♯) and can help resolve a conflict between equidistant pitches. While occasionally imperfect, the pitch spelling algorithm in *Terraformation* satisfies the need to make quick and efficient decisions when the performer advances the notation.

[77] The performer selection mechanism was inspired by Ricardo Climent's "*Russian Disco* (2007), for flute, clarinet, and electronic expression," http://citizenurge.blogspot.com/2013/10/russian-disco-2007-cl-flute-and.html (accessed February 17, 2017).

[78] David Meredith, "The *ps13* Pitch Spelling Algorithm," *Journal of New Music Research* 35, no. 2 (2006): 121–159.

E♭♭	B♭♭	F♭	C♭	G♭	D♭	A♭	E♭	B♭	F	C	G	D	A	E	B	F♯	C♯	G♯	D♯	A♯
						✔				■								✗		
							✔			■									✗	
				–						■						–				
						✔					■							✗		
							✔				■								✗	
				✗							■					✔				
		Totals:		0		2	2									1				
						✔	✔			✔	✔					✔				

Figure 3.25: The pitch spelling algorithm in *Terraformation* maps given pitches along a continuum of fifths and selects the closest grouping.

Each press of the foot switch also triggers qlist data to control the computer's audio output. This involves several processes that are somewhat predictable in their behavior. The sample playback engine, for example, triggers pre-recorded sounds. One type of trigger is through direct scheduling, so that when the musician steps on the foot switch for the *n*th time, sample *m* begins to play. Another type of trigger is a random triggering sub-process called *sns.woggle* after the Wiard Synthesizers hardware module the Woggle Bug.[79] Once activated, the *sns.woggle* process can randomly play samples from a table of choices, the playback rate and density of samples based on parameters associated with the *sns.woggle*'s two asynchronous clocks. The type of sample playback behavior is therefore somewhat predicable and composed into the qlist.

The live processing applied to the viola's output is also controlled by messages sent from the qlist and triggered by the musician. One audio processing module, called

[79] Wiard Synthesizers, "Model 371 Dual Woggle Bug," http://wiard.com/modular/300series/wogglebug/ (accessed February 11, 2017); The Wiard Woggle Bug is based on the Buchla Model 265/266 "Source of Uncertainty," https://buchla.com/product/266e-source-of-uncertainty/ (accessed February 11, 2017).

sns.combreson, is a 10-band comb filter bank with independently adjustable fundamental frequencies for each band. A triggered qlist message to the *sns.combreson* sets the initial parameters and routes the viola's output into the processor. In addition, the control message turns on a random low-frequency oscillator that slowly modulates the fundamental frequencies of each band. Like the sample player and notation engine, the behaviors of the live processing engine can be exactly specified or subjected to chance conditions within pre-determined limits.

Conclusion

The complex decision-making processes described in my compositions (and in algorithmic art generally) are made possible by increasingly capable deterministic machines: research mainframes, personal computers, mobile devices, and embedded microchips to name a few. Early adopters and proponents of machine-generated art realized that the process of creation was strikingly different from that of traditional art-making. In their seminal article on computer art, Arnold Rockman and Leslie Mezei observed:

> In the twentieth century we now have much more elaborate machines which can be so arranged that they not only produce the final work of art but can also assist the artistic practitioner during the actual process of creation…. During the process of 'programming' the computer, we are forced to reconsider the whole process of artistic creation and response.[80]

One of the principle differences of computer-assisted creation is noted by A. Michael Noll when he says, "Some time ago it became apparent that the computer should be doing

[80] Arnold Rockman and Leslie Mezei, "The Electronic Computer as an Artist," *Canadian Art* 94 (1968): 265.

things which would be very difficult otherwise."[81] In other words, computer-generated artwork grew out of the idioms associated with the strengths of the machine: speedy calculation, unmatched precision, and the tantalizing possibility of an algorithm that learns from previous experience.

The computer, then, becomes a collaborative partner to the artist, participating in the creative art to whatever extent the programmer allows. The machine can simply realize the artist's vision, as in Noll's *Computer-Generated Ballet* (1965), where Noll programmed dance choreography to be rendered by the computer into a stereoscopic animation.[82] On the other hand, the machine can be given complete autonomy to create as in Harold Cohen's *AARON*, where a computer equipped with a robotic arm and a learning algorithm applies paint to canvases with brushes.[83] In the latter case, the authorial distinction between human and machine is difficult to discern. Cohen describes the issue himself:

> The central idea that has really been at the root of everything I've done [is] that I could externalize what I knew about art-making into a form where a machine could do it; eventually, even, that it should be possible for a computer program to have enough knowledge to function autonomously as an artist.[84]

In a similar way, my work using algorithms to generate musical compositions requires that I provide the computer with rules about composing derived from my own artistic practice.

---

[81] A. Michael Noll quoted in Arnold Rockman and Leslie Mezei, "The Electronic Computer as an Artist," *Canadian Art* 94 (1968): 266.

[82] A. Michael Noll, "The Beginnings of Computer Art in the United States: A Memoir," *Computers and Graphics* 19, no. 4 (1995): 501.

[83] Pamela McCorduck, *AARONS's Code: Meta-Art, Artificial Intelligence, and the Work of Harold Cohen* (New York: Macmillan, 1991).

[84] Harold Cohen, "A Sorcerer's Apprentice: Talk at the Tate Modern," (2004), http://www.aaronshome.com/aaron/publications/tate-final.doc (accessed February 15, 2017).

The computer, then, is given the ability to make arbitrary decisions in the process of creating the work. My role as composer is to shape the general outcome of those decisions. In addition, I also invite the performer(s) to manipulate the resulting music. In these situations, who is the author of the composition? I began the process by programming the computer and enlisting the musicians; but I do not complete the creative act alone.

Compositions using real-time notation embrace the central premise that all music is fundamentally variable. No amount of score detail or performer instruction can ensure that a work is replicable; even fixed media electroacoustic works are subject to changes in media technologies, playback hardware, room acoustics, and psychological factors affecting listeners. The variation, nuance, and interpretation of a work imbue musical performance with the ephemerality of the moment. A work that is cartographically composed specifically seeks this fleeting experience by designing a system that allows specific control where desired and variable chance operations everywhere else.

The principles of cartographic composition are inspired by algorithmic art in the larger visual, literary, and performative domains. The nature of the algorithmic system and the control structures guiding the outworking of the system are of particular interest. The earliest procedures remain the most common: combination, permutation, and random selection. The rise of the computer's role in making artistic decisions is in part due to the computer's ability to implement these procedures efficiently and invite increasingly complex functions. These include statistical-based models, grammatical models, chaotic functions, genetic simulations, neural networks, artificial intelligence, and more. In all cases, the essential character of the resulting artwork can be found in the underlying system that generated the work.

A cartographic approach to composition is primarily concerned with designing the topography of a piece, ensuring that important features that define the character of the work are prominent, and leaving the performer to explore the detailed reality of the land. This inadvertently involves the use of algorithms to generate material that can be reliably consistent, wildly variable, or, more likely, something in-between those two extremes. My explorations using this approach resulted in three works: *Law of Fives*, *Polytera II*, and *Terraformation*. The underlying systems and methods of control used in each piece are markedly different and illustrate a progression toward both more compositional control and greater performer agency.

Chapter 4

Performer Action Modeling in Real-Time Notation

Introduction

Physical gestures are perhaps the oldest form of human communication, predating vocal language. Recent anthropological research points to the universal phenomenon of manual sign languages and their ease of adoption by infants to suggest that such gestures were the primary communication mode of early bipedal hominins.[85] Similarly, the notation of manual action precedes any notation resembling common practice notation (CPN). Clay tablets dating to the Old Babylonia period (ca. 2000–1700 B.C.E.) depict scales on a four-stringed lyre using cuneiform tablature notation, arguably making action-based music notation the oldest form of music notation.[86]

While tablatures for specific instruments (lute, guitar, organ, etc.)[87] have existed for centuries, a generalized approach to action-based music notation has only been attempted in the twentieth century. For centuries before, CPN focused on notation suitable for describing the resultant sound. Action notation is typically subsumed under the more general category of graphic music notation or text-based music notation, both of which act as extensions or replacements of CPN. These additions and expansions developed concurrently with similar

---

[85] Michael C. Corballis, "Did Language Evolve from Manual Gestures," in *The Transition to Language*, ed. Alison Wray (New York: Oxford University Press, 2002), 162.

[86] Schoyen Collection, "MS 5105," http://www.schoyencollection.com/music-notation/old-babylonia-cuneiform-notation/oldest-known-music-notation-ms-5105 (accessed February 10, 2017).

[87] Charles Francis Abdy Williams, *The Story of Notation* (London: Walter Scott Publishing, 1903), 31.

trends in the visual art world. This chapter will describe several ways composers have notated performer action rather than resultant sound.

Action-based music notation is a viable solution for a major problem in real-time notation (RTN), namely the need for efficient notation in order to facilitate quick and accurate sight-reading. "Pure action-based scores in fact utilize images that suggest clear instructions at first sight and need no further explanation. Such scores could literally be sight-read!"[88] My RTN work, *Terraformation* (2016–17) for viola and computer,[89] uses a combination of action-based notation and CPN. The action-based elements are generated from a model of the physical actions required to produce sounds on the viola. The notation is designed to evoke complex and expressive musical outcomes while being as visually efficiently as possible. In this way, I propose that the application of action-based notation to RTN is both a fruitful extension of the action-based experiments in notation and a solution to one of the key problems of real-time composition.

Notating Action

Music notation mediates the relationship between composition and performance. Expansions of notational language correspondingly expand and modulate those relationships. The following discussion explores different expansions of CPN through the addition of abstract

[88] Juraj Kojs, "Notating Action-Based Music," *Leonardo Music Journal* 21 (2011): 67.

[89] Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

graphics or textual direction and their effect on compositional process and performance practice.

Resultant Sound Notation

Many notations have been developed through the twentieth and early twenty-first centuries, but not all of them refer to action. Like CPN, some notations invent new ways of notating resultant sound. John Cage's score for *Aria* (1958),[90] for instance, uses line contours plotted on a Cartesian pitch/duration axis colored in such a way as to represent different styles of vocalization. The notation uses symbols distilled from CPN to address traditional parameters of music rather than performer action.

Karlheinz Stockhausen's *Plus Minus* (1963)[91] is another example of new notation that only addresses the resultant sound. The score for *Plus Minus* asks the performer to construct the details of the piece by reconciling a complex set of instructions with several pages of abstract graphics. The work is a set of instructions for making an indeterminate number of compositions based on the number of performers and order in which the graphics are combined. Like *Aria*, Stockhausen's use of graphics and text is directed toward musical parameters like pitch, duration, tempo, dynamics, and articulation. These, and many other such examples of notational innovation, seek to address the resulting sound rather than directing performer action.

---

[90] John Cage, *Aria*, for voice, any range, (New York: C.F. Peters, 1963).

[91] Karlheinz Stockhausen, *Plus Minus,* 2 x 7 Seiten für Ausarbeitungen (London: Universal Edition, 1963).

Performative Action in Notation

Directing a musician's action in performance is a relatively new development in the history of notation. One of the earliest forms of performative action in CPN can be traced to textual stage directions in theatrical works.[92] Before that, several types of action-based notation existed for the purpose of communicating and preserving dance choreography.[93] Many experimental notation systems in the twentieth and twenty-first centuries ask the performer to engage in detailed bodily or instrumental action. The range of action techniques and notational language demonstrates the variety of reasons for such use: music as theater, sound production, indeterminate parameters, notational efficiency, intentional complexity, or performer freedom to name a few.

The notation of some actions is directly correlated with playing. This is often the case when writing for a new instrument without an established tradition of performance practice. Luigi Russolo in *Risveglio di una città* (*Awakening of a City*) (1913–14),[94] for example, notates the speed, pressure, and resulting dynamic of his crank-driven *intonarumori* instruments.

---

[92] See, for example, Richard Wagner's staging directions as documented in Katherine Syer, "From Page to Stage: Wagner as *Regisseur*," in *Richard Wagner and His World*, ed. Thomas S. Grey (Princeton, NJ: Princeton University Press, 2009), 3–22.

[93] Ann Hutchinson Guest, *Choreo-Graphics: A Comparison of Dance Notation Systems* (New York: Routledge, 1998).

[94] Luigi Russolo, *Risveglio di una città* (1913–14), only surviving excerpt found in his article "Grafia enarmonica per gli intonarumori futuristi," *Lacerba* 2, no. 6 (March 1914).

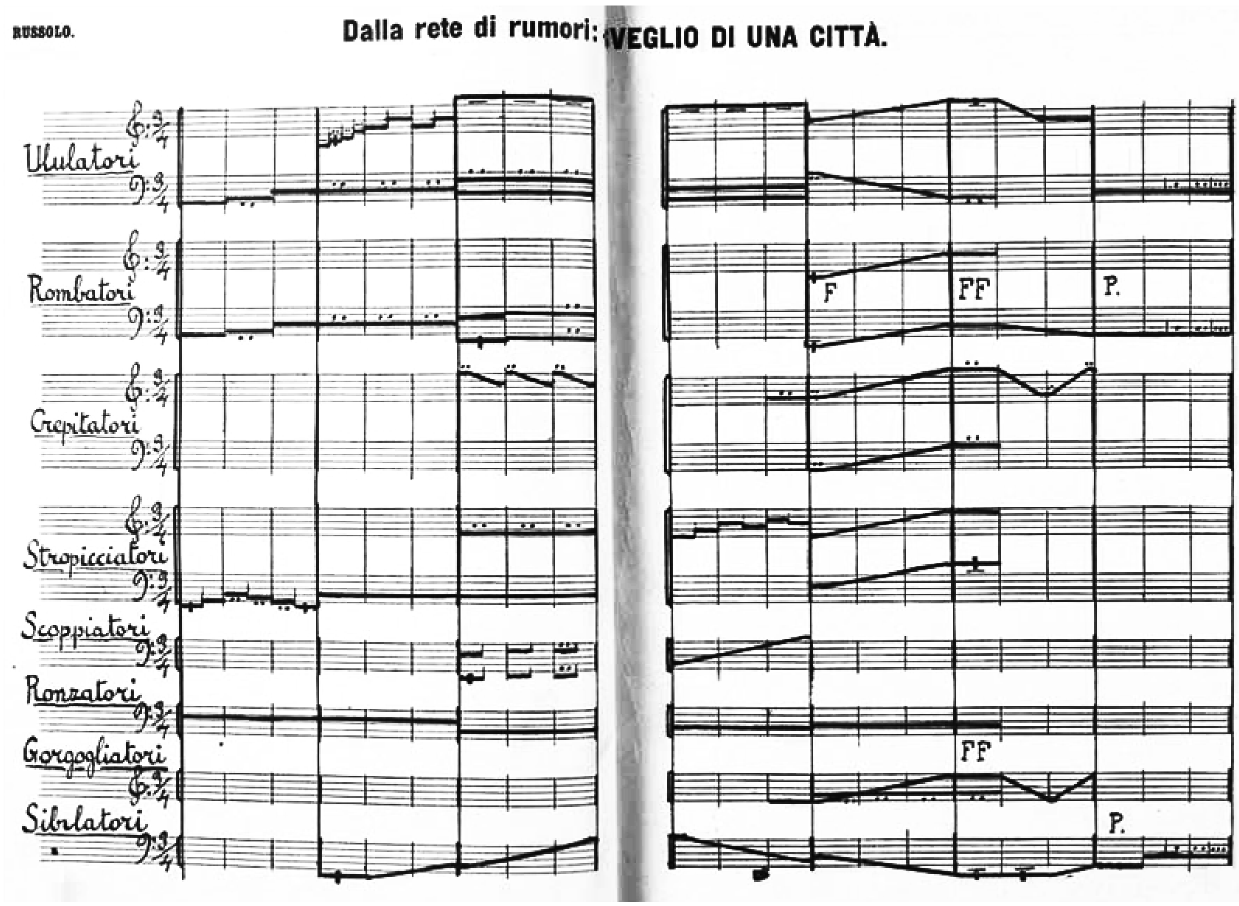

Figure 4.1: Action notation in Luigi Russolo’s *Risveglio di una città* (1913–14).

Russolo combines familiar CPN elements like five-line staves and time signatures with graphic line contours similar in appearance to Cage’s line contours for *Aria*. The difference, however, is that Cage’s contours implicitly rely on the interpretation of musical parameters while Russolo’s notations act as instrumental tablature.

Notating performative action is a central feature of the Fluxus movement. Inspired by John Cage’s courses at The New School in the 1950s, artists such as George Brecht, Al Hansen, Allan Kaprow, and Alison Knowles began to write text-based scores that often directed a performer’s actions. An example is Brecht’s *Drip Music, Fluxversion 1* (1959):

> First performer on a tall ladder pours water from a pitcher very slowly down into the bell of a French horn or tuba held in the playing position by a second performer at floor level.[95]

This work, and many others created by Fluxus composers, is entirely dependent on actions that are difficult to notate in CPN. A textual description of the actions is the most efficient means of communication from composer to performer.

Notating action also opens new avenues for performance techniques on instruments with established performance traditions. Helmut Lachenmann's *Pression* (1972, rev. 2010)[96] and *Gran Torso* (1971–72)[97] employ a mixture of CPN and tablature notation in order to explore new instrumental sounds in his pursuit of *musique concrète instrumentale*. Lachenmann's introduction of the "bridge clef" and "string clef" enable the notation to directly mediate a non-standard action on the instrument. The resulting sounds of Lachenmann's actions are innately connected with the action required to produce the sound. The sound of ricochet bowing, for example, is impossible to produce using any other technique. Actions themselves are sometimes unintuitively related to the resulting sound. In his 2010 revision of *Pression*, "action dynamics," notated as dynamics in quotation marks, suggest the physical force of an action required to produce a sound with a disproportionate dynamic outcome.

---

[95] George Brecht, "*Drip Music, Fluxversion 1* (1959)," in *The Fluxus Performance Workbook*, ed. Ken Friedman, Owen Smith, and Lauren Sawchyn, 22 (Performance Research e-Publication, 2002), https://thestudio.uiowa.edu/fluxus/sites/default/files/FluxusWorkBook.pdf (accessed March 14, 2017).

[96] Helmut Lachenmann, *Pression* (Wiesbaden: Breitkopf & Härtel, 1969, rev. 2010).

[97] Helmut Lachenmann, *Gran Torso* (Wiesbaden: Breitkopf & Härtel, 1971, rev. 1988).

The discrepancy between action and sound and the discovery of new modes of sound production is a hallmark of Aaron Cassidy's work. He accomplishes this by decoupling and modulating a large set of action parameters in tablature notation:

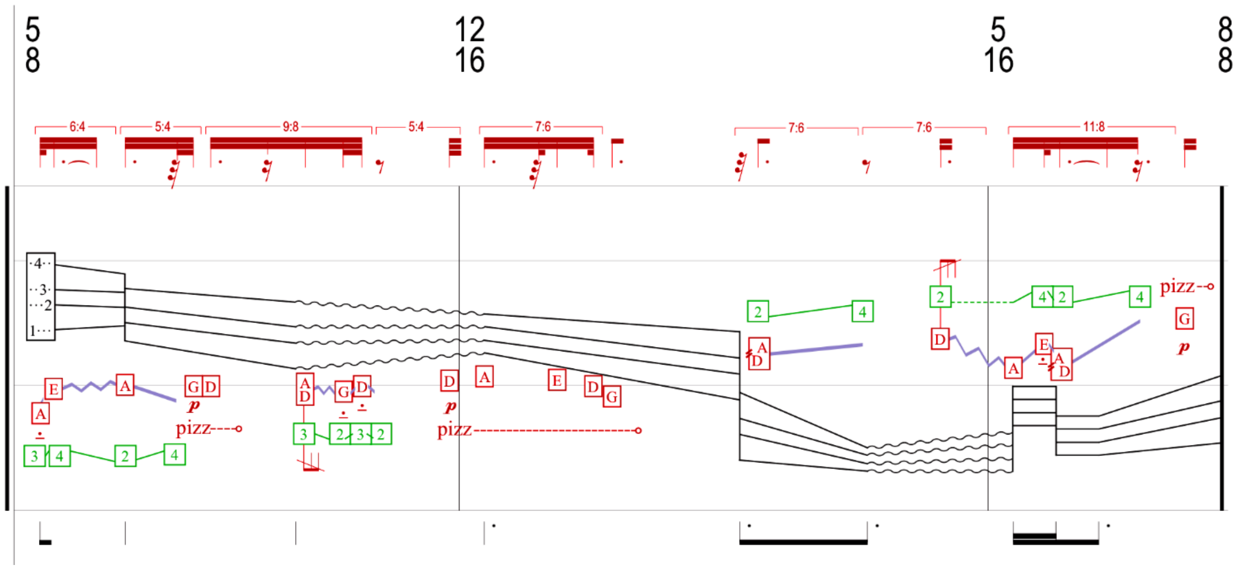

Figure 4.2: Decoupled string tablature in the second violin part from Aaron Cassidy's Second String Quartet (2009–10).

In his indeterminate string work *The Crutch of Memory* (2004)[98] and his Second String Quartet (2009–10),[99] Cassidy loosely specifies pitch information by providing the performer with a graphic contour of left hand position, variable finger width, and fingerboard location. Hand positions, fingerings, and pitches become less precise and more gestural as a consequence of this unusual approach to notating the left hand.

[98] Aaron Cassidy, "*The Crutch of Memory* (2004), for indeterminate solo string instrument," http://aaroncassidy.com/music/crutchofmemory.htm (accessed February 12, 2017).

[99] Aaron Cassidy, "Second String Quartet (2009–10)," http://aaroncassidy.com/music/secondquartet.htm (accessed February 12, 2017).

Cognitive Attention Balancing

A composer might employ action-based notation for the purpose of cognitive attention balancing. This constitutes an admission by the composer that each parameter addressed in the notation requires a portion of the performer's finite cognitive function. The more parameters specified in the notation, the higher demand required of the performer's brain.

Due to the limitations of CPN, action-based notation is a potential solution to simplifying performance instructions. One might imagine how cumbersome Juraj Kojs's directions in *Revelations* (2005) to scrape, bounce, and roll a variety of circular toys across resonant plates would be if notated in CPN.[100] The opposite position, that action-based notation requires more attention from a performer, is also plausible. Take, for example, Lachenmann's use of invented clefs. Tablature notation such as the bridge clef or string clef has the potential to ignore or subvert a performer's highly developed skills of reading CPN and playing their instrument. In some regards, very little prior knowledge of notation and performance technique is required or even relevant. Contemporary experiments in tablature intend to question the validity of CPN and traditional performance practice itself; this posits a potentially oppositional relationship between composer and trained performer, which is itself a determinant of the musical result.

It comes as no surprise, then, that through notation some composers purposely create a work of staggering difficulty, overwhelming the performer with a multitude of (sometimes contradictory) tasks. This is often the case in the works of Brian Ferneyhough, Richard Barrett, and others composing so-called complex scores, and is almost inevitable in the "decoupled"

---

[100] Juraj Kojs, "Language of Action and Cyberaction," *Journal of New Music Research* 38, no. 3 (2009): 287.

notation of Cassidy and others. The opposite situation of requiring very little specific parameter control from the musician leaves room for performers to interpret, improvise, and interact with other performers. There is evidence of this in the text-based works of John Cage–such as *Empty Words* (1974)–,[101] the group improvisation pieces of Christian Wolff–such as *For 1, 2, or 3 People* (1964)–,[102] and jazz lead sheets. Here the composer relies on the performer's creative abilities to collaboratively complete the music. A wide breadth of creative work lies between the extremes of notational vacuum and parameter overload, with composers often attempting to balance one difficult parameter by making the other remaining parameters correspondingly easier. This is my approach to action-based notation in my work *Terraformation*.

Performative Action in Real-Time Notation

Purposes of Action-Based Notation in Real-Time Notation

Many of the earliest works using RTN are action-based. Gerhard E. Winkler's *Hybrid II (NetWorks)* (1996, rev. 2001),[103] for example, uses several real-time line contours to direct the solo violist's glissandi, bow contact position, and dynamic profile.

---

[101] John Cage, *Empty words: Writings '73–'78* (Middletown, CT: Wesleyan University Press, 1979).

[102] Christian Wolff, *For 1, 2, or 3 People* (New York: C.F. Peters, 1964).

[103] Gerhard E. Winkler, "*Hybrid II (NetWorks)*—or: At the edge of musical self-organization," in *Electronics in New Music*, eds. Claus-Steffen Mahnkopf, Frank Cox, and Wolfram Schurig (Hofheim: Wolke Verlag, 2006), 236–249.

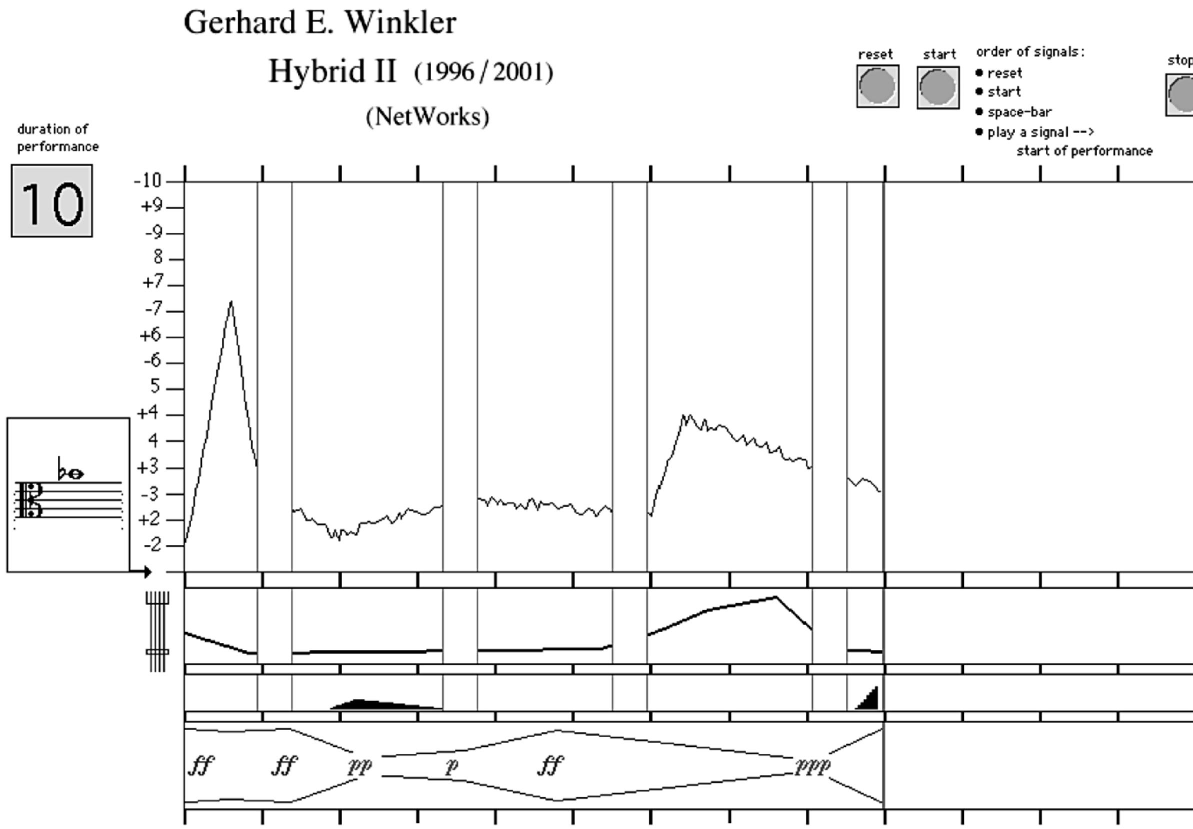

Figure 4.3: Action notation in Gerhard E. Winkler's *Hybrid II (NetWorks)* (1996, rev. 2001).

Likewise, Karlheinz Essl's *Champ d'Action*,[104] uses descriptive on-screen text to direct a group improvisation. The choice to use text and moving line segments was no doubt partially due to computer limitations. However, these early works reveal an attempt to streamline the notational elements in order create compelling music that is efficient to sight-read. As Winkler states, "In general a mixture of *symbolic* (e.g. a "main-pitch") and *graphic elements* (e.g. Glissando-lines) has turned out to be the clearest way of Realtime-notation. It depends on the idea of the piece and the aesthetics of the composer, which elements these will be…. Which *aspects of playing* have to be notated up to which extent of precision (The range goes from *full*

[104] Karlheinz Essl, "*Champ d'Action* (1998), Realtime Composition Environment for Computer-Controlled Ensemble" http://www.essl.at/works/champ.html (accessed February 15, 2017).

realtime-notation, – using all the "in-time" – possibilities of the computer-screen –, to *partly fixed and pre-notated elements*, – e.g. rhythmic patterns, which can be prepared in advance –, up to *fully notated score-fragments*)."[105] These first RTN works demonstrated efficient notation methods and prefigured a fascination with directing performer action in real-time.

Composers currently writing RTN pieces continue to use the techniques established by Winkler, Essl, and others. The radial scores of David Kim-Boyle[106] and Ryan Ross Smith,[107] for example, which display a clock hand-like play head sweeping over attack points situated on a clock face, the intersection of two graphic elements is an immediately clear paradigm for complex rhythmic actions.

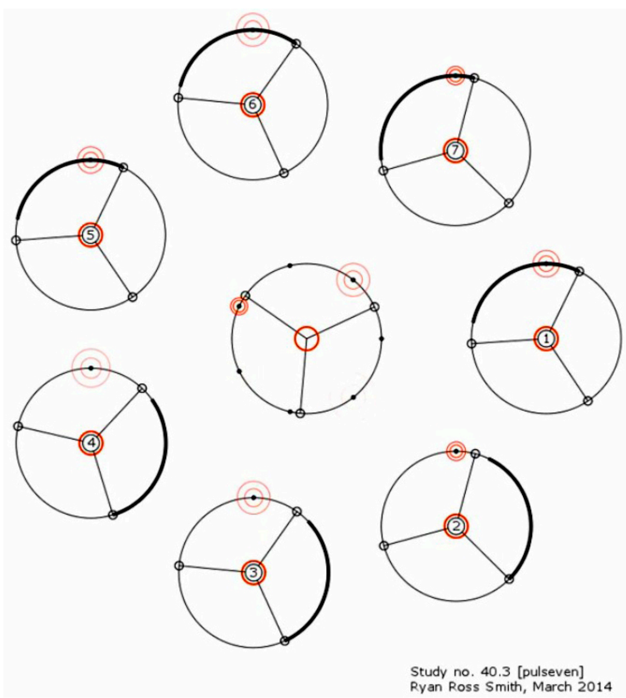

Figure 4.4: Radial notation in Ryan Ross Smith's *Study no. 40.3* (2014).

[105] Gerhard E. Winkler, "The Realtime-Score: A Missing-Link in Computer-Music Performance." *Proceedings of the 2004 Sound and Music Computer Conference* (2004): 3.

[106] David Kim-Boyle, "*Point Studies No. 4* (2013, rev. 2016), for voice and electronics," http://www.davidkimboyle.net/point-studies-no-4-20141.html (accessed February 15, 2017).

[107] Ryan Ross Smith, "*Study no. 40.3 [pulseven]* (2014), for seven instrumentalists," http://ryanrosssmith.com/study40_3.html (accessed February 15, 2017).

In their simplest form, these radial scores tell the musician when to perform an action. When duplicated to direct large ensembles, the radial score efficiently notates dense polyrhythmic textures.

When musical parameters are decoupled through an efficient graphical language, the performer is freed to focus their attention on the most musically challenging elements on a momentary basis. As described above, decoupling performative actions has the potential for revealing new modes of sound production. One drawback is that it also has the potential for increasing strain on the performer. Finding the equilibrium between these two objectives in RTN is a delicate task.

Performer Action Modeling in *Terraformation*

Terraformation for viola and computer uses action-based notation to increase efficiency in sight-reading, to enable an interactive formal structure, and to reveal new modes of sound production. The performative actions required in the piece are based on a study of physical and psychological mechanisms at work in the musician’s physical contact with the instrument. The resulting notation is carefully designed to ease the cognitive translation from graphic representation to bodily action.

The notation used in *Terraformation* resulted from an active collaboration with violist Michael Capone. His experiences and reactions in reading early versions of the work helped determine the present state of the piece. In particular, Capone helped me rank the difficulty of left hand positions and balance the weighting applied to the algorithm when moving the left hand from one position to another. He also narrated his sight-reading thought process as he correlated the different forms of notation used in *Terraformation*, relating when certain notations were beneficial and when they were extraneous. His guidance regarding an open-ended parameterization of the physical actions required to play the viola helped determine the parameters I chose to address in the work.

Overview of the Notation Used in *Terraformation*

There are three distinct forms of notation in *Terraformation*. One type of notation is a five-line staff with standard clefs capable of showing common music notation symbols. Elements of this staff can be hidden so that one of three different modes can be displayed at any given time: specific pitches and rhythms using standard symbols, specific pitches with proportionally spaced rhythms, or approximate pitches (displayed as stems without note heads) with specific rhythms.

The second type of notation is a tablature depiction of the viola's fingerboard. Instead of fret-like gradations of position, just the one-, two-, and three-octave positions and the approximate end of the fingerboard are marked. Each of the musician's fingers is notated on the fingerboard as a color-coded encircled number. An open or unplayed string is shown as a grayed out zero at the far left-hand side of the diagram. In addition, the lowest string with a finger down is marked with the letter name of the specific pitch for quick reference.

The third type of notation consists of two sets of color gradients. The first stretches across the horizontal width of the five-line staff and is used to indicate bow contact position. The specific position at any given moment corresponds to the color sharing the same vertical alignment as the current rhythm on the five-line staff. The color blue indicates *molto sul tasto*, green is *normale*, red is *molto sul ponticello*, and yellow is behind the bridge. Any gradient between those colors represents a bow contact point between the endpoints of that continuum. The second color gradient is applied to each of the note heads on the five-line staff. Ranging from black to light green, these indicate a continuum between normal left hand finger pressure to light finger pressure (as light as possible indicates slightly lighter than harmonic finger pressure).

These three types of notation comprise an aggregate notational system, although two of the three types are subject to display at any given moment:

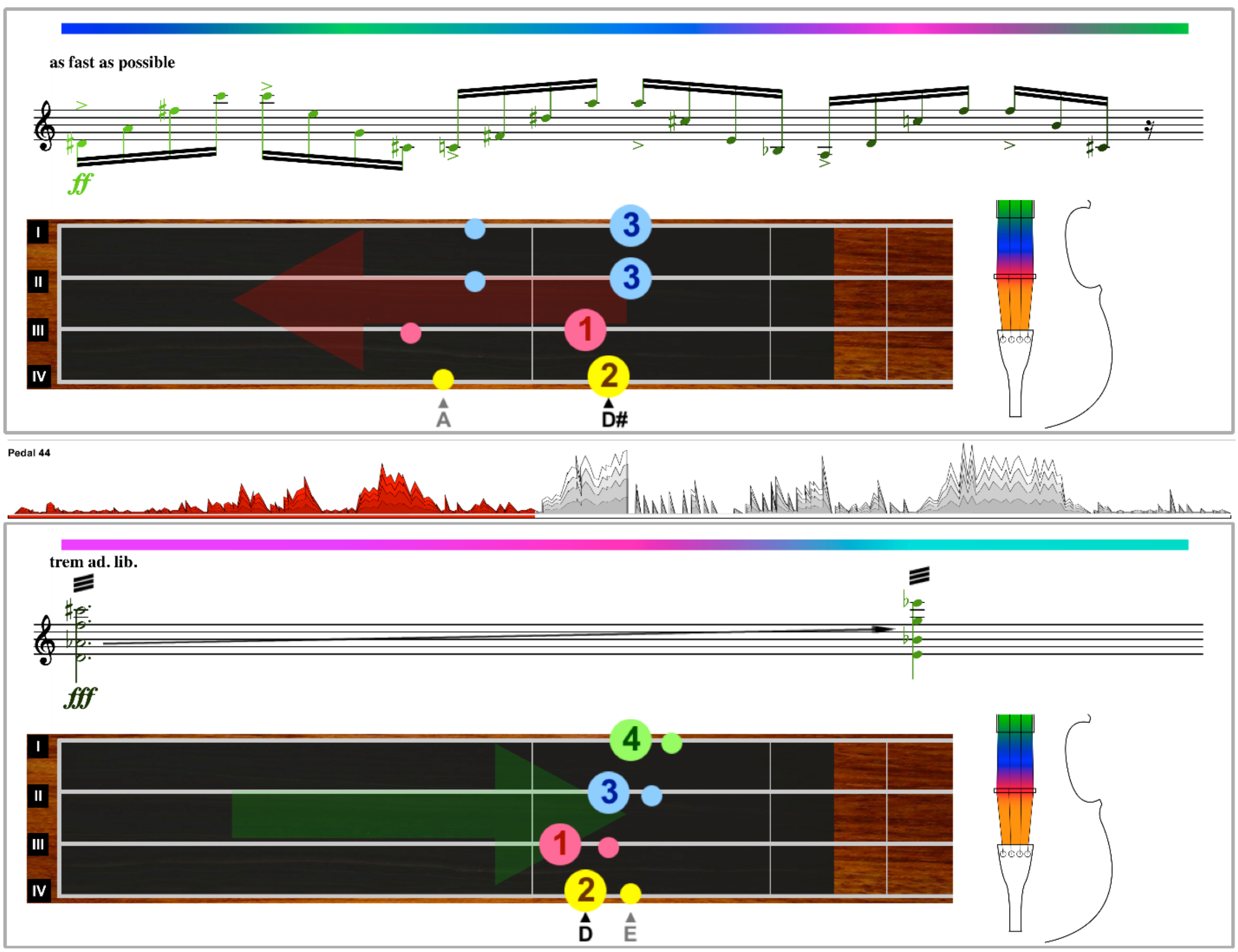

Figure 4.5: The aggregate notation and performance interface for *Terraformation*.

While the five-line staff system remains on-screen throughout the piece, the fingerboard and color gradients can be independently hidden when not required. Additionally, two aggregate systems of notation occupy the performer's screen-based score. The top aggregate system shows the notation for the current musical activity and the bottom system shows the subsequent material. Between the two aggregate systems is a graphic indicating the performer's current location in the form.

Fingering Positions on the Fingerboard

The algorithm driving musical material in *Terraformation* is built on a series of constraints that model the physical action required to produce a quadruple stop on the viola, referred to hereafter simply as a "chord." The general sequence of chord creation is illustrated below:

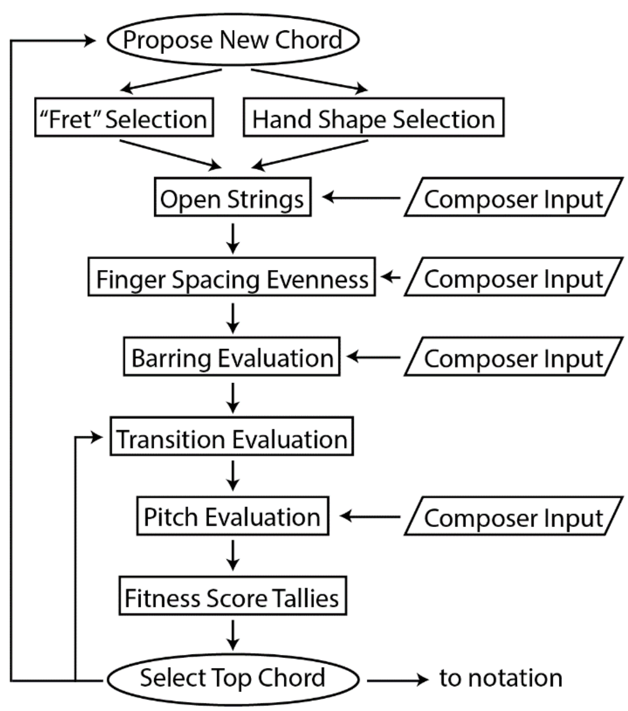

Figure 4.6: A block diagram of the algorithm for generating new chords in *Terraformation*.

This sequence of operations iterates a number of times to generate a pool of potential chord candidates. At the end of the process, the algorithm proposes the best possible choice to follow the current chord based on inputs governing the model. The action-based logic behind each of these subroutines is explained below.

"Fret" Selection and Maximum Finger Stretch

The term "fret" is used here as a method of conveniently locating the finger on the fingerboard and also as a way to avoid more conventional position-based string pedagogical practice. The model first randomly selects a fret and assigns it to the lowest-fretted finger:[108]

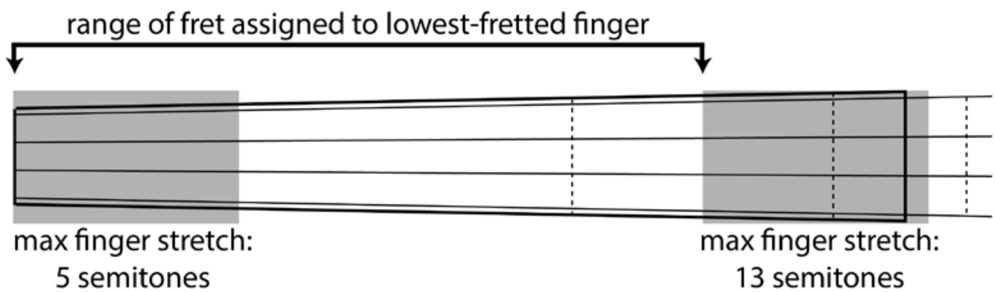

Figure 4.7: The range of possible "fret" positions and maximum finger stretch in those positions.

The possible range for the lowest-fretted finger is bounded on one end by the open strings at fret-0 and at the other end by fret-18. Based on the selected fret, the algorithm randomly chooses the distance between the lowest- and highest-fretted fingers. At the lowest end of the fingerboard, the maximum stretch between the lowest- and highest-fretted finger is

[108] For all practical purposes, the lowest-fretted finger in a quadruple stop is always the first finger. Similarly, the highest-fretted finger is always the fourth finger. The exact fingers are not specified in the algorithm so as to allow for non-quadruple stopped possibilities where an open string or rest on the first or fourth fingers changes which finger is playing the lowest fret.

five semitones. This stretch increases to thirteen semitones at fret-18, extending approximately to the end of the fingerboard.[109]

Hand Shape Selection

A parallel process chooses a hand shape from a predetermined set of twenty-one options ranked by difficulty. A total of twenty-four (4! = 24) hand positions are theoretically available, but three are physically impossible.

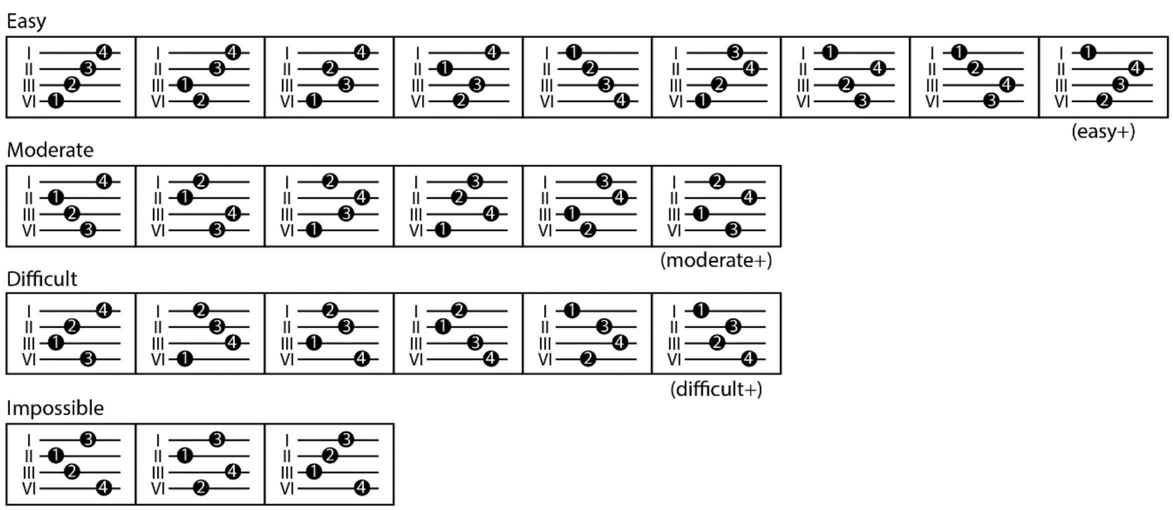

Figure 4.8: All twenty-four unbarred left hand positions on the viola ranked by difficulty.

The hand positions are selected based on a weighted transition table that prefers easier hand positions. Once chosen, the hand position is coupled with the fret selection and finger stretch parameters described above to generate a specific finger and fret combination for the lowest- and highest-fingered frets. The two central fingers' exact positions remain to be determined. In addition, a corresponding penalty is applied to the chord's overall fitness score

109 Paul Rolland, *The Teaching of Action in String Playing: Developmental and Remedial Techniques* (Urbana, IL: Illinois String Research Associates, 1974).

based on the difficulty of the hand position. This score is tallied and ranked at the very end of the process.

Open Strings

Before fixing the exact fret positions of all of the fingers, the algorithm randomly allows for the selection of open strings. Open strings are applied to both finger and string, ignoring the fret parameter in the subsequent routines related to finger spacing. An input value governs the percentage chance of open strings at each chord request.

Evenness of Finger Spacing

The remaining two central fingers' positions are determined in connection with an input value that corresponds to finger spacing evenness. At low-fretted positions on the fingerboard, little variation is possible for the central fingers due to the limitations of the maximum finger stretch. At higher-fretted positions, a higher concentration of pitch possibilities in condensed physical space yields more options. Two factors govern the evenness of the finger spacing. The first is a decreasing amount of flexibility between adjacent pairs of fingers from the first-second pair to the third-fourth pair. In other words, the variation in finger spacing is most flexible between the first and second fingers and least flexible between the third and fourth fingers. The second factor is that asking the player to stretch the space between one pair of fingers inhibits stretching in other fingers.

On the whole, even spacing of the two central fingers between the outer fingers is the most comfortable and therefore the more playable solution. Increasing the uneven input value randomly deviates away from even spacing using the two-factor model of finger spacing just described.

Barring Evaluation

At this point, the algorithm has generated a complete chord with specific finger and fret locations. Many chords are still highly impractical from a physical perspective or undesirable from a musical perspective. Several evaluation processes examine the fitness of the chord and assign it a score that when tallied rates its viability.

The first evaluation looks for chords with two fingers on the same fret, commonly referred to as barring. An input value controls whether or not two adjacent strings can be barred. Chords with three or more fingers on the same fret or with two non-adjacent strings on the same fret are immediately rejected.

Chord Transition Evaluation

The second stage of evaluation examines the transition between the current chord and the proposed subsequent chord. The algorithm tracks the movement of each finger from the current chord to the proposed chord and generates a score that considers the following: whether or not a finger changes strings, the direction of the move up or down the fingerboard, and the dexterity of each finger. Moving a finger from one string to another incurs a significant scoring penalty, with changes going from a higher-numbered string to a lower-numbered string being more severe than a lower- to higher-numbered string.[110] The reason for this is that it is more difficult to contract a finger to a new position than to extend a finger. Next, the average fret positions of the current and proposed chord are compared. More distant average fret movement acquires a higher scoring penalty. Finally, each of the scoring mechanisms accounts for differences in finger dexterity by using a finger-specific weighting, with movements in the fourth finger generating higher penalties. This finger-specific weighting reflects an overall ease of movement in the first finger with each subsequent finger diminishing in dexterity.

Pitch Evaluation

The third stage in the evaluation process scores each chord according to a specified pitch-class, pitch-class set, or combination of both. Chords that contain one or more matches are given a higher ranking as more desirable. Each evaluation routine – pitch-class, pitch-class set, or both – can be activated or deactivated. In any given iteration of the algorithm, the pitch-class evaluator finds the most matches and the both evaluator finds the least. By requesting that chords fulfill both pc and pcs requirements, the algorithm will sacrifice ease of chord transition and playability for more desirable pitch content.

Fitness Score Tallies

Following generation and evaluation, a list of proposed chords are finally collected and their corresponding fitness scores tallied. The list is sorted first by chords that fit the requested

[110] To be clear, a lower-numbered string (ie. String I) produces higher pitches than a higher-numbered string (ie. String IV).

pitch requirements. Within that list, chords are arrangement by the difficulty of the chord's physical production. The chord with the top score (ie. the least amount of penalties) is displayed for the musician to perform and is fed back into the chord algorithm for comparison with subsequent chord candidates. In addition, the fitness score follows the chosen chord through the creation of the remaining musical parameters – rhythmic figures, dynamic contour, bow contact position, and left hand finger pressure, to name a few. The difficulty of these parameters is inversely related to the chord's fitness score. So, for example, as the difficulty of the chord increases, the difficulty of the rhythmic figure decreases. In this way, the fitness score mediates the amount of attention that the performer is likely to spend on any single parameter.

Efficiency in Hybrid Notation

The performative action model in *Terraformation* attempts to balance the cognitive demands on the musician by using a hybrid combination of notation types. The aggregate notation display is designed to give the performer instructions that are immediately readable while also providing a depth of detail. Comments from violist Mike Capone following a rehearsal of *Terraformation* revealed the specific sequence of information gathering that he executes each time the display is refreshed. The performer first deduces the hand position from the fingerboard diagram. While he generally replicates the hand position on the instrument he is assessing the position of the lowest-fretted finger. He then finalizes hand position by checking it against the CPN, making small adjustments where necessary. The moment he spends looking at the CPN also gives him an approximate understanding of the rhythmic character of the current staff system. As he begins to perform the material, he is constantly correlating the four-color gradient that represents bow contact position and the two-color gradient that represents left hand finger pressure with the current rhythmic figure, pitch, and dynamic. Finally, in moments of minimum cognitive strain – in rests or during repeating figures, for example – he may look below the current aggregate staff system to the upcoming system in order to read ahead. In this way, through efficiency of a hybrid notation display, the musician is able to link information gleaned from different types of notation into a cohesive, continuous performance.

Conclusion

While incorporating action-based music notation into a work using RTN is not a new endeavor, the methods and benefits of doing so are an incredibly rich area for exploration. The fascination with action-based music notation in the twentieth and twenty-first centuries has yielded a variety of alternate ways of mediating musical performance. Some of the key benefits of this category of notation include clarity of sound production techniques and immediately recognizable instructions that reduce cognitive strain on the performer. These are important factors when asking a musician to sight-read during performance as in the case of RTN. In *Terraformation*, an algorithm modeling the physical actions required to produce sound creates, ranks based on difficulty and pitch content, and notates musical material. Finally, by using several types of notation to instruct the performer – a combination of action-based notation and CPN – the musician is able to efficiently extract and unify the instructions into a cohesive musical gesture.

Chapter 5

Performance Practice of Real-Time Notation

Absence of Performance Protocols for Real-Time Notation

The performance practice issues of real-time notation (RTN) share connections with open form music, indeterminacy, complexity, free improvisation and interactivity. These issues, in combination with larger political and educational problems in the musical world, pose a formidable hurdle for many would-be performers.[111] Over the past decade or so, a growing number of composers have incorporated RTN in their practice despite the inherent difficulties—recent or nascent software, lack of standardization of notational approaches and performance practice, and more. Some have written extensively on the topic of RTN in an effort to detail new software in the field or to explain the technological or theoretical underpinnings of a new work.[112][113] With some notable exceptions, few have presented the problems and newfound freedoms that the performer faces in performing such works. The final portion of Jason Freeman's 2008 article "Extreme Sight-Reading, Mediated Expression and Audience Participation: Real-time music notation in live performance" contains an excellent first attempt at developing a comprehensive guide for the performer.[114] Following his argument that RTN can dynamically connect audiences to live performers, Freeman describes the difficulties

---

[111] Jef Chippewa, "Practicalities of a Socio-Musical Utopia," originally published in *Nutida Musik* 3–4 (2012–2013), http://newmusicnotation.com/chippewa/texts/chippewa_spahlinger_2013-06.pdf (accessed January 10, 2017).

[112] Christopher McClelland and Michael Alcorn, "Exploring New Composer/Performer Interactions Using Real-Time Notation," *Proceedings of the 2008 International Computer Music Conference* (2008): 1–4.

[113] Sang Won Lee and Jason Freeman, "Real-Time Music Notation in Mixed Laptop-Acoustic Ensembles," *Computer Music Journal* 37, no. 4 (Winter 2013): 24–36.

[114] Jason Freeman, "Extreme Sight-Reading, Mediated Expression and Audience Participation: Real-time music notation in live performance," *Computer Music Journal* 32, no. 3 (2008): 25-41.

involved in designing a score suitable for the sight-reading musician and the rehearsal time needed to familiarize oneself with the types of notation a piece is likely to produce. However, he does not go into detail when describing performer psychology in both the rehearsal and performance experience. In addition, his discussion of RTN is focused on synchronized ensemble improvisation and audience participation. Freeman does not address RTN scores that employ common practice notation (CPN) symbols.

Composers of RTN works tend to include small reports of performance practice in their research, often mentioned as an ancillary issue. Such remarks are sometimes quite vague and may perhaps be too general for an effective description of a non-standard and emerging performance practice: "The best way to approach the playing of a Real-Time-Score seems to be that of a relaxed, playful 'testing' of the system."[115] While benevolent and potentially helpful to some performers, anecdotes and superficial suggestions ignore serious barriers for performers approaching RTN. The trust required between a composer, performer and a work that exhibits notational agency is not something that should be taken lightly and with a significant and growing body of works that make use of RTN now in existence, we can begin to look more in depth at these and other aspects of the practice.

New Freedoms for Musical Expression

Freedom From Replication

The composer or performer considering whether or not to engage in the practice of RTN

---

[115] Gerhard E. Winkler, "The Real-Time-Score: Nucleus and Fluid Opus," *Contemporary Music Review* 29, no. 1 (2010): 99.

might understandably wonder in what ways the added challenges can ultimately benefit a composition. RTN affords both composer and performer with new freedoms in live performance and new means for musical expression.

One freedom is the rejection of replicating the performance of a work. Since the advent of the phonograph, recorded performances have imparted an increasingly burdensome tradition on the shoulders of each generation of performers. This problem is not isolated to the hallowed ranks of common practice music. Recordings of contemporary compositions, particularly composer-endorsed recordings and recordings by esteemed new music performers, become authoritative in a way that was perhaps unintended. Such documents become a type of *Urtext* (an *Urklang* perhaps) and an immediate arbiter of what is an "authentic" performance of a piece.

Bruce Haynes's remarks about authenticity and values in common practice music are increasingly applicable to new music:

> The shortlist of "Masterpieces" that it plays over and over, repeatability and ritualized performance, active discouragement of improvisation, genius-personality and the pedestal mentality, the egotistical sublime, music as transcendent revelation, Absolute Tonkunst…ceremonial concert behavior, and pedagogical lineage.[116]

Those ideals are in contrast to those that Haynes asserts ruled musical events before the nineteenth century:

> That pieces were recently composed and for contemporary events, that they were unlikely to be heard again (or if they were, not in quite the same way), that surface details were left to performers, that composers were performers and valued as craftsmen rather than celebrities…and that audiences behaved in a relaxed and natural way.[117]

[116] Bruce Haynes, *The End of Early Music* (New York: Oxford University Press, 2007), 223.

[117] Ibid.

Perhaps a concerted effort should be made to return to these ideals in some part today. Among other things, these ideals reveal the lack of contemporary expectations on the performer to be a necessary creative agent in the music making process. Paul Thom affirms this line of thinking when he says, “An ideology of replication leaves no room for interpretation; and yet interpretation is a necessity…in performance.”[118] Works using RTN offer freedom from the shackles of so-called “authenticity” and the burden of being measured against recordings. It does this through works that defy singular documentation due to the mutable nature of the notation from one performance to the next. Therefore, the performer is freed of the burden of replicating something that cannot be fully documented, authorized, or deemed “authentic,” and can instead be charged with the task of engaging their own creative energies to play music.

Improvisational Freedom

With the performer reinstated as a valued musical interpreter and the weight of the obligation of replication lifted, works that use RTN also grant a degree of creative license to the performer directly through improvisation. This can manifest itself as anything from works with designated moments of aleatory or choices left to the performer to works using abstract graphic notation to guide the performer through improvisation, or that provide a contextual framework with varying degrees of interpretational freedom left to the performer. Karlheinz

[118] Paul Thom, “Authentic Performance Practice,” in *The Routledge Companion to Philosophy and Music*, ed. Theodore Gracyk and Andrew Kania, 91–100 (New York: Routledge, 2011), 97.

Essl's *Champ d'Action* (1998),[119] for example, uses a combination of on-screen text and graphic symbols to elicit group improvisation:

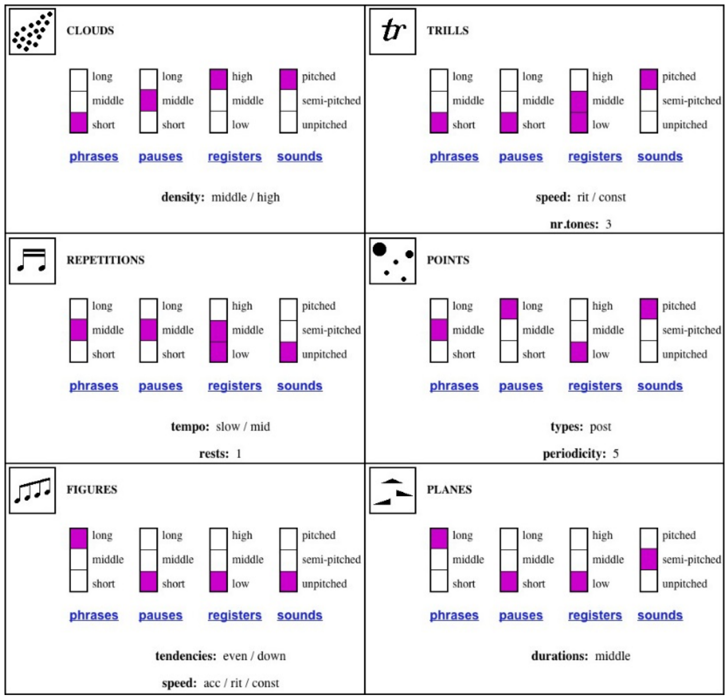

Figure 5.1: Computer-generated instructions in Karlheinz Essl's *Champ d'Action*.

Written for an unspecified ensemble of between 3 to 7 soloists, the musicians respond to live-generated instructions that couple commonly understood parameters (within Western music performance practice, at least) like phrase length, pause length, pitch register, and noisiness with descriptive classes like "clouds," "points," and "planes." These textual instructions must first be translated to the performer's specific instrument before attempting the loftier goal, "to create relationships by listening and reacting to the sounds that are produced by the other players which could lead to dramatic and extremely intense situations."

[119] Karlheinz Essl, "*Champ d'Action,* Realtime Composition Environment for Computer-Controlled Ensemble," http://www.essl.at/works/champ.html (accessed February 15, 2017).

Essl describes the piece as a, "real-time composition environment for computer-controlled ensemble,"[120] indicating the open-form nature of the work and his relinquished compositional agency to computer spontaneity and performer creativity.

Interactive Freedom

The interaction between score and performer is one of the most apparent benefits of RTN. Some works allow the performer to assume direct control over the content of their own notation or the notation of another performer. This is the case in Jason Freeman's *SGLC* (2011) for computers and acoustic instruments, in which an ensemble of laptop performers choose and modify pre-composed musical fragments for an instrumental ensemble to perform in real-time.[121] While Freeman urges each performer to familiarize themselves with the pre-composed material, he gives complete freedom to the laptop "re-composers" to create loops, add or subtract notes, change dynamics, transpose, or otherwise alter the notation. In *SGLC*, the relationship between the two ensembles might appear adversarial as the instrumentalists are at the notational mercy of the laptop performers. Freeman attempts to make this hierarchy more egalitarian by encouraging pairs of laptop and instrumental performers to rehearse as a duet, becoming familiar with each other's behaviors and abilities, before rehearsing as an ensemble: "This unusual setup encourages all of the musicians to share their musical ideas with each other, developing an improvisational conversation over time."[122]

---

[120] Ibid.

[121] Jason Freeman, "*SGLC* (2011) for Laptop and Instrumental Ensemble," http://distributedmusic.gatech.edu/jason/music/sglc-2011-for-laptop-and/ (accessed February 10, 2017).

[122] Ibid.

Freeman's approach to notational improvisation is representative of new interactions made possible in RTN. One might call this type of interaction *permutative interaction* because pre-composed segments are reordered through some mediated process. Such practices evoke a range of new categories of interaction between the performer and their notation. These include, but are not limited to, *formal interaction*, where the performer can influence aspects of the large-scale structure of a piece; *temporal interaction*, where rhythmic augmentation, diminution, or tempo modulation can change dynamically; and *local interaction*, where surface details of a piece like pitches, rhythms, dynamics, articulations, and other expressive elements become dependent on performer input. In presenting models of interactive performance in computer music, Todd Winkler acknowledges the limitations inherent in the undertaking: "No traditional model is either complete or entirely appropriate for the computer…. Simulation of real-world models is only a stepping stone for original designs idiomatic to the digital medium."[123] Perhaps these new types of interaction using RTN move closer toward Winkler's goal of original, idiomatic designs for interactive computer music.

Ephemerality and Multiplicity

In the age of affordable multi-track audio recorders and digital video cameras, and social media hosting platforms like YouTube and SoundCloud, artists are building (and indeed *expected* to build) more and more personal online portfolios offering glossy documentation of their work. This manner of documentation often comes across as "litmus tests" of the individual

---

[123] Todd Winkler, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, MA: MIT Press, 2001), 23.

works' existence, against which future performances should be judged. In addition, the societal pressures to package, brand, and sell a finished artwork potentially choke out the ephemeral moment of actual music making, as quite often a significant amount of the time is required to create and produce the documentation—in some cases it can take longer than the creative act itself! Works using RTN can address and exploit the beauty of impermanence that is so often absent in such approaches to creation and offer a solution to this philosophical and moral problem in the form of multiplicity: each performance presents only one possible version of a work that exists in plurality. To witness one performance is to know only part of a much larger whole. From the performer's standpoint, each performance is unique, free from any historical burden and any comparative critique in the future. The music exists only as it is performed, in that particular moment. Any documentation of an individual performance will inherently fail to fully represent the work.

Problems in Rehearsal and Performance

Traditional Purposes of Practice and Rehearsal

With new freedoms for interaction and improvisation, and without concerns about replication in light of the ephemerality and multiplicity of works performed using RTN, come the practical issues that face musicians in rehearsal and performance. Before exploring new approaches on the subject, the purposes of practice and rehearsal in fixed CPN works should be explored. The most self-evident purpose of individual practice is to learn the details of a piece that are needed for its performance. Some performers describe their practice trajectory as a process in which they first translate notational language into physical gestures, and then gradually link larger and larger musical units together, culminating in a large-scale coherent interpretation.[124] Other performers may follow the opposite path, beginning from a theoretical understanding of the entire work and moving towards mastering the details of each moment. In either case, what is necessary is an understanding of the specific and the general, the micro and the macro.

Except in the case of solo works, the rehearsal process involves interaction with other players. Each individual's micro-macro knowledge of the piece—gained during their own private learning process and practice—contributes to the development of a cohesive understanding of ensemble interaction that is relevant to that particular musical work or context. Rehearsal with a computer-mediated audio component like sample playback or interactive digital signal processing adds further complication. Often in the case of works exploring computer interactivity, an important part of the available rehearsal time is spent navigating the technological prosthetics involved. These include microphones, loudspeakers, pedals, sensors, and other devices manipulated, directly or indirectly, by the performer. And then of course further time is spent—individually and as a group—in problem solving, in order to ensure all the devices are compatible and functioning properly. Rehearsal time is also needed to determine the temporal modalities employed in the piece. This crucial aspect defines how time is controlled and by whom. A fixed temporal modality means that the performer should ensure that their performance conforms to the computer's model of time, whereas a fluid modality gives the computer and performer independent command. Some pieces use a

---

[124] Elizabeth McNutt, "A Postscript on Process," *Music Theory Online* 11, no. 1 (2005).

kind of interactive accompaniment where the computer attempts to conform to the performer's model of time. Finally, rehearsal time is spent discovering and exploring the behaviors of the computer agent. This can take many forms including traditional score following, coordinated live-input processing, and active human-computer joint improvisation to name a few.[125] The challenges to the performer's ability to negotiate an interactive computer part in rehearsal become even more complex when the work involves RTN.

New Purposes of and Approaches to Rehearsal with Real-Time Notation

Many of the traditional purposes for practice and rehearsal become irrelevant with works using RTN. One of the primary problems for newcomers and veterans alike is the unfamiliar process of rehearsing a piece with a dynamic score. One might ask, "Why rehearse when the notation changes in the moment of performance?" The following is an attempt to answer that question.

Behavior

While in RTN works the performers may still be expected to incorporate more "traditional" aspects of performance practice (translating symbolic notation, surface details and annotations into an informed interpretation of the whole), they must now also *engage* with the score, paying attention to behaviors that may be quite different or altogether absent from performance contexts they are more familiar with: for example, performing solely on acoustic

[125] Elizabeth McNutt, "Performing Electroacoustic Music: A Wider View of Interactivity," *Organised Sound* 8, no. 3 (2003): 297–304.

instruments. In a similar way to rehearsing with an interactive computer part, where the performer must investigate the behaviors of the computer agent, a work using RTN might behave in a number of ways. The score might, for example, have specific responses to performer input. Alternatively, a tempo-based clock or real-time clock might drive an event scheduler that in turn generates notation. Temporal and behavioral correlations may take a myriad of forms, but they can be studied in two broad ways: with an eye for general local detail and with an eye for general large-scale form. The local detail can be as simple as discovering a set of pre-composed fragments, or it can be as complex as deducing the frequency of rhythmic figures, probability of pitches, or variety of graphic indications.

As described earlier, my quartet for viola, bass clarinet, marimba and computer, *Law of Fives* (2015), uses a limited number of predetermined pitches selected by a probability table to be merged with rhythms designed in a permutation algorithm.[126] In this piece, the pitches are predictable while their order and associated rhythms are variable. In addition, the probability tables and rhythmic algorithm are modified in real-time by control signals coming from microphones near the performers. The amplitude envelope of the viola, for example, influences both the probability of rests in the marimba part and a random permutation factor in the bass clarinet's rhythms.

Each acoustic instrument influences the other players' notation simultaneously as well as the parameters driving a synthesizer played by the computer. Since local details depend on

[126] Seth Shafer, "*Law of Fives* (2015), for viola, bass clarinet, marimba, and computer," http://sethshafer.com/law_of_fives.html (accessed February 12, 2017).

multiple performers' input, the rehearsal process involves understanding how the ensemble's collective performance affects the each individual's notational output.

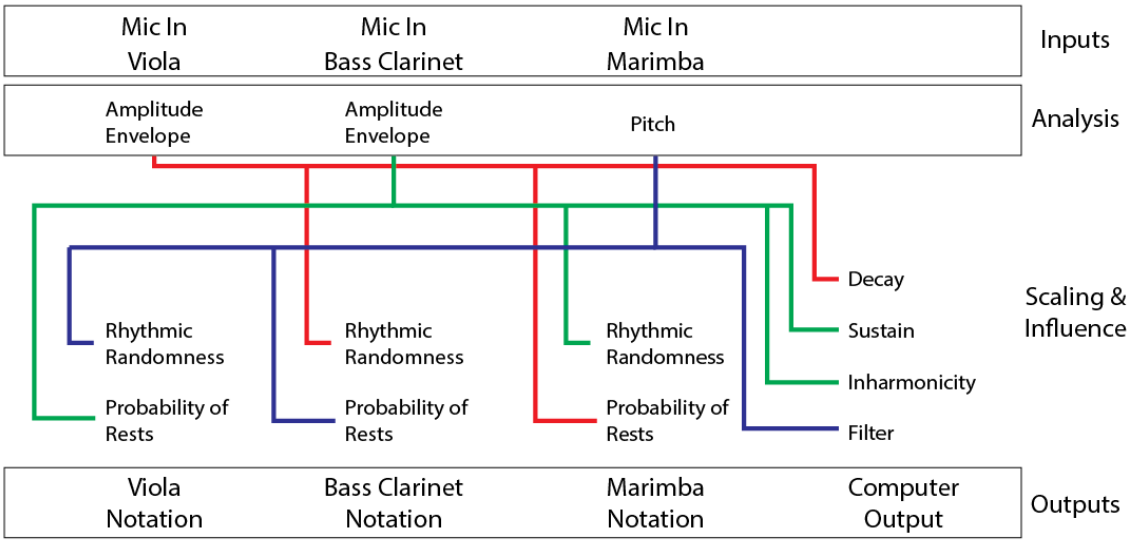

Figure 5.2: Notational variability from live performer influence in Seth Shafer's *Law of Fives* (2015).

If local detail defies the predictability just described, the performer might benefit from studying the large-scale form. Rehearsal should afford the performer an opportunity to play the piece multiple times in order to gain a sense of pre-planned or emergent forms. One possibility is that the notational behavior changes significantly at certain time-points. This is a strategy employed in *Polytera II*, where large-scale changes in tempo, texture, orchestration, and tessitura are predictable throughout the course of the piece and are repeatable from one performance to the next.[127] In order to help the musicians follow these large-scale changes, I developed a system of graphic icons to be displayed in the score and explained with an accompanying document that details the formal plan:

[127] Seth Shafer, "*Polytera II* (2016), for flute, piano, and computer," http://sethshafer.com/polytera_2.html (accessed February 12, 2017).

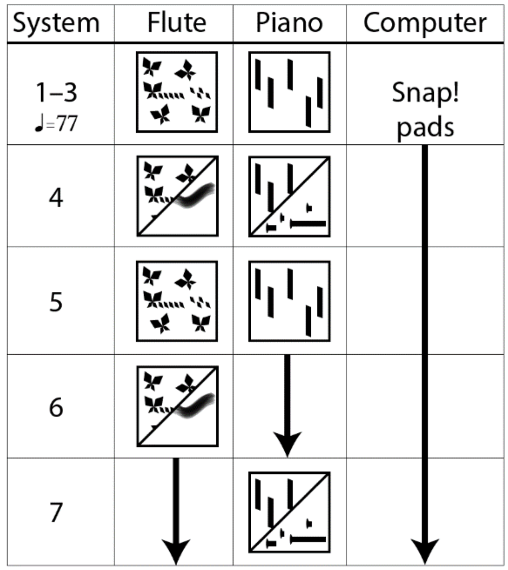

Figure 5.3: An excerpt of the formal plan for *Polytera II* showing a system of graphic icons that represent large-scale changes in tempo, texture, orchestration, and tessitura.

These embedded graphics along with the formal plan help the musicians quickly internalize the modulation of musical parameters as the form unfolds.

In a similar way, the large-scale form of *Terraformation* is displayed on-screen for the performer as an abstract graphic.[128] A play head corresponding to the current MIDI foot switch location is integrated into the graphic:

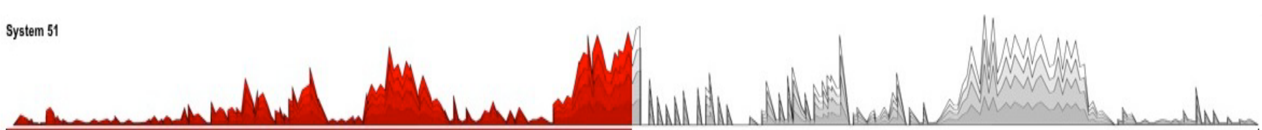

Figure 5.4: A graphic depiction of the form displayed on-screen in *Terraformation*.

The height of the graphic corresponds to rhythmic activity and dynamic intensity. In this way, the performer is able to track their progress through the piece and anticipate upcoming moments in the form.

[128] Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

In other works, one might find that a certain behavior *x* always follows a different behavior *y*, or some more sophisticated arrangement. Another attribute that one can study is the general difficulty level and the modulation of that difficulty throughout the work. Perhaps the piece begins simply and moves in a trajectory toward complexity in a process of accumulation, or moves conversely from complexity toward simplicity in a process of iltering.

Some behaviors lie outside of either composer or performer control. A work like Nick Didkovsky's *Zero Waste* (2001) involves a sight-reading pianist whose performance is recorded by a computer and re-notated for the performer to play again.[129] The piece creates a performer-computer-notation feedback loop that highlights inaccuracies in human performance and errors in the computer analysis of the performance, not to mention inadequacies in symbolic notation. Didkovsky describes the outcomes of this process:

> *Zero Waste* amplifies the resonances of a system which is characterized by the limitations of human performance and CPN music notation. If the performer were perfect, and if music transcription and notation were both theoretically and practically perfect, then *Zero Waste* would consist of identical repetitions of the first two measures. Of course, no sight reader is perfect, and notation must strike a balance between readability and absolute accuracy, so each new pair of measures diverges and evolves a bit more from the last.[130]

The notation mechanism in *Zero Waste* creates the possibility for a chaotically deterministic form in which certain long-term behaviors can emerge. One behavior is an increased number of rests near the beginning of each system due to the small size of the analysis window and performer hesitation. Another is the oscillating accumulation and

[129] Nick Didkovsky, "Recent Compositions and Performance Instruments Realized in Java Music Specification Language," *Proceedings of the 2004 International Computer Music Conference* (2004): 1–2.

[130] Ibid., 1.

dissolution of chord clusters regulated by rhythmic quantization error, at one limit, and the difficulty of playing such chords, at the other.

Audience Interaction

Another situation that evades composer and performer control is that of audience participation. Works like Kevin Baird's *No Clergy* (2005)[131] and Jason Freeman's *Graph Theory* (2005)[132] crowd-source certain compositional decisions, making the rehearsal of such works difficult. In this case, simulating audience feedback in rehearsal, while far from optimal, can at least help clarify which parameters might be able to be anticipated and which are more likely to be subject to chance. Whatever strategy the composer employs, a major purpose of rehearsal is deducing notational behavior.

Sight-Reading

A common experience in the performance of RTN is that some amount of sight-reading is necessary. Rehearsal provides the performer time to practice sight-reading the notational output from the system. Even performers confident in their abilities can balk at the prospect of sight-reading live in front of an audience. Therefore, substantial time must be dedicated to this task—perhaps beyond that which is needed for "traditional" performance from fixed-notation scores—to aid in both the behavioral analysis described previously and developing quick music

[131] Kevin Baird, "Real-Time Generation of Music Notation Via Audience Interaction Using Python and GNU Lilypond," *Proceedings of the 2005 Conference on New Interfaces in Musical Expression* (2005): 240–41.

[132] Jason Freeman, "Extreme Sight-Reading, Mediated Expression, and Audience Participation: Real-Time Music Notation in Live Performance," *Computer Music Journal* 32, no. 3 (2008): 25–41.

reading skills. Performers must keep in mind that what occurs in rehearsal is just as likely to occur in performance. If a particular rehearsal of a piece yields a moment of markedly more difficulty notational behavior, that troublesome behavior can be expected to potentially emerge in performance. Paraphrasing Murphy's Law,[133] whatever *can* be difficult in rehearsal, *will* be difficult in performance.

Triage

Another important consideration in rehearsal (and in the eventual performance) is the strong likelihood that the performer's interpretation will vary—expectedly or not—to some degree from what is actually notated in the score. Some pieces, particularly those with graphic elements, may require a great deal of improvisation from the performer who must therefore be well prepared not only for unexpected notation to appear in the moment of the performance but also to make spontaneous interpretive decisions in a constantly changing notational or performance context. Another piece might use traditional musical symbols that leave little to no room for improvisation from the performer. Whatever the case, the performer should be prepared for situations where, whether as a direct result of the notational design or the pressures and human limits of fast music reading, some deviation from the score and preparations made in rehearsal will be necessary. Flutist Elizabeth McNutt refers to this "carefully practiced reaction to a potentially overwhelming set of concerns" as *triage*, invoking the strict hierarchy of degrees of urgency used to assign and schedule emergency medical

[133] Robert A. J. Matthews, "Tumbling Toast, Murphy's Law and the Fundamental Constants," *European Journal of Physics* 16 (1995): 172–76.

care.[134] In the heat of performance, "mistakes" will occur (again, this may be by design) and the musician must triage the various problems presented in order to assess which musical elements take priority. While this might result in a degree of creative improvisation, it could also involve listening instead of playing, skipping ahead several beats in the music, repeating several beats of music, sacrificing pitch or rhythmic accuracy, or other strategies. Familiarity with the composer's style and the nature of the piece can guide the performer through this learning, performance and decision-making process. If the general effect of the work is of prime importance, triage is clearly preferable to a defeated silence if the performer's sight-reading skills falter. Conversely, in some cases formal connections may need to be sacrificed to meet the demands of local detail. These realities must be faced directly, ideally with composer guidance, so that the performer knows what options are most appropriate when an "ideal" rendering is unavailable.

Graphical User Interface (GUI)

Performers should be encouraged to become as familiar as possible with the on-screen graphical user interface (GUI) of the work. Every piece is different in this respect and the performer must acclimate themselves to the GUI in order to glean every useful bit of information possible. The notational display might follow one of several design considerations. The notation might have a *degree of on-screen movement*. Constant horizontal or vertical motion in the style of the so-called scrolling score is very common. Another option is that the

[134] Elizabeth McNutt, interview by author, Denton, TX, January 10, 2017.

notation slides periodically every beat, bar, system, or pre-determined span of time so that the current time point remains on-screen. On the other hand, the notation might remain stationary, dividing the notation into virtual "pages" that turn or refresh periodically (similar to page turns in print scores). All on-screen notation designs should be concerned about the possibility of knowing what lies just ahead. The method by which a performer can read ahead is highly variable and could be related to the degree of on-screen motion. Similarly, the *time synchronization* and *location tracking system* can behave in a number of ways. The most common methods include a smooth scrolling tracker, a tempo-quantized tracker, or a bouncing-ball type tracker. Familiarizing oneself with these on-screen designs will undoubtedly increase the efficiency and accuracy of score reading.

Practical Considerations

Finally, the performer must be able to read the notation comfortably from their desired playing position, adjusting the music size and screen distance accordingly. A number of solutions for positioning laptops at sitting or standing height are commonly available. External computer display mounts provide an option for installing a screen on an adjustable height stand with a repositionable boom arm. In addition, as the power of mobile processors increase and software options become more proliferate, the display of RTN will increasingly transfer to conveniently mountable mobile devices and tablets. One example of mobile software written specifically for RTN is the *Decibel ScorePlayer*.[135] Remaining practicalities such as who or what

[135] Cat Hope, Aaron Wyatt, and Lindsay Vickery, "The Decibel ScorePlayer: New Developments and Improved Functionality," *Proceedings of the 2015 International Computer Music Conference* (2015): 314–317.

triggers the piece to start, how the piece ends, and whether or not the performer interacts with the screen or software in any unusual ways must also be addressed in rehearsal.

Performer-Composer Trust in Performance

A successful performance of a work using RTN hinges on the trust a performer places in the composer and computer-mediated notation. While there is no formula for building such confidence, the following factors can help create a more optimal situation for the performer and composer.

Many factors that lead to an ideal RTN experience for the performer revolve around the difficulty of the score and the sufficiency of information about the piece provided by the composer. Ideally, the notation should strike a balance between several competing factors: the *general difficulty* of the mechanical instructions like pitches, rhythms, dynamics, and articulations; the *visual layout* of the score including the size of the notation font, the use of non-standard symbols, and whether the performer reads from a part or a score; the *clarity of the timekeeping* mechanism and how tempo modulations are implemented; the amount of *expressive interpretation* desired by the composer; the amount of *improvisation*; and the difficulty of *ensemble coordination*. As the complexity of one parameter increases, the remaining parameters should correspondingly decrease in complexity to let the performer divert maximal effort to the most difficult elements. The performer can be best prepared if the composer provides clear and ample information about hardware and software requirements, the GUI, notational conventions, a formal behavioral outline, sample scores, and/or documentation of past performances.

The composer may purposefully overwhelm the performer if failure is a conceptual component of the work. Failure in performance is a theme explored by many composers in what some have termed the "post-digital" aesthetic.[136] Any performer might understandably be alarmed at such a prospect. The optimal experience for a performer put in that situation is one that does not make them appear foolish. Didkovsky creates a performance situation in *Zero Waste* that requires failure in a way that is uncritical of the pianist. In that piece, a *real* failure would be a performance by a musician who is able to sight-read perfectly—this would effectively undermine the fundamental concept of the work and result in an uninteresting musical experience.

Performer failure is just one of many unusual demands that RTN can potentially require of musicians. Many RTN scores benefit from performers who develop trust in the composer and with the RTN system. RTN works are enhanced by a performer who is willing to risk sight-reading from the stage, who makes mistakes and continues to engage, and who knows that some performance errors are apparent to the audience while others are not. Above all, the successful musician will attempt to transcend the high demands of RTN and ultimately make music.

Comparing Real-Time Notation with the Complex Score

A brief examination of the complex score and the associated musical movement called New Complexity provides some historical and aesthetic perspective on many of the

[136] Kim Cascone, "The Aesthetics of Failure: 'Post-Digital' Tendencies in Contemporary Computer Music," *Computer Music Journal* 24, no. 4 (2000): 12.

performance practice issues related to RTN. To begin with, the complex score shares some striking similarities to RTN. Music by composers such as Brian Ferneyhough, Richard Barrett, and Aaron Cassidy (among others) ask players to perform at the limit of what is possible. This is often accomplished by presenting the performer with conflicting instructions or goals represented in a meticulously detailed, high-density fixed score. The result is a collision of actions with a variable sonic outcome from one performance to the next.

Richard Barrett explains these collisions by saying that the processes underlying his music, "are dramatized in the resulting music, and they're also dramatized in the way that the performer will react to the notation and what it notates."[137] In his solo cello piece *Ne songe plus à fuir* (1985–86),[138] for example, a collision of parameters result in a variety of sonic outcomes when performed "accurately." Barrett illustrates the sonic unpredictability of his notation: "There's a muted section in which you can put your fingers in the right places quite easily, yet it always comes out sounding differently…. There are all kinds of adequate and very different aural realizations of that section, since built into the composition is a certain instability."[139]

Brian Ferneyhough similarly describes the unstable sonic landscape his notation evokes: "In my model, important interference phenomena arise when individual strata come into contact. These chaotic fluctuations are, I suppose, what my music is really 'about.' At the other

---

[137] Richard Barrett, "Richard Barrett in a Conversation with Derek Bermel and Joshua Cody," The Ensemble Sospeso New York (February 1996), Archived by Internet Archive. http://web.archive.org/web/20031004032714/http://www.sospeso.com/contents/articles/barrett_p1.html (accessed February 12, 2017).

[138] Richard Barrett, *Ne songe plus à fuir*, for amplified solo cello (Bury St Edmunds, UK: United Music Publishing, 1985–6).

[139] Richard Barrett, "Richard Barrett in a Conversation with Derek Bermel and Joshua Cody."

extreme, I am fascinated by parallel distortions effected on adjacent strata by some form of external formal force."[140] Here Ferneyhough invokes concepts in *assemblage theory*, developed by philosophers Gilles Deleuze and Félix Guattari,[141] to describe the uncertainty of surface material and the shaping of that uncertainty by large-scale parameter control. The composition process could be described as cartographic; the "external formal force" guides the background unfolding of foreground "chaotic fluctuations."

Like Ferneyhough and Barrett, Aaron Cassidy's music also deals with a collision of materials. He attempts to articulate structural moments in his work by varying "the relative 'force' of the collision[s]."[142] These collisions often occur as a result of his extreme use of tablature notation, which he developed in order to represent and communicate his interest with using *physicality as material*.[143] As described in Chapter 4, Cassidy's work for solo string instrument, *The Crutch of Memory* (2004), is constructed as a choreography of physical gestures performed on the instrument rather than designed through the manipulation of common parameters such as pitch, rhythm, timbre, and so forth.[144] The gestures themselves emerge as the result of correlating several independently notated streams of information: the placement of the left hand on the fingerboard, the width of the fingers, the interaction between the

---

[140] Brian Ferneyhough, "Brian Ferneyhough in a Conversation with Joshua Cody," The Ensemble Sospeso New York (Fall 1996), Archived by Internet Archive, http://web.archive.org/web/20030424094218/http://www.sospeso.com/contents/articles/ferneyhough_p1.html (accessed February 12, 2017).

[141] Gilles Deleuze and Félix Guattari, *A Thousand Plateaus: Capitalism and Schizophrenia*, trans. Brian Massumi (Minneapolis: University of Minnesota Press, 1987).

[142] Aaron Cassidy, "The String Quartet as a Laboratory and Playground for Experimentation and Tradition (or, Opening Out/Closing In)," *Contemporary Music Review* 32, no. 4 (2013): 322.

[143] Ibid., 312.

[144] Aaron Cassidy, "*The Crutch of Memory* (2004), for indeterminate solo string instrument," http://aaroncassidy.com/music/crutchofmemory.htm (accessed February 12, 2017).

fingers and the strings, and the actions of the right hand and bow. By using an action-based notation that decouples performance instructions, Cassidy invents new modes of sound production that are unstable, and whose results may be unreproducible.

My own use of RTN has striking similarities to these examples, presenting the musician with a variety of potentially conflicting goals: to relinquish the security of a fixed score while embracing new performance freedoms; to sight-read in front of an audience while performing musically; to expose the limits of ability while performing confidently. These dichotomies are different from those of the complex score, but their collision similarly produces unpredictable aural outcomes. Due to this inherent variability, the complex score and RTN both celebrate the beauty of ephemerality and multiplicity. Both also present ensemble coordination issues, albeit for different reasons. Finally, both present problems in rehearsal strategies. In some ways, the real-time score is a logical extension of the complex score in which Barrett's concepts of *notation as freedom* and *improvisation as a method of composition* can be fully realized.[145]

Conclusion

As with any emerging practice, the subject of RTN performance practice continues to expand and evolve at a rapid pace. The elucidation of its performance praxis follows immediately behind the creation and performance of new works. The preceding discourse builds upon the foundation already established by Winkler, Didkovsky, Freeman, and many others by offering insight into the new freedoms available to the performer. Similarly, this

[145] Richard Barrett, "Notation as Liberation," *Symposium: Notation in Contemporary Music: Composition, Performance, Improvisation.* Goldsmiths University of London (October 18–20, 2013).

nascent field will also rely upon the ongoing work of many practitioners as RTN system research and development evolves. The liberation of the performer from a fixed score affords more room for expression, improvisation, and the fleeting moment of music making. The solutions presented here for practice and rehearsal stem from long-standing traditions in *musique mixte* practice and rehearsal, the goal of which is clear: the production of new types of music. As Earle Brown remarked:

> Although there have been numerous scores written which have utilized nontraditional notation, there are relatively few in which the notation has played a really functional role in the essential nature of the musical conception of the work. By "real functional role," I mean that the piece could *not* be notation traditionally and that the sound of the work is of an essentially different character because of the new notation. The "decorative" value of a score is in itself a pleasure buy I am more concerned with the possibilities of a notation system that will produce an aural world which defies traditional notation and analysis and creates a performance "reality" which has not existed before.[146]

The performer is an utterly essential component to bringing RTN works to life. It is crucially important to address this collaborative relationship in the design of the work. This is a central feature of *Terraformation*, based both on past projects and on active collaboration with violist Michael Capone, whose experiences and reaction in reading early versions of the work became fundamental composition determinants.

[146] Earle Brown, "The Notation and Performance of New Music," *The Musical Quarterly* 72, no. 2 (1968): 180-1.

Chapter 6

Future Directions

Introduction

The contextual framework for understanding a work that uses real-time notation (RTN) spans the subjects of on-screen notation, algorithmic art, music notation and representation, and the performance practice of notational interactivity. My compositions *Law of Fives* (2015),[147] *Polytera II* (2016),[148] and *Terraformation* (2016–17)[149] confront these issues, each in a different way. These works are unified through their use of the screen as a notational medium and through a compositional approach I have termed cartographic. This approach pre-plans large-scale features while allowing small-scale surface details to be subject to algorithmic variability and/or performer choice.

Terraformation is perhaps the most liberal of these works in terms of algorithmic variability, relying on a computer model of the violist's physical movements in order to suggest a progression of musical material. The notation is displayed as a kind of hybrid instruction, blending common practice notation (CPN) symbols with action-based tablature and abstract graphics. The goal of this notation is to create an efficient sight-reading platform for the musician.

---

147 Seth Shafer, "*Law of Fives* (2015), for viola, bass clarinet, marimba, and computer," http://sethshafer.com/law_of_fives.html (accessed February 12, 2017).

148 Seth Shafer, "*Polytera II* (2016), for flute, piano, and computer," http://sethshafer.com/polytera_2.html (accessed February 12, 2017).

149 Seth Shafer, "*Terraformation* (2016-17), for viola and computer," http://sethshafer.com/terraformation.html (accessed February 12, 2017).

The fact that one must sight-read in the concert situation is one of several unusual demands that RTN makes of the performer. The novel situation of RTN affords opportunities to engage performers as creative partners in the music-making process. Each of my three pieces gives the performer agency in shaping the outcome of the work through additional layers of interactivity. *Law of Fives* and *Polytera II* use networks of control that allow one musician's performance to effect parameters of another musician's notation. *Terraformation* gives the performer choice over multiple notational paths that have branching repercussions over the entire work.

Ephemerality and multiplicity are the primary motivation behind this body of work. The solutions that I have proposed all balance elements of variation with a degree of compositional identity. While presenting many challenges to the traditional processes of composing, preparing, and presenting concert music, these works celebrate the act of live performance, embracing variation and difference as the multifaceted nature of a prismatic whole.

Combining Real-Time Notation with the Complex Score

The future of my creative work can perhaps be located at the intersection of RTN and the complex score. As I briefly described in Chapter 5, the complex score shares many similarities with RTN by presenting the performer with a collision of goals that result in unpredictable sonic outcomes. In RTN, the collision focuses on the variable nature of the notated music and the inherent difficulty of accurate sight-reading. In the complex score, the collision focuses on the juxtaposition of multiple, conflicting musical parameters that keeps perfection just out of the performer's reach.

My work developing an action-based notation for *Terraformation* was primarily concerned with efficiency in sight-reading. I found a solution using a fingerboard tablature that shows hand/finger placement and glissandi movement across the fingerboard. This notational strategy bears some resemblance to Aaron Cassidy's string writing in *The Crutch of Memory* (2004), where musical materials are conceived as instrumental choreography.[150] An area of future interest includes adding additional notational layers to my real-time action-based notation. This would likely require the removal of the five-line staff and other CPN symbols to make space for new notational mechanisms. Where the notational elements in *Terraformation* are primarily used to reinforce each other, perhaps they could be decoupled to allow for the emergence of potentially new modes of sound production.

The procedure used to model the violist's physical actions could also be extended to support an exploration of the real-time complex score. *Terraformation* was conceived for solo viola, but an adaptation for solo violin is quite possible and is already underway. The most critical alterations include the pitches of the instrument's strings and the position of the hand on the fingerboard. Adapting the system to accommodate other string instruments would require further research into the difficulty of each physical action and the specific parameters of the instrument. The system could conceivably be adapted to generate and display hand positions on a cello or tablature for a guitar. Non-string instruments would require a new algorithm and notational display, but the principles behind such an approach appear to be

[150] Aaron Cassidy, "*The Crutch of Memory* (2004), for indeterminate solo string instrument," http://aaroncassidy.com/music/crutchofmemory.htm (accessed February 12, 2017).

possible and musically interesting.[151] In addition, small chamber ensembles using on-screen displays for real-time complex notation could further benefit from visual synchronization aids, such as described in Chapter 2, and interactive networks of notational influence.

The Machine is the Artist–Authorship in Algorithmic Art

The use of algorithms to generate notation appears to be fertile territory for my future work. As such, the questions posed in Chapter 3 regarding authorship in algorithmic art continue to interest me. In my assessment, the construction of the algorithm and its driving control structures sheds light on the nature of the resulting work. By examining the control mechanisms and underlying algorithms one can analyze not only the outcomes already in existence but speculate on what other possibilities lie in wait.

In order to catalog my continued research in the area of authorship in algorithmic art, I have designed a web resource called *the machine is the artist—is the artist the machine*. [152] This interactive, annotated timeline depicts important figures, events, writings, and works of visual, textual, sonic, and performance art in the history of algorithmic art. The website also acts as an ongoing repository for innovations in the field and will document my ongoing investigation into the ways in which machines extend the abilities of human creativity.

[151] See, for example, Aaron Cassidy, "*What then renders these forces visible is a strange smile (or, First Study for Figures at the Base of a Crucifixion)* (2008), for solo trumpet," http://aaroncassidy.com/music/smile.htm (accessed February 12, 2017).

[152] Seth Shafer, "the machine is the artist—is the artist the machine," http://sethshafer.com/themachineistheartist.html (accessed February 12, 2017).

The Future of Notation?

In light of the present study and its implications for further research, I conclude with a question*: what is the future of notation?* In chapter 2, I suggested that ubiquitous use of computers in the composition process—including computer engraving software, MIDI sequencers, digital audio workstations, audio analysis tools, and specialized computer-assisted composition environments—has led composers and performers alike to acclimate to, if not always embrace, the computer display as a notational medium. Performances using fixed on-screen notation with software like VizScore prove the viability, not to mention the many advantages, of using the computer display in this way. At the same time, RTN has yet to gain universal appeal in concert halls all over the world. Is it possible that a growing popularization of fixed on-screen notation intended for performance will lead to a more widespread employment of RTN in composition? Could fixed on-screen notation and RTN shape the future of music notation?

If so, I propose that a multifaceted exploration of the techniques and possibilities afforded by RTN will also transform compositional and collaborative practices. This dissertation represents an exploration of music that challenges our preconceptions and pushes the limits of creation and performance. Just as the proliferation of fixed paper notation was the product of incremental advancements in printing technology throughout the last few centuries, likewise fixed on-screen notation, RTN, cartographic composition and their associated performance practices are outcomes of our current technology. As technology becomes more powerful and accessible, the body of work in this area of music making will continue to expand and differentiate. It is my hope that the research presented here builds solidly upon the foundations already by other artists, and invites many others into the discussion.

Bibliography

Agostini, Andrea and Daniele Ghisi. "Bach: An Environment for Computer-Aided Composition in Max." *Proceedings of the 2012 International Computer Music Conference* (2012): 373–78.

———. "bach: automated composer's helper." http://www.bachproject.net/ (accessed February 10, 2017).

Ariza, Christopher. "An Open Design for Computer-Aided Algorithmic Music Composition: athenaCL." Ph.D., New York University, 2005.

Ascott, Roy and Edward A. Shanken. *Telematic Embrace: Visionary Theories of Art, Technology, and Consciousness*. Berkeley: University of California Press, 2003.

Assayag, Gérald, Camilo Rueda, Mikael Laurson, Carlos Agon, and Olivier Delerue. "Computer-Assisted Composition at IRCAM: From PatchWork to OpenMusic." *Computer Music Journal* 23, no. 3 (1999): 59–72.

Baird, Kevin. "Real-Time Generation of Music Notation Via Audience Interaction Using Python and GNU Lilypond." *Proceedings of the 2005 Conference on New Interfaces in Musical Expression* (2005): 240–41.

Barrett, Richard. *Ne songe plus à fuir*, for amplified solo cello. Bury St Edmunds, UK: United Music Publishing, 1985–6.

———. "Notation as Liberation." *Symposium: Notation in Contemporary Music: Composition, Performance, Improvisation.* Goldsmiths University of London (October 18–20, 2013).

———. "Richard Barrett in a Conversation with Derek Bermel and Joshua Cody," The Ensemble Sospeso New York (February 1996). Archived by Internet Archive. http://web.archive.org/web/20031004032714/http://www.sospeso.com/contents/articles/barrett_p1.html (accessed February 12, 2017).

Beales, Chris. "Explaining Probability using Dice." http://www.chrisbeales.net/environment/consequences_probability%20dice.html (accessed January 12, 2017).

Birchfield, David Andrew. "Evolving Intelligent Musical Materials." D.M.A., Columbia University, 2003.

Boden, Margaret A., and Ernest A. Edmonds. "What is Generative Art?." *Digital Creativity* (March 2009): 1–31.

Brecht, George. "*Drip Music, Fluxversion 1* (1959)." In *The Fluxus Performance Workbook*, edited by Ken Friedman, Owen Smith, and Lauren Sawchyn, 22. Performance Research e-Publication, 2002. https://thestudio.uiowa.edu/fluxus/sites/default/files/FluxusWorkBook.pdf (accessed March 14, 2017).

Brown, Earle. “The Notation and Performance of New Music.” *The Musical Quarterly* 72, no. 2 (1968): 180-201.

Buchla Electronic Musical Instruments. “Model 265/266 Source of Uncertainty.” https://buchla.com/product/266e-source-of-uncertainty/ (accessed February 11, 2017).

Bumgardner, Jim. “The Whitney Music Box.” http://whitneymusicbox.org (accessed October 31, 2015).

Burroughs, William S. *Word Virus: The William S. Burroughs Reader*. New York: Grove Press, 1998.

Cage, John. *Aria*, for voice, any range. New York: C.F. Peters, 1963.

———. *Empty words: Writings ’73–’78.* Middletown, CT: Wesleyan University Press, 1979.

———. *John Cage*. Edited by Richard Kostelanetz. New York: Praeger, 1970.

Cascone, Kim. “The Aesthetics of Failure: ‘Post-Digital’ Tendencies in Contemporary Computer Music.” *Computer Music Journal* 24, no. 4 (2000): 12–18.

Cassidy, Aaron. “Second String Quartet (2009–10).” http://aaroncassidy.com/music/secondquartet.htm (accessed February 12, 2017).

———. “*The Crutch of Memory* (2004), for indeterminate solo string instrument.” http://aaroncassidy.com/music/crutchofmemory.htm (accessed February 12, 2017).

———. “The String Quartet as a Laboratory and Playground for Experimentation and Tradition (or, Opening Out/Closing In).” *Contemporary Music Review* 32, no. 4 (2013): 305–23.

———. “*What then renders these forces visible is a strange smile (or, First Study for Figures at the Base of a Crucifixion)* (2008), for solo trumpet.” http://aaroncassidy.com/music/smile.htm (accessed February 12, 2017).

Chippewa, Jef. “Practicalities of a Socio-Musical Utopia.” Originally published in *Nutida Musik* 3–4 (2012–2013). http://newmusicnotation.com/chippewa/texts/chippewa_spahlinger_2013-06.pdf (accessed January 10, 2017).

Chow, Ka Nin. “An Embodied Cognition Approach to the Analysis and Design of Generative and Interactive Animation.” Ph.D., Georgia Institute of Technology, 2010.

Clarke, Bruce and Linda Dalrymple Henderson. *From Energy to Information: Representation in Science and Technology, Art, and Literature*. Writing Science. Stanford, Calif.: Stanford University Press, 2002.

Climent, Ricardo. “*Russian Disco* (2007), for flute, clarinet, and electronic expression.” http://citizenurge.blogspot.com/2013/10/russian-disco-2007-cl-flute-and.html (accessed February 17, 2017).

Cohen, Harold. “A Sorcerer’s Apprentice: Talk at the Tate Modern.” (2004) http://web.archive.org/web/20160714153206/http://www.aaronshome.com/aaron/publications/tate-final.doc (accessed February 15, 2017).

Collins, Nick. “The Analysis of Generative Music Programs.” *Organised Sound* 13, no. 3 (2008): 237–48.

Cont, Arshia. “Antescofo,” IRCAM, http://repmus.ircam.fr/antescofo (accessed February 10, 2017).

Cope, David. *Virtual Music: Computer Synthesis of Musical Style*. Cambridge, MA: MIT Press, 2001.

———. *The Algorithmic Composer*. Computer Music and Digital Audio Series. Vol. 16. Madison, Wis.: A-R Editions, 2000.

Corballis, Michael C. “Did Language Evolve from Manual Gestures.” In *The Transition to Language*, edited by Alison Wray, 161–179. New York: Oxford University Press, 2002.

Dannenberg, Roger, and Hirofumi Mukaino. “New Techniques for Enhanced Quality of Computer Accompaniment.” *Proceedings of the 1988 International Computer Music Conference* (1988): 243–49.

Dauphinais, Michael David. “Real-Time Composer-Performer Collaboration as Explored in *Wilderness*: A Dance and Audio Installation.” D.M.A., Arizona State University, 2012.

Davidovsky, Mario. *Synchronisms No. I*. New York: McGinnis & Marx, 1966.

Dawkins, Richard. *The Blind Watchmaker.* New York: Norton, 1996.

Deleuze, Gilles, and Félix Guattari. *A Thousand Plateaus: Capitalism and Schizophrenia*. Translated by Brian Massumi. Minneapolis: University of Minnesota Press, 1987.

Di Scipio, Agostino. “Formalization and Inuition in *Analogique A et B*.” *International Symposium Iannis Xenakis* (2005): 95-108.

Didkovsky, Nick, and Georg Hajdu, “MaxScore, music notation in Max/MSP.” *Proceedings of the 2008 International Computer Music Conference* (2008): 1-5; http://www.computermusicnotation.com/ (accessed February 10, 2017).

Didkovsky, Nick. “Recent Compositions and Performance Instruments Realized in Java Music Specification Language.” *Proceedings of the 2004 International Computer Music Conference* (2004): 1–4.

Draves, Scott. “*Electric Sheep*: Crowdsourced Evolving Art.” http://electricsheep.org/ (accessed February 11, 2017).

Dunbar-Hester, Christina. “Listening to Cybernetics: Music, Machines, and Nervous Systems, 1950–1980.” *Science, Technology, & Human Values* 35, no. 1 (2010): 113–39.

Eacott, J. F. “Contents may Vary: The Play and Behaviour of Generative Music Artefacts.” Ph.D., Westminster College (United Kingdom), 2007.

Eigenfeldt, Arne. “Generative Music for Live Performance: Experiences with Real-Time Notation,” *Organised Sound* 19, no. 3 (December 2014): 279.

Eldridge, Alice. “Collaborating with the Behaving Machine: Simple Adaptive Dynamical Systems for Generative and Interactive Music.” Ph.D., University of Sussex, 2008.

Essl, Karlheinz. “*Champ d’Action*, Realtime Composition Environment for Computer-Controlled Ensemble.” http://www.essl.at/works/champ.html (accessed February 15, 2017).

Ferneyhough, Brian. “Brian Ferneyhough in an Conversation with Joshua Cody,” The Ensemble Sospeso New York (Fall 1996). Archived by Internet Archive. http://web.archive.org/web/20030424094218/http://www.sospeso.com/contents/articles/ferneyhough_p1.html (accessed February 12, 2017).

Finer, Jem. “Longplayer.” The Longplayer Trust. http://longplayer.org/ (accessed February 10, 2017).

Fleischer, Richard. *Out of the Inkwell: Max Fleischer and the Animation Revolution*. Lexington, Kentucky: The University Press of Kentucky, 2005.

Fober, Dominique. “Time Synchronization in Graphic Domain: A New Paradigm for Augmented Music Scores.” *Proceedings of the 2010 International Computer Music Conference* (2010): 458–61; http://inscore.sourceforge.net/ (accessed February 10, 2017).

Fontana, Bill. “Harmonic Bridge.” http://resoundings.org/Pages/Harmonic_Bridge1.htm (accessed February 10, 2017).

Franke, Herbert. *Konforme Abbildung*. compart: Center of Excellence Digital Art. http://dada.compart-bremen.de/item/artwork/341 (accessed February 11, 2017).

Freeman, Jason and Andrew Colella. “Tools for Real-Time Music Notation,” *Contemporary Music Review* 29, no. 1 (2010): 101–113.

Freeman, Jason. “Extreme Sight-Reading, Mediated Expression, and Audience Participation: Real-Time Music Notation in Live Performance.” *Computer Music Journal* 32, no. 3 (2008): 24–41.

———. “*SGLC* (2011) for Laptop and Instrumental Ensemble.” http://distributedmusic.gatech.edu/jason/music/sglc-2011-for-laptop-and/ (accessed October 31, 2015).

Funkhauser, Christopher Thomas. *Prehistoric Digital Poetry: An Archeology of Forms, 1959–1995*. Tuscaloosa, AL: University of Alabama Press, 2007.

Galanter, Philip. “What is Generative Art? Complexity Theory as a Context for Art Theory.” *International Conference on Generative Art* (2003): 4.

Gilman, Elizabeth and Geoffrey Underwood, “Restricting the Field of View to Investigate the Perceptual Spans of Pianists,” *Visual Cognition*, vol. 10, no. 2, pp. 201–32, 2003.

Gleick, James. *Chaos: Making of a New Science*. New York: Penguin, 1987.

Goethe, Johann Wolfgang von. “Introduction to the ‘Propyläen.’” In *Goethe on Art*, edited and translated by John Gage. London: Scolar Press, 1980.

Goolsby, Thomas. “Eye-Movement in Music Reading: Effects of Reading Ability, Notational Complexity, and Encounters.” *Music Perception* 12 (1994): 77–96.

Guest, Ann Hutchinson. *Choreo-Graphics: A Comparison of Dance Notation Systems*. New York: Routledge, 1998.

Guo, Ruiwen. “Algorithmic Menagerie.” M.F.A., Rensselaer Polytechnic Institute, 2014.

Haynes, Bruce. *The End of Early Music*. New York: Oxford University Press, 2007.

Hiller, Lejaren and Leonard Isaacson. *Experimental Music: Composing with an Electronic Computer*. New York: McGraw-Hill, 1959.

Hiller, Lejaren. “Composing with Computers: A Progress Report.” *Computer Music Journal* 5, no. 4 (Winter 1981): 7–21.

Hoadley, Richard. “Notating Algorithms.” *Symposium for Performance of Electronic and Experimental Composition*. University of Oxford, January 6–7, 2012.

Honing, Henkjan. “Structure and Interpretation of Rhythm and Timing,” *Dutch Journal of Music Theory* 7, no. 3 (1998): 227–32.

Hope, Cat, Aaron Wyatt, and Lindsay Vickery. "The Decibel ScorePlayer: New Developments and Improved Functionality." *Proceedings of the 2015 International Computer Music Conference* (2015): 314–317.

Keene, Jeff. "Composing Music in Constrained Search Environments." M.S., University of Denver, 2010.

Kim-Boyle, David. "*Point Studies No. 4* (2013, rev. 2016), for voice and electronics." http://www.davidkimboyle.net/point-studies-no-4-20141.html (accessed February 15, 2017).

———. "The Ontology of Live Notations Through Assemblage Theory." *Proceedings of the 2016 International Conference on Technologies for Music Notation and Representation* (2016): 92–7.

Klein, Yves Amu. "*Octofungi*." Living Sculpture. http://www.livingsculpture.com/works/octofungi (accessed February 11, 2017).

Kojs, Juraj. "Language of Action and Cyberaction." *Journal of New Music Research* 38, no. 3 (2009): 285–294.

———. "Notating Action-Based Music." *Leonardo Music Journal* 21 (2011): 65–72.

Lachenmann, Helmut. *Gran Torso.* Wiesbaden: Breitkopf & Härtel, 1971, rev. 1988.

———. *Pression.* Wiesbaden: Breitkopf & Härtel, 1969, rev. 2010.

Laurson, Mikael, Mika Kuuskankare, and Vesa Norilo. "An Overview of PWGL, a Visual Programming Environment for Music." *Computer Music Journal* 33, no. 1 (Spring, 2009): 19–31; http://www2.siba.fi/PWGL/index.html (accessed February 10, 2017).

Lee, Sang Won and Jason Freeman. "Real-Time Music Notation in Mixed Laptop-Acoustic Ensembles." *Computer Music Journal* 37, no. 4 (Winter 2013): 24–36.

Luck, Geoff, Petri Toiviainen, and Marc Thompson. "Perception of Expression in Conductors' Gestures: A Continuous Response Study." *Music Perception* 28, no. 1 (2010): 47–57.

Mandelbrot, Benoît. *Fractals: Form, Chance, and Dimension*. New York: W. H. Freeman, 1977.

Manouray, Phillippe. *Jupiter: Pour Flûte & Électronique En Temps Réel* (1987, Rev. 1992, 2008). Paris: Amphion, 2011.

Matthews, Robert A. J. "Tumbling Toast, Murphy's Law and the Fundamental Constants." *European Journal of Physics* 16 (1995): 172–76.

May, Andew. "Bouncing Ball Conductor Patch." http://andrewmaymusic.com/Software/conductor.zip (accessed February 16, 2017).

———. "*Chant/Songe* (2004), for clarinet and computer," http://andrewmaymusic.com/Works/Chant_Songe.html (accessed February 10, 2017).

———. "*Vanishing* (2000, rev. 2017), for chamber ensemble and computer," http://andrewmaymusic.com/Works/Vanishing.html (accessed February 10, 2017).

McClelland, Christopher and Michael Alcorn. "Exploring New Composer/Performer Interactions Using Real-Time Notation." *Proceedings of the 2008 International Computer Music Conference* (2008): 1–4.

McCorduck, Pamela. *AARONS's Code: Meta-Art, Artificial Intelligence, and the Work of Harold Cohen.* New York: Macmillan, 1991.

McNutt, Elizabeth. "A Postscript on Process." *Music Theory Online* 11, no. 1 (2005).

———. "Performing Electroacoustic Music: A Wider View of Interactivity." *Organised Sound* 8, no. 3 (2003): 297–304.

Meredith, David. "The *ps13* Pitch Spelling Algorithm." *Journal of New Music Research* 35, no. 2 (2006): 121–159.

Nees, Georg. *Generative Computergrafik.* Berlin: Siemens AG, 1969.

Nierhaus, Gerhard. *Algorithmic Composition*. New York: Springer, 2009.

———. *Patterns of Intuition*. Dordrecht: Springer Netherlands, 2015.

Noll, A. Michael. "The Beginnings of Computer Art in the United States: A Memoir." *Computers and Graphics* 19, no. 4 (1995): 495–503.

Orio, Nicola, and François Déchelle. "Score Following Using Spectral Analysis and Hidden Markov Models." *Proceedings of the 2001 International Computer Music Conference* (2001): 1–5.

Packer, Randall and Ken Jordan. *Multimedia: From Wagner to Virtual Reality*. New York: Norton, 2002.

Paul, Christiane. *Digital Art*. New York: Thames & Hudson, 2008.

Pearl, Judea. "Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning." *Proceedings of the 7th Conference of the Cognitive Science Society* (1985): 1–20.

Picking, Richard. "Reading Music from Screens vs Paper." *Behaviour and Information Technology* 162 (1997): 72–8.

Pinkston, Russell. "*Lizamander* (2003), for flute and Max/MSP," http://russellpinkston.com/?portfolio=item-three (accessed February 10, 2017).

Rockman, Arnold, and Leslie Metzei. "The Electronic Computer as an Artist." *Canadian Art* 94 (1964): 365–7.

Rolland, Paul. *The Teaching of Action in String Playing: Developmental and Remedial Techniques.* Urbana, IL: Illinois String Research Associates, 1974.

Priestley, John. "Poiesthetic Play in Generative Music." Ph.D., Virginia Commonwealth University, 2014.

Prigogine, Ilya and Isabelle Stengers. *Order Out of Chaos: Man's New Dialogue with Nature*. New York: Bantam Books, 1984.

Russolo, Luigi. "Grafia enarmonica per gli intonarumori futuristi." *Lacerba* 2, no. 6 (March 1914).

———. *The Art of Noises*. Translated by Barclay Brown. New York: Pendragon Press, 1986.

Schnell, Norbert, Riccardo Borghesi, Diemo Schwarz, Frederic Bevilacqua, and Remy Müller. "FTM: Complex Data Structures for Max." *Proceedings of the 2005 International Computer Music Conference* (2005): 9–12; http://ftm.ircam.fr/index.php/Main_Page (accessed February 10, 2017).

Schoyen Collection. "MS 5105." http://www.schoyencollection.com/music-notation/old-babylonia-cuneiform-notation/oldest-known-music-notation-ms-5105 (accessed February 10, 2017).

Shafer, Seth. "*Law of Fives* (2015), for viola, bass clarinet, marimba and computer." http://sethshafer.com/law_of_fives.html (accessed February 12, 2017).

———. "Performance Practice of Real-Time Notation." *Proceedings of the 2016 International Conference on Technologies for Music Notation and Representation* (2016): 65–70.

———. "Performing Real-Time Notation." *eContact!* 18, no. 4 (forthcoming).

———. "*Polytera II* (2016), for flute, piano, and computer." http://sethshafer.com/polytera_2.html (accessed February 12, 2017).

———. "*Pulsar [Variant II]* (2014), for trumpet and computer." http://sethshafer.com/pulsar_variant_2.html (accessed February 12, 2017).

———. "*Terraformation* (2016-17), for viola and computer." http://sethshafer.com/terraformation.html (accessed February 12, 2017).

———. “the machine is the artist—is the artist the machine.” http://sethshafer.com/themachineistheartist.html (accessed February 12, 2017).

———. “*Ursa Minor* (2015), for euphonium and computer,” http://sethshafer.com/ursa_minor.html (accessed February 12, 2017).

———. “VizScore: An On-Screen Notation Delivery System for Live Performance.” *Proceedings of the 2015 International Computer Music Conference* (2015).

Silver, David, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot,Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. “Mastering the Game of Go with Deep Neural Networks and Tree Search.” *Nature* 529 (January 2016): 484–9.

Smith, Ryan Ross. “*Study no. 40.3 [pulseven]* (2014), for seven instrumentalists.” http://ryanrosssmith.com/study40_3.html (accessed February 15, 2017).

Stephton, Matt. “Christopher Strachey *Loveletters*.” http://www.gingerbeardman.com/loveletter/ (accessed February 11, 2017).

Stockhausen, Karlheinz. *Plus Minus*, 2 x 7 Seiten für Ausarbeitungen. London: Universal Edition, 1963.

Strachey, Christopher. “The ‘Thinking’ Machine.” *Encounter: Literature, Arts, Politics* 13 (October 1954): 25–31.

Surges, Gregory. “Generative Audio Systems: Musical Applications of Time-Varying Feedback Networks and Computational Aesthetics.” Ph.D., University of California, San Diego, 2015.

Syer, Katherine. “From Page to Stage: Wagner as *Regisseur*.” In *Richard Wagner and His World*, edited by Thomas S. Grey, 3–22. Princeton, NJ: Princeton University Press, 2009.

Taylor, Grant D. *When the Machine Made Art: The Troubled History of Computer Art*. New York: Bloomsbury, 2014.

Thom, Paul. “Authentic Performance Practice.” In *The Routledge Companion to Philosophy and Music*, edited by Theodore Gracyk and Andrew Kania, 91–100. New York: Routledge, 2011.

Todd, Stephen, and William Latham. *Evolutionary Art and Computers*. New York: Academic Press, 1992.

Vercoe, Barry. “The Synthetic Performer in the Context of Live Performance.” *Proceedings of the 1984 International Computer Music Conference* (1984): 199–200.

Vickery, Lindsay. “The Evolution of Notational Innovations from the Mobile Score to the Screen Score,” *Organised Sound* 17, no. 4 (2012): 128–136.

———. “The Limitations of Representing Sound and Notation on Screen.” *Organised Sound* 19, no. 3 (2014): 215–27.

Wakefield, Graham. “Real-Time Meta-Programming for Interactive Computational Arts.” Ph.D., University of California, Santa Barbara, 2012.

Whitelaw, Mitchell. *Metacreation: Art and Artificial Life.* Cambridge, MA: MIT Press, 2004.

Whitney, John. *Digital Harmony: On the Complementarity of Music and Visual Art* Peterborough New Hampshire: Byte Books, 1980.

Wiard Synthesizers. “Model 371 Dual Woggle Bug.” http://wiard.com/modular/300series/wogglebug/ (accessed February 11, 2017).

Wiener, Norbert. *Cybernetics: or, Control and Communication in the Animal and Machine*. 2nd ed. Cambridge, MA: MIT Press, 1965.

Williams, Charles Francis Abdy. *The Story of Notation*. London: Walter Scott Publishing, 1903.

Winkler, Gerhard E. “*Hybrid II (NetWorks)*—or: At the edge of musical self-organization.” In *Electronics in New Music*. Edited by Claus-Steffen Mahnkopf, Frank Cox, and Wolfram Schurig, 236–49. Hofheim: Wolke Verlag, 2006.

———. “The Realtime-Score: A Missing-Link in Computer-Music Performance.” *Proceedings of the 2004 Sound and Music Computer Conference* (2004): 1–6.

———. “The Real-Time-Score: Nucleus and Fluid Opus.” *Contemporary Music Review* 29, no. 1 (2010): 89–100.

Winkler, Todd. *Composing Interactive Music: Techniques and Ideas Using Max*. Cambridge, MA: MIT Press, 2001.

Wolff, Christian. *For 1, 2, or 3 People.* New York: C.F. Peters, 1964.

Wu, Xinxing and Guanrong Chen. “Central Limit Theorum and Chaoticity.” *Statistic and Probability Letters* 92 (2014): 137–142.

Xenakis, Iannis. *Formalized Music: Thought and Mathematics in Composition*. Harmonologia Series. Revised edition Vol. 6. Stuyvesant, NY: Pendragon Press, 1992.

Young, La Monte and Marian Zazeela. “Dream House: Sound and Light Environment.” MELA Foundation. http://www.melafoundation.org/ (accessed February 10, 2017).