Deep Decline

Marcin Paczkowski

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2016

Reading Committee:

Juan Pampin, Chair

Richard Karpen

James Coupe

Program Authorized to Offer Degree:

Digital Arts and Experimental Media (DXARTS)

University of Washington

Abstract

Deep Decline

Marcin Paczkowski

Chair of the Supervisory Committee:
Associate Professor Juan Pampin
Digital Arts and Experimental Media (DXARTS)

Deep Decline is a musical piece for chamber ensemble and interactive computer-generated sound that uses custom motion-sensing devices for controlling sound transformations. The work focuses on organic integration of playing the instrument and influencing audio processing, combining improvisation elements, computer-assisted composition, accelerometer sensors, and live sound transformations. *Deep Decline* was premiered by Ensemble Dal Niente on October 30, 2015 in Seattle, Washington.

This document describes the creative process leading up to composing this piece, as well as preparation of the performance with the musicians. Conceptual and technical aspects of the motion-sensing devices and computer-generated sound are documented, enabling future implementations of the technologies used.

TABLE OF CONTENTS

List of Figures......vi

Chapter 1. Introduction......1

1.1 *Deep Decline* – overview......1

1.2 Organization of the dissertation......2

Chapter 2. Research trajectory......3

2.1 Personal background in conducting......3

2.2 Motion interfaces for musical performance......3

2.3 Improvisation......5

2.4 Computer-assisted composition......8

Chapter 3. Previous works......12

3.1 *Still, early* for four horns and live electronics......12

3.2 *Restrained meters* for dancer and electronic media......13

3.3 *...where odd things are kept...* for sensors and live electronics......16

3.4 *Percussivometers* for percussion and live electronics......17

3.5 *Tide* for orchestra......19

Chapter 4. Deep Decline......21

4.1 Introduction......21

4.2 Form and Musical Material......22

4.2.1 Overview......22

4.2.2 Section 1 22
4.2.3 Section 2 25
4.2.4 Section 3 26
4.2.5 Section 4 27
4.3 Instrumentation 28
4.3.1 Overview 28
4.3.2 Relations between instruments 29
4.3.3 Use of percussion 29
4.3.4 Doublings 30
4.3.5 The choice of oboe or trumpet 31
4.4 Computer sound processing 31
4.4.1 Overview 31
4.4.2 Accelerometer system 32
4.4.3 Granulator 32
4.4.4 Pitch shifter 34
4.4.5 Triggering 35
4.4.6 Sustaining effect 35
4.4.7 Recording, playback, and performance illusion 35
4.5 Algorithmic material generation 36
4.5.1 Environment for computer-assisted composition 36
4.5.2 Serial procedures 37
4.5.3 Rhythmic density in section 2 38
4.5.4 Rhythmic density in section 4 38

4.6 Preparation of the improvised sections..39
4.6.1 Individual rehearsals...39
4.6.2 Source material suggestions...40
4.6.3 Phrases for wind instruments..40
4.6.4 Phrases for string instruments...41
4.6.5 Phrases for piano...41
4.6.6 Phrases for percussion..42
4.7 Notes on the preparation of the performance..42
4.7.1 Advancing of sections for the live electronics system.......................................42
4.7.2 Large-scale musical gestures...44
4.7.3 Transition between recording and playing in section 4......................................45
4.7.4 Live sound realization during the performance..45
Chapter 5. Performance notes..47
5.1 Premiere performance at Meany Theater (October 2015)..47
5.1.1 Performers and the space..47
5.1.2 Technical setup..48
5.1.3 Rehearsal process...49
5.2 Performance at the composer's recital at the Chapel Performance Space (May 2016).......50
5.2.1 Performers and the space..50
5.2.2 Technical setup..51
5.2.3 Rehearsal process...51
Chapter 6. Conclusion..53

6.1 Current state of the project........53
6.2 Future work........53
Bibliography........55
Appendix A: Technical documentation........58
A.1 Hardware and computer configuration........58
A.1.1 Overview........58
A.1.2 Computer and operating system........58
A.1.3 Audio interface........58
A.1.4 MIDI pedal........59
A.2 SuperCollider environment........59
A.3 Accelerometer system........60
A.4 Running the piece........62
A.5 Code reference........62
A.5.1 File structure........62
A.5.2 Code structure and initial settings........63
A.5.3 Parameters of the processing modules........64
A.5.4 Workflow for adjusting parameters on-the-fly........67
A.6 Accelerometer device design........70
A.6.1 Overview........70
A.6.2 Code for the ESP8266-based board........70
A.6.3 Wiring of the ESP8266-based board and the MPU6050 accelerometer........71
A.6.4 OpenSoundControl (OSC) data transmission........71

A.6.5 Adjusting settings over serial interface........ 72

A.7 Sound engineering........ 73

A.7.1 Mixing board........ 73

A.7.2 Microphone setup........ 73

A.7.3 Speaker setup........ 74

Appendix B: Score........ 76

LIST OF FIGURES

Figure 1: The score of Still, early (excerpt)..12
Figure 2: The score of Restrained Meters..14
Figure 3: The performance of ...where odd things are kept... in May 2016 at the Chapel Performance Space...16
Figure 4: The score of Percussivometers (excerpt)..18
Figure 5: The score of Tide (excerpt)... 19
Figure 6: Overall form of Deep Decline...22
Figure 7: 12-tone row from section 1 of Deep Decline.. 23
Figure 8: Culmination of section 1 in Deep Decline.. 24
Figure 9: An excerpt from section 2 of Deep Decline - single note repetition............................25
Figure 10: An excerpt from section 3 of Deep Decline - improvisations.................................. 26
Figure 11: Harmonic structure in section 4 of Deep Decline..27
Figure 12: An example of a burst of activity in section 4 of Deep Decline................................28
Figure 13: Tremolo on inverted cymbal placed on top of timpano, accompanied by flute and violin... 30
Figure 14: Raymond Larsen preparing his improvisation with the accelerometer......................39
Figure 15: The Graphical User Interface of the system created for Deep Decline, with the current event number indicated on top..43
Figure 16: Ensemble Dal Niente performing Deep Decline..47
Figure 17: Preliminary stage plot for the performance at Meany Theater..................................48
Figure 18: Performance of Deep Decline at the Chapel Performance Space (film still by Adam Hogan)...50
Figure 19: The prototype accelerometer device...61
Figure 20: Schematic of the accelerometer system..71
Figure 21: Pinout for the serial port on the Adafruit Huzzah board..72
Figure 22: Stage setup for Deep Decline using 2 stereo pairs of speakers................................. 74

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to Juan Pampin for supporting me throughout my studies at the University of Washington, for sharing his expertise and for helping me to grow as a composer and as an artist. I would like to thank the DXARTS faculty: Richard Karpen for sharing his artistic vision and providing inspiration, and James Coupe for his valuable feedback on my work.

I would like thank many friends from DXARTS, School of Music, and Dance Program, with whom I shared my academic journey: Robert Twomey, Hana Lee, Tivon Rice, Martin Jarmick, Michael McCrea, Chun Shao, Daniel Peterson, Adam Hogan, Joel Ong, Inmi Lee, Josh Archibald-Seiffer, Ania Stachurska, Yigit Kolat, Jeff Bowen, Ivan Arteaga, Luke Fitzpatrick, Andrew Angell, and Wilson Mendieta. I'm grateful for your friendship, feedback, and many occasions to collaborate on a variety of artistic endeavors.

I would also like to thank my parents, Zbigniew Pączkowski and Stanisława Pączkowska, and my sister Oktawia Pączkowska, for their love and support during my time pursuing the doctoral degree at the University of Washington.

CHAPTER 1. INTRODUCTION

1.1 *Deep Decline* – Overview

This dissertation describes composition and realization, as well as artistic research leading up to creating a musical piece for a chamber ensemble, which involves motion-controlled computer-transformed sound. The piece, titled *Deep Decline*, is scored for flute, oboe or trumpet, clarinet, piano, percussion, violin, violoncello, and live electronics. All performers, including the conductor, wear custom accelerometer-based motion sensing devices that are used to control sound transformations in real time.

This work is a culmination of my research in fusing instrumental performance and electronic sound transformations using motion detection techniques. Leading up to composing this piece a number of other works have been created, using different subsets of the techniques employed in this project.

The main artistic and technical aspect was the development of sound transformation processes that can be used by the musicians while playing their instrument. The goal was to provide organic control of the transformed sound, so the player can perform with it intuitively. This was achieved by creating custom, lightweight motion sensing devices and performing multiple tests focused on playing ergonomy. Finally, fine-tuning of the control parameters resulted in a system that allows for musically expressive instrumental sound extension and can be easily performed with and learned by the players.

Certain sections of the piece rely on the players' ability to improvise. In these sections in-depth exploration of sonic qualities of the instruments extended with live electronics system was undertaken. Both the musical material and the motion detection nuances were developed and shaped in the course of individual and ensemble rehearsals.

Deep Decline was realized employing computer-assisted composition techniques. The melo-rhythmic material of the piece was largely shaped using specifically designed algorithms, according to pre-composed parameter changes. This allowed the composer to have a more direct control over long-term musical development, as well as direct engagement with musical gestures as a whole.

The piece was premiered by Ensemble Dal Niente on October 30, 2015 in Meany Theater in Seattle. The composer led the subsequent performance of a revised version on May 26, 2016 in the Chapel Performance Space, also in Seattle.

1.2 Organization of the dissertation

This dissertation describes the process of creation of the piece, starting with the research trajectory, indicating other projects that provide context for my work. Next, brief descriptions of my previous projects highlight particular techniques that ended up being used and extended in the piece. A separate chapter is dedicated to the analysis of the piece, including a description of the electronic processes and the work with the musicians on developing the sections involving interaction with live electronics. The premiere and subsequent performance are described and finally the future trajectory for further development is indicated. In the first appendix, the technical setup is described in order to document all of the aspects of the piece, but also to enable subsequent performances, particularly in case the available technical means change over time. The full score is included in the second appendix.

CHAPTER 2. RESEARCH TRAJECTORY

2.1 Personal background in conducting

Prior to my doctoral studies at the University of Washington, I received degrees in Music Composition as well as Orchestral and Opera Conducting from the Academy of Music in Kraków, Poland. Initially, the two specializations were quite separated, however the idea of making a closer connection between performing as a conductor and creating computer music gradually became more and more prevalent. While real-time performance was always in the center of my interest, the question of choosing appropriate control techniques was often raised. My work at the Electroacoustic Music Studio at the Academy lead to composing a piece for symphony orchestra and electroacoustic layer – *Synchrophony*. In this piece fixed-media file playback is adjusted in real time to the tempo of the performance by the orchestra through a time-stretching algorithm controlled with a tap-tempo mechanism. This required a separate performer, who was responding to the conductor's gestures by tapping the correct tempo, enabling the custom synchronization mechanism to make adjustments during the performance.

Although in *Synchrophony* the conductor was not directly controlling the electroacoustic layer, this experiment inspired the current research in using conducting-like gestures for controlling and performing computer music.

2.2 Motion interfaces for musical performance

Conducting can be perceived as the original wireless motion interface for musical performance. Conductor's gestures influence the tempo, dynamics, articulation, and expression. This information is transmitted using the hands and usually a baton and is interpreted by the musicians playing in an orchestra or an ensemble. This naturally relies on the visual contact between the conductor and the musicians[1].

---

[1]Incidentally, the first remote electromechanical control device was used by a conductor: Hector Berlioz in his Orchestration treatise [1], in the chapter devised to conducting, describes an "electric metronome" in a concert hall in Brussels, set up by M. Verbrugghe. This device consisted of a button, resembling a piano key, located at the conductor's stand and connected to a battery and wires. Pressing this key would make a mechanical baton at a remote location oscillate on its board. It was used numerous times during performances conducted by Berlioz to signal tempo to the chorus or musicians offstage.

One of the first conducting-like motion interfaces for electronic music is Max Matthews' Radio Baton, also called Radio Drum [2]. This device uses 2 mallets, which are operated above a flat surface. The mallets act as antennas connected to a transmitter, while the flat surface contains an antenna connected to a receiver. The position of the mallets can be read by measuring capacitance between the transmitting and receiving antennas. Originally this device was used with Max Mathews' Conductor program, which enabled playback of MIDI sequences while controlling their tempo and loudness [3]. The system was able to detect both beats when the mallets were operated in a drum-like fashion, as well as continuous changes of position when the mallets were dragged on top of the receiving antenna's surface. Such continuous change could be used to gradually control certain aspects of music, like dynamics. Number of other composers have used Radio Drum in their work: Richard Boulanger, Fernando Lopez-Lezcano, David A. Jaffe and W. Andrew Schloss [4]. In some cases, the device is used in a way that extends beyond the original control paradigm: Fernando Lopez-Lezcano created the PadMaster program, which divides the flat surface of the Radio Drum into number of sections - pads. "Pads can control the playback of MIDI information or soundfiles either through sequences or algorithms." [5] The ability to control multiple processes, including algorithmically generated structures as well as sound playback, greatly extended the expressive capabilities of the Radio Drum.

So far the conducting-like control of the musical material have been discussed. While they give authority over certain aspects of performance, they don't allow the performer to control basic elements of music, such as pitch and rhythm. The first gestural musical instrument allowing to control these parameters was the Theremin, invented in 1920 by Léon Theremin [6]. The instrument, consisting of an enclosure with electric circuits and 2 perpendicular antennas, was played by moving the hands closer and farther in relation to both antennas, changing pitch and loudness respectively. One of the appealing aspects of this instrument is that it is being played without actually touching it. While it provides direct control over the musical material, the element of remote control is also similar to the act of conducting. This performance aspect has been used by a number of composers/performers, for example by Joel Chadabe, who used a large Theremin and conducting-like gestures to control a digital synthesizer in his piece *Solo* [7]. The Radio Drum is related to the Theremin, as both instruments are based on a similar physical principle of capacitance sensing of electromagnetic waves [8]. Other electric conductor batons were developed over time, some of them using an accelerometer [9] - a device that senses movement along one or more axes. Taking advantage of accelerometers, other composers became interested in using dancers for controlling electronic sound. One of the early examples is Gordon Mumma's Telepos written for Merce Cunningham's dance piece TV Rerun (1972) [10]. The dancers were wearing belts with accelerometers attached to them; signals from the devices were transmitted wirelessly to the electronic instruments and thus the dancer's movements were influencing the sound throughout the piece.

The use of hands for gestural control, spanning over a broader spectrum between *conducting* and *playing*, is central to Michel Waiswisz's *The Hands* [11], developed at the Studio for Electro-Instrumental Music (STEIM), originally in 1984. This was the first attempt to create a glove-like device for controlling sound. The device contains push keys, mercury switches (for detecting movement/tilt), an ultrasonic transmitter/receiver, a potentiometer, as well as a microphone, pressure sensors and a small display in later versions. The multitude of sensors embedded in this device enabled detecting movement, the distance between hands, buttons pressed, and in turn allowed control over pitch and loudness, among other elements.

In case of *The Hands* and the Radio Baton, as well as many subsequent digital controllers, the development of dedicated software was a central part of developing the performing capabilities of the device. Whether it's conducting a MIDI sequence or analyzing signal from multiple sensors to properly detect tilt, the mapping between multiple sensors and desired musical gesture was a non-trivial task.

2.3 Improvisation

The spontaneous creation of music is fundamental for all of human musical activity. Early musical practice didn't have the strict division into groups readily separate in the Western tradition - creative musicians, performing musicians and listeners; the role of the creator and performer usually belonged to the same person [12]. Moreover, when creators would play by themselves and just for themselves, they would be the sole listeners, which presumably led to changes and variations made to the music. The invention of musical notation, which eventually emphasized fixed composition as its ultimate goal, came to the Western tradition relatively late. [2]

[2] Musical notation is still known at most in the rudimentary form to the many peoples of the Near and Far East [12].

Although the knowledge of music in the first centuries of the Common Era is based on Christian liturgical chants, which were written down in imprecise notation long time after they were first conceived, the development of certain forms indicates the role of improvisation in musical development. The plainchant, being the official chant of Christian liturgies, has been gradually extended through melismatic flourish [13]. One of the most common is *Jubilus*, performed on the last syllable of certain alleluias [12]. Over time, some extended melismatic melodies started disconnecting from the plainchant and evolved into separate forms.

While these early examples were monophonic, the first treatise describing rudimentary polyphonic singing, *Musica enchiriadis* from the 9th century, describes how to double a chant in perfect intervals. From this early, not explicitly notated polyphonic practice, more developed forms emerged, like the organum and the motet. The plainchant became the basis for these forms as *cantus firmus* ("Fixed melody" in Latin) [14]. The second voice was to be performed according to predefined rules, it was not notated. These rules for counterpoint were gradually developed and more complex structures were employed by the composers, but it often appears that the compositions were still not fully worked out when they were written down. Johannes Tinctoris, in his treatise *Liber de arte contrapuncti* from 1477, states that the counterpoint may be performed either written or improvised [12]. This of course required a great skill and knowledge from the performer. The practice of improvised counterpoint was also described by other authors later in the 16th century. The popular form of two-part singing in thirds (or sixths) of the 15th century, *Fauxbourdon*, sometimes appears as three-voice composition, where only two out of three voices were notated.

It is worth noting that vocal and instrumental performance practice in the Renaissance and Baroque was constantly changing and the compositions that survived in manuscript or print often represent a pale outline of how the music actually sounded. One of the noted examples of this practice is the richly ornamented performance of Gregorio Allegri's *Miserere* at the Sistine Chapel. The manuscript of the piece, composed in the early 1600, was kept secret, while the piece was meant to be performed only by the papal choir. The special fame of the performances was reported on numerous accounts. Leopold Mozart wrote from Rome in 1770: "The manner of its performance must contribute more to it than the composition itself" [12]. Leopold I, Holy Roman Emperor, was supposedly disappointed when he received an (authorized) copy, which turned out be to a "simple fauxbourdon".

Improvisation on keyboard instruments became very prominent in the baroque period. It was not only used for embellishment of a vocal or instrumental melody, but developed into the first autonomous forms of instrumental music, like preambles, preludes and toccatas. These compositions took advantage of the instrument's ability to play diverse structures, both homophonic and polyphonic. The motoric component was also an important element of these pieces. Another improvisation-dependent technique of the Baroque period was *basso continuo* (thorough-bass) [15]. It employed notation of the bass line, with accidentals and numerals indicating harmony, over which the player would realize chordal accompaniment. This technique was used in numerous pieces of the 17th and 18th century and sometimes later, notably in the recitatives of the operas.

Solo instrumental music of the Baroque period also indicates a developed tradition of improvisation. Ornamentation in the slow movements of sonatas and concerti is present in certain publications, for example in Arcangelo Corelli's Sonatas Op. 5 [12]. These scores include both plain and ornamented versions, most likely for educational purposes. Extensive ornamentation is also seen in Johann Sebastian Bach's transcriptions of solo concerti, prepared for clavier or organ. Another improvised addition to a solo piece, both instrumental and vocal, is a *Cadenza*, a virtuoso passage inserted near the end of a movement or an aria. It underwent considerable development throughout the 19th century. By the end of the century, the improvisatory character of *Cadenza* was somewhat diminished through the practice of precise notation.

Instrumental improvisation saw a peak in the middle of the 19th century, followed by a quick decline after around 1840. The piano virtuosos touring Europe at the time were to include at least one improvised piece, usually at the end of the concert. The themes for the improvisations were popular melodies and arias, sometimes provided by the audience, partly to counter the critics' charges that these were merely memorized pieces. Fryderyk Chopin was known to improvise on Polish national melodies. Ludwig van Beethoven and Franz Liszt were among others praised for their extempore performances. Another composer and pianist, Daniel Steibelt, visiting Vienna during his year-long European tour in 1799, entered into an improvisation contest with Beethoven [16]. Steibelt was clearly defeated and reportedly mocked by Beethoven's playing, which impaired his success in the city.

The beginning of the 20th century saw almost no improvisation in Western classical music. New genres however emerged: accompaniment for social dance, for silent films, and most importantly the whole new category of jazz. While improvisation in jazz is indispensable, and jazz influenced many classical composers, the tradition of extempore playing was not re-introduced into the composition process until 1960s [17]. Around that time, jazz improvisation became so elaborate that it reached beyond the earlier "jazz" idiom and the distinction between "jazz improvisation" and "free improvisation" became arbitrary. George Lewis and Roscoe Mitchell are among many players, whose jazz-rooted improvisations defy simple categorization; these players use a variety of materials, processes and techniques in their playing. Other contributing factors for including improvisatory elements in Western music in the second half of the 20th century were composers' growing knowledge of non-European music, as well as the introduction of aleatoric procedures in the compositional process.

Since the second half of the 20th century, a substantial repertoire of electronic and computer music featuring instruments started to include improvisatory elements. These may be employed to facilitate the exploration of electronically generated sound, to embrace randomness programmed into the electronic part, or to extend the available body of knowledge of performing on a given instrument, in an attempt to extend its capabilities. The latter approach was employed by Richard Karpen in his *Solo/Tutti: Variations on an Irrational Number* (2002), scored for viola and real-time computer processing, and in number of his subsequent pieces [18]. The composer aims at drawing from the centuries of complex knowledge of playing the instrument, in order to transform and extend it, opening up new possibilities for exploration.

2.4 Computer-assisted composition[3]

The conceptualization of processes has always been present in musical thinking. Early models of procedures that accompany the beginnings of polyphonic singing have been mentioned in the previous chapter. The procedural view of music and music composition has

[3] The term "computer-assisted composition" refers to the use of the computer to support a particular compositional idea originating from the composer, for example to carry out processes like transforming the 12-tone row. This has been the way of using the computer for composition in my artistic practice so far and it is different from, and possibly a subset of the process of "algorithmic composition", which implies that the computer is used to generate and represent the compositional formalism [19].

been developing over centuries. While perception of music is a complex phenomenon involving memory, emotional response and understanding of musical language, there usually is an element of expectation, or of the hypothesis of a musical process. The listener's projection of the development of the piece is then juxtaposed with what the piece actually contains, resulting in a complex network of tensions and releases. Some of these processes can be formalized into a symbolic representation [20].

One example of a developed set of rules governing the generation of musical material is the technique of counterpoint and forms relying on it, having its peak of importance in the music of late Baroque. Such forms, like the canon or the fugue, provide the composer with rules for creating valid relations between voices, as well as possible transformations of the thematic material. Masterfully composed polyphonic compositions embrace these rules and juxtapose or support the expectations set up by them.

In the development of Western music, another significant application of algorithmic procedures can be seen in the 12-tone technique at the beginning of the 20th century, and even more so in the serialism of the post-World War II Europe [21]. In these techniques, the composer is provided with a set of rules for organizing pitch material (in the case of 12-tone technique) based on a source row, and later for organizing other elements of the composition as well. This highly structured way of composing provided, on the one hand, more support for the composing process, providing the composer with a certain amount of musical material, but on the other hand, the procedure focused solely on the minute details of the music: individual pitches, rhythms, dynamics and articulations, leaving out considerations for the large-scale formal structure or musical texture.

At the same time, the second half of 20th century brought interest among composers to include elements of chance into the music creation process. These elements were indicated earlier, when discussing the history of improvisation. One approach was to use aleatoric elements, asking the musicians to develop a musical phrase according to given instructions. Another way was to use chance in the compositional process. John Cage was inspired by Morton Feldman, who in the piece *Extensions 1* for violin and piano indicated only the register of the instrument to use and the number of note repetitions [22]. The exact pitch and rhythm was decided by the performers. Cage in his Concerto for piano and chamber orchestra, used a coin toss to arrange elements of the composition, effectively avoiding any preconceptions he might have had about the form of the piece. The use of chance in music composition is not without precedence: Wolfgang Amadeus Mozart, among other composers, created a piece that was meant to be "assembled" by rolling a dice: his Musikalisches Würfelspiel [musical dice game] consists of precomposed single measures [23]. The performer chooses the order in which they are to be performed by rolling a dice, which gives a considerable number of possible resulting Minuets (and Trios).

More elaborate combinatorial techniques required a considerable pre-composition work by the composer. With the advent of the digital computer, it became possible to automate some of these procedures. The first experiments using computers for compositional purposes are attributed to Lejaren Hiller who, together with Leonard Isaacson, programmed the digital computer at the University of Illinois, Urbana-Champaign to compose the Illiac Suite for String Quartet (1956) [20]. Other pieces followed, including Computer Cantata (1964) and HPSCHD (1968). Other composers at that time also started working on generating musical material according to mathematical rules, among them James Tenney, Gottfried Michael Koenig and Iannis Xenakis.

Xenakis used stochastic formulas to generate musical material at first by hand, for example for his piece *Metastasis* [20]. Soon after he created the Stochastic Music Program (SMP) running on an IBM computer. The formulas he used were originally developed to describe the behavior of particles in gases. The SMP program was used to create a number of pieces, for example *Eonta* (1964) for piano and brass.

Xenakis' interest laid in controlling aggregates, or "clouds" of sounds. The processes he used can be described as stochastic (probabilistic): the composer provided the density of the notes, their pitch range and distribution in terms of timbre. The individual notes were generated randomly, according to these rules. The element of chance was used for material generation. However, what was perceived as a result was not a random succession of notes; instead, the changing parameters of the stochastic processes became audible as compositional elements. This technique is in contrast with deterministic procedures, as the ones used by Hiller, where the algorithm carries out a fixed, but possibly complicated compositional task. In this case, random parameters are not used. Instead, the composer provides all the required input data. Hiller's deterministic and Xenakis' stochastic approaches also coincide with their treatment of the material after it has been generated by the machine: Hiller claims that the computer's output should not be altered in the process of transcribing into musical notation. On the other hand Xenakis allows alterations, when his musical intuition demands them [20].

To date, a large number of programs for algorithmic and computer-assisted composition have been developed. Some of the ones currently available are PatchWork and Common Music. PatchWork has been used, among others, by Tristan Murail and Gérard Grisey for composing structures derived from an analysis of an instrument's spectrum [24]. Richard Karpen used Common Music in the process of composing numerous works throughout his career.

Besides environments specifically dedicated to computer-assisted and algorithmic composition, it is now possible to carry out variety of procedural techniques using existing computer programming tools, often with the aid of dedicated libraries of extensions. One example of this approach is an extension (a "Quark"[4]) for the SuperCollider [25] environment called *Ctk* (The Composition Tool Kit) [26]. It primarily enables generating musical events for non-real-time composition using the object-oriented programming paradigm. This allows for using the scripting language to generate musical events according to any rule that can be turned into an algorithm, and thus into a programming function.

[4]The extensions for the SuperCollider language, that are hosted in a public repository, are called Quarks.

CHAPTER 3. PREVIOUS WORKS

3.1 *STILL, EARLY* FOR FOUR HORNS AND LIVE ELECTRONICS

My earlier work at the University of Washington focused on opening the compositional process to include elements that were not explicitly notated in the score. Such approach has numerous aspects: it enables including musical elements that are particular to the given performer that would not necessarily be feasible to notate, it allows to approach composing in a more direct way, thorough the process of rehearsal and directed improvisation, and it allows for leaving more freedom up to the musician. All of this distributes the creative process more evenly between the composer and the performer, requiring the latter to be more creatively engaged than in the case of the thoroughly notated music. In return, this methodology facilitates for the composer to create works drawing more directly from the embodied performing knowledge of the performer.

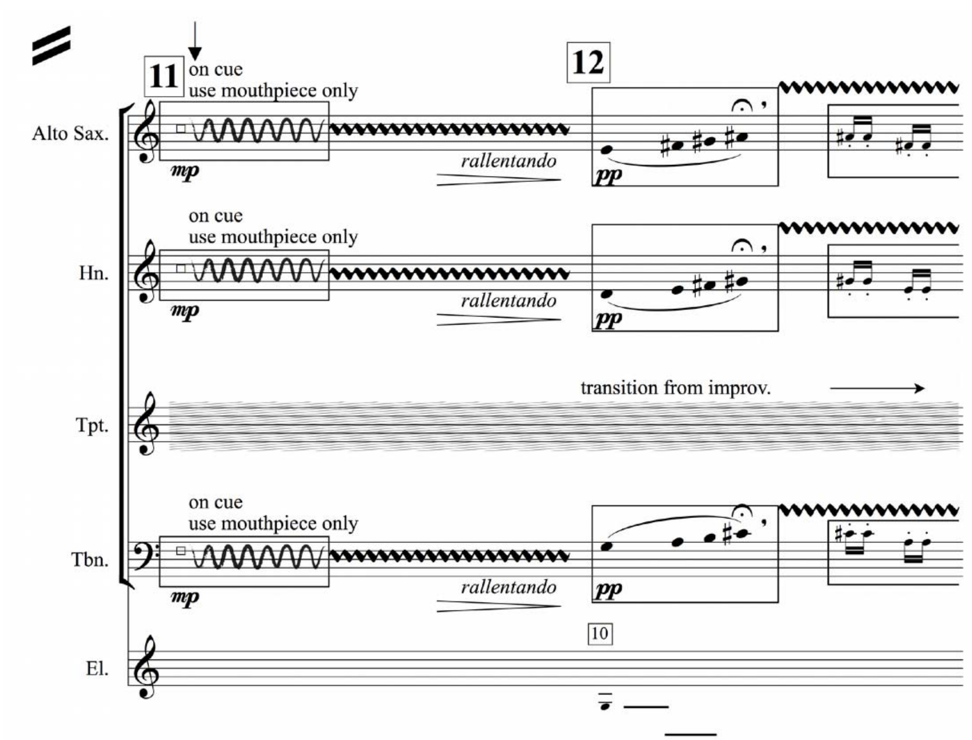


Figure 1: The score of Still, early *(excerpt)*

Still, early (2011) [27] is scored for alto saxophone, french horn, trumpet, trombone and live electronics. The notation indicates phrases to play and kinds of transitions to apply to them during transitions from one section to another (see Figure 1). The performers have a considerable freedom in realizing these instructions, however the resulting piece has its own identity, thanks to indicated musical structures and still limited range of possible interpretations.

Working with the performers in the process of rehearsing and adjusting live electronics' parameters was an intrinsic element of the composition. Even though the notation indicated a number of possibilities, the exercise of playing, improvising, and the composer's curatorial role in these tasks gave shape to the final form and material of the performance.

Finally, this piece is focused on the computer's involvement in improvisation. One of the main aspects of the live electronics system is the ability of the computer to adjust its sonic output based on the performer's playing. The system is capable of recognizing pitch, loudness, and rhythmic density, but also the melodic direction of the phrase, articulation character, as well as some timbral aspects of the sound. These elements are mapped to various parameters of the computer-generated sound, contributing to the creation of a reactive system, supporting a variety of musical gestures that the performer can play. This mechanism is featured in extended improvised sections of individual instruments, accompanied by the computer.

3.2 *Restrained meters* for dancer and electronic media

The following chapter describes my increasing interest in using motion interfaces for controlling musical processes. The first experiments to analyze movement were realized in 2012 using a handheld computer, that had an accelerometer built in. With the aid of available software, it was possible to stream sensor data wirelessly using a wireless network using the Open Sound Control (OSC) protocol. These first experiments had a twofold goal: to facilitate physical gesture recognition, and to enable motion-controlled timbre generation. Gesture recognition itself was not pursued further after some initial tests. Timbre generation, however, became one of the techniques involved in the first finished piece using accelerometers: *Restrained meters* (2013) for dancer and electronic media.

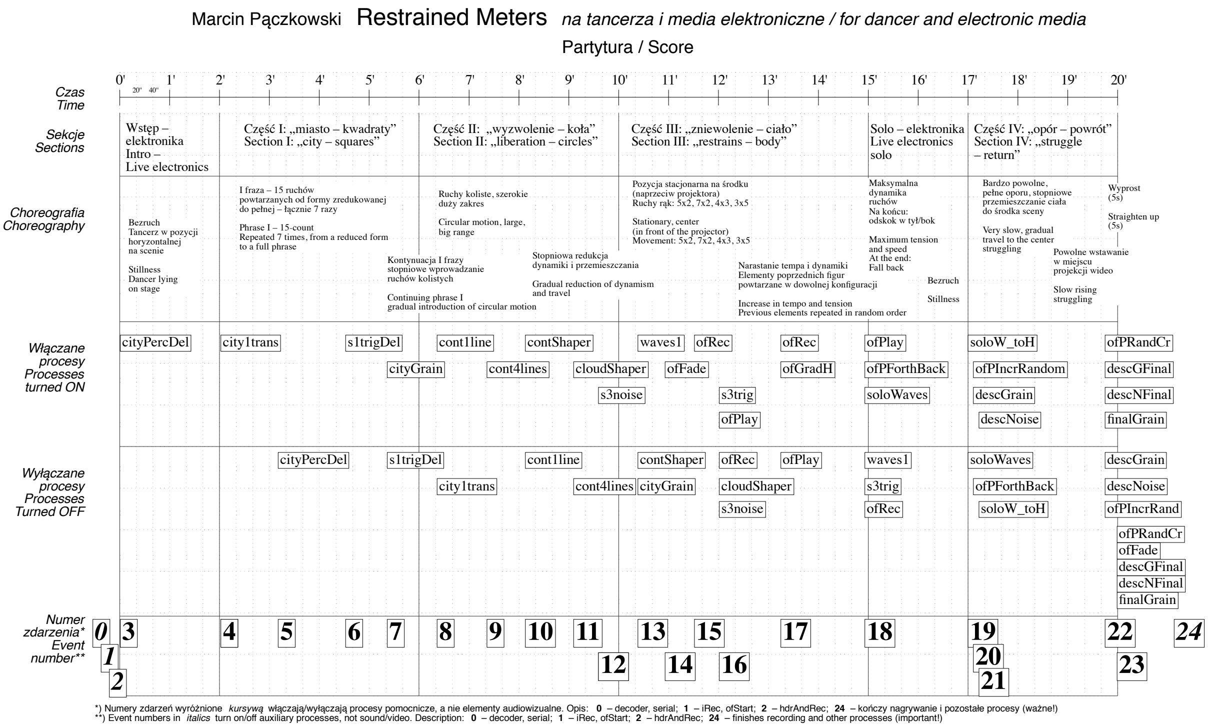


Figure 2: The score of Restrained Meters

This piece integrates dance with interactively generated sound and video. The dancer wears a custom accelerometer system consisting of four accelerometers positioned on his hands and ankles. The signal from the sensors is transmitted wirelessly to the computer, where it's being processed in numerous ways, including frequency domain Fourier analysis and waveshaping. The project also uses a depth camera and a projector, enabling selective projection of the dancer's body.

The custom accelerometer system was build using Arduino LilyPad microcontrollers, that were used to gather data from the accelerometers, and to send them using a serial protocol through an xBee radio to the computer. This first iteration of the custom hardware provided good performance, however required wired connections running under the performer's clothes, between accelerometers on his limbs and the microcontroller with the wireless transmission device, located in the lower abdominal area. The current updated accelerometer system is using self-contained sensors with transmitters, without extensive wiring. This current system, used also for *Deep Decline,* is described in detail in the appendix.

Sound in this piece was realized with Ambisonics[5]. All sounds are positioned in the virtual space around the audience and can be reproduced by a variable number of speakers. The sound movements are related to the movements of the dancer. This links together the performing space with the sonic space of the audience. The use of spatialized sound was later employed in my other works involving accelerometer systems.

The inspiration for the piece comes largely from works by Merce Cunningham and his artistic method of combining the choreography with the music shortly before the performance. In this piece, choreography and music were developed independently throughout a large portion of the creative process. They were however both created in response to similar formal ideas, which provided a higher-level connection between the two. At the time when they were combined together, real-time control using the accelerometers was leveraged.

The choreography for the piece was developed together with Wilson Mendieta, who was the dancer who performed the piece. It is divided into four sections, each consisting of a main phrase that is gradually developed by the dancer. The first section deals with sharp edges and rectangular motion, while the second one involves circular motion, both in terms of body movement, as well as travel on stage. The third section reintroduces sharp-edged figures, but it is performed in one place; in this section video projection provides counterpoint to the dancer's movements. The last section consists of one extremely slow gesture in which the performer moves back to the stage center position. The score for the piece is presented in Figure 2.

The main concept in this piece revolves around the dancer being the only performer, not just in terms of the choreography, but also controlling the sound. In order to do that, numerous ways of data analysis were being performed, aiming to connect the characteristics of the movement with the characteristics of sound. One of the techniques having a considerable potential is to use accelerometer data to create wavetables, used in the synthesis processes. That way the movement can be directly shaping the timbre, and a wide array of gestures can generate varied sonic results. Another process aiming at timbral control was frequency domain analysis of the accelerometer data. The chosen analysis windows were used to create spectral filters transforming other sounds. Finally, more direct mappings between motion and sound were also

[5]Ambisonics is a surround-sound system that enables recording, manipulation and reproduction of a full 3-dimensional sonic space [28].

employed: in the section of the piece involving circular movement, the changes in rate and direction of the motion are reflected in smoothly changing pitch of numerous sonic layers.

3.3 *…WHERE ODD THINGS ARE KEPT…* FOR SENSORS AND LIVE ELECTRONICS

After the initial experiments with *Restrained meters*, I started the development of a system for my solo performance. During this research *Electronic study 02* was composed. This was the first solo performance project using accelerometers.

Figure 3: The performance of ...where odd things are kept... *in May 2016 at the Chapel Performance Space*

In terms of sound, the main material was taken from 4-channel surround recordings made on the campus of the University of Washington. The main focus of the piece was to develop a vocabulary of movement and applicable analysis techniques that would enable a performance resembling the one of a conductor, as opposed to the one of a dancer. This involved triggering sounds on the downbeat gesture, for example. Other gestures and sounds were also developed, as well as direct control over spatialization of the sound.

This first solo project was later developed into a new piece, *...where odd things are kept…* (2014). In this piece, a number of elements was introduced: many new sounds recorded in

Ambisonic format, manipulation of partials of the recorded sounds using Juan Pampin's Analysis, Transformation and Synthesis (ATS) toolkit [29], resynthesis from pre-recorded and pre-analyzed materials, and continuous amplitude control of sounds.

The main achievement of this piece has been the further development of the movement and sonic vocabulary, focused on solo performance. The character of the gestures was geared towards a musician performer, as opposed to a dancer performer. The piece is also more substantial than the initial study, through a number of different kinds of interaction with the sound and a larger number of musical materials used.

3.4 *PERCUSSIVOMETERS* FOR PERCUSSION AND LIVE ELECTRONICS

Real-time sound transformation has been one of the main aspecs of computer music I have always been involved with, starting from my studies in the Electroacoustic Music Studio at the Academy of Music in Kraków. Both *Restrained meters* and *...where odd things are kept...* investigate the relationship between physical movement and various forms of sound generation, but without real-time sound processing. At the same time, the movement vocabulary of the accelerometer projects became more closely related to conducting-like performance, as well as other musical performing techniques. This coincided with a request by percussionist Andrew Angell for the composition of a percussion piece including electronic sound transformations. This opportunity was used to take elements already developed for gestural control and to implement real-time sound transformations controlled directly by the player. The resulting piece, *Percussivometers*, was commissioned, composed for and developed with Andrew Angell.

The technique of playing percussion has some similarities to conducting. One example is the downbeat gesture, usually expressed with the hand traveling quickly down and rebounding off an invisible point located around the waist height. This can be thought of as hitting an invisible drum. In the case of a percussion performer, the instrument is obviously visible and real, however the character of the gesture is still similar to some conducting gestures. Moreover, during the act of playing, the performer's hands can perform other gestures when they are momentarily not engaged in playing the instrument. This characteristic enabled both re-using some of the motion detection algorithms, as well as developing new sound transformation effects organically controlled by the performer.

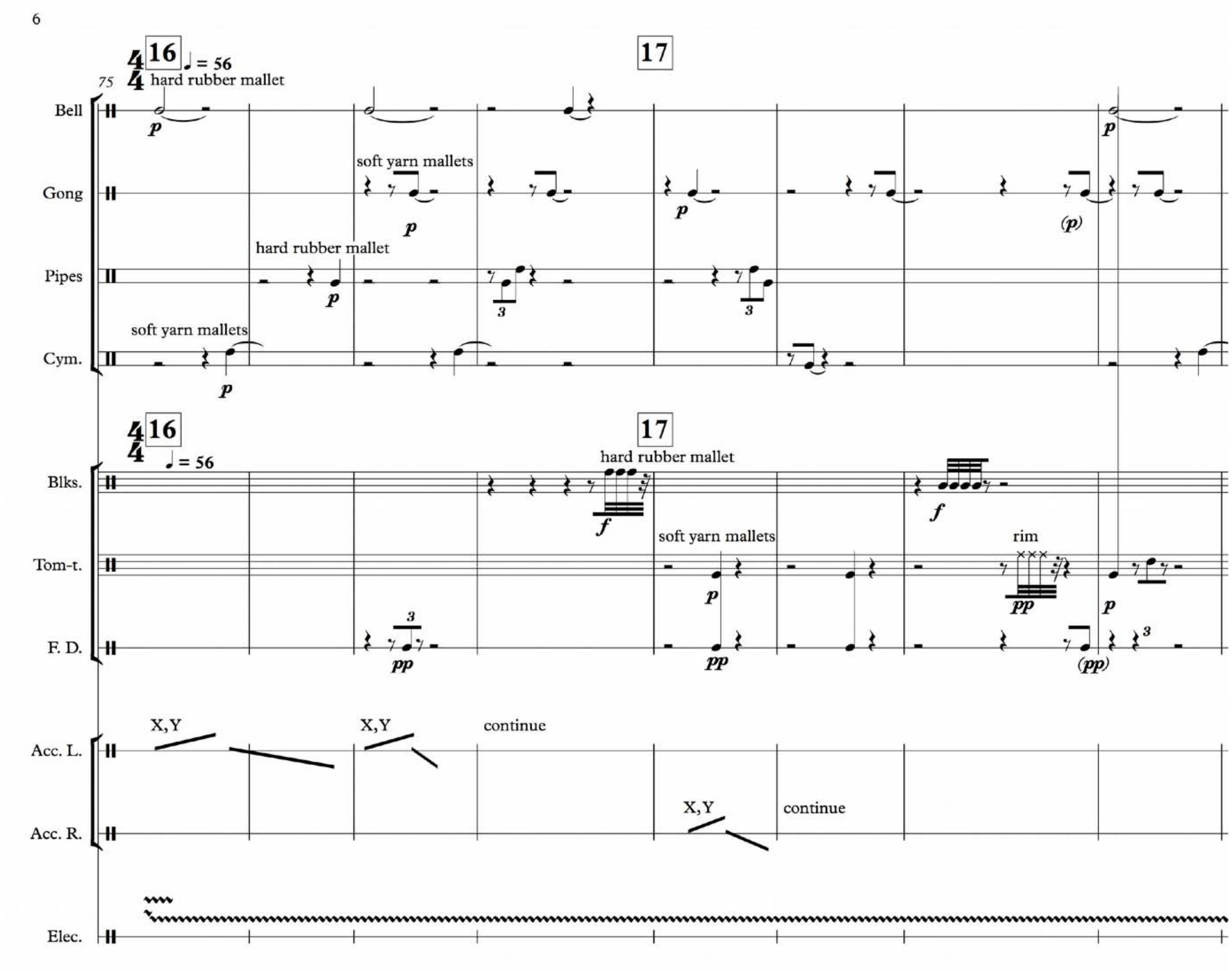


Figure 4: The score of Percussivometers *(excerpt)*

Percussivometers explores the combination of acoustic percussion sounds with motion-controlled sound transformations, extending the instruments' sound. Some of the processes include:

- changing pitch of the instrument with the tilt of the hand; this is often performed on the decay of cymbals and other metal instruments with long decay,
- recording and playback of the sound, with the tilt of the hand controlling the speed of the granulation process; this enables the performer to replay a phrase and "freeze" it at any point, or skip through some portion of it,
- granulation and extension of the live sound, changing its pitch with the hand's tilt.

The expressive power of these processes lies in the percussionist's ability to control them while playing, in a thoroughly organic way. For certain effects, one hand can be hitting an instrument with a mallet, while the other gesturally controls the sound transformation. In other cases, both hands can alternate between actively playing the instrument and influencing sound processing.

3.5 *TIDE* FOR ORCHESTRA

Tide (2015) uses only acoustic instruments, without the addition of electronically generated sound. The piece is scored for a symphony orchestra: 2 flutes, 2 oboes, 2 clarinets, 2 bassoons, 2 horns, 2 trumpets, 2 trombones, tuba, timpani, percussion and string section. The title is related to the prevalent kind of musical gesture used in the piece: increasing and decreasing rhythmic density and pitch range.

The harmonic layer of the piece is based on a harmonic series starting from C♯2. All the melodic phrases thus use a subset of the pitches following that series. In higher register and slow to medium tempo, quarter tones are used to more closely approximate the desired harmonic.

Figure 5: The score of Tide *(excerpt)*

The pitch and rhythmic material was composed with the aid of specially designed algorithms. A basic algorithm was programmed to generate a single phrase. Its control parameters include:

- envelope of ranges of rhythmic density changing over time, also indicating the slowest and the fastest rhythmic values,
- envelope for note durations over time,
- envelope of pitch range over time, independently for low and high bounds of the gesture,
- pitch quantization, enabling control over harmonic content of the phrase.

On top of the single phrase generation, another algorithm was built to create multiple phrases in relation to each other. This gave me a high-level control over the creative process, enabling creation of longer segments of musical development.[6]

In case where low and high bounds for a given parameter were provided, the resulting value (pitch or rhythm) was chosen randomly within the given bounds. This yielded slightly different results each time the algorithm was used. I would evaluate the result of each instance of the generated material and either decide to keep it, generate it again or make minor changes. An excerpt from the score is presented in Figure 5.

This curatorial approach to material is an emerging artistic method employed by me across numerous works. It very much pertains to the pieces using field recordings, like *Electronic Study 02* or *...where odd things are kept…*, where choices need to be made to decide what to keep and what to reject from the recorded sound. This approach also concerns the process of directed improvisation and the work with the performers throughout the creative process, like in *Still, early*, or *Deep Decline*, where the performer would play a phrase according to the given instructions, but then that phrase would be modified by the composer. The ability to take a preexisting material and manipulate it resembles not only a sculptor's work, but also subtractive sound synthesis techniques, in which a broadband signal is filtered to obtain the desired sonic effect.

[6]The higher-level algorithm sequentially generates the single phrase numerous times with predefined envelopes (rising and falling for the pitch and rhythmic values). The ranges of for the individual phrases' envelopes are defined as envelopes themselves, so the generated structure evolves over time, while maintaining the character of the individual phrase.

CHAPTER 4. *DEEP DECLINE*

4.1 INTRODUCTION

My previous works, described earlier, indicate the research trajectory for *Deep Decline*. In this piece a number of formerly investigated techniques was further developed and refined: instrument extension through accelerometer-controlled sound processing, computer-assisted composition, as well as the inclusion of improvisation in the creative process. These elements were then combined with intuitively composed harmonic and melodic structures.

My main focus was on the refinement of the motion-controlled sound transformation techniques. As mentioned earlier, the goal is to extend the playing capabilities on the given instruments, maintaining ergonomic control of the processed sound with little or no change in the playing technique. All of the players, as well as the conductor, had custom motion sensors attached to their hands. In certain sections of the piece the movement influences real-time sound processing of their own instrument. The conductor controls sound played by all instruments, either as a featured soloist (when the instruments are not playing), or as another performer, adding a sonic layer on top of what's being played by the other performers.

In the process of creating the piece, algorithmic procedures were used to aid with the compositional process, ranging from transformations of a 12-tone row, through stochastic control over rhythmic repetition density, to the distribution of harmonic material between instruments and the superposition of rhythmic structures. These techniques are used distinctively in different sections of the piece, marking the formal structure and musical development.

Improvisation plays an important role in the creative process. Improvised sections contain indications of musical material to use in the process of developing individual solo parts. However, the main point was to explore the musical possibilities of the hybrid instruments combined with the motion-controlled sound transformations. This process focused on individual work done with the players in order to acquaint them with the sonic possibilities of the computer sound processing, as well as to develop a musical vocabulary of their now hybrid instruments, so that they could fully utilize it during the performance.

Illusion is yet another element that I am increasingly fascinated by. In particular the illusion of performance, juxtaposed with sounds that could not be played by the instruments, was inspired by the ending of *.Eals (Oomsu)*, an orchestra piece by Paweł Szymański, written in 2009 and premiered at the Warsaw Autumn festival the same year. In his piece, Szymański imperceptibly transitions from the orchestra's acoustic sound to a previously recorded material. The music that is played back is characterized by subtle microtonal shifts, but is otherwise perfectly reproducing the sound of the acoustic instruments. During playback, the orchestra still maintains the appearance of being actively playing. At the very end, playback stops abruptly, while the orchestra keeps mimicking performance. It's not until this moment, when the illusion is broken, that what happened becomes apparent.

4.2 Form and Musical Material

4.2.1 *Overview*

Deep Decline has four distinguishable sections. Each of them is characterized by a unique musical material, as well as a distinct procedure for its development. The climax of the piece occurs at the end of the third section, which is dedicated to individual solos and thus it can vary in length between performances. The graph in Figure 6 indicates approximate times based on the second performance of the piece.

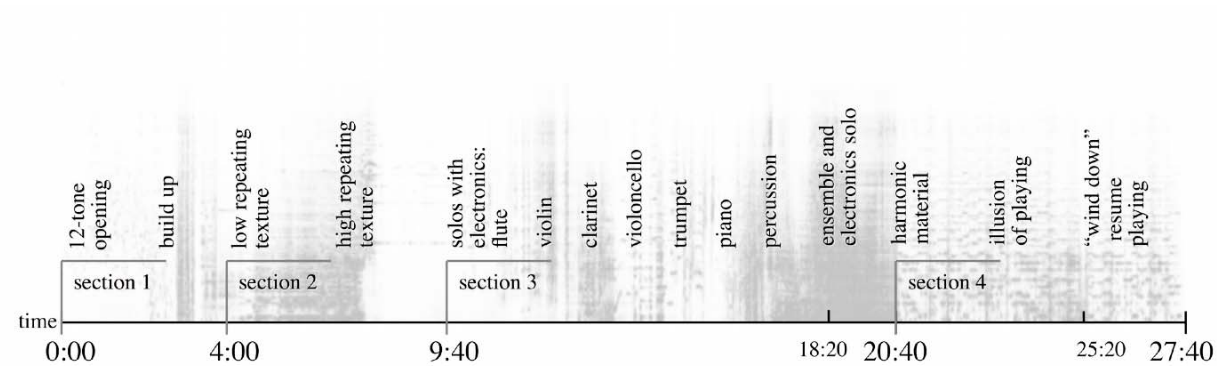


Figure 6: Overall form of Deep Decline

4.2.2 *Section 1*

The first section spans from the beginning of the piece until the measure 57 (rehearsal letter B). The music is based on a 12-tone row (see Figure 7). The phrase starts with an ascending minor sixth interval, followed by four descending semitones and then an ascending tritone. This structure was chosen by analyzing my own previous works, which lead to realization that the ascending sixth, followed by the descending motion, was intuitively chosen on numerous occasions[7].

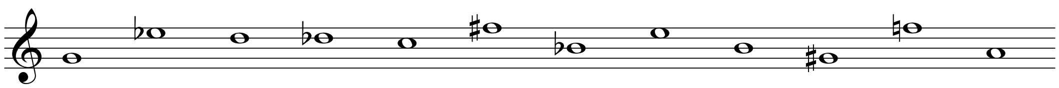


Figure 7: 12-tone row from section 1 of Deep Decline

After conceiving this opening structure, an algorithm was used to complement the row using the remaining pitches of the dodecaphonic scale. The algorithm chooses the complementing pitches randomly and was executed numerous times. Of all the generated structures, one that used an interval of sixth extensively was chosen, establishing and strengthening the intervallic identity of the row.

The whole first section is using only this row, in its prime, inverted, retrograde and inverted retrograde forms, in all twelve transpositions. The pitch material is not subject to any development beyond permuting the row, however other elements were algorithmically controlled: the number of the instruments playing, low and high bounds of the pitch register, as well as rhythmic speed and intensity. The overall increase of these parameters lasts until measure 40, however the development is interrupted by manually removing the material between measures 28 and 35. This thinning of the structure is accompanied by the differentiation of dynamic and articulation markings, and by a ritardando. After reaching a momentary climax in measure 40 (see Figure 8), subsequent repetitions of the motive use smaller and smaller sections of the row, eventually employing only two pitches, and finally transitioning to the new section, which focuses on repetitions of a single pitch. Through section 1 instruments play without any electronic sound processing.

[7] An example of a significant use of this figure can be found in my composition *Wieczorem* for chamber choir (2006).

Figure 8: Culmination of section 1 in Deep Decline

4.2.3 *Section 2*

The second section, starting at measure 57 (rehearsal letter B) and ending at measure 146 (rehearsal letter E), focuses on rhythmic density development of structures that initially used very limited pitch material (single note repetitions in the low register). Density rises gradually but in a nonlinear fashion, with momentary increases and decreases throughout the section. From measure 86 on (rehearsal letter C), the increased intensity is maintained, while the pitch material changes from single note repetition to a range of random pitches gradually moving higher and higher in the register. That transition culminates in measure 116 (rehearsal letter D), after which the pitch structure reverts to a single note repetition or to a diad repetition (this time in the high register), while the rhythmic density and the dynamics gradually diminish.

Figure 9: An excerpt from section 2 of Deep Decline - *single note repetition*

Throughout the second section the notes played by the instruments are short, and the rhythmic density varies by changing the time interval of note onsets, and not their durations. The computer sound processing extends these sounds through a granular process, which creates a continuous texture, accentuated by the sharp articulation of the instruments.

4.2.4 *Section 3*

The third section is dedicated to the solos of the individual performers, exploring motion-controlled sound transformations. It spans from measure 146 (rehearsal letter E) to measure 155 (rehearsal letter N), although in this section meter is not used and measure numbers refer to larger musical structures. Each instrument has a loosely defined amount of time to perform, which is approximately 30 seconds. Subsequent instrument entrances are indicated by the players themselves or by the conductor. In the transitional moments, the two instruments can briefly interact musically, creating small duets. The timing of this section is indicated chronometrically; there is no meter employed.

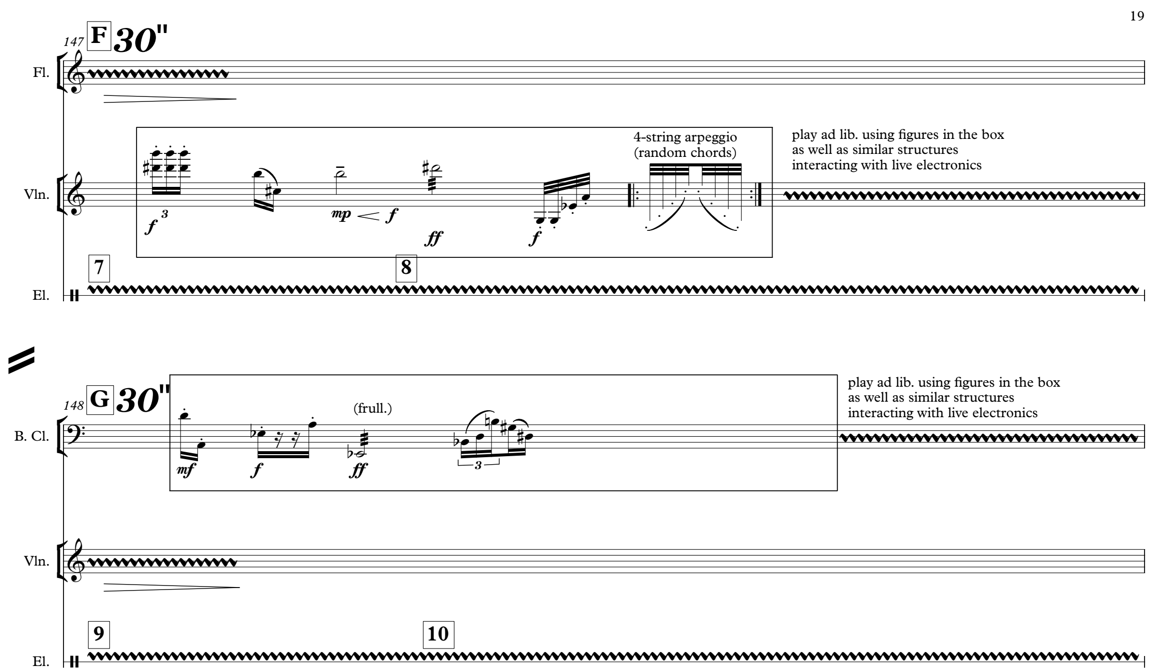


Figure 10: An excerpt from section 3 of Deep Decline - *improvisations*

The score provides motives and figures that can be used as starting points for developing improvisations. These figures are taken from other sections of the piece. The preparation of the improvisations is described in detail in a later chapter, however it should be noted that the performers had considerable freedom in developing their parts. The main objective of this section is to leverage their ability to control sound processing through the hand motion.

After their individual solos, all the performers resume improvising. In this section the conductor controls the overall intensity of the texture. Like in the music of Earle Brown, the conductor is encouraged to suggest the character of musical material for the individual players, starting and stopping their parts to shape a musically meaningful narrative. The ensemble improvisation culminates with the conductor taking charge of electronic sound processing, at which point the performers stop playing. The section concludes with a solo by the conductor.

4.2.5 *Section 4*

The last - fourth - section starts in measure 155 (rehearsal letter N) and lasts until the end of the piece. It employs a new harmonic language, that is in contrast with the previous materials. Five chords, chosen intuitively, are the only harmonic framework of this section (see Figure 11). The harmonic pace is fairly uniform and the chordal structure is repeated over and over again. While these chords are meant to serve as an underlying harmonic structure, the rhythmic intensity, the instrumentation and the resulting melodic motives of individual parts are generated algorithmically. The goal for such procedure was to independently control the harmonic pace and the momentary phrases.

Figure 11: Harmonic structure in section 4 of Deep Decline

Bursts of increased rhythmic activity, supported by louder dynamics, are a characteristic of this section. These bursts do not change the pace of the harmonic structure; instead they embellish the chordal voicing fitting a given place in the harmonic phrase. An example of one of the bursts in piano, violin, and violoncello is presented in Figure 12.

Starting from the beginning of this section, all instruments are being recorded. At measure 193, the playback of the recorded sound starts, after the recording stops shortly beforehand. The playback fades in very slowly and is played through a separate pair of speakers, positioned behind the ensemble. Since the harmonic pace is similar throughout the section, the recorded sound should blend seamlessly with the acoustic sound. At this time, the musicians must maintain physical movement while they play. They should also perform a gradual diminuendo and their sound should gradually diminish. Eventually, at measure 206 (rehearsal letter P) acoustic sound ceases completely, while the recorded sound continues. All musicians, as well as the conductor, should maintain the appearance of playing, making sure that their hand with the accelerometer device is kept in motion.

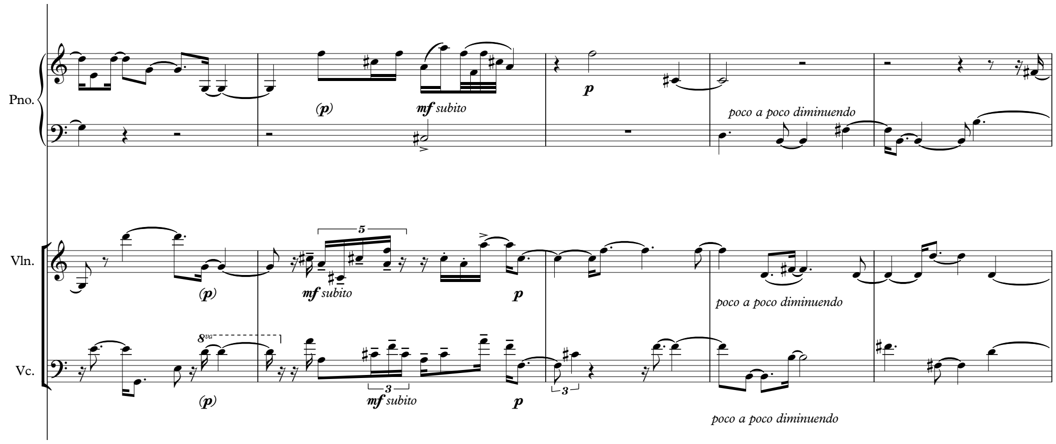


Figure 12: An example of a burst of activity in section 4 of Deep Decline

The whole ensemble continues mimicking the performance for about 10-20 seconds and then, right before the rehearsal letter Q, everyone ceases to move. At this time the recorded sound "winds down", breaking the illusion of playing. After a few seconds the ensemble resumes movement and the playback resumes, too. While the musicians do not resume playing, their movement starts controlling additional sound processing, which is used to perform another ensemble improvisation, this time using exclusively physical movement, without actually playing the instrument. The kinds of sound transformations are the same as in the third section, where the musicians were improvising individually.

After a rise and a fall of overall intensity, performed at the discretion of the players, the conductor resumes beating time in measure 209 (rehearsal letter R). At that time, musicians resume playing their parts and the recorded sound fades out. In this last segment of the piece, the conductor uses the left hand in a downbeat-like gesture to trigger sustained sounds, originating from the current acoustic sound.

4.3 Instrumentation

4.3.1 *Overview*

Deep Decline is scored for a variation of a Pierrot ensemble [30]: flute (doubling on piccolo), oboe or trumpet, clarinet (doubling on bass clarinet), violin, violoncello, piano, percussion, and live electronics. This heterogenous ensemble provides a multitude of timbral possibilities, as well as a wide range of pitch registers.

4.3.2 *Relations between instruments*

The ensemble plays primarily tutti in the first and fourth sections of the piece. At the beginning, instruments are introduced sequentially, leading up to measure 39, where all of them are playing. In the second section of the piece (measures 57 through 146) instruments are grouped together based on their pitch range: the low range is assigned to bass clarinet, piano and violoncello, while piccolo, oboe (or trumpet), piano and violin cover the high range. Percussion also participates in this section accompanying the low instruments, using a tam-tam and a cymbal, and it contributes to shape the rhythmic development, while being pitch-agnostic.

The third section of the piece (rehearsal letters E through N) is dedicated mostly to solo playing. Aside from the ensemble improvisation at rehearsal letter L, instruments play individually. The order in which they play is based on a principle of alternation between the instrumental families. The order is as follows: flute, violin, clarinet, violoncello, trumpet/oboe, piano, percussion.

4.3.3 *Use of percussion*

The percussion set for *Deep Decline* consists of: 1 timpano (28”), crotales (one octave), snare drum, 2 tom-toms, 2 cymbals and a tam-tam. The role of percussion changes throughout the piece, at times it serves as an accentuated marker at the climax or the ending of a given motivic structure, for example in measures 40, 42 and 45. This treatment is very much like the role of timpani in a symphony from the classical period. As mentioned before, in the second section of the piece, tam-tam and cymbal contribute to the rhythmic layer of the ongoing texture. In the third section, the instrumentalist is free to use all the instruments available in improvisation.

There are two notable uses of percussion in the piece, in which it contributes to the overall pitch structure. Starting at measure 48, the percussionist plays a tremolo with soft mallets on an inverted cymbal that is placed on top of the timpano, while using the pedal to change its tuning back and forth. This results in perceived timbral change with a considerable pitch identity, which is then reflected and complemented by the violin with a slow glissando up and down with an artificial harmonic (starting at measure 49), and also by the flute with a slow bending of the pitch up and down (starting at measure 51). An excerpt from the score, showing this section in flute, percussion, and violin is presented in Figure 13.

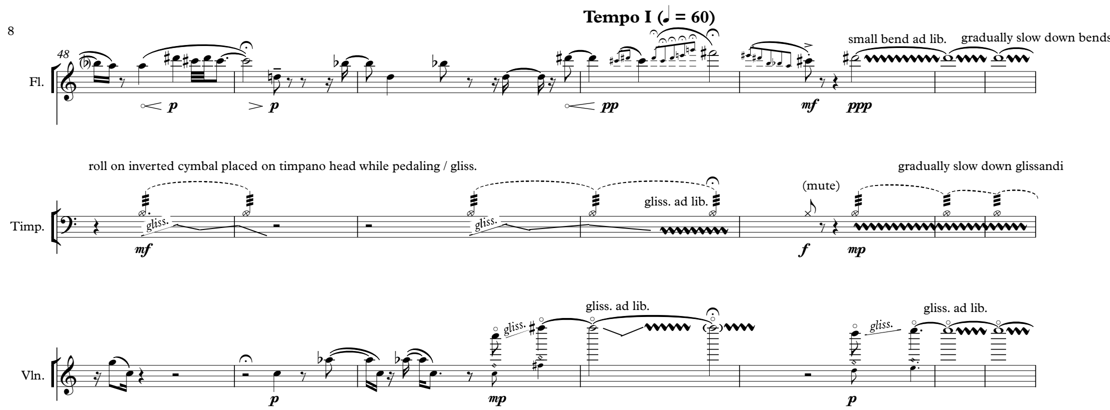


Figure 13: Tremolo on inverted cymbal placed on top of timpano, accompanied by flute and violin

The last section of the piece (from measure 155 to the end) is where the percussion contributes the most to the pitch material. Pitches are played on crotales, as well as timpano. While all percussion instruments contribute to the moderately-paced structures, they also support the bursts of activity, particular in this section of the piece. An example of pitch support can be seen in measures 171-172, which employs glissando on timpano. In other places, these bursts are supported by non-pitched instruments, like tom-toms and snare drum in measures 184-185.

4.3.4 *Doublings*

As mentioned earlier, the score requests certain doublings: piccolo for flute and bass clarinet for clarinet. This approach greatly enriches the low and high ends of the pitch range of the ensemble. The low and high extremes are used to a great degree in the second section of the piece. Also, the bass clarinet's timbral richness gives a certain character to the improvised solo in the third section.

4.3.5 *The choice of oboe or trumpet*

The instrumentation of the initial version of the piece included oboe, and not trumpet. This was dictated by the setting of the ensemble which was to be performing the piece. In the process of preparing the second performance, I had the opportunity to include a trumpet in the ensemble. After some consideration and making the necessary adjustments, I decided to substitute the oboe part with the trumpet.

The timbral qualities of the oboe were very fitting for number of melody-oriented elements of the piece. The phrases requiring quick jumps in extreme registers were easy to execute. On the other hand, the trumpet provided a larger range of possible timbres, especially when using a variety of mutes, and also a great range of dynamics, and a little more freedom in physical hand manipulation when interacting with live electronics.

The process of transforming the oboe part into the trumpet part consisted of changing the register of certain sections, as well as limiting the number of large melodic jumps originally included in the part. Mute indications were also added. Most importantly though, the solo in the third section of the piece was prepared with the performer, employing musical phrases that take advantage of the instrument's timbral capabilities.

After evaluating both versions, I consider them equivalent. Despite some differences in the parts, they both convey unique qualities, supporting the overall character of the piece.

4.4 Computer sound processing

4.4.1 *Overview*

The computer sound processing in *Deep Decline* consists of processes that manipulate sound of the acoustic instruments, either through real-time transformations, or through a variation of recording and playback. Most of them are controlled in real-time through physical gestures. The piece is meant to be primarily amplified using a stereo speaker setup, however the internal sound processing is realized in the Ambisonics format [28], similarly to the number of my other pieces. Working with spatial sound is part of a larger trajectory in my work and maintaining consistent workflow facilitates easier development from piece to piece.

The process of capturing instrument's sound during the performance and using it for playback, granulation and for other transformations was largely inspired by Richard Karpen, who uses similar approach in a number of his works for solo instrument with live electronics, for example *Solo/Tutti: Variations on an Irrational Number* (2002) [18] *Aperture* (2006) [31] and *Strand Lines* (2008) [32].

4.4.2 *Accelerometer system*

Most sound processing algorithms in the piece are controlled using a custom accelerometer-based motion interface. The accelerometer is a device used to measure linear acceleration [9]. The particular device used in this project is capable of performing measurement along three perpendicular axes.

One could imagine that the primary use for such device would be to measure an abruptly changing movement. However, because of the Earth's gravity, tilting the device gives a clear change in the readout. Depending on the character of the movement, here associated with playing on a given instrument, either the tilt or the rate of change are used to control musical parameters.

All performers wear this device during the performance. In most cases the right hand was chosen as the one providing more independent control (like in the case of the flute and the clarinet), or because it enabled to better capture the inherent movement associated with performing on a given instrument (like bow movement on violin and violoncello). The exception is the trumpet; the performer is able to use the left hand to operate the accelerometer independently from the right hand, which is used actively to play the instrument. The conductor wears devices on both hands to enable control of two different sets of parameters.

The digital accelerometer sensor interfaces to a microcontroller with embedded radio transmission capability, which sends the data through a wireless network to the computer. A more detailed description of this system can be found in the appendix.

4.4.3 *Granulator*

One of the main processes used in conjunction with accelerometer-based motion-controlled devices is a granulator. It records sounds into a circular buffer and plays back short segments extracted from it. The recording is activated only when the sound's amplitude is above a settable threshold, while the granulating process is continuous. The main parameters of the process are:

- audio input,

- rate at which new grains are created,
- the low bound for the range of pitches (playback rates) of the grains,
- the high bound for the range of pitches (playback rates) of the grains,
- amplitude threshold, above which recording is activated.

In order to provide a more natural control over the resulting texture, playback rate range is expressed in a *midiratio*. This unit defines the resulting transposition in semitones, as opposed to playback rate or frequency ratio: (0) means there is no transposition (frequency ratio of 2^(0/12) = 1); (1) means the sound is transposed one semitone up (frequency ratio of 2^(1/12) ≈ 1.0595); (-12) means the sound is transposed one octave down (frequency ratio of 2^(-12/12) = 0.5). That way, the control is linear in regard to the perceived pitch change. The resulting transposition for each grain is chosen randomly in uniform distribution between low and high bounds. That choice can be further quantized to a multiple of any musical interval; this function is usually used to make all grains quantized to the nearest semitone.

These parameters are controlled in real time by the accelerometer device. The two primary modes of control are:

- tilting the hand to change the rate of granulation and to change low and high bounds for the pitch, and
- analyzing the rate of change of the accelerometer's movement and mapping it to both the rate of granulation and the pitch range.

The first method is used on flute, clarinet, oboe/trumpet and piano. The musicians are able to tilt their hand while playing the instrument. In some cases it is feasible to tilt the whole instrument, particularly the flute, so the position of the hand in relation to the instrument doesn't change. The granulation process controlled by the conductor also uses this mode of control.

The second mode is used on instruments which employ considerable movement as part of their regular playing technique: violin, violoncello, and percussion. In their case, the amplitude of the movement (how abruptly does the direction of the movement change) was mapped to the granular process. The mapping was performed so that relaxed or slow movements would not trigger any granulation (the grain generating frequency would be equal to zero). As the players increase the intensity of their phrases the granulation process is introduced proportionally to the intensity of the bow or the mallet movement.

As mentioned before, the granulation process happens constantly for the duration of a given section, whether the instrument is making sound or not. This enables the players to create contrapuntal structures, that continue even if the instrument stops being played, in which case the control over the processed material is maintained. In this way, string players can sustain their processed sound by moving the bow in the air, while wind players can achieve the same result by moving their hand and the instrument without blowing in it.

4.4.4 *Pitch shifter*

Another process used extensively in conjunction with the detection of the tilt of the accelerometer is a pitch shifter.

The algorithm of this real-time pitch shifter was implemented by the request and under the supervision of Juan Pampin, who developed the initial real-time version of it. The current implementation combines Pampin's approach with Joseph Anderson's non-real-time implementation of Keith Lent's pitch shifting algorithm [33]. This algorithm is characterized by being able to change the pitch, without changing the formant structure of the sound. While it works only for the monophonic and periodic (pitched) sound sources, it yields very good results for instruments like the flute, clarinet and trumpet.

The control for this process is similar to the tilting control of the granular process: the tilt is mapped to control the transposition of the processed sound. On top of that, the amplitude of the processed sound is controlled by tilting the hand along a different axis.

The pitch shifter is used on the flute, trumpet (oboe), clarinet, and percussion. In certain sections of the piece, this process is active simultaneously with the granulator. In order to enable control over both kinds of processing, the parameters are chosen in a way that the range of the tilt controls only pitch shifting, while the granulator is silent. Tilting the hand past a certain point activates the granulator, while the pitch shifting is attenuated. This enables gradual transition from one process to another, performed seamlessly while also playing the instrument.

4.4.5 *Triggering*

Triggering sounds is used in section 3, where the conductor takes over the control of the sound processing. The gesture used in this case resembles a regular conducting downbeat. The accelerometer detects a rapid change of direction when the conductor's hand bounces off an invisible point in the air and this change is interpreted as a trigger.

In the section at rehearsal letter M, this process is used to trigger granular synthesis on sounds previously played by the instruments. After triggering this texture, the tilting of the conductor's hand adjusts its pitch range, very much like in the case of the granular process applied to the individual instruments. The ability to trigger numerous layers and to shape their parameters afterwards provides an expressive performing medium for the conductor in the live electronics solo section.

Another more subtle process that uses triggering is activated in the last section of the piece (from measure 209, rehearsal letter R, to the end). Throughout this segment, the conductor is able to sustain sounds based on the momentary harmonies in the ensemble. While the conductor is leading the ensemble with the right hand, the left hand triggers the sustaining part as if giving cues to an invisible player - the computer system.

4.4.6 *Sustaining effect*

A variation of the granular process, described earlier, is used to sustain instrumental sounds in the second section of the piece, starting at the rehearsal letter B. This particular process is not influenced by the accelerometers.

In this section, instrumental parts contain primarily short sounds. These sounds are used as the source material for granulation, but without changing their pitch, maintaining the harmonic layer of the section. This effect can be thought of as a more diverse version of a piano's sustain pedal.

4.4.7 *Recording, playback, and performance illusion*

Between measures 155 (rehearsal letter N) and approximately 190, all instruments are being recorded. This recording starts to be played back in measure 193; it is however introduced very quietly and the loudness increases very slowly and gradually over the course of about 10 measures. As this whole section (starting in measure 155) is based on a similarly paced and repeated harmonic structure, the introduced recording blends well with the acoustic sound of the instruments[8]. Also starting at measure 193, the instruments gradually diminish the dynamics. The diminuendo should be gradual and should result in making no sound by the measure 206 (rehearsal letter P). It is important, however, that the whole ensemble maintains the appearance of everyone still being playing, since in this section the playback of the previously recorded sound depends on sustaining the movement of the players.

The playback mechanism maintains the original playback speed as long as the performers keep moving, once the performers stop their movement the sound “winds down”. This effect resembles someone gradually slowing down a tape or a record player. The physical movement should be as natural for playing the given instrument as possible, although it might even need to be slightly more elaborate in order to ensure that the sensors perceive enough change. Because the percussion part is somewhat sparse in this section, and because it is very much trigger-oriented, an additional playback algorithm is employed in this part. It uses percussionist's gesture of striking the instrument to advance playback position to the nearest section of the recording that holds percussion sounds.

Throughout rehearsal letter P, all the musicians and the conductor should maintain active performing appearance. Ideally, the audience should not be aware that the sound is coming only from the speakers. After about 20 seconds, the conductor gives a cue and everyone ceases to move: at this time the playback slows down greatly and the illusion of playing is broken.

4.5 Algorithmic material generation

4.5.1 *Environment for computer-assisted composition*[9]

In the past, my compositional process usually employed some classic algorithmic elements, such as imitation, motivic development, and common serial procedures. These procedures were usually implemented manually, up until composing my piece *Tide*. Since then, the algorithmically conceived musical structures became more often realized with the aid of a computer program. Algorithmic procedures, used to aid in the process of composing, were

---

[8]The seamless blending is also facilitated by the fact that the instruments are amplified.

[9]The difference between computer-assisted and algorithmic composition is discussed in footnote 3 on page 8.

implemented in the SuperCollider environment, separately from the real-time performance system. Particular functions were programmed in response to the compositional choices made during the process of writing the piece.

These functions generated arrays containing information about all the notes for a given gesture. Each note had five parameters: start time, pitch, duration, velocity, and track number. These arrays were used to generate MIDI files using a *SimpleMIDIFile* class (part of the *wslib* Quark [34]). The generated MIDI files were then opened in the Sibelius [35] score editing software, where quantization settings were chosen manually. As the generated MIDI files contained complex rhythms, setting the shortest interpreted note duration and kinds of tuplets allowed was essential in obtaining a readable musical notation. Optimal settings were chosen in the process of trial and error and in most cases the generated notation required only moderate corrections. As sections of the length of few measures were generated at a time, they were first put into a temporary score, which was evaluated for readability and musical results. Afterwards, the music was copied from the temporary into the main score, where final editing took place.

4.5.2 *Serial procedures*

As mentioned earlier, the first section of the piece was composed using the 12-tone technique. First, the row itself was conceived partially through the influence of my previous works, and partially with the aid of an algorithm, which complemented the row. While this could be easily achieved without the computer, this process revealed a particular relationship between the musical material and the act of composing: the composer makes choices of a curatorial character, evaluating and accepting or rejecting materials that already exist, or were algorithmically created. A parallel approach can be found in my treatment of the field recordings used in my other pieces: the preexisting material is evaluated for its sonic qualities and aesthetic appropriateness for a given musical moment.

After creating the initial row, another algorithm was used to develop the first section. This algorithm takes an array of pitches as an argument, treating them as an ordered set, while other arguments specify rhythmic (onset) density, pitch range, note durations, and number of generated parts. These parameters can be specified as envelopes (values changing over time) or ranges of possible values, also changing over time (tendency masks). The algorithm takes the row and randomly chooses its transposition and form (prime, retrograde, inverted and inverted retrograde). Then, the pitch range at a given point in time is obtained and the proper octave for the next pitch of the row is chosen, with the appropriate note duration and appropriate position in the measure, according to the rhythmic density value. Such procedure helps to directly compose the large scale musical development.

The musical material was generated in sections, appropriately to changes in the development. The assignment of the generated parts to the particular instruments was done manually in the process of transferring the material to the music notation software. Similarly, in this process dynamic and articulation markings were added to the score. At times, the material was manually reduced, imposing another layer of musical development on top of the generated structures.

4.5.3 *Rhythmic density in section 2*

The process of composing the second section of the piece was done employing a similar algorithm to the one used in the first section. This algorithm also enabled control over pitch and rhythmic material over time. The difference with the first section is that the pitch structure was not provided. Instead, the range of the pitches to use was indicated.

4.5.4 *Rhythmic density in section 4*

The last section of the piece required yet another algorithm. The harmonic layer is based on five precomposed chords, which are being repeated. The pitches of a given chord are distributed among all the instruments. While the first section was employing a serial procedure, where each subsequent pitch for every instrument was taken from the same instance of the row, in the last section the harmonic structure was provided instead, so the pitches needed to be chosen from a set, appropriate for the current chord in the progression. Moreover, the pace of advancing between chords was a separate parameter, independent from the rhythmic density.

A recognizable gesture in this section is a burst of rhythmic intensity, occurring numerous times. The algorithm enabled to control at which point or how often the bursts would be created, while the harmonic pace was maintained at a constant level.

4.6 Preparation of the improvised sections

4.6.1 *Individual rehearsals*

The third section of the piece (rehearsal letters E through M) features individual solos, focused on the exploration of the sonic capabilities of the real-time computer sound processing, controlled through the accelerometer-based system. In this section, the performers are asked to creatively develop the musical material in an improvisatory manner.

Figure 14: Raymond Larsen preparing his improvisation with the accelerometer

Individual work with every performer is crucial for developing these improvisations. The main task is to acquaint them with the sonic qualities and the control ergonomics of the live electronics system. For this reason, it is desirable to allow each performer to explore the system individually. This was possible for the second performance of the piece: I held individual meetings with each player outside of the ensemble rehearsal time, which allowed them to practice with the system. During this process, the mappings between movement and the sonic result were fine-tuned. More information about the possible adjustments of the system can be found in the appendix.

For these sessions, a portable setup was used, consisting of a computer with an audio interface, one microphone, two pairs of headphones, one sensor and a WiFi access point. This enabled meeting with the musicians outside of the regular rehearsal space.

Holding individual meetings with the performers was possible since they all are a part of the community of musicians in Seattle I have been working with for the last few years. For performances in the near future, I would like to continue this approach and meet with the musicians individually to work on the improvised sections. Eventually, I would like to refine the vocabulary of the instruments accompanied by motion-controlled sound processing to the point where I will be able to clearly notate particular phrases to be used in the improvisation, while still leaving the performers a considerable amount of freedom.

4.6.2 *Source material suggestions*

The starting point for improvisation are the figures indicated in the score. These figures are extracted from other parts of the piece. Each figure may be used as a source phrase for further development, however the resulting improvisation is by no means bound to using only this material. Furthermore, a capable improviser might rely very little on these notated structures, however employing them is desirable in order to support the integrity with the rest of the piece. I would like to point out, that extensive exploration of the live electronics system is in the focus of the improvised sections. This approach favors greater performers' freedom and I prefer it over the extensive use of the notated structures.

4.6.3 *Phrases for wind instruments*

Flute, bass clarinet, and trumpet (oboe), have two processes available to them: a pitch shifter and a real-time granulator. The control mechanism is generally set up as follows: when the performer's hand wearing the accelerometer is oriented horizontally, tilting it left or right changes the pitch, both for the pitch shifter, as well as the granular process. Tilting the hand forward and backward transitions between pitch shifter and the granular process. The particular angular positions of the hand slightly differs for each instrument, in order to enable the most natural control when playing.

Pitch shifting is very effective on long notes and in slow phrases, while granular process fits well with faster and more abrupt ones. The player can use the tilt of the hand to extend their playing with appropriate process.

There's also one characteristic of the granular process that enables using it independently from the pitch shifter. While the latter works only while the sound is being played, the former keeps in memory the sound that was most recently played (up to two seconds). This allows controlling the granular process while not actively playing the instrument; moreover, short sounds, played while the granulator is producing sound, enable gradual change of the generated texture.

4.6.4 *Phrases for string instruments*

For violin and violoncello, only the granulating process is available. The choice of using just one process was dictated by the availability of control gestures that were the most natural for the players. In the case of these instruments, tilting the left hand is not feasible, and the right hand is moving constantly with the bow, making the tilt recognition ineffective. Since the right hand is already moving considerably, it was determined that measuring the amplitude of its movement could be used as a control signal.

The magnitude of changes of the bow direction is roughly mapped to the rate of grain generation and also to the pitch range of the generated sound. This enables creating gestures that are accelerating both in terms of acoustic sound, as well as the processed sound. Similarly to the wind instruments, the granular process is working constantly, not only when the instrument is playing. The possible uses of this feature include starting playing a phrase and then continuing the bow movement in the air (without touching the instrument), which will cause the processed sound to continue. Another example takes advantage of the ability to gradually replace the process' memory: after playing a phrase with the bow (arco), the player may introduce single pizzicato notes, separated by moving the bow in the air. This will result in the electronic sound gradually changing from the bowed to the pizzicato sound.

4.6.5 *Phrases for piano*

The piano sound is also extended with the granular process only. In this case, however, the control is tilt-based, and not movement rate-based. The sensor is adjusted to react to changes gradually, which enables the pianist to have the wrist position change the intensity of the granular process while playing the instrument. Tilting the hand front and back (or the wrist up and down) while having the hand on the keyboard, controls the rate at which the grains are being generated. Tilting the hand right and left changes the pitch of the generated cloud of sound. As described previously, short sounds can be used to gradually change the timbre of the generated cloud of sound.

4.6.6 *Phrases for percussion*

The percussion has both the pitch shifter and the granulator available for extending its part. The pitch shifter works very much like with the wind instruments: tilting the hand changes the pitch of the sound. This is particularly useful for modulating the decay portions of the cymbal and crotales.

The granular process is controlled by the magnitude of the motion of the hand with the mallet. This way, a tremolo or an increasingly fast phrase will be supported by an increasingly dense granulated sound.

4.7 Notes on the preparation of the performance

4.7.1 *Advancing of sections for the live electronics system*

The live electronics system is organized in so-called processing modules. Each module performs functions like:

- recording or playback of sound,
- pitch shifting controlled by a defined sensor,
- real-time granulation controlled by a defined sensor, and
- sustaining sound with the granulation effect.

Throughout the piece, these modules are turned on and off in appropriate groups. These groups are being organized into numbered cues, also called events, triggered throughout the piece. The system needs a person to advance from one event to the next in the appropriate places in the piece, indicated with a number in the score in the live electronics part.

The implementation of these processing modules uses the *ProcMod* extension for SuperCollider, created by Joshua Parmenter as a part of his research for Richard Karpen at the

Center for Digital Arts and Experimental Media at the University of Washington [36]. *ProcMod* facilitates control over large-scale musical gestures, providing global amplitude envelopes and efficient management of the computer's resources, as well as organization of groups of processes into events.

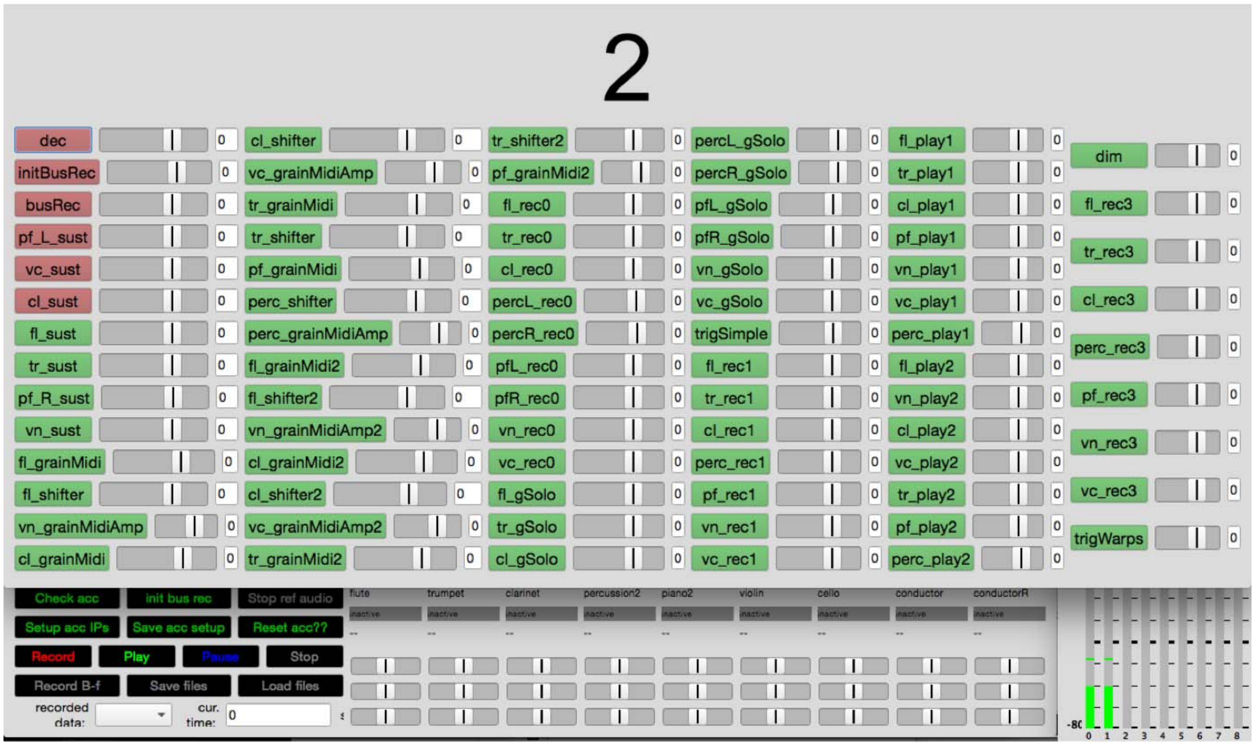


Figure 15: The Graphical User Interface of the system created for Deep Decline, with the current event number indicated on top.

The cueing may be performed by a dedicated person controlling the system, or by the conductor. Advancing to the next cue can be triggered by pressing a MIDI pedal, or pressing a button embedded in the accelerometer device of the conductor. In the case the conductor performs the cueing, a MIDI pedal might be provided as a backup in case of wireless system failure. Visual feedback for the conductor is also needed to display the current event number. This has been realized during the performance with the aid of a Virtual Network Computing (VNC) screen sharing service running on a second computer positioned on stage. This computer displayed the screen of the main computer running the performance software, which was positioned in the hall, next to the mixing desk (see the photograph from the performance at the Chapel Performance Space, Figure 18 on page 50).

4.7.2 *Large-scale musical gestures*

One of the features of the piece is the gradual development of the musical material over a longer period of time. Such process is quite apparent in the very first section: the rhythmic intensity increases gradually up to rehearsal letter A. This development starts around measure 19, where all the instruments (except the percussion) contribute to the musical texture. Measures 29 through 35 momentarily disassemble this process, but it resumes right after, leading up to the climax in measure 40. The directionality of this process, as well as the temporary nature of the structure's breakdown in measure 35 needs to be maintained by the performers.

Another gesture that spans over a long period of time can be found between measure 57 and 116. First, the rhythm gradually accelerates until measure 86. Afterwards, the pitch structure gradually changes in register. After measure 116, the rhythm decelerates until the next section which is accompanied by the diminishing dynamics.

In the last segment of the piece, from measure 155 until the end, the melo-rhythmic intensity and pace are maintained at a similar level. The important elements are the bursts of activity occurring in all the instruments at the same time (for examples in measures 166, 171-172, 184-186). These structures should be clearly exposed, contrasting with the rest of the music in that section. An example of this is presented in Figure 12 on page 28.

4.7.3 *Transition between recording and playing in section 4*

Starting at measure 155, all the instruments are being recorded. This recording starts to be played back in measure 193. It is important, that both cues - the one for the recording and the one for the playback - are triggered at the same place in relation to the beginning of the phrase (which is at the beginning of measures 155 and 193, respectively). Moreover, from measure 193 on, when the sound is starting to be played back, all musicians need to keep simulating the performance. Their movement however might need to be slightly more elaborate to ensure that the sound playback is maintained. The diminuendo, indicated in the score, should be as gradual as possible. In the last few measures before ceasing to make sound (while keeping the physical appearance of playing), performers might start omitting single notes to facilitate even a more gradual transition to the illusion of performing. This illusion lasts for about 20 seconds until all the musicians stop the movement (one measure before rehearsal letter Q) and playback "winds down", breaking the illusion that was maintained up to this point.

4.7.4 *Live sound realization during the performance*

All the instruments have a microphone next to them, primarily facilitating real-time sound processing. The processed sound is carried through the main pair of speakers in stereo configuration, positioned in line or downstage from the ensemble. Additionally, an additional pair of speakers should be used positioned behind the ensemble, possibly on shorter stands or set directly on the floor of the stage. All computer-generated sounds up to measure 155 (rehearsal letter N) should be directed to the main pair of speakers. After measure 155, the sound from the computer should be directed to the pair of speakers behind the ensemble. There is no processed sound being played back between measures 160 and 190, which gives time for the sound engineer to adjust the mixer settings as needed.

The signal from the microphones should also be routed through the mixing console. This enables slight amplification of the acoustic sound, which should be employed in a very subtle manner. The main reason for amplifying the acoustic signal is to bring the acoustic and processed sound timbrally closer together. Moreover in section 4, where the sound is recorded and played back, it is important to ensure that the timbre of the amplified instruments is sonically and spatially as close as possible to the sound that is being played back.

The sound engineer needs to overlook the levels of sound, making sure that the balance between acoustic and processed sound is maintained. One place where the engineer needs to actively adjust the levels is the transition from the acoustic to the recorded sound, approximately from measure 193 on. As noted before, the recorded sound should be directed to the speakers behind the ensemble. Moreover, the engineer should support the gradual transition between the acoustic and recorded sound by slowly fading in the signal fed to the speakers after measure 193. The transition from acoustic to recorded sound should last about 30-40 seconds.

CHAPTER 5. PERFORMANCE NOTES

5.1 Premiere performance at Meany Theater (October 2015)

5.1.1 *Performers and the space*

The premiere took place on October 30, 2015 in Meany Theater in Seattle. The concert was part of the Music of Today series, dedicated to performances of contemporary music. The piece was performed by the Chicago-based contemporary music collective Ensemble Dal Niente: Anne Donaldson (violin), Peter Ferry (percussion), Emma Hospelhorn (flute), Mabel Kwan (piano), Michael Lewanski (conductor), Andrew Nogal (oboe), Katherine Schoepflin (clarinet) and Christopher Wild (violoncello). The full program of the concert was as follows:

- Deep Decline (2015) by Marcin Pączkowski (b. 1983)
- Triplum (1995) by Franco Donatoni (1927-2000)
- Dmaathen (1976) by Iannis Xenakis (1922-2001)
- Un feu distinct (1991) by Joël-François Durand (b. 1954)
- Apophenia (2014) by Huck Hodge (b. 1977)

Figure 16: Ensemble Dal Niente performing Deep Decline

Meany Theater is a multipurpose venue located on the University of Washington campus. It has a capacity of 1,206 seats. It hosts a variety of events throughout the year, including international performers through the UW World Series program, as well as performances from the School of Drama, School of Music, Dance Program and the Center for Digital Arts and Experimental Media.

5.1.2 *Technical setup*

For the premiere performance, I was controlling the live electronics system and the Midas PRO3 mixing board, so that the conductor did not have to dedicate his attention to the computer system. The computer with the audio interface was located in the middle of the hall, for optimal mixing position.

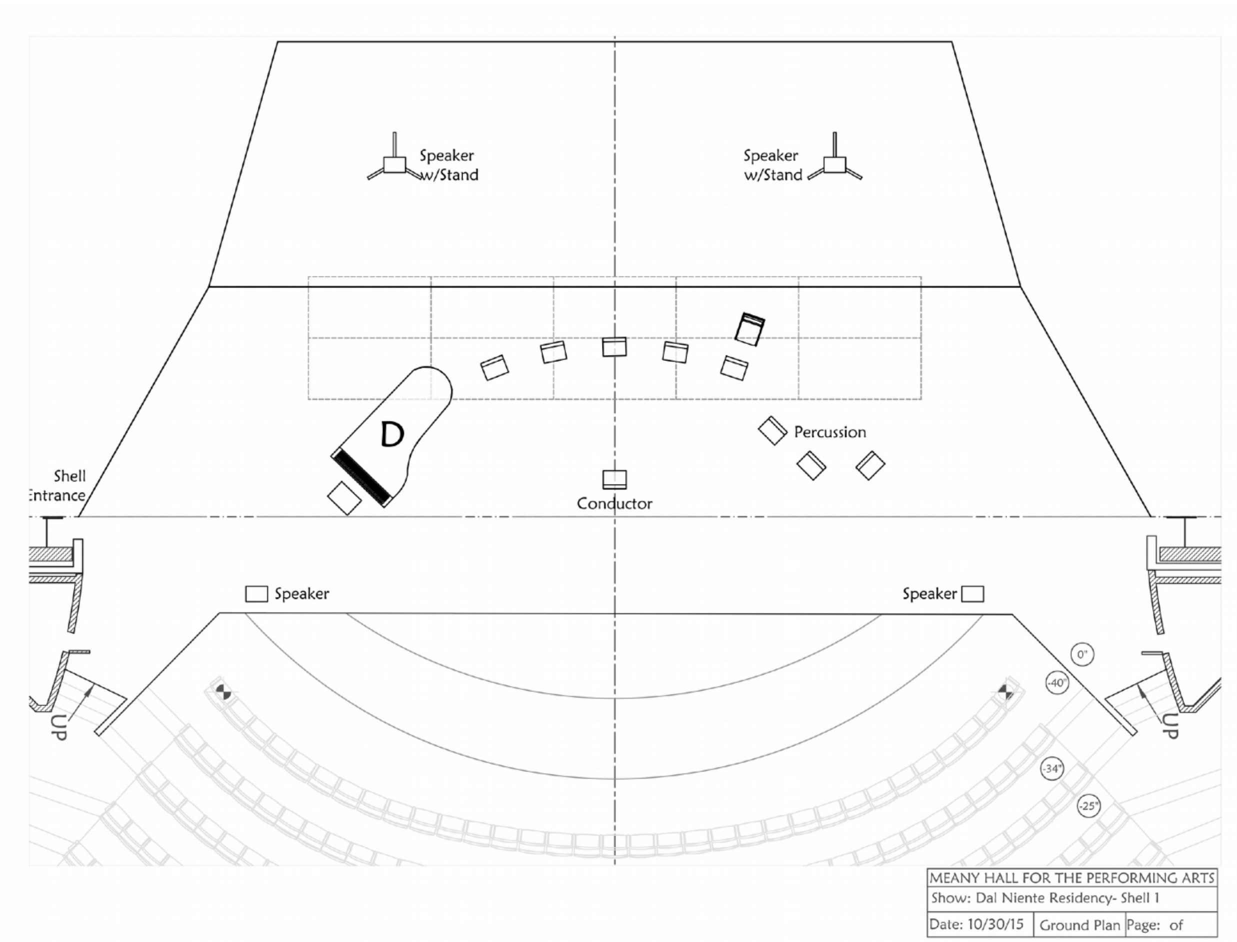


Figure 17: Preliminary stage plot for the performance at Meany Theater

All the instruments were equipped with a microphone. Piano and percussion were using each a regular microphone on a stand, gathering sound from the whole instrument. Violin and violoncello each had a small microphone attached to the instrument. Clarinet and oboe used a lavalier-style microphone attached to the performers' clothes, and the flutist was wearing a headset microphone. All signals were routed to the mixing board located in the back of the hall.

Signal from the microphones was routed through direct outputs from the mixing console to the table in the middle of the hall. Moreover, a wireless controller, using Midas Mixtender application running on Apple iPad, was provided to enable control over mixing parameters from this location.

Two sets of speakers were used: the main pair downstage and an additional pair upstage, behind the ensemble. Mixer settings enabled independent control over the balance between sound coming out of both pairs of speakers.

The WiFi router and access point for the accelerometer system was placed on the edge of the stage, in order to minimize radio interference. An ethernet cable connected the router to the station in the middle of the hall.

5.1.3 *Rehearsal process*

The Ensemble held a week-long residency at the University of Washington. Throughout that time, a total of four rehearsals (including the dress rehearsal) was scheduled. The first rehearsal was devoted largely to acquaint the ensemble with the system and to preliminarily tune the control parameters of the accelerometer system. During the first session the ensemble also worked on the acoustic parts of the piece.

During the second and the third rehearsals, further refinements were introduced in the control of the live electronics part. Individual rehearsals with the performers were not planned during this residency, however part of the time was dedicated to individual work on the improvised sections during regular rehearsals.

The accelerometers worked properly throughout the first three rehearsals, although they refused to connect to the WiFi router at the dress rehearsal, on the day of the concert. This was finally solved by changing the settings of the router to serve the dedicated network on a different channel. After solving this problem, the performance did not encounter other technical obstacles.

5.2 Performance at the composer's recital at the Chapel Performance Space (May 2016)

5.2.1 *Performers and the space*

The second performance took place at the Chapel Performance Space, located in the Good Shepherd Center in Seattle. It was part of the ongoing Wayward Music series, dedicated to contemporary and experimental music. The piece was performed by Natalie Ham (flute), Ivan Arteaga (clarinet), Raymond Larsen (trumpet), Andrew Angell (percussion), Josh Archibald-Seiffer (piano), Luke Fitzpatrick (violin), David Balatero (violoncello), Marcin Pączkowski (conductor)

Figure 18: Performance of Deep Decline at the Chapel Performance Space (film still by Adam Hogan)

The concert featured solely my own music. The full program included:

- ...where odd things are kept... (2014)
- [no title] (improvisation with Ivan Arteaga) (2016)
- Percussivometers (2014)
- Deep Decline (2015/2016)

The Chapel Performance Space is booked through Nonsequitur, an organization dedicated to promoting “adventurous music”. It is a vital part of Seattle’s new music and modern jazz scene, presenting about 10 nights a month of concerts of non-commercial music.

5.2.2 *Technical setup*

For this performance I was personally conducting the ensemble, as well as triggering the events of the live electronics system.

The main computer with the audio interface was positioned in the hall, next to the mixing console. There was another computer positioned on stage, which was displaying the screen from the main machine through a Virtual Network Computing screen sharing service.

All the instruments had a microphone on a stand positioned next to them. The signal from the microphones was sent to the mixing console, and then forwarded through direct outputs to the audio interface. Michael McCrea was controlling the sound levels at the board.

Two sets of speakers were used: a main set next to the ensemble on stage, and a second set behind the ensemble. The balance between them was controlled at the mixer. As indicated in the previous chapters, the sound in the fourth section of the piece was directed to the speakers behind the ensemble.

The WiFi router was again placed on stage, with wired ethernet connection running to the main computer. The sensor system worked flawlessly throughout the performance.

5.2.3 *Rehearsal process*

For this performance, it was feasible to work with the musicians independently, which was beneficial for developing further the improvised sections. Three full rehearsals were scheduled (including the dress rehearsal). On top of that, I have met with each player for a session dedicated to working on the improvised solos.

The first rehearsal was dedicated to work on the ensemble parts of the piece, particularly from the beginning to rehearsal letter B, and then from rehearsal letter N onwards. The first and second rehearsals were separated by over a week, and the individual meetings with the performers happened during this time. The control over sound transformations was greatly improved over this process. The second ensemble rehearsal allowed the performance of all of the improvised parts seamlessly throughout the piece.

The dress rehearsal was focused on shaping the overall form. Few technical glitches were addressed, as well as the issue of balance between acoustic and recorded/played back sound in rehearsal letter P.

CHAPTER 6. CONCLUSION

6.1 Current state of the project

Deep Decline is ready for future performances. The project in its current state provides an infrastructure for expressive exploration of the accelerometer-based system, while the computer-assisted composition framework yielded satisfying musical results.

The working prototype of the sensor system works reliably. The technique of playing with it is being constantly developed, however the experiments undertaken in the process of preparation of the two performances resulted in the emergence of a vocabulary for the sensor-extended instruments.

Two families of processes were used extensively: a sound granulator and a pitch shifter. The mappings between instrumentalists' movement and sound transformations enabled players to perform intuitively and expressively, maintaining the ergonomy of playing their instrument.

A new framework for computer-assistted composition developed for *Deep Decline,* provided flexibility to generate a variety of materials. The algorithms are easily extensible to realize my current and future musical goals[10].

6.2 Future work

I intend to work further with the technologies developed so far. The first goal is to further develop the hardware prototype, so it can be sent out and easily used during other performances of *Deep Decline*, possibly without the presence of the composer. In order to do so, some durability improvements need to be made, as well as more testing in varied environments to ensure reliable radio transmission. The possibility of using embedded computers for sound processing would possibly make the whole system even more portable and reliable to be sent off to the performers.

The immediate continuation of my work with accelerometers alone will be to use the Motion Processing Units, already present in the hardware, for more refined motion analysis. This would involve calculating real-world acceleration values, independent from the orientation of the

[10]The current framework for computer-assisted composition is an extension of the system used to compose *Tide.*

device, as well as tilt information, independent from linear acceleration. Such separation would provide even more refined control over the musical gestures.

Independently from the hardware improvements, I'm already exploring the usability of the accelerometer system in the context of improvised music. This exploration will continue, with the goal of developing an expressive gestural instrument.

I would also like to further pursue the idea of illusion, not only by using sensors and sound processing, but also by introducing physical actuators to the system. Such actuators would enable embedding the sounds of the physical space in the performance, as well as creating mechatronic sound-generating counterparts to the acoustic instruments.

Deep Decline wouldn't have its current shape without collaboration with the performers, without their feedback, and their willingness to try a variety of performing strategies. Such process not only boosts creativity, but helps me to get a new perspective on my musical ideas. I intend to continue developing future works in such collaborative creative setting.

BIBLIOGRAPHY

[1] Berlioz, Hector, *Berlioz's orchestration treatise: a translation and commentary*. Cambridge ; New York: Cambridge University Press, 2002.

[2] "The Mathews Radio Baton." [Online]. Available: https://ccrma.stanford.edu/radiobaton/. [Accessed: 21-Jun-2016].

[3] M. V. Mathews, "The Radio Baton and Conductor Program, or: Pitch, the Most Important and Least Expressive Part of Music," *Comput. Music J.*, vol. 15, no. 4, pp. 37–46, 1991.

[4] D. A. Jaffe and W. A. Schloss, "The Computer-Extended Ensemble," *Comput. Music J.*, vol. 18, no. 2, pp. 78–86, 1994.

[5] F. Lopez-Lezcano, "PadMaster: Banging on Algorithms with Alternative Controllers," *Int. Comput. Music Conf. Proc.*, vol. 1996, 1996 1996.

[6] R. Orton and H. Davies, "Theremin," *Grove Music Online*.

[7] J. Chadabe, "Interactive Composing: An Overview," *Comput. Music J.*, vol. 8, no. 1, pp. 22–27, 1984.

[8] B. Boie, M. V. Mathews, and W. A. Schloss, "The Radio Drum as a Synthesizer Controller," *Int. Comput. Music Conf. Proc.*, vol. 1989, 1989 1989.

[9] "Accelerometer - Oxford Reference." [Online]. Available: http://www.oxfordreference.com/view/10.1093/acref/9780199534463.001.0001/acref-9780199534463-e-0039. [Accessed: 03-Jul-2016].

[10] D. Vaughan, *Merce Cunningham: Creative Elements*. Routledge, 2013.

[11] E. Dykstra-Erickson and J. Arnowitz, "Michel Waisvisz: The Man and the Hands," *interactions*, vol. 12, no. 5, pp. 63–67, Sep. 2005.

[12] Ernest T. (Ernest Thomas) Ferand, *Improvisation in nine centuries of western music; an anthology with a historical introduction.* Köln, A. Volk Verlag, 1961.

[13] K. Levy, J. A. Emerson, J. Bellingham, D. Hiley, and B. M. Zon, "Plainchant," *Grove Music Online*.

[14] M. J. Bloxam, "Cantus firmus," *Grove Music Online*.

[15] P. Williams and D. Ledbetter, "Continuo," *Grove Music Online*.

[16] F. Dawes, K. A. Hagberg, and S. D. Lindeman, "Steibelt, Daniel," *Grove Music Online*.

[17] B. Nettl, R. C. Wegman, I. Horsley, M. Collins, S. A. Carter, G. Garden, R. E. Seletsky, R. D. Levin, W. Crutchfield, J. Rink, P. Griffiths, and B. Kernfeld, "Improvisation," *Grove Music Online*.

[18] "Solo/Tutti: Variations on an Irrational Number (2002)," *Richard Karpen*, 17-Feb-2015. [Online]. Available: https://richardkarpen.com/solotutti-variations-on-an-irrational-number-2002/. [Accessed: 23-Jun-2016].

[19] H. Taube, *Notes from the metalevel: introduction to algorithmic music composition*. London ; New York: Taylor & Francis Group, 2004.

[20] C. Roads, *The computer music tutorial*. Cambridge, Mass: MIT Press, 1996.

[21] P. Griffiths, "Serialism," *Grove Music Online*.

[22] J. Pritchett, L. Kuhn, and C. H. Garrett, "Cage, John," *Grove Music Online*.

[23] Wolfgang Amadeus Mozart, *Musikalisches Würfelspiel: eine Anleitung "Walzer oder Schleifer mit zwei Würfeln zu componieren ohne Musikalisch zu seyn, noch von der Composition etwas zu verstehen*. Mainz: New York, 199AD.

[24] G. Assayag, C. Rueda, M. Laurson, C. Agon, and O. Delerue, "Computer-Assisted Composition at IRCAM: From PatchWork to OpenMusic," *Comput. Music J.*, vol. 23, no. 3, pp. 59–72, 1999.

[25] "SuperCollider » SuperCollider." [Online]. Available: http://supercollider.github.io/. [Accessed: 25-Jul-2016].

[26] "supercollider-quarks/Ctk," *GitHub*. [Online]. Available: https://github.com/supercollider-quarks/Ctk. [Accessed: 23-Jul-2016].

[27] M. Paczkowski, *Still, early*. Thesis --University of Washington, 2011.

[28] P. Fellgett, "Ambisonics. Part one: general system description," *Studio Sound*, pp. 20–22, 40, Aug-1975.

[29] J. Pampin, "ATS." [Online]. Available: https://ccrma.stanford.edu/~juan/ATS_manual.html. [Accessed: 22-Jul-2016].

[30] C. Dromey, "Pierrot ensemble," *Grove Music Online*.

[31] "Aperture (2006)," *Richard Karpen*, 17-Feb-2015. [Online]. Available: https://richardkarpen.com/aperture-2006/. [Accessed: 25-Jul-2016].

[32] "Strand Lines (2007)," *Richard Karpen*, 17-Feb-2015. [Online]. Available: https://richardkarpen.com/strand-lines-2007/. [Accessed: 23-Jun-2016].

[33] K. Lent, “An Efficient Method for Pitch Shifting Digitally Sampled Sounds,” *Comput. Music J.*, vol. 13, no. 4, pp. 65–71, 1989.

[34] “supercollider-quarks/wslib,” *GitHub*. [Online]. Available: https://github.com/supercollider-quarks/wslib. [Accessed: 24-Jul-2016].

[35] “Sibelius | Avid.” [Online]. Available: http://www.avid.com/sibelius. [Accessed: 19-Jun-2016].

[36] J. Parmenter, “Dynamic large-scale Gestural control in supercollider server,” *Int. Comput. Music Conf. ICMC 2007*, pp. 129–132, 2007.

[37] “Teensy LC (Low Cost).” [Online]. Available: https://www.pjrc.com/teensy/teensyLC.html. [Accessed: 25-Jul-2016].

[38] “Download » SuperCollider.” [Online]. Available: http://supercollider.github.io/download.html. [Accessed: 10-Jul-2016].

[39] “supercollider/sc3-plugins,” *GitHub*. [Online]. Available: https://github.com/supercollider/sc3-plugins. [Accessed: 10-Jul-2016].

[40] “www.dd-wrt.com | Unleash Your Router.” [Online]. Available: http://www.dd-wrt.com/site/index. [Accessed: 25-Jul-2016].

[41] “Adafruit HUZZAH ESP8266 Breakout ID: 2471 - $9.95 : Adafruit Industries, Unique & fun DIY electronics and kits.” [Online]. Available: https://www.adafruit.com/product/2471. [Accessed: 21-Jul-2016].

[42] “Espressif Systems - Wi-Fi and Bluetooth chipsets and solutions.” [Online]. Available: https://espressif.com/en. [Accessed: 25-Jul-2016].

[43] “MPU-6050 | InvenSense.” [Online]. Available: https://www.invensense.com/products/motion-tracking/6-axis/mpu-6050/. [Accessed: 21-Jul-2016].

[44] “Introduction to OSC | opensoundcontrol.org.” [Online]. Available: http://opensoundcontrol.org/introduction-osc. [Accessed: 19-Jul-2016].

[45] “esp8266/Arduino,” *GitHub*. [Online]. Available: https://github.com/esp8266/Arduino. [Accessed: 11-Jul-2016].

[46] “jrowberg/i2cdevlib,” *GitHub*. [Online]. Available: https://github.com/jrowberg/i2cdevlib. [Accessed: 11-Jul-2016].

APPENDIX A: TECHNICAL DOCUMENTATION

A.1 HARDWARE AND COMPUTER CONFIGURATION

A.1.1 *Overview*

Performance of *Deep Decline* requires the following elements:

- personal computer running SuperCollider software
- an audio interface with at least seven inputs and two outputs
- 9 custom accelerometer-based sensors
- WiFi router and access point
- a footswitch with a MIDI interface (optional)

A.1.2 *Computer and operating system*

The personal computer used in the two performances of the piece to date was an Apple Macbook Pro 13", model 9.2, also referenced as model "mid-2012". This computer was equipped with a 3rd generation Intel Core i7 processor clocked at 2.9GHz, as well as 16GB of memory (RAM). The operating system was Mac OS X, version 10.10.5.

It is likely that the performance would be possible on a somewhat slower computer as well, however the lower limits of such configuration haven't been tested. Due to the open source nature of SuperCollider, it should also be possible to run the code under the Linux operating system, likely requiring some minor adjustments. This however has not been tested.

A.1.3 *Audio interface*

The audio interface used for the performance was a RME Fireface 800. This hardware provided stable performance with a relatively small audio buffer size (128 samples) and thus reasonably small processing latency. The sampling rate was set to 48kHz. All the discrete input signals were sent from the direct outputs of a mixing console to the line inputs of the audio interface.

A.1.4 *MIDI pedal*

One of the ways of advancing the events in the piece is by pressing a MIDI pedal. For this purpose any device visible to the computer as a MIDI interface can be used, as long as it sends Control Change message number 64 (which is assigned to sustain pedal in the MIDI specification).

I used a custom-built device based on a microcontroller Teensy-LC [37]. This board is capable of simulating a USB MIDI interface. The footswitch pedal was connected between one of its pins and the ground. I wrote a simple program triggering the MIDI message whenever the connection between the pin and the ground was made, which happened when the pedal was depressed.

A.2 SuperCollider environment

The piece was realized with SuperCollider [38] versions 3.7.2 (first performance) and 3.8 beta (second performance). In order to recreate the performance, a working installation of SuperCollider is necessary, and extended with the following elements:

- Synthesis Plugins and Extensions, called SC3-plugins [39],
- a set of custom classes (provided with the code attached to this dissertation),
- a selection of language extensions (Quarks).

Both the SC3-plugins, as well as the custom classes, should be copied to an appropriate Extensions folder, particular to the given operating system. The proper path can be found using the command

```
Platform.userExtensionDir
```

executed in SuperCollider. The indicated folder might need to be created manually if it doesn't exist.

The custom classes are provided in one package with the main code and include the following:

- BufferedControl.sc - a wrapper enabling recording of the control signals
- ControlPlotter.sc - a class developed by Michael McCrea for plotting control signals, used for monitoring accelerometer signals

- ControlProcessor.sc - a class enabling flexible mapping of values for the process modules
- Esp8266server.sc - a class mainaining communication with the accelerometers
- PSGPitchShift - a pitch shifter
- Quantize.sc - a pseudo-UGen, enabling quantization of control signals

The language extensions - Quarks - can be installed using a graphical interface from within SuperCollider. The interface is invoked by executing

```
Quarks.gui
```

The following Quarks should be activated:

- Ctk
- NetLib
- SenseWorld
- Tendency
- XML

After installing the Quarks, the class library should be recompiled. This can be done with the button provided in the graphical interface invoked by the Quarks.gui command.

A.3 Accelerometer system

The accelerometer system consists of custom wearable sensors that use a wireless computer network (WiFi) for communication. Figure 19 presents the prototype device, where the serial port is located on top to the left, and the charging cable is on the right. In addition to the sensors themselves, a dedicated WiFi router is required for reliable transmission. To minimize packet loss, the router should be connected to the computer using a wired (Ethernet) connection. The wired and wireless networks should be using the same IP address space. The network needs to operate on the 2.4GHz band.

The name of the network (SSID), as well as the password, need to be provided to the sensors. These settings are stored in the memory even when the batteries are disconnected. If changes need to be made to the wireless network name or password, they can be transmitted to the sensors using a serial interface.

The sensors need to be provided with the IP address and port number of the computer running the SuperCollider code. This information is also stored between device resets, however it might change due to network configuration. For this reason, it is best to use a router that keeps track of the assigned IP addresses to hardware identifiers and consistently assigns the same IPs even after rebooting the router. The device used in this project was Linksys E2000, running dd-wrt firmware [40]. It did store in its memory the assigned IP addresses, always allocating the same address to the given device.

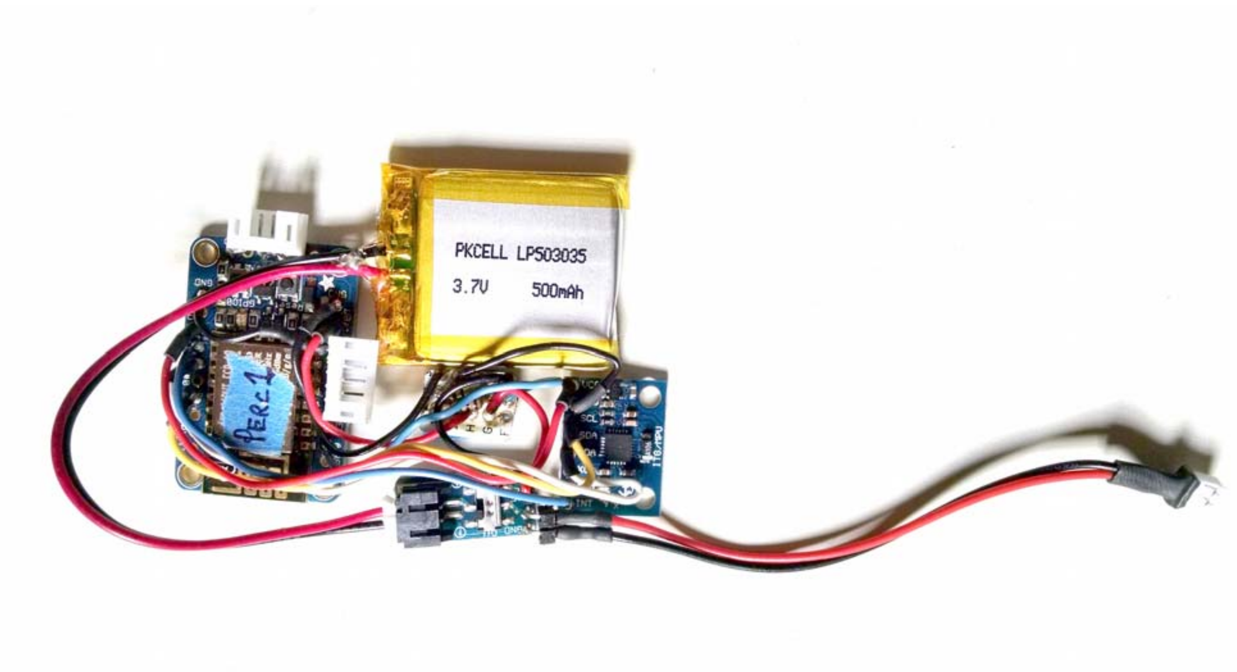


Figure 19: The prototype accelerometer device.

After starting the main code in SuperCollider, the IP addresses of the sensors may be set by pressing the “Set acc IPs” button on the user interface on the screen. This might need to be performed numerous times for the information to reach all the sensors. This button triggers the “setupAccelerometers” function in the code (line ~180). This function has parameters of the lowest and highest IP address number to send the setup information to. This should correspond to the DHCP server settings in the WiFi router. Additionally, networkDevice needs to be specified, as a string (name) reported by the ifconfig utility: “en0” is the wired network device on Apple’s Macbook Pro with a built-in ethernet port.

In case the sensors fail to connect to the WiFi network (the blue light on the board does not come on), it is possible that the radio traffic on a given wireless channel is too heavy. This can be usually solved by adjusting the network’s channel in the WiFi router’s settings. The channels the least susceptible to interference are usually numbers 1, 6, and 11.

A.4 RUNNING THE PIECE

Once all the required extensions for SuperCollider are installed, the audio interface is connected and the WiFi network is set up, it is possible to proceed with starting the live electronics system. In order to run the piece, the file “deep decline.scd” needs to be opened in SuperCollider. Then, the main code block needs to be executed. In the SuperCollider IDE on Mac OS X this can be done by placing the cursor inside the code block and pressing Command-Return on the keyboard. This will run the code, start the SuperCollider server and display windows of the graphical user interface (GUI) (see Figure 15 on page 43).

The main window contains an indication of the accelerometer sensors state, as well buttons for providing them with the computer’s IP address (button “Setup acc IPs”). Ideally, this should only be required once for a given network setup.

The “practice” window, listing all process modules, provides information about processes that are currently running, as well as the current event number. The smaller window enables advancing to the next event by pressing the on-screen button.

During the performance, advancing to the next section of the piece is performed either by pressing a MIDI pedal, or by pressing a button positioned in the inner side of the conductor’s left accelerometer. The debounce mechanism prevents accidentally skipping over to the next section by allowing a maximum of one event change per second.

A.5 CODE REFERENCE

A.5.1 *File structure*

The file with the code for performing the piece is named “deep decline.scd”. This file can be moved to any writable location on the filesystem. In the same folder as this file, another writable folder needs to exist, named “recordings”. In this folder the automatically created recordings will be stored. The code reference provides information that might be useful in case some parameters of the piece need changing.

A.5.2 *Code structure and initial settings*

The main code is structured as follows:

- Variable declarations (beginning)
- Settings (line ~50)
    - This is the place to adjust parameters like audio input/output numbers, sample rate, low pass and high pass filtering for speakers, buffer size for audio processing and others
- Initialization (line ~100)
- Function definitions (line ~180)
- Starting the SuperCollider Server, and when done: (line ~635)
    - Synth definitions (SynthDefs) (line ~670)
    - Functions definitions (line ~1720)
    - Graphical user interface (GUI) code (line ~1910)
    - Process modules (ProcMods) template functions (line ~2180)
    - ProcEvents, containing instances of all ProcMods (line ~5895)
        - This is the place to modify parameters of the ProcMods, like mapping between sensor data and processing values
    - Cue advancing and MIDI pedal initialization (line ~6910)
- Below the main code block, delineated by the parentheses (line ~6940), the code for ControlProcessor instances used for adjusting control parameters during rehearsals can be found; adjusting parameters is described in the section dedicated to preparation of the improvisations

The parameters in the settings section can be changed to adjust the system to different hardware configurations. This is the list of chosen variable names, indicating value ranges:

- firstAudioOutput: number of the audio output of the audio interface for the left speaker; zero (0) is the first output
- audioInputArray: an array of input channel numbers, corresponding to the instruments in the following order: flute, oboe/trumpet, clarinet, percussion left, percussion right (may be the same number as percussion left when using a single microphone), piano left, piano

right (may be the same number as piano left when using a single microphone), violin, violoncello

- saveInputData: value should be true or false; when true, all the input channels and output signal (in Ambisonic B-format) will be recorded as a multichannel file, placed in the folder named “recordings” in the same folder as the “deep decline.scd” file
- decoderType: use \uhj for stereo setup
- sr: sample rate of the system; 48000 is the default
- hardwareBuffer: buffer size for audio processing; 128 samples was used providing reasonable latency and good stability
- accelerometerNamesSensors: an array of Symbol objects, containing names of the sensors worn by the musicians; for the order of instruments, refer to the array in the accelerometerNames variable

A.5.3 *Parameters of the processing modules*

The parameters of the most significant modules are described, in order to facilitate further refinement of the processes.

The main granular process is defined in the function stored at

```
procs[\grainIn2midi]
```

It has following parameters:

- id: unique symbol identifying the process;
- amp: global amplitude, 0-1 (possibly higher; use .dbamp method to specify the value in decibels)
- audioInput: number of audio input, in relation to audioInputArray; usually defined as a value from the audioInputs array, like audioInputs[0]
- tFreq: an instance of ControlProcessor defining the frequency (rate) of grain triggering
- grainRateLo: an instance of ControlProcessor defining the lower limit of the grain playback rate (transposition), expressed in midiratio (0 = no transposition, 12 = one octave up, -12 = one octave down)
- grainRateHi: an instance of ControlProcessor defining the upper limit of the grain playback rate (transposition), expressed in midiratio (see grainRateLo)

- ampThreshold: an instance of ControlProcessor specifying the amplitude above which signal will be recorded into the internal buffer
- grainDur: an instance of ControlProcessor specifying grain duration in seconds
- azimuthsLo: an instance of ControlProcessor specifying the lower limit for azimuth values for the grains (in the ambisonic system); expressed in radians (use .degrad to use degrees); 0 is center straight ahead, 30.degrad is 30 degrees *left* (positive values move counterclockwise)
- azimuthsHi: an instance of ControlProcessor specifying the upper limit for azimuth value for the grains
- elevationsLo and elevationsHi: see azimuthsLo/Hi; 0 is center, 30.degrad is 30 degrees above (this value is not effective in the default stereo setup)
- recordGate: an instance of ControlProcessor allowing to block the recording of the new sounds when its value is 0; by default it is kept at a constant value of 1
- ampFollowerFallTime: if nil, the granulated sound will continue indefinitely; when a number is provided, the granulated sound will fade out for the specified time since the last sound occurred above the amplitude threshold
- rateQuantizeQuantum: an instance of ControlProcessor specifying quantum for pitch quantization of the grains; 1 will quantize grain playback rates (transpositions) to a semitone
- rateQuantizeTolerance: an instance of ControlProcessor specifying tolerance for pitch quantization of the grains; 1 will enable influencing the pitch within a semitone of the next multiple of the quantum
- rateQuantizeStrength[11]: an instance of ControlProcessor specifying quantize strength; usually set between 0.5 and 0.9
- bufLength: length of the internal buffer used for granulation; usually 2 seconds
- att: attack (fade in) time of the whole process
- rel: release (fade out) time of the whole process

[11]The quantizing algorithm works as follows: if the absolute difference between the input value rounded to the nearest multiple of quantum and the input value itself is smaller than the tolerance, then the said difference, multiplied by quantize strength, is added to the input signal; the code representation of this mechanism can be found in the *quantize* method of the SimpleNumber class

The main pitch shifting process is defined in the function stored at

`procs[\shifterFormantAmpMidi]`

Pitch shifter-specific parameters are as follows:[12]

- formantRatio: an instance of ControlProcessor specifying formant shift[13], in midiratio values
- pitchRatio: an instance of ControlProcessor specifying pitch shift, in midiratio values
- ampCtl: an instance of ControlProcessor specifying the amplitude of the process
- azimuth: an instance of ControlProcessor specifying the azimuth value for the processed sound in radians; negative values go counterclockwise (see notes for azimuthsLo of the previous process)
- Elevation: an instance of ControlProcessor specifying elevation value for the processed sound
- pfQuantizeQuantum: an instance of ControlProcessor specifying quantum for pitch quantization of the processed sound; 1 will quantize to a semitone
- pfQuantizeTolerance: an instance of ControlProcessor specifying tolerance for pitch quantization of the processed sound; 1 will enable influencing the pitch within a semitone of the next multiple of the quantum
- pfQuantizeStrength: an instance of ControlProcessor specifying quantize strength; usually set between 0.5 and 0.9

The strings and percussion instruments use a variation of the granular process, that depends on the rate of movement of the performer's hand. The process is defined in the function stored at

`procs[\grainIn2midiFreqCtl]`[14]

Grain creation rate is controlled by the movement amplitude of the accelerometer. The movement amplitude value is then scaled and offset. Similarly, grain playback rate

[12]Refer to the previous process for the common parameter names.

[13]Formant shift does not change the pitch of the sound; instead, the reinforcing of particular harmonics, characteristic for a given sound source, is shifted. This enables timbral changes, while maintaining the original pitch. Although in *Deep Decline* formant shifting is not used, this parameter is included since the underlying process allows for such manipulation.

[14]Refer to the regular granular process for the common parameter names.

(transposition) is controlled by multiplying and offsetting the value from the amplitude of movement. This can be expressed as:

```
<grainTriggerFrequencey> = amplitude(tFreqBus) * tFreqMul + tFreqAdd

<grainRateLo> = amplitude(tFreqBus) * grainRateLoMul + grainRateLoAdd

<grainRateHi> = amplitude(tFreqBus) * grainRateHiMul + grainRateHiAdd
```

The process' parameters are as follows:

- tFreqBus: bus number that carries the signal from the accelerometer; usually accessed through accDict dictionary, like: accDict[\violin].bufferedBus.bus + 2 (this means "z" axis of the violin accelerometer; each device outputs values into 3-channel bus, correspoding to the 3 axes: x, y, and z)
- tFreqMul: an instance of ControlProcessor specifying a multiplier for the amplitude follower value, influencing grain triggering frequency
- tFreqAdd: an instance of ControlProcessor specifying an offset for the amplitude follower values, influencing grain triggering frequency
- tFreqFollowerRel: an instance of ControlProcessor specifying the release time for the amplitude follower
- grainRateLoMul: an instance of ControlProcessor specifying a multiplier for the amplitude follower values, influencing the lower limit of grain playback rate (midiratio value)
- grainRateLoAdd: an instance of ControlProcessor specifying an offset for the amplitude follower values, influencing the lower limit of grain playback rate (midiratio value)
- grainRateHiMul: see grainRateLoMul, but influencing upper limit
- grainRateHiAdd: see grainRateLoAdd, but influencing upper limit

A.5.4 *Workflow for adjusting parameters on-the-fly*

For the improvised sections, where the performers control their sound processing with sensors, it is desirable to adjust the mappings between the parameters on the fly, during the rehearsal. For this purpose the class ControlProcessor was created.

ControlProcessor enables defining a parameter as one of the following types: a static value, a continuously changing value (an envelope), a tendency mask, or a mapping of a control bus signal with appropriate scaling. This last function was extensively used for mapping the parameters from the sensors.

The syntax for this class is as follows:

```
ControlProcessor.new(id, inArgs, type)
```

The “id” should be a symbol, that is later used for identifying an active ControlProcessor. In my code, this id is prepended with the particular process id the ControlProcessor is used in. In this case if the id of the ControlProcessor is set to “\tFreq” and it is used inside a process module with the id “\cl_play2”, the ControlProcessor’s id will be changed to “\cl_play2_tFreq”, which enables identifying it across all the processes.

The “inArgs” argument might be a static value, an Envelope, a Tendency, and finally an array of parameters. When it is an array, the “type” argument specifies the way of treating the values.

Currently, only the “\linlin” type is used. In this case, the inArgs array is expected to consist of: [input bus number, input min value, input max value, output min value, output max value, lag time, lag curve (number or shape symbol)].

In order to adjust the parameters on the fly, one needs to run the main code block, then advance to the appropriate section of the piece (event number) which involves improvisation on a given instrument. While the section is running, the following code should be executed (see line ~6945 in “deep decline.scd”):

```
"---all ControlProcessors---".postln;
ControlProcessor.allInstances.asSortedArray.do({|thisArr| thisArr[0].postln;});
"".postln;
```

This will list the names of all active ControlProcessors. Checking settings of a given instance can be invoked in the following manner:

```
ControlProcessor.allInstances[<id>].inArgs
```

for example

```
ControlProcessor.allInstances[\fl_grainMidi_tFreq].inArgs
```

Running this code, while the “fl_grainMidi” process is running, should return:

```
[ 1, -0.5, 1, 0, 200, 0.2 ]
```

Please note that the first number in the array is the control bus number. When adjusting the parameters, this number shall not be changed. This number might change after re-running the main code and invoking the .inArgs method again. In this case, while making the adjustments, the updated bus number should be used.

In the current example, the values following the bus number are: input min value, input max value, output min value, output max value, and lag time. In order to change any of them, the whole array needs to be passed to the "inArgs" argument:

```
ControlProcessor.allInstances[\fl_grainMidi_tFreq].inArgs_(
[1, -0.5, 1, 0, 200, 0.2]
)
```

The parameters above indicated that the input values between -0.5 and 1 will be scaled to the range between 0 and 200, with the lag time of 0.2s. To change the parameters, for example so that the input range stays the same but the output range is between 100 and 400 with the lag time of 0.5s, the ControlProcessor should be updated with the following values:

```
ControlProcessor.allInstances[\fl_grainMidi_tFreq].inArgs_(
[1, -0.5, 1, 100, 400,0.5]
)
```

After assigning new values to the ControlProcessor, the change will be reflected in the sound immediately. Once the desirable settings are achieved, the new parameters need to be manually transferred to the code. For this example, the appropriate place in the code is around line 6270. The current settings are:

```
tFreq: ControlProcessor(\tFreq, [
accDict[\flute].bufferedBus.bus + 1, -0.5, 1, 0, 200, 0.2
], \linlin)
```

In order to change the range of values, the second, third, fourth, fifth, sixth (and optionally seventh, if used) element of the array may be changed. Other values should be left unchanged. It is important to maintain the syntax structure when making changes to the code; changes in the placement of brackets, parentheses, and commas, will likely cause an error

A.6 ACCELEROMETER DEVICE DESIGN

A.6.1 *Overview*

The central part of the accelerometer system is the Adafruit Huzzah [41], a breakout board for the ESP8266 microcontroller manufactured by Espressif [42]. The accelerometer device is based on the MPU-6050 chip by InvenSense, which supports the I^2C interface for communication [43].

The ESP8266 microcontroller is programmed to read the accelerometer data and to send its values over the Open Sound Control protocol [44], running on the UDP transport layer over the wireless network.

A.6.2 *Code for the ESP8266-based board*

The Adafruit Huzzah board was programmed in the Arduino environment, with the aid of the Arduino core for ESP8266 project [45]. The interface to the MPU-6050 device uses Jeff Rowberg's library [46], which facilitates communication with the sensor. Only the raw accelerometer data is used currently, however it is planned to use the MPU-6050's embedded motion processing unit in the future.

The use of the Arduino environment should enable easy porting of the code to other platforms, in case the current hardware becomes unavailable in the future.

A.6.3 *Wiring of the ESP8266-based board and the MPU6050 accelerometer*

The wiring between the Adafruit Huzzah board and the MPU-6050 accelerometer consists of connecting power, 2 data lines, and an interrupt signal. On top of that, a power switch, and a voltage divider for measuring the voltage of the battery is included.

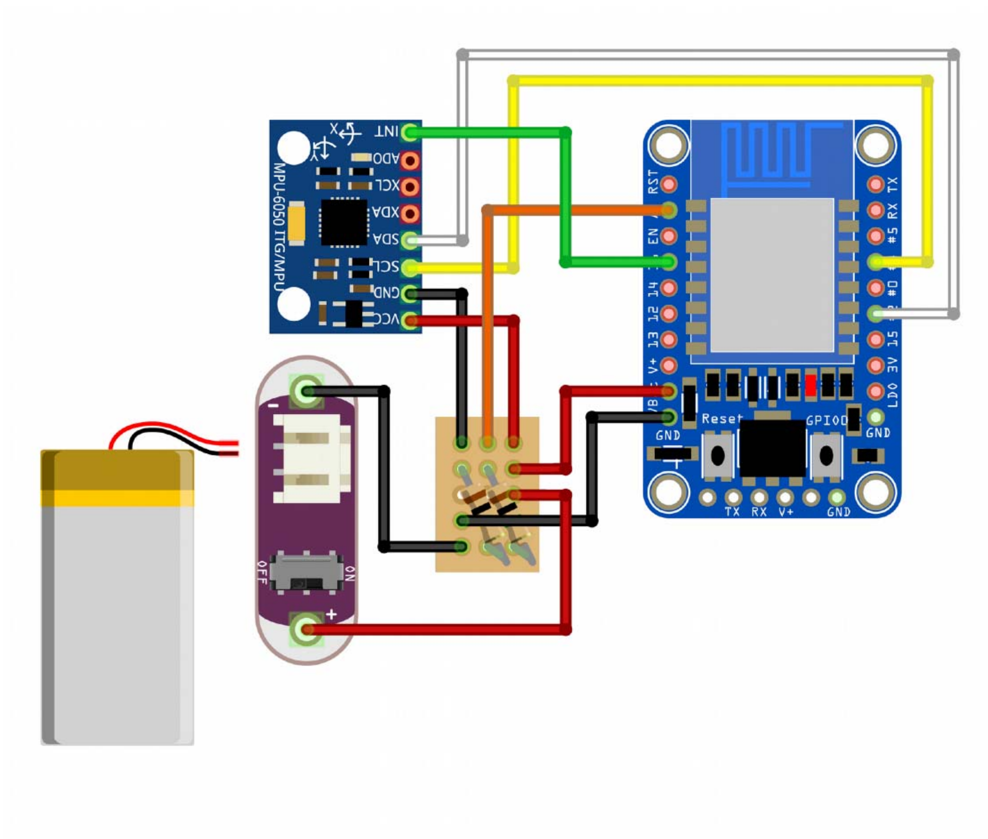


Figure 20: Schematic of the accelerometer system

A.6.4 *OpenSoundControl (OSC) data transmission*

Each device is continuously streaming accelerometer data to the indicated IP address and port. This is currently occurring at the rate of about 200 messages per second. On top of that, status messages indicating battery voltage and number of messages sent per second is transmitted with a lower frequency (one message per second).

The format of the accelerometer data sent over OSC is

```
/acc x y z
```

where x, y, and z are the acceleration values in the three axes. The device also sends a status message in the form

```
/status voltage percentage numMessages
```

reporting battery voltage, the calculated battery life percentage, and the number of messages sent per second.

A.6.5 *Adjusting settings over serial interface*

Settings of the device can be adjusted over the serial interface. A USB to serial adapter needs to be connected to the appropriate pins on the Adafruit Huzzah board. The connection is established with a 115200 baudrate.

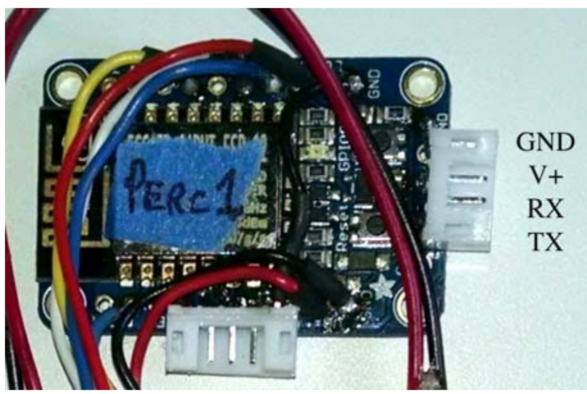


Figure 21: Pinout for the serial port on the Adafruit Huzzah board

The communication uses ASCII characters. Each command should be followed by a newline or a carriage return character. Enter "?" (without parentheses) to receive the list of possible commands. From the available parameters, the wireless network name (SSID) and the password are the two that are likely to need changing, depending on the setup.

To read the current network name, enter "ssid" [return]. To change the network name, enter "ssid new-network-name" [return]. Similarly, the command "pwd" enables reading and writing the password for the network.

Beside other parameters, the command "name" can be used to check or set the name of the current sensor. This should be the same name as the one written physically on the device, in order to enable its identification and proper assignment to a given instrument.

A.7 SOUND ENGINEERING

A.7.1 *Mixing board*

The mixing board needs to accommodate all the microphone inputs, enabling forwarding their signal through direct outputs to the audio interface. The signal from the microphones should be directed to the speakers at low volume. The sound of the instruments should be primarily acoustic. The amplification should only support the blending of the acoustic sound with the processed and recorded sound.

A.7.2 *Microphone setup*

All instruments have a microphone assigned to them. The piano and percussion may use one or two microphones, depending on the number of inputs available in the mixer and the audio interface. The microphone assignment to the audio interface inputs, defined in the SuperCollider code, is as follows:[15]

- 1: flute
- 2: oboe or trumpet
- 3: clarinet
- 4: percussion
- 5: piano
- 6: violin
- 7: violoncello

If a 2-microphone setup for piano and percussion is available, then the channel assignment would be as follows:

- 1: flute
- 2: oboe or trumpet
- 3: clarinet
- 4: percussion left
- 5: percussion right

[15]Input numbers are 1-based and assume the use of a single microphones for piano and percussion.

- 6: piano left
- 7: piano right
- 8: violin
- 9: violoncello

Both clip-on microphones as well as traditional ones on stands may be used for amplification and processing. Clip-on microphones are usually less susceptible to feedback, which might occur when using the pitch shifting processes.

A.7.3 *Speaker setup*

The piece requires two pairs of speakers. The main pair, positioned to the sides of the ensemble and possibly downstage, should be used to carry the majority of sound throughout the piece. The supporting pair, positioned behind the ensemble, should be used in the fourth section of the piece, after measure 155, when the acoustic and played back sound gradually crossfade.

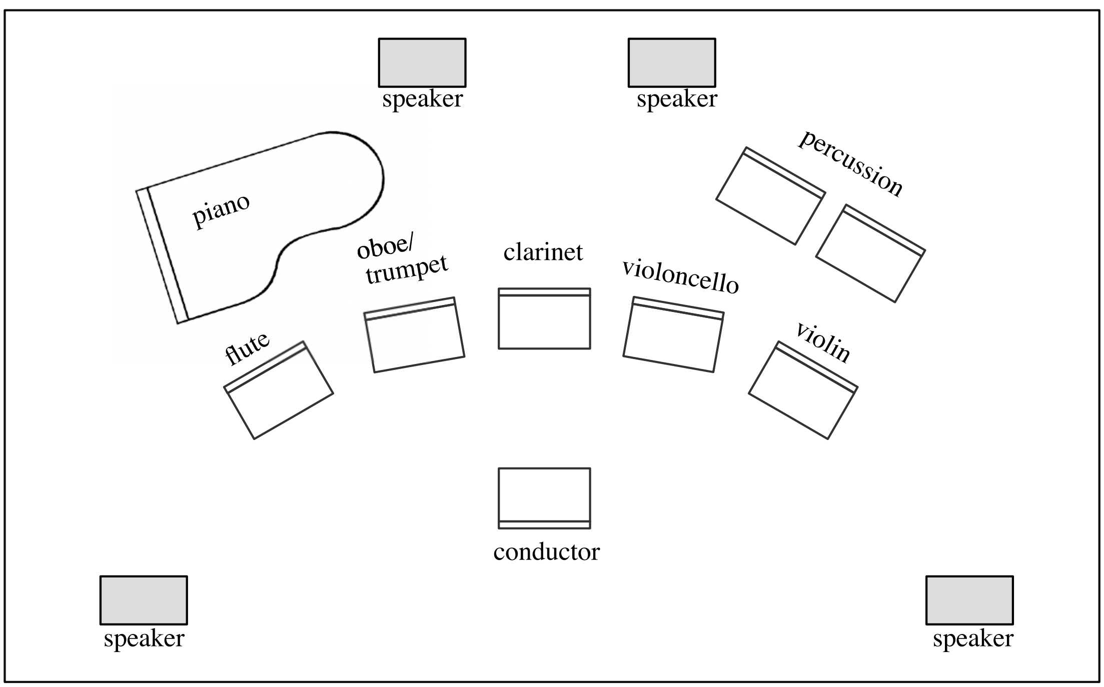


Figure 22: Stage setup for Deep Decline using 2 stereo pairs of speakers

It is also possible to realize the piece in the ambisonic setup, using six speakers positioned around the audience. The ambisonic setup would be used, like the main stereo pair, throughout majority of the piece. The additional pair behind the ensemble should still be used, to better blend with the acoustic sound after measure 155, as described earlier. The ambisonic version provides the audience with a more immersive experience of the live electronics, at the cost of increasing the separation between the instrumental and the processed sound. In the future, I intend both versions to be performed.

APPENDIX B: SCORE

Marcin Pączkowski

DEEP DECLINE

for chamber ensemble

[2015/2016]

Revised in May 2016

Duration: approx. 25 minutes

Instrumentation:

Flute / Piccolo

Oboe or Trumpet in B ♭ (see notes)

Clarinet in B ♭ / Bass Clarinet in B ♭

Percussion:
- Timpano (1 – 28”)
- Crotales
- Snare Drum
- Tom-toms (2)
- Cymbals (2: 1 on stand, 1 placed inverted on timpano head)
- Tam-tam

Piano

Violin

Violoncello

Performance notes:

Score is notated in concert pitch (octave transpositions still apply).

Accidentals apply throughout the measure.

Oboe and Trumpet parts are interchangeable, however should not be performed simultaneously. Either Oboe or Trumpet should be used in a given performance.

Tremolo and tremolando notated as 32nds should be performed as fast as possible (unmeasured).

Sections without time signature have approximate length provided in seconds. This time is only a guideline, as improvised material can be extended/compressed at the discretion of the performers.

The piece requires a live electronic system, consisting of microphones, wearable sensors, computer with a sound card and custom software, as well as PA equipment. Sensors and software is available from the composer at marcin@paczkowski.art.pl

Rehearsal letters E through M, as well as P and Q are focused on interaction with the live electronics system.

Deep Decline

Marcin Pączkowski (2015/2016)


8
Fl.
Ob.
Cl.
Tpt.
Pno.
Vln.
p
3
15
Fl.
Ob.
Cl.
Tpt.
Pno.
Vln.
Vc.
8va
p
3


21
Fl.
Ob.
Cl.
Tpt.
Pno.
Vln.
Vc.
27
Fl.
Ob.
Cl.
Tpt.
Pno.
Vln.
Vc.
sf
mf
p
sf
sf
p
f
mf
p
mf


32
molto rit.
Più mosso
accel.
Fl.
Ob.
Cl.
Tpt.
Pno.
Vln.
Vc.
sf
p
cresc.
3
5


37
♩ = 100
Fl.
Ob.
Cl.
Tpt.
Crot.
Pno.
Vln.
Vc.
f
cresc.


39
A
Fl.
Ob.
Cl.
Tpt.
S. D.
Tom-t.
Cym.
Pno.
Vln.
Vc.
soft yarn mallets
(mute w/l.h.)
mf
fff
ff


Meno mosso (♩ = 80)
42
Fl.
Ob.
Cl.
Tpt.
S. D.
Tom-t.
hard rubber mallet mallets
(l.v.)
Cym.
Pno.
gliss.
Vln.
Vc.


46
Fl.
Ob.
Cl.
Tpt.
roll on inverted cymbal placed on timpano head while pedaling / gliss.
Timp.
gliss.
Tom-t.
soft yarn mallets
(l.v.)
T.-t.
Pno.
approximate pitches
(w/cymbal's decay)
gliss.
Vln.
Vc.


Tempo I (♩ = 60)
51
small bend ad lib.
gradually slow down bends
B
♩ = 52
Fl.
pp
mf
ppp
Ob.
mf
Cl.
To Bass Cl.
Bass Clarinet in B♭
mf
Tpt.
mf
gradually slow down glissandi
(mute)
gliss. ad lib.
remove cymbal
Timp.
f
mp
Pno.
mf
p
l.v. sempre
gliss. ad lib.
gliss.
gliss. ad lib.
gradually slow down glissandi
Vln.
p
Vc.
p
2
Extending low pitches
El.
61
Ob.
B. Cl.
(frull.)
3
p
l.v.
T.-t.
p
Pno.
(8)
Vc.
El.


67
slow gliss.
as before
Fl.
pp
Ob.
B. Cl.
poco a poco cresc.
3
(mute)
hard snare sticks
l.v. sempre
Cym.
T.-t.
poco a poco cresc.
Pno.
poco a poco cresc.
(8)
slow gliss.
as before
Vln.
mp
Vc.
poco a poco cresc.
El.
accel.
73
B. Cl.
Cym.
T.-t.
Pno.
3
(8)
Vc.
El.


78
B. Cl.
Cym.
T.-t.
Pno.
(8)
Vc.
El.
82
♩ = 108
C
B. Cl.
Cym.
Pno.
(8)
Vc.
El.


87
B. Cl.
Cym.
Pno.
Vln.
Vc.
El.
slow gliss.
tempo as before
91
Fl.
B. Cl.
Cym.
Pno.
Vln.
Vc.
El.
slow gliss.
tempo as before
pp
mp

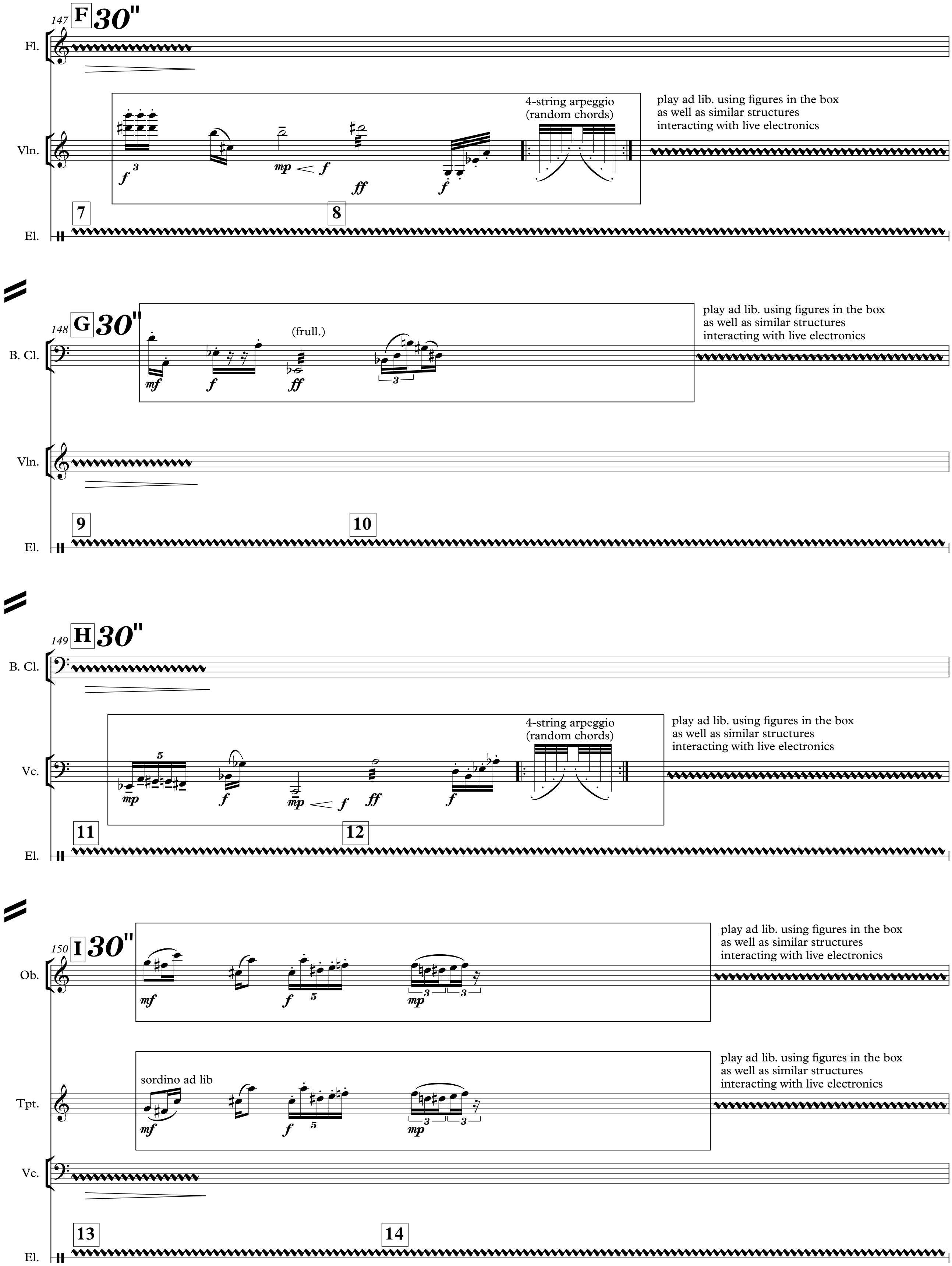

95
To Picc.
Fl.
B. Cl.
Cym.
Pno.
Vc.
El.
99
Ob.
B. Cl.
senza sordino
Tpt.
Cym.
Pno.
Vln.
Vc.
El.

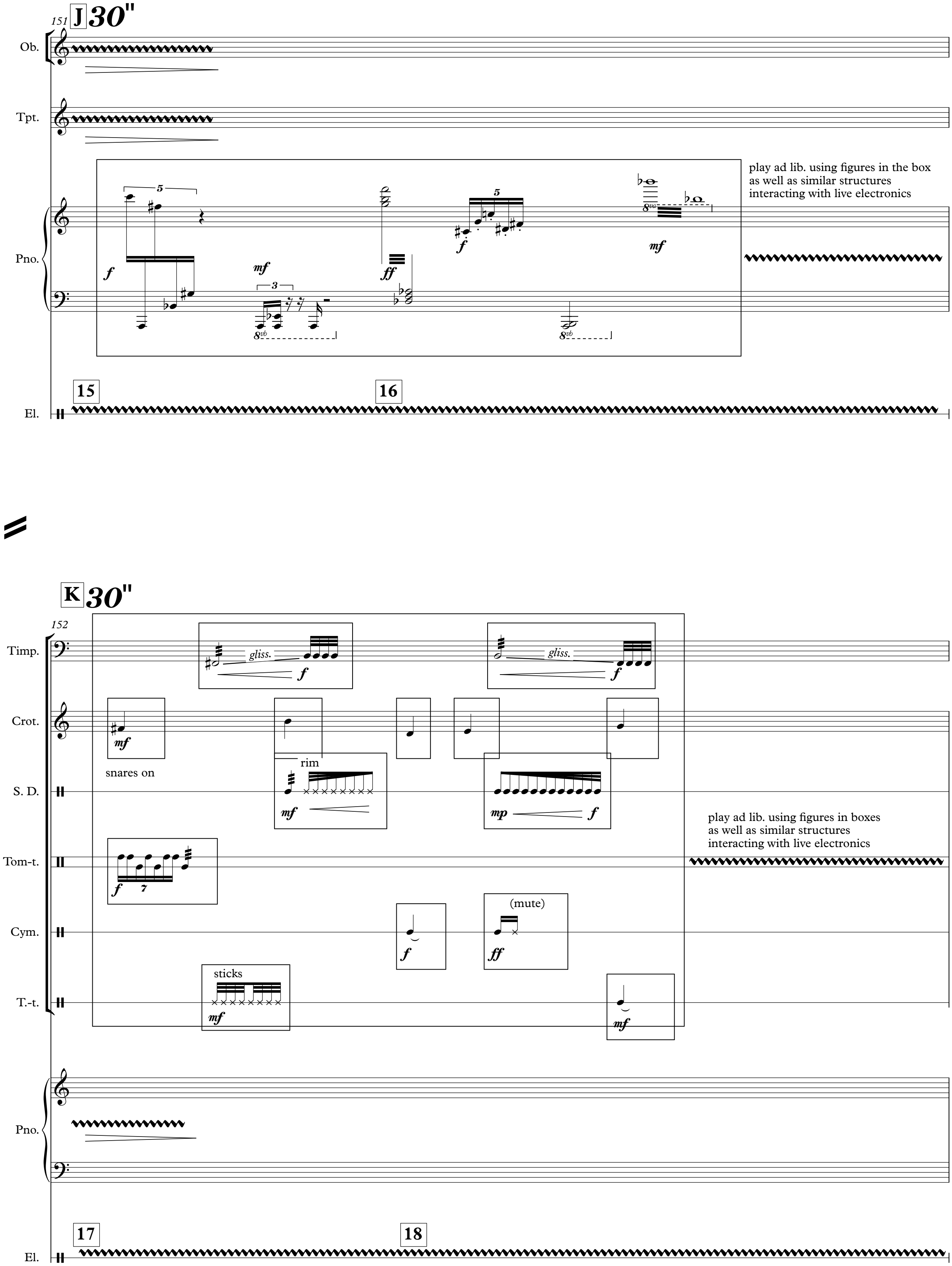

103
B. Cl.
Cym.
Pno.
Ped.
Vc.
El.
107
Piccolo
Fl.
mf
cresc.
Ob.
B. Cl.
f
Tpt.
Cym.
Pno.
Vln.
Vc.
3

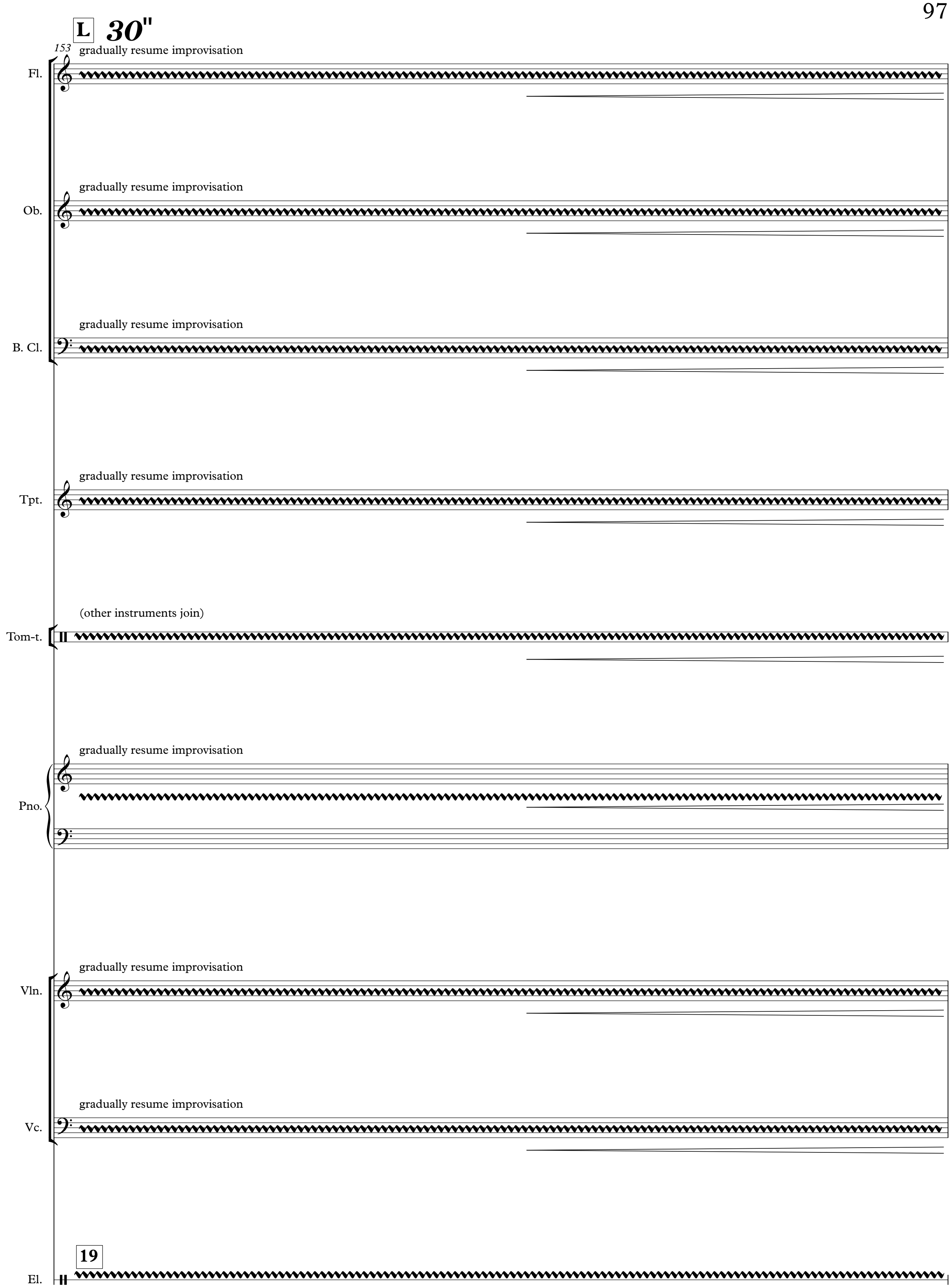

111
Picc.
Ob.
B. Cl.
Tpt.
Cym.
Pno.
Vln.
El.
8va
f
3
4
115
D
frull.
rit.
ff
5
poco a poco dim.
Extend high pitches

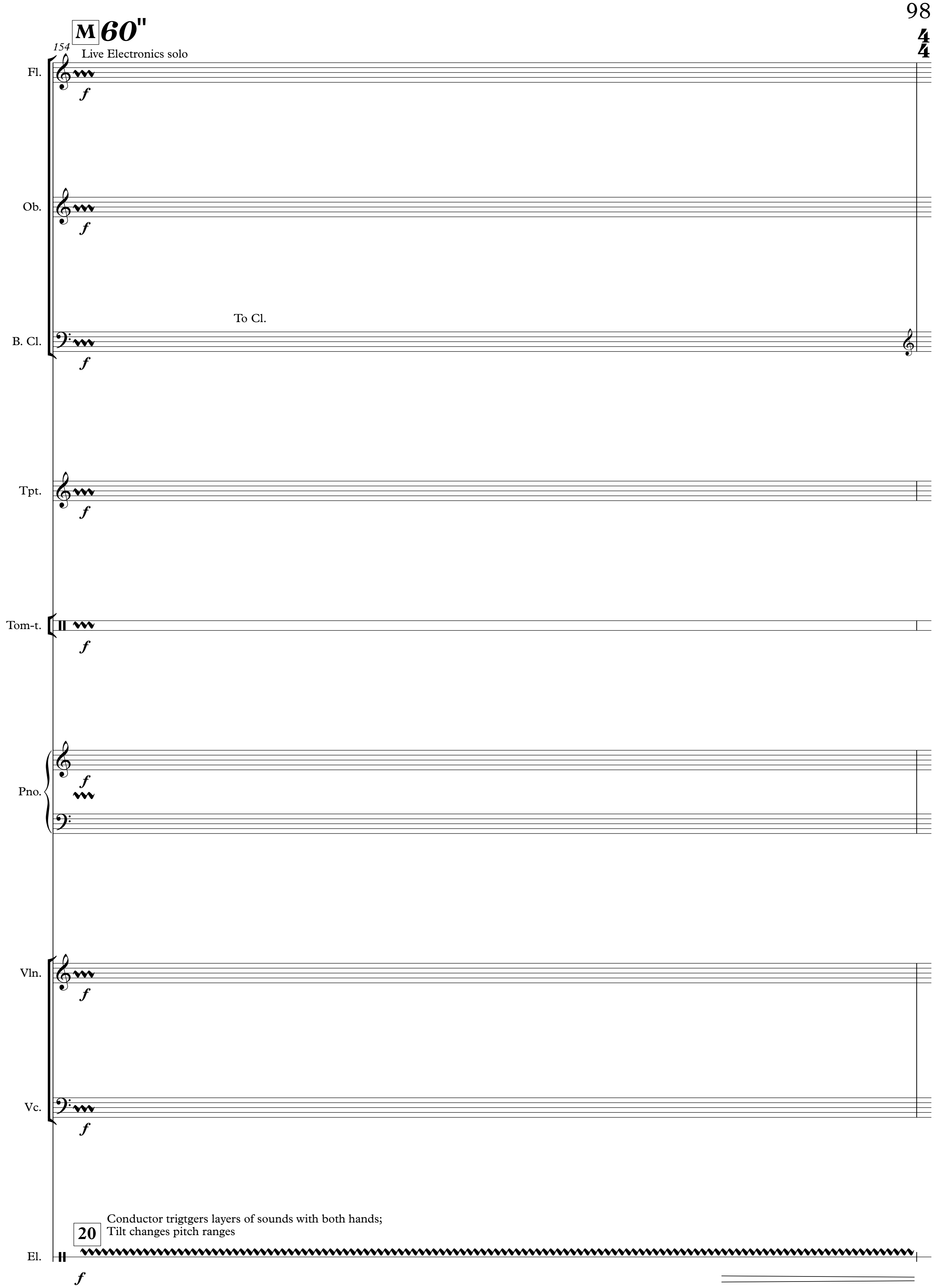

120
♩ = 60
Picc.
Ob.
Tpt.
con sordino (cup mute)
(8)
Pno.
Vln.
El.
mp
128
Picc.
Ob.
Tpt.
(8)
Pno.
mp
Vln.
mp
3
El.


135
Picc.
Ob.
Tpt.
Pno.
Vln.
El.
140
To Flute
Flute
5
E 30"
146
Fl.
mf
mp
f
play ad lib. using figures in the box
as well as similar structures
interacting with live electronics
6 Interact/extend individual solos


147 F 30"
Fl.
4-string arpeggio (random chords)
play ad lib. using figures in the box as well as similar structures interacting with live electronics
Vln.
7
8
El.
148 G 30"
(frull.)
play ad lib. using figures in the box as well as similar structures interacting with live electronics
B. Cl.
Vln.
9
10
El.
149 H 30"
B. Cl.
4-string arpeggio (random chords)
play ad lib. using figures in the box as well as similar structures interacting with live electronics
Vc.
11
12
El.
150 I 30"
play ad lib. using figures in the box as well as similar structures interacting with live electronics
Ob.
sordino ad lib
play ad lib. using figures in the box as well as similar structures interacting with live electronics
Tpt.
Vc.
13
14
El.


151
J 30"
Ob.
Tpt.
Pno.
play ad lib. using figures in the box
as well as similar structures
interacting with live electronics
15
16
El.
K 30"
152
Timp.
gliss.
Crot.
snares on
rim
S. D.
play ad lib. using figures in boxes
as well as similar structures
interacting with live electronics
Tom-t.
(mute)
Cym.
sticks
T.-t.
Pno.
17
18
El.


L
30"
153
gradually resume improvisation
Fl.
gradually resume improvisation
Ob.
gradually resume improvisation
B. Cl.
gradually resume improvisation
Tpt.
(other instruments join)
Tom-t.
gradually resume improvisation
Pno.
gradually resume improvisation
Vln.
gradually resume improvisation
Vc.
19
El.

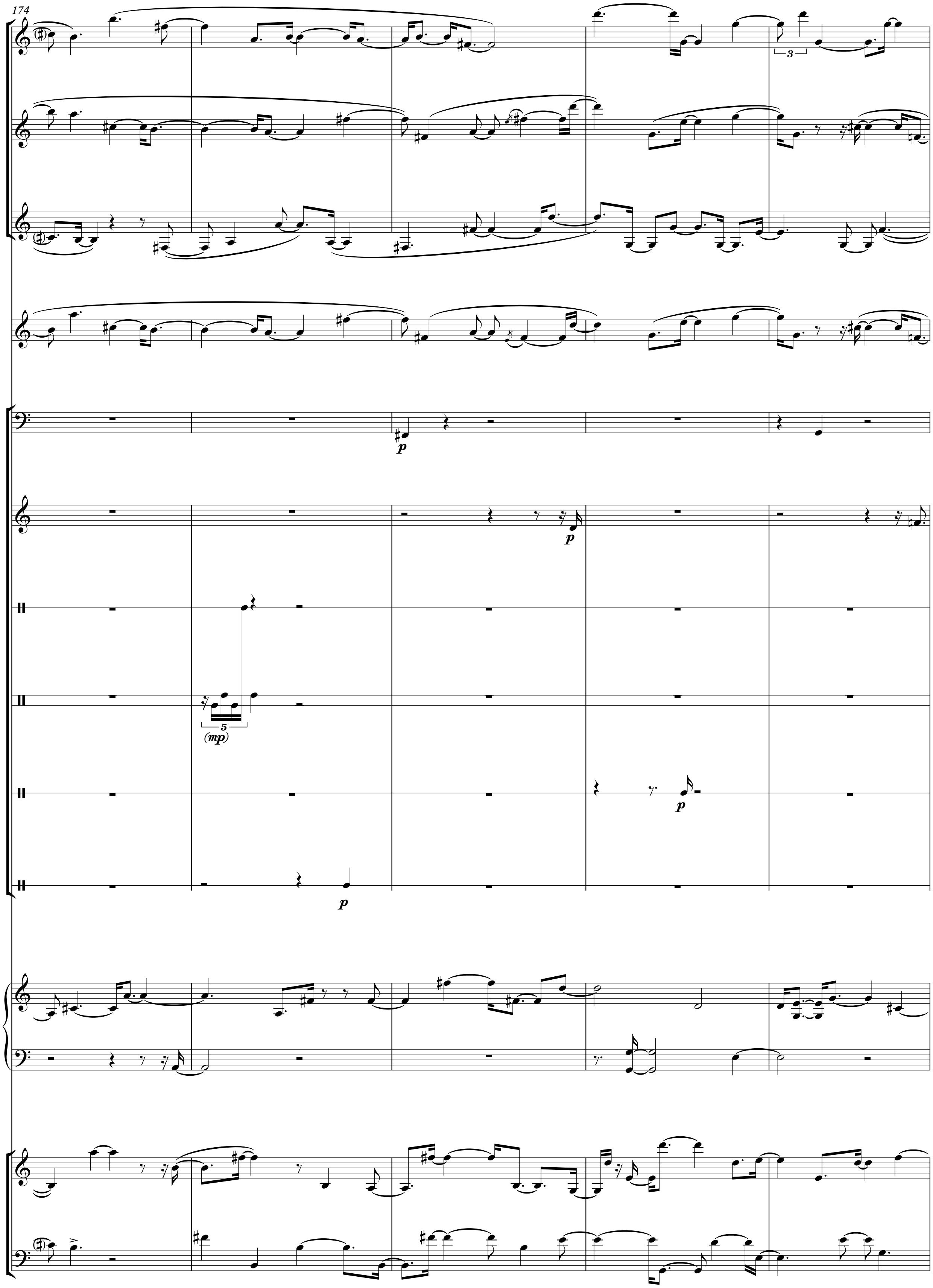

M
60"
154
Live Electronics solo
Fl.
Ob.
To Cl.
B. Cl.
Tpt.
Tom-t.
Pno.
Vln.
Vc.
20
Conductor trigtgers layers of sounds with both hands;
Tilt changes pitch ranges
El.


N
= 60
155
Fl.
Ob.
Clarinet in B♭
Cl.
con sordino
Tpt.
soft yarn mallets l.v. sempre
Timp.
l.v. sempre
Crot.
snares off
S. D.
Tom-t.
l.v. sempre
T.-t.
Pno.
Vln.
Vc.
El.


160
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
T.-t.
Pno.
Vln.
Vc.


165
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
S. D.
Tom-t.
Cym.
T.-t.
Pno.
Vln.
Vc.
f
pp
mp
mf


170
Fl.
Ob.
Cl.
Tpt.
(alt. for smaller timpani: B2-D3)
Timp.
gliss.
Crot.
(mp)
Tom-t.
Cym.
mf
Pno.
Vln.
Vc.
f


174
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
S. D.
Tom-t.
Cym.
T.-t.
Pno.
Vln.
Vc.
p
(mp)


179
O
Fl.
Ob.
Cl.
Tpt.
Crot.
Cym.
T.-t.
Pno.
Vln.
Vc.
3


183
Fl.
Ob.
Cl.
Tpt.
S. D.
Tom-t.
Cym.
Pno.
Vln.
Vc.
(mp)
f subito
mf
3
5
6
7

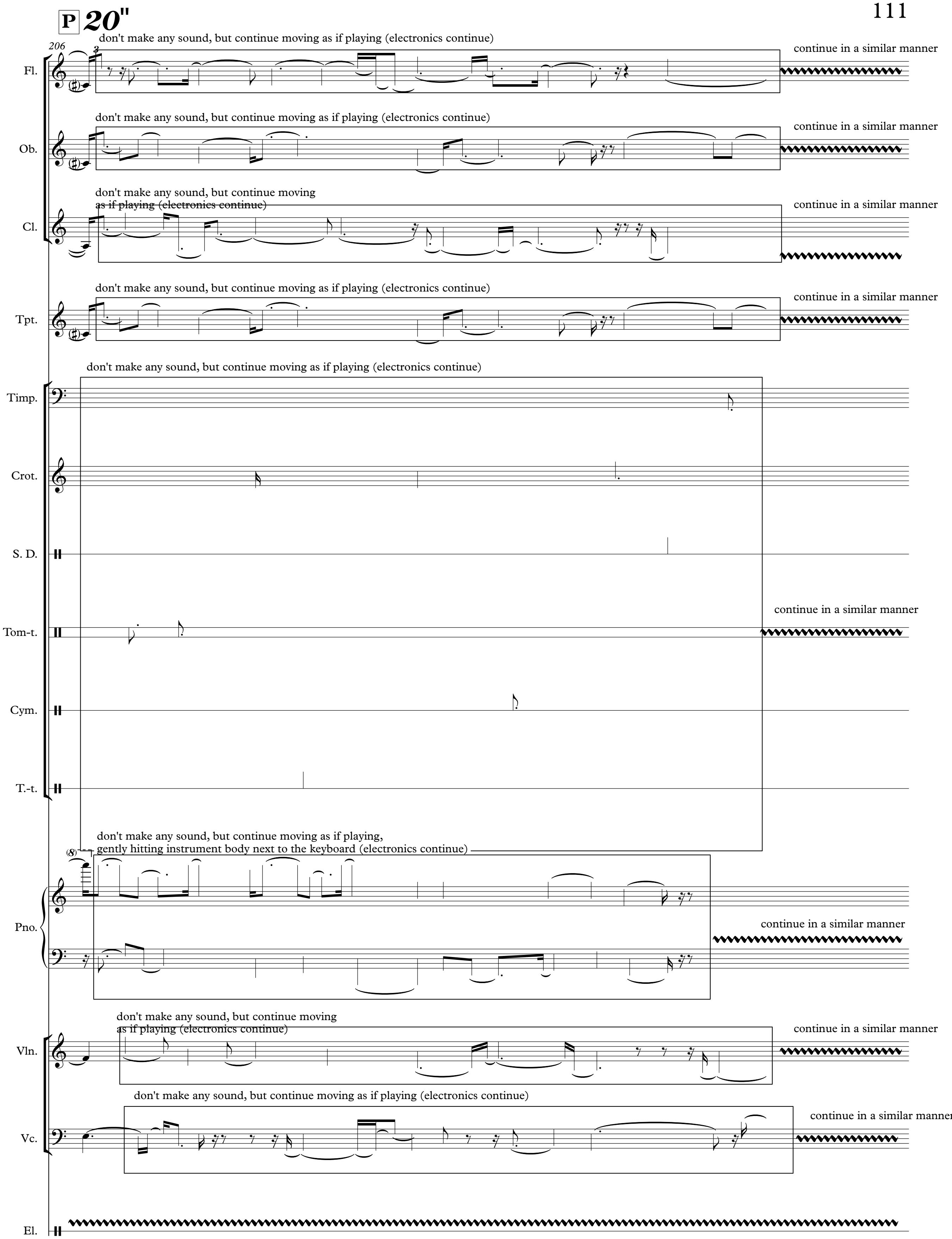

186
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
Cym.
T.-t.
Pno.
Vln.
Vc.
mp
mp
mp
mp
p
(p)
p
mf
p
mp
mp
mp

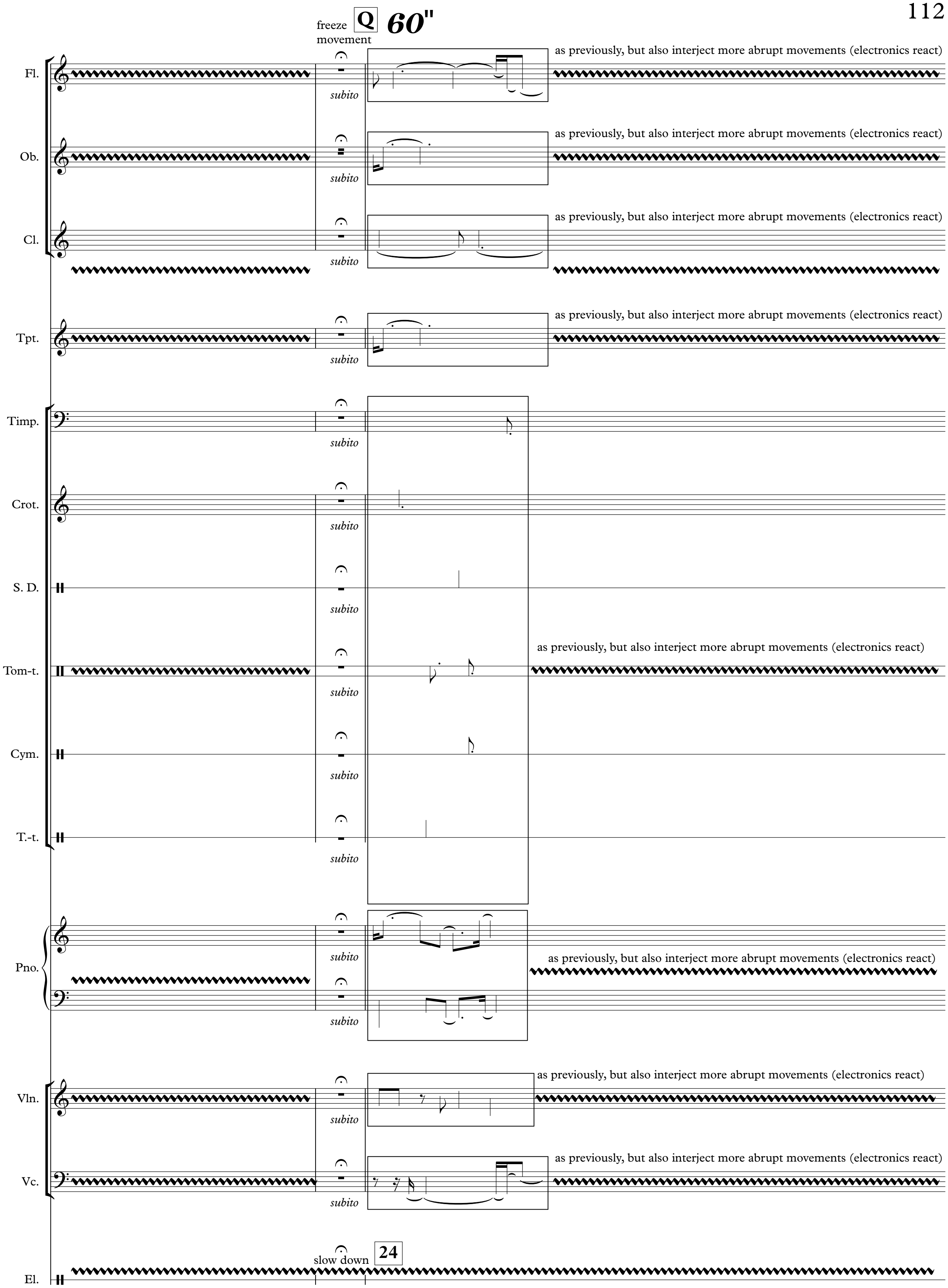

190
Fl.
Ob.
Cl.
Tpt.
Crot.
S. D.
Tom-t.
Cym.
Pno.
Vln.
Vc.
El.
(p)
mf subito
p
poco a poco diminuendo
mf
8va
22
23
Gradually enter, mimicking instruments

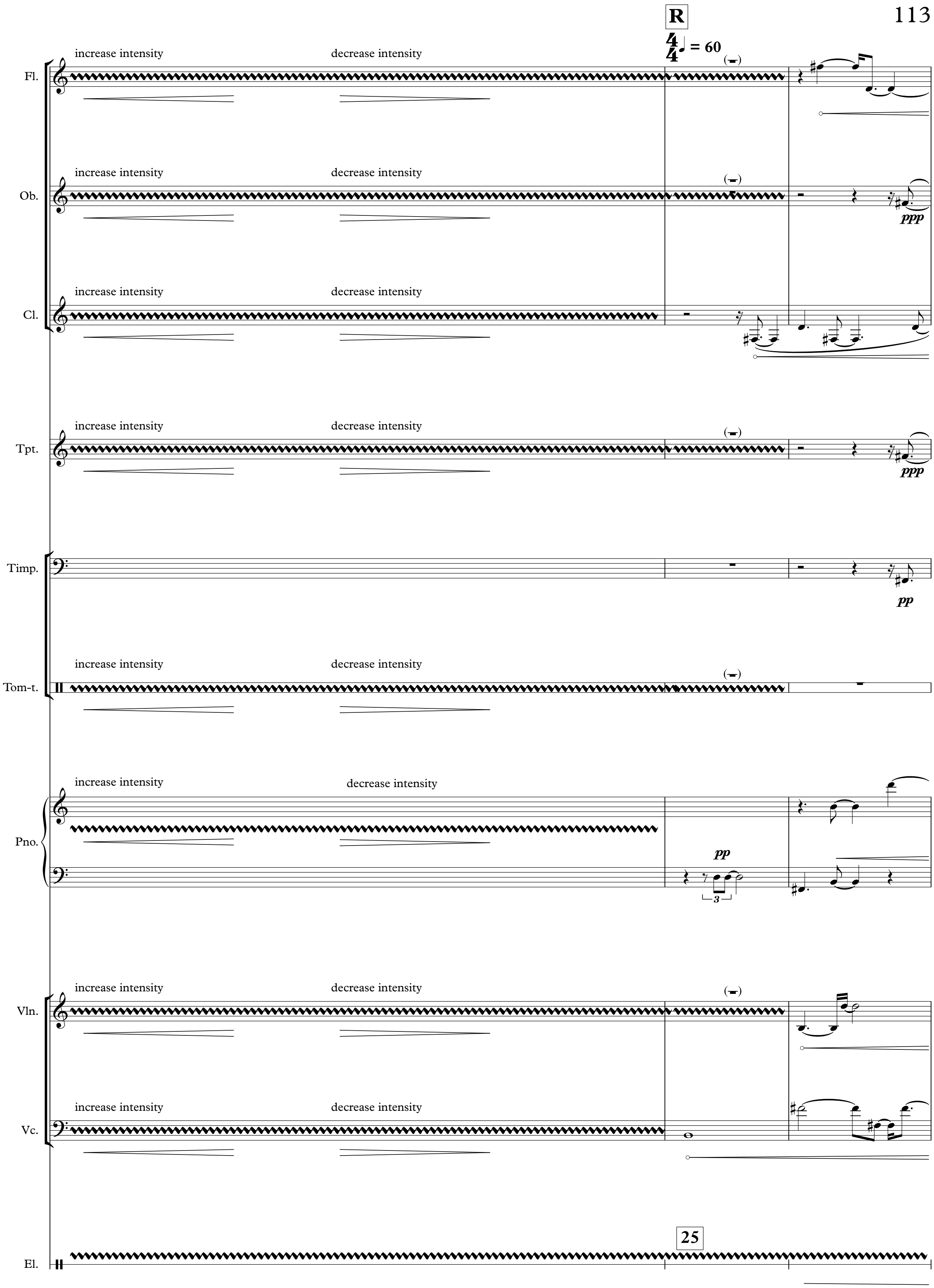

195
Fl.
Ob.
Cl.
Tpt.
Timp.
p
Crot.
S. D.
p
Cym.
Pno.
Vln.
Vc.
El.


200
3
Fl.
Ob.
Cl.
Tpt.
Timp.
p
Tom-t.
p
T.-t.
pp
Pno.
Vln.
Vc.
El.


203
Fl.
Ob.
Cl.
Tpt.
Crot.
pp
Cym.
ppp
8va
Pno.
8vb
Vln.
Vc.
El.
3


P 20"
206
don't make any sound, but continue moving as if playing (electronics continue)
continue in a similar manner
Fl.
don't make any sound, but continue moving as if playing (electronics continue)
continue in a similar manner
Ob.
don't make any sound, but continue moving as if playing (electronics continue)
continue in a similar manner
Cl.
don't make any sound, but continue moving as if playing (electronics continue)
continue in a similar manner
Tpt.
don't make any sound, but continue moving as if playing (electronics continue)
Timp.
Crot.
S. D.
continue in a similar manner
Tom-t.
Cym.
T.-t.
don't make any sound, but continue moving as if playing, gently hitting instrument body next to the keyboard (electronics continue)
continue in a similar manner
Pno.
don't make any sound, but continue moving as if playing (electronics continue)
continue in a similar manner
Vln.
don't make any sound, but continue moving as if playing (electronics continue)
continue in a similar manner
Vc.
El.


freeze movement
Q
60"
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
S. D.
Tom-t.
Cym.
T.-t.
Pno.
Vln.
Vc.
El.
subito
as previously, but also interject more abrupt movements (electronics react)
slow down
24


R
♩ = 60
increase intensity
decrease intensity
Fl.
Ob.
Cl.
Tpt.
Timp.
Tom-t.
Pno.
Vln.
Vc.
El.
ppp
pp
3
25


211
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
Tom-t.
Cym.
T.-t.
Pno.
Vln.
Vc.
El.
mp
p
pp
conductor triggers sustaining harmonies by hitting downbeat with left hand


216
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
S. D.
mp
Tom-t.
mp
Cym.
T.-t.
Pno.
Vln.
Vc.
El.


221
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
S. D.
Tom-t.
Cym.
Pno.
Vln.
Vc.
El.


225
rit.
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
rim
S. D.
Tom-t.
T.-t.
Pno.
Vln.
Vc.
El.


229
Fl.
Ob.
Cl.
Tpt.
Timp.
Crot.
S. D.
Tom-t.
Cym.
T.-t.
Pno.
Vln.
Vc.
El.
p
26

VITA

Marcin Pączkowski is a composer, conductor, and digital artist, working with both traditional and electronic media. As a composer, he is focused on developing new ways of creating and performing computer music. His pieces involving real-time gesture control using accelerometers have been performed at the Music of Today concert series in Seattle, Washington, Northwest Percussion Festival in Ashland, Oregon, and at the Audio Art festival in Kraków, Poland.

As a conductor he is involved in performing new music and premiering new works. He is the conductor and co-director of Inverted Space, a Seattle-based new music collective. He is also the music director of the Evergreen Community Orchestra in Everett, Washington and co-founder of the contemporary chamber vocal ensemble Pogratulujmy Mrówkom in Kraków, Poland.

Before starting his studies in DXARTS, Marcin received his Masters' degrees from the Academy of Music in Kraków, Poland (composition and conducting), and from University of Washington in Seattle, Washington (composition). He was a grant recipient from Polish Institute of Music and Dance and from Lesser Poland Scholarship Foundation Sapere Auso.