HYPERCOMPRESSION

Stochastic Musical Processing

Charles J. Holbrow

Bachelor of Music
University of Massachusetts Lowell, 2008

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning, in partial fulfillment
of the requirements for the degree of **Master of Science**
at the **Massachusetts Institute of Technology**
August 2015

Author: CHARLES HOLBROW

Program in Media Arts and Sciences

September 7, 2015

Certified by: TOD MACHOVER

Professor of Music and Media
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by: PATTIE MAES

Interim Academic Head
Program in Media Arts and Sciences

HYPERCOMPRESSION

Stochastic Musical Processing

Charles Holbrow

The following person served as a reader for this thesis:

Thesis Reader: Joseph A. Paradiso

Associate Professor of Media Arts and Sciences

Program in Media Arts and Sciences

Massachusetts Institute of Technology

HYPERCOMPRESSION

Stochastic Musical Processing

Charles Holbrow

The following person served as a reader for this thesis:

Thesis Reader: James A. Moorer

Principal Scientist

Adobe Systems, Incorporated

Contents

1 Introduction

In the 6th century B.C., Pythagoras discovered that dividing a resonating string into simple mathematical ratios produced harmonious musical intervals, while arbitrary ratios produced dissonance. His observation is probably the first of the many explicit parallels between math and music that have been identified since his time. Today, we describe musical pitches as integers within a given tuning system. We describe the tuning system with a mathematical formula that relates frequency to pitch. Musical time, rhythm and meter are commonly described numerically. Musical transposition and inversion both mirror mathematical functions and borrow their names directly from mathematics.

As computers, amplifiers, and electronics become our primary tools for creating manipulating, and performing music, mathematics and music necessarily become more interconnected. Nearly every modern musical recording, broadcast, and stream is the summation of many digital recordings that have been individually discretized, sampled, mathematically encoded, decoded, and digitally processed numerous times before ever reaching our ears.[1] It is tempting to describe music today as applied mathematics, but doing so betrays a fundamental quality of music: Musicality does not correspond to mathematical elegance or precision. A musician will diverge from a musical score to accomplish a particular artistic objective. A vocalist does not abruptly change a pitch, but gently and carefully lands on a pitch. A jazz musician might intentionally play slightly behind the beat. A classical performer knows how to hold a fermata just long enough. These intentional human artifacts are characterized more by a feeling than by a formula.

The computer's inability to understand feeling has led to new genres of music like EDM[2], Black MIDI[3], and Demoscene[4], but these styles of music feature (rather than fix) the inhuman nature of computers. If we want to integrate a computer into the performance or production of truly expressive music, we must capture perceived feelings formulaically and program the computer to reproduce them. This thesis describes three different, but related projects that confront this challenge from contrasting perspectives: Reflection Visualizer,

[1] Alex Case. *Sound FX: Unlocking the Creative Potential of Recording Studio Effects*. Focal Press, 2007. ISBN 978-0240520322

[2] EDM (Electronic Dance Music) features formulaic and repetitive grooves locked to a temporal grid and often incorporates aggressive use of digital pitch correction, further exaggerating a robotic quality.

[3] Black MIDI is a musical genre that uses low fidelity audio samplers with a large number of MIDI notes over a short time. A single three minute Black MIDI track is likely to have over 100,000 MIDI notes. The name refers to the solid black appearance of the piano score.

[4] Demoscene music celebrates digital synthesis of compositionally complex electronic music and audio visualizations, using low level software interfaces and including the design and programming of the music synthesizers as part of the composition.

Stochastic Tempo Modulation, and Hypercompression.

1.1 *Reflection Visualizer*

Music and space are intimately connected. The first project, described in chapter 2, explores how we can compose music using acoustic reflections in architectural space as a medium. Reflection Visualizer is a software tool that lets us design and experiment with abstract acoustic lenses or "sound mirrors" in two dimensions. It is directly inspired by the music and architecture of Iannis Xenakis, a 20th century composer, music theorist, architect, and engineer. His collected works provide guidance and perspective to all the projects in this thesis.

1.2 *Stochastic Tempo Modulation*

Music and Time are inseparable. All music flows through time and depends on temporal constructs - the most common being meter and tempo. Accelerating or decelerating tempi are common in many styles of music, as are polyrhythms. Music with multiple simultaneous tempi or *polytempic music* is less common, but still many examples can be found. Fewer examples of music with simultaneous tempi that shift relative to each other exist, however, and it is difficult for musicians to accurately perform changing tempi in parallel. Software is an obvious choice for composing complex and challenging rhythms such as these, but existing compositional software makes this difficult. Stochastic Tempo Modulation offers a solution to this challenge by describing a strategy for composing music with multiple simultaneous tempi that accelerate and decelerate relative to each other. In chapter 3 we derive an equation for smoothly ramping tempi to converge and diverge as musical events within a score, and show how this equation can be used as a stochastic process to compose previously inaccessible sonorities.

1.3 *Hypercompression*

We usually think of compression in terms of *reduction*: We use data compression to reduce bit-rates and file sizes and audio compression to reduce dynamic range. Record labels use of dynamic range compression as a weapon in the *loudness war*[5,6], has resulted in some of today's music recordings utilizing no more dynamic

[5] "Loudness War" is the popular name given to the trend of increasing percieved loudness in music recordings. Beginning in the 1990s, record labels have attempted to make their music louder than the competition, at the expense of audio fidelity.

[6] Emmanuel Deruty and Damien Tardieu. About Dynamic Processing in Mainstream Music. *AES: Journal of the Audio Engineering Society*, 62(1-2):42–56, 2014. ISSN 15494950

range than a 1909 Edison cylinder.[7] A deeper study of dynamic range compression, however, reveals more subtle and artistic applications beyond that of reduction. A skilled audio engineer can apply compression to improve intelligibility, augment articulation, smooth a performance, shape transients, extract ambience, de-ess vocals, balance multiple signals, or even add distortion.[8] At its best, the compressor is a tool for temporal shaping, rather than a tool for dynamic reduction.

[7] Bob Katz. *Mastering Audio: The Art and Science*. Focal Press, 2nd edition, 2007. ISBN 978-0240808376

[8] Alex Case. *Sound FX: Unlocking the Creative Potential of Recording Studio Effects*. Focal Press, 2007. ISBN 978-0240520322

Hypercompression expands the traditional model of a dynamic range compressor to include spatial shaping. While unconventional, spatial processing is a very natural fit for the compression paradigm. Sound is a medium that exists in time as well as in space.[9] The mathematics and implementation of the Hypercompressor are described in detail in chapter 4.

[9] Converting measurement of sound from the cycles per second (in the temporal domain) to wavelength (in the spatial domain) is a common objective in acoustics and audio engineering practices. See *The Sound Reinforcement Handbook* by G. Davis for examples.

Hypercompression was used in the live performance of *De L'Expérience*, a new musical work by composer Tod Machover for Narrator, Organ, and Electronics. During the premier at the Maison Symphonique de Montréal in Canada, Hypercompression was used to blend the electronics with the organ and the acoustic space. A detailed description of how Hypercompression featured in this performance is also discussed in chapter 4.

1.4 Background

These projects build on the work and ideas of Iannis Xenakis, a 20th century composer, architect, and engineer. Xenakis spent his youth reading about astronomy, archeology, ancient literature, and mathematics.[10] He studied music and engineering at the Polytechnic Institute in Athens, Greece. By 1948, Xenakis had graduated from the university and moved to France where he began working for the French architect, Le Corbusier. The job put his engineering skills to use, but Xenakis also wanted to continue studying and writing music. While searching for a music mentor, he approached Oliver Messiaen[11], and asked for advice on whether he should study harmony or counterpoint. Messiaen later described his conversation with Xenakis:

[10] Peter Hoffmann. Xenakis, Iannis. *Grove Music Online, Oxford Music Online, Oxford University Press*, 2015. URL http://www.oxfordmusiconline.com/subscriber/article/grove/music/30654

[11] Messiaen was a prolific French composer known for rhythmic complexity. He was also regarded as a fantastic music teacher, and his students include Karlheinz Stockhausen, Pierre Boulez, and Quincy Jones.

> "I think one should study harmony and counterpoint. But this was a man so much out of the ordinary that I said: No, you are almost 30, you have the good fortune of being Greek, of being an architect and having studied special mathematics. Take advantage of these things. Do them in your music."[12]

[12] Tom Service. A Guide to Iannis Xenakis's Music, 2013. URL http://www.theguardian.com/music/tomserviceblog/2013/apr/23/contemporary-music-guide-xenakis

In essence, Messiaen was rejecting Xenakis as a student, but we can see how Xenakis ultimately drew from his disparate skills in his compositions. The score for his 1945 composition *Metastasis* (figure 1.1) resembles an architectural blueprint as much as it does a musical score.

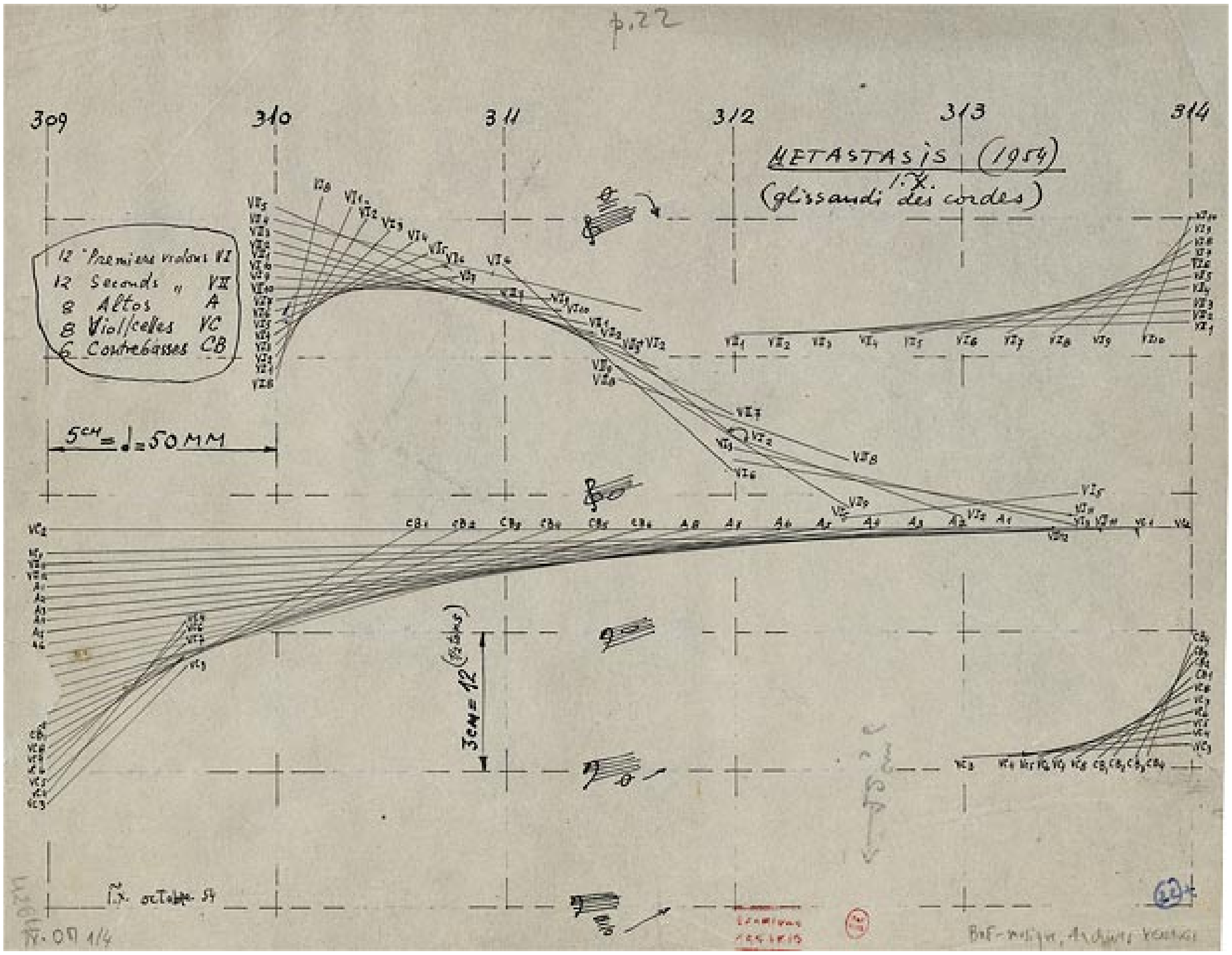

Figure 1.1: Excerpt from Iannis Xenakis' composition, *Metastasis* (1954), measures 309-314. This score in this image was then transcribed to sheet music for the orchestral performance.

The Philips Pavilion

In 1956, Le Corbusier was approached by Louis Kalff (Artistic Director for the Philips corporation) and asked to build a pavilion for the 1958 World's Fair in Brussels. The pavilion was to showcase the sound and lighting potential of Philips' technologies. Le Corbusier immediately accepted, saying:

> "I will not make a pavilion for you but an Electronic Poem and a vessel containing the poem; light, color image, rhythm and sound joined together in an organic synthesis."[13]

The final product lived up to Le Corbusier's initial description. It included:[14]

[13] Oscar Lopez. AD Classics: Expo '58 + Philips Pavilion / Le Corbusier and Iannis Xenakis, 2011. URL http://www.archdaily.com/157658/ad-classics-expo-58-philips-pavilion-le-corbusier-and-iannis-xenakis/

[14] Vincenzo Lombardo, Andrea Valle, John Fitch, Kees Tazelaar, and Stefan Weinzierl. A Virtual-Reality Reconstruction of Poeme Based on Electronique Philological Research. *Computer Music Journal*, 33(2):24–47, 2009

Figure 1.2: The Philips Pavilion at the 1958 Brussels World Fair as shown in Volume 20 of the *Philips Technical Review*, 1959.

1. A concrete pavilion, designed by architect and composer Iannis Xenakis
2. *Interlude Sonoire* (later renamed *Concret PH*), a tape music composition by Iannis Xenakis, approximately 2 minutes long, played between performances, while one audience left the pavilion and the next audience arrived
3. *Poème Électronique*, a three channel, 8 minute tape music composition by composer Edgard Varèse
4. A system for spatialized audio across more than 350 loudspeakers distributed throughout the pavilion
5. An assortment of colored lighting effects, designed by Le Corbusier in collaboration with Philips' art director, Louis Kalff
6. Video consisting mostly of black and white still images, projected on two walls inside the pavilion
7. A system for synchronizing playback of audio and video, with light effects and audio spatialization throughout the experience

Role of Iannis Xenakis During the initial design stage, Le Corbusier decided that the shape of the pavilion should resemble a stomach, with the audience entering through one entrance and exiting out another. He completed initial sketches of the pavilion layout and then delegated the remainder of the design to Xenakis.[15]

The architectural evolution of the pavilion from Le Corbusier's early designs (figure 1.4) to Xenakis' iterations (figure 1.5), illustrates

[15] Joseph Clarke. Iannis Xenakis and the Philips Pavilion. *The Journal of Architecture*, 17(2):213–229, 2012. ISSN 1360-2365. DOI: 10.1080/13602365.2012.678641

the profound impact that Xenakis had on the project. An article in the *Philips Technical Review*[16] gives a wonderfully detailed account of Xenakis' process in restructuring the design:[17]

[16] Philips Natuurkundig Laboratorium. *Philips Technical Review*. Number v. 20. Philips Research Laboratory, 1959

[17] **TODO: Clean this section.**

1. Xenakis was aware that parallel walls and concave spherical walls would both negatively impact audio perceptibility due to repeated or localized acoustic reflections.

2. To accommodate musical purpose of the space he decided to explore surfaces with varying curvature...

3. ...leading him to consider ruled surfaces such as the conoid and hyperbolic paraboloid.

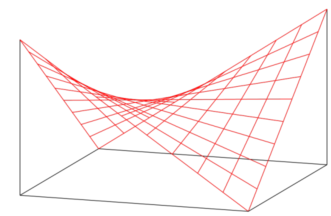

Figure 1.3: A ruled surface. For a surface to be considered "ruled" every point on the surface must be on a straight line, and that line must lie on the surface. In Xenakis' time, ruled surfaces were useful in architecture, because they simplified the construction of curved surfaces by using straight beams.

Through this process, we see Xenakis utilizing the skills that he learned at the Polytechnic Institute and continued to develop while working with Le Corbusier. He also understood the mathematical formation of the ruled surfaces that make up the structure. These surfaces even look familiar to the Metastasis score (figure 1.1). In his 1963 book, *Formalized Music*, Xenakis explicitly states that the Philips Pavilion was inspired by his work on *Metastasis*.

1.5 *Architecture and Music in Space and Time*

In *Formalized Music*[18], Xenakis describes how developments in music theory mimic equivalent developments in philosophy, mathematics, and the sciences. Plato, for example, believed that all events transpire as determined by cause and effect. While Plato and Aristotle both described causality in their writing, it was not until the 17th century that controlled experiments and mathematics corroborated the theory.[19] Similarly, music theory has historically employed causal rules to describe counterpoint, tonality, and harmonic movement.[20]

[18] Iannis Xenakis. *Formalized Music*. Pendragon Press, 1992. ISBN 0-945193-24-6

[19] In 1687, Isaac Newton published *Philosophiæ Naturalis Principia Mathematica* (*Mathematical Principles of Natural Philosophy*), in which he compiled the 3 laws of motion that set the foundation for the study of *classical mechanics*.

[20] **TODO: Add example**

Causality was largely used to describe physical phenomena until the 19th century when statistical theories in physics began to include probabilistic notions.[21] Xenakis noticed that more contemporary fields like *probability theory* generalize and expand on the antecedent theories of causality. Xenakis thought that music composition should naturally follow the progression that physics did, with music theory generalizing and expanding on causal rules that had existed previously. Indeed, starting in the late 19th century and early 20th century, composers like Strauss and Debussy began to bend the existing rules of music theory, composing music that branched away from the causal and tonal theories of the time. With the rise of serialism[22] and indeterminate music[23], composers such as Stockhausen, Boulez, John Cage, Aaron Copland, and Béla Bartók began to use probability

[21] The Maxwell-Boltzmann distribution, which was first derived by James Clerk Maxwell in 1860, describes the probability distribution for the speed of a particle within an idealized gas. For more see `http://plato.stanford.edu/entries/statphys-statmech/`

[22] Serialism is a technique for musical composition in which instances of musical elements (such as pitch, dynamics, or rhythm), are given numerical values. Sequences built from the values are ordered, repeated and manipulated throughout the composition.

[23] In music, indeterminacy refers to the use of chance (such as rolling dice or flipping coins) as part of the compositional process.

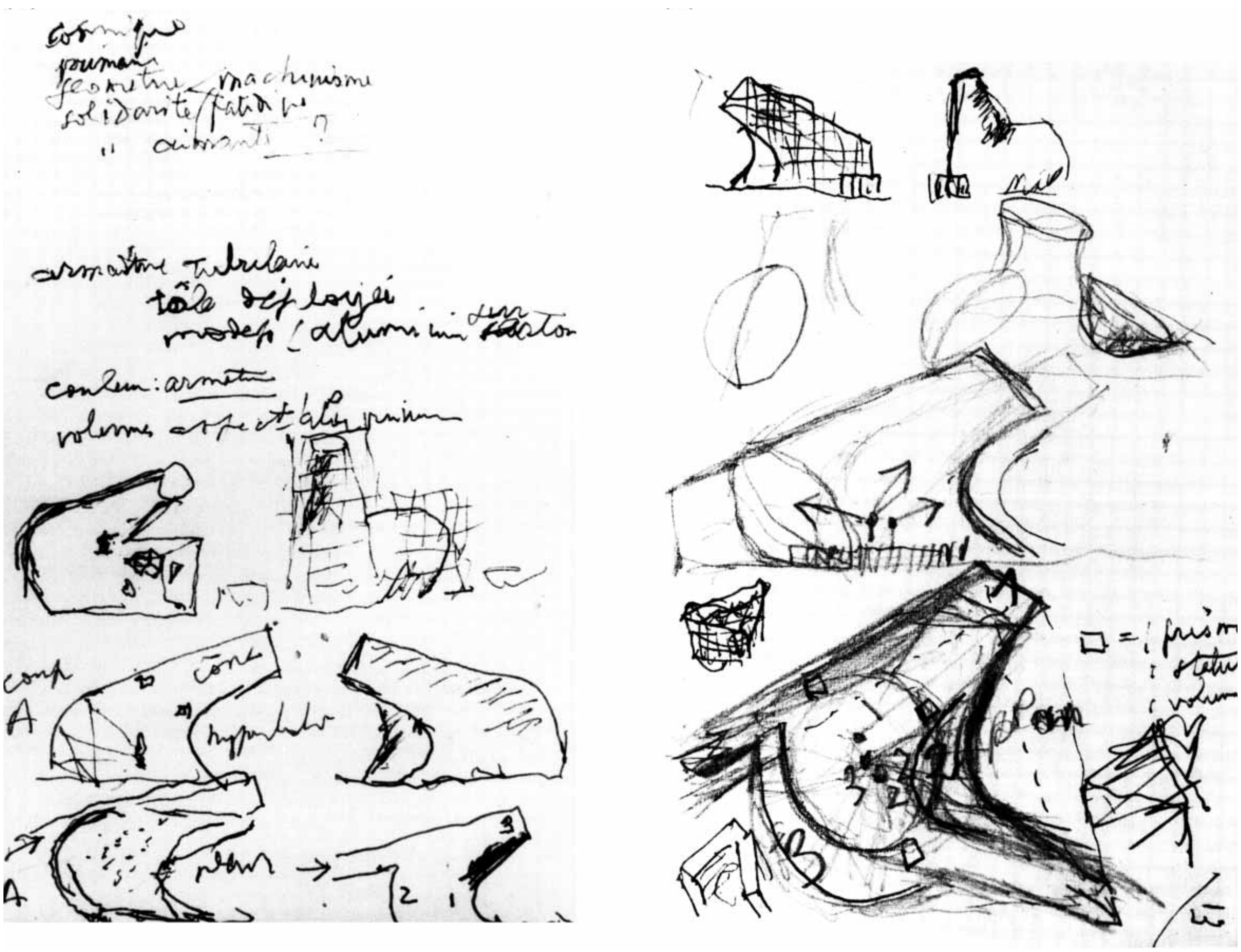

Figure 1.4: Le Corbusier's design sketches for the Philips Pavilion, September – October, 1956 (© 2012 Artists Rights Society, New York/ADAGP, Paris/FLC)

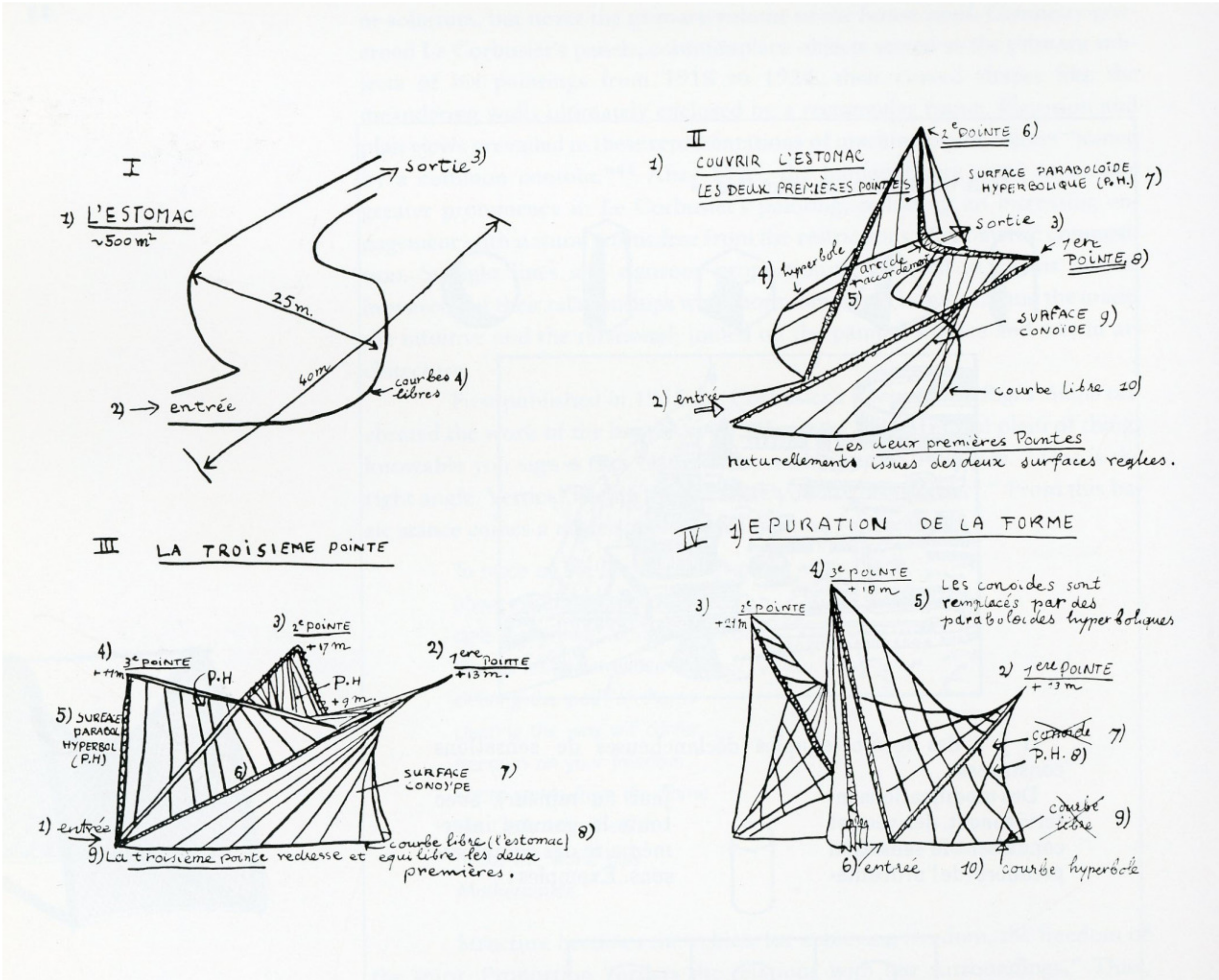

Figure 1.5: Xenakis' early drawings of the Philips Pavilion as documented in volume 20 of the *Philips Technical Review*.

and chance in composition, the same way that physicists were using probability to describe the material world.

To Xenakis' mind, serial music was no less causal than the music it intended to supersede. He described serial music as embodying "virtually absolute determinism."[24] Xenakis saw music theory as a sub-set of mathematics and algebra: While musicians have a different vocabulary, they also use mathematical principles to describe and compose music. Because Xenakis understood mathematics as well as music, he was able to identify how even in serialism and indeterminate music, composers were only utilizing a small subset of algebraic theory. In his own music, Xenakis wanted to generalize and expand the causal framework that musicians and theorists had been using to compose and understand music, paralleling similar

[24] Iannis Xenakis. *Formalized Music*. Pendragon Press, 1992. ISBN 0-945193-24-6

developments in physics and mathematics. As a reference to *chance*, or *stochos*, Xenakis coined the term *stochastic music* to describe his development.[25]

[25] **TODO: Clarify**

Xenakis' book, *Formalized Music* gives a verbose explanation of stochastic music. Some authors have interpreted his description more explicitly. In *Audible Design*, Trevor Wishart describes the stochastic process used to compose stochastic music as:

> "A process in which the probabilities of proceeding from one state, or set of states, to another, is defined. The temporal evolution of the process is therefore governed by a kind of weighted randomness, which can be chosen to give anything from an entirely determined outcome, to an entirely unpredictable one."[26]

[26] Trevor Wishart. Audible Design, 1994

Xenakis' Reflection In the Spring of 1976, while defending his doctoral thesis at the University of Paris, Xenakis emphasized the relevance of seemingly unrelated disciplines to the creative process. A translation of his defense includes this statement:

> "The artist-conceptor will have to be knowledgeable and inventive in such varied domains as mathematics, logic, physics, chemistry, biology, genetics, paleontology (for the evolution of forms), the human sciences, and history; in short, a sort of *universality*, but one based upon, guided by and oriented toward forms and architectures."[27]

[27] L Russolo. *The Art of Noises*. Monographs in musicology. Pendragon Press, 1986. ISBN 9780918728227

From Xenakis' drawings we can deduce that he used the same tools, skills, and philosophy to imagine and conceive both music and architecture. His approach elevated both forms and blurred the distinction between the two. Perhaps if we had kept using pen and paper to design buildings and write music, the reality today would be closer to the ideal that he imagined.

As the ideas that inspired Xenakis and other progressive 20th century composers were taking root in contemporary music, the culture of artistic form and composition was already beginning the transition into the digital domain. There is no reason why digital tools cannot favor stochastic processes to linearity; there is no reason why digital tools cannot treat music and architecture as equals. However, even today, software for composing music still favors static pitches to glissandi; software for architectural design still favors corners to curves. Most importantly, the software skills that we use to design and manipulate space, and the skills that we use to compose music, mutually exclude each other.

This is where the projects described here make a contribution. By drawing from music, mathematics, computer science, acoustics, audio engineering and mixing, sound reinforcement, multimedia production, and live performance, we can create tools that allow us to indiscriminately compose with space and sound.

1.6 Universality

At the MIT Media Lab, we celebrate the study and practice of projects that exist outside of established academic disciplines. The Media Lab (and the media) have described this approach as interdisciplinary, cross-disciplinary, anti-disciplinary, or post-disciplinary; rejecting the cliché that academics must narrowly focus their studies learning *more and more about less and less*, and eventually knowing *everything about nothing*. The projects described here uphold the vision of both Xenakis and the Media Lab. Each chapter documents the movitaions and implementation of a new tool for manipulating space and sound. Each project draws from an assortment of fields including music, mathematics, computer science, acoustics, audio engineering and mixing, sound reinforcement, multimedia production, and live performance.

2

Spatial Domain: Reflection Visualizer

It was Xenakis' goal for the curved surfaces of the Philips Pavilion to reduce the sonic contribution of sound reflections as much as possible.[1] He knew that reflections and the resulting comb filtering could impair intelligibility and localization of music and sounds. The pavilion was to have hundreds of loudspeakers, and an elaborate custom sequencer electronically selected which speakers Edgard Varèse's music played from. However, large concave surfaces like the ones on the inside of the pavilion can have a focussing effect on acoustic reflections,[2] which may result in severe filtering and phase cancellations. There is evidence that carefully constructed curved surfaces such as stage shells can be effective for performers of acoustic music.[3] In the context of loudspeaker playback, the advantages of curved surfaces over flat (non-parallel) surfaces are ambiguous at best.[4] When we hear an acoustic sound reflection off a concave surface, the sound can arrive at our ears in two possible states:

1. The path of the sound from the source to the reflecting surface to our ears is equidistant for each point on the reflecting surface. Ignoring any direct sound, the reflection arrives in-phase, and the surface acts as acoustic amplifier of the reflection.

2. The path of the sound from the source to the reflecting surface to our ears is slightly different for each point on the surface. All the reflections arrive out of phase with each other.

Parabolic microphones[5,6], and acoustic whispering chambers[7] fall into the first category. Both use surfaces that are angled to reflect all sound emanating from one direction to converge at a certain point. If a concave surface reflecting surfaces is not carefully designed to focus sounds *in phase* it is much more likely that the sounds will arrive out of phase. The curves of the Philips Pavilion probably created an *unusual* acoustic space, rather than a good space for critical listening.

[1] Philips Natuurkundig Laboratorium. *Philips Technical Review*. Number v. 20. Philips Research Laboratory, 1959

[2] Martijn Vercammen. The reflected sound field by curved surfaces. *The Journal of the Acoustical Society of America*, 123(5):3442, 2008. ISSN 00014966. DOI: 10.1121/1.2934246

[3] Peter D'Antonio. Performance Acoustics: The Importance of Diffusing Surfaces and the Variable Acoustics Modular Performance ShelI–VAMPS (TM). In *91st Audio Engineering Society Convention*, New York, 1991

[4] T.J. Cox. Acoustic Diffusers: The Good, the Bad and the Ugly. *Proceeding of the Institute of Acoustics*, 2006

[5] Parabolic microphones use a specially designed parabolic reflector that focuses sound arriving from one direction on the microphone capsule. Handheld models are commonly used in birdsong recording, and on the sidelines of football games. They typically have a diameter of two feet or less, and can capture sound up to 500 feet away. Due to the size of the reflector, commercial parabolic microphones cannot capture sounds below approximately 1kHz.

[6] G Davis and R Jones. *The Sound Reinforcement Handbook*. Recording and Audio Technology Series. Hal Leonard, 1989. ISBN 9780881889000

[7] A room built such that the walls are angled to direct sound from one corner to another. If two people stand in the correct spaces in a whispering chamber, they can clearly hear each other whispering even though they may be at opposite ends of the room.

However, this may have been advantageous to the project: Part of the spectacle was seeing, hearing, and experiencing something completely unprecedented and unlike anything else.

2.1 Composing with Space

If Xenakis had been able to model the reflections and compose them directly into the piece, what would the tools be like? How can we make the difference between in phase reflections and out of phase reflections intuitive? It we had tools available to compose with acoustic reflections, how could we use controlled filtering creatively, and what kind of music, and architectural spaces would we make? The Xenakis inspired Reflection Visualizer is an abstract software tool for experimenting with architectural acoustic lenses. It is intended more as an experiment for architectural or musical brainstorming than as a simulation for analysis of sound propagation. For example:

1. It illustrates sound projection in only two dimensions.
2. It is frequency independent. Real sufaces reflect only wavelengths much smaller than the size of the reflector.[8]
3. Diffraction is ignored.
4. Acoustic sounds waves of higher frequencies propagate more directionally than lower frequencies. This property is ignored.

[8] Zhixin Chen and Robert C. Maher. Parabolic Dish Microphone System, 2005. URL http://www.coe.montana.edu/ee/rmaher/publications/maher_aac_0805.pdf

2.2 Implementation

The Reflection Visualizer was implemented as a web app using the HTML5 Paper.js[9] vector graphics library. Try it out online at http://web.media.mit.edu/~holbrow/mas/reflections/

[9] http://paperjs.org/

2.3 Reflection Visualizer Architectural Example

Assume we are creating the floor plan for a new architectural space and accompanying electronic music performance. We want to use acoustic reflections to create certain sounds at certain locations in the performance space. We can use the this tool to prototype potential layouts. The curved black line in the user interface (figure 2.1) represents a reflective surface. The black dot with emanating red lines represents a sound source, and the sound propagation. Click and drag on any black dot to move the object. Black dots connected by grey lines are handles that re-orient (instead of move) objects. On reflection surfaces, the handles adjust the angle and shape of the surface curve. Handles connected to sound sources adjust the angle and length of

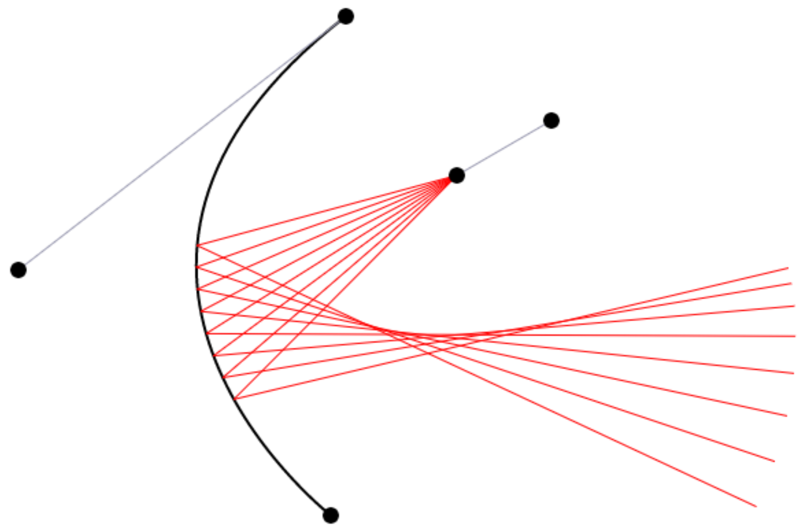

Figure 2.1: Reflection Visualizer user interface.

the sound beams. Each red line emanating from a sound source is the same length, no matter how many times it has been reflected. If it is possible to adjust the length of the red lines such that each one ends at the same spot, it shows that reflections will arrive at that spot in phase. Figures 2.2 and 2.3 show how we can adjust the curve of a surface to focus reflections on a point.

2.4 *Reflection Visualizer Musical Example*

The red emanating lines can also be thought of as stochastic pitch swarms, similar to those Xenakis wrote for Metastasis in 1954 (figure 1.1).
TODO: Add Musical Example

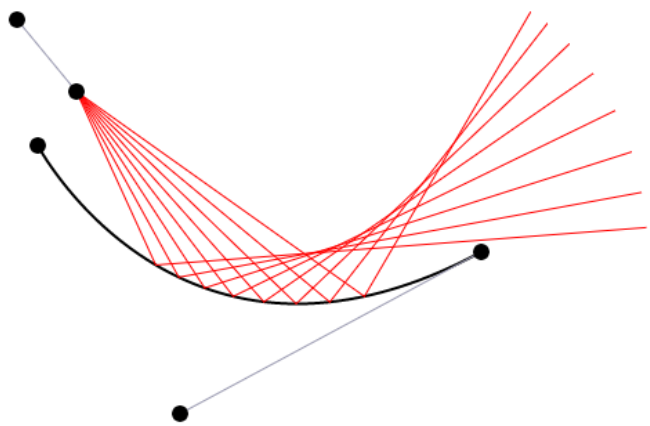

Figure 2.2: Reflections from a 30° loudspeaker arriving out of phase.

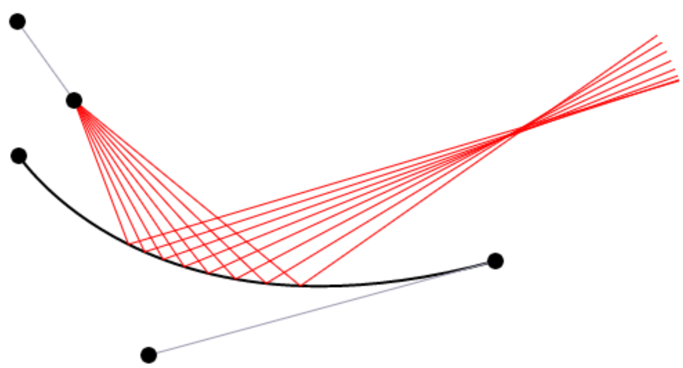

Figure 2.3: By adjusting the curvature of the reflective surface, we can focus the audio reflections.

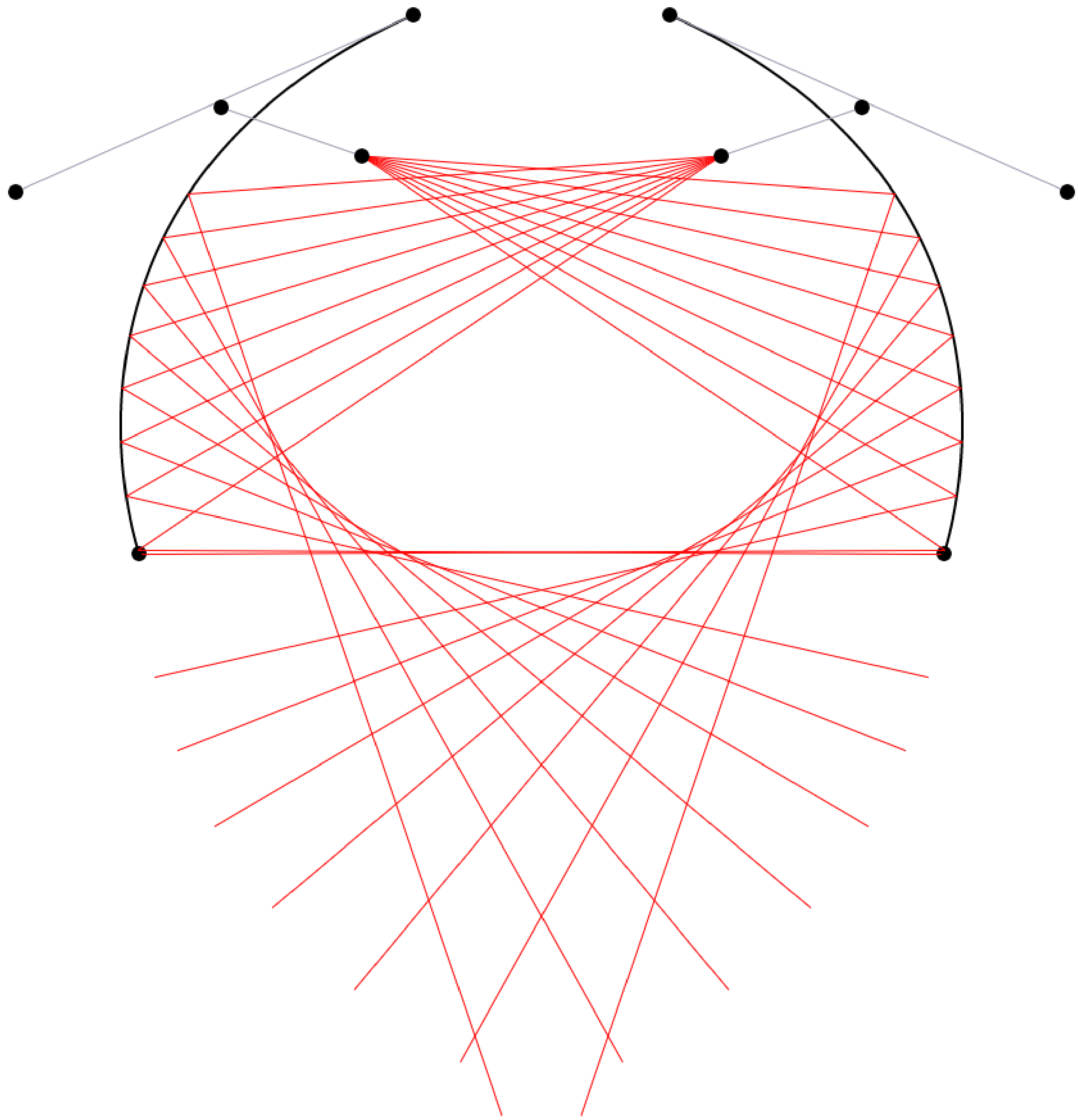

Figure 2.4: Reflection Visualizer user interface.

3
Temporal Domain: Stochastic Tempo Modulation

In the 20th century there was a revival of complexity in contemporary music composition.[1] As performers developed the virtuosic skills required to play the music, composers also wrote increasingly difficult scores to challenge the performers.[2] One composer writing very complex music in this period was Elliot Carter. Carter developed a technique he called *metric modulation*, in which his music would transition from one musical meter to another through a transitional section that shared aspects of both the original meter and the new meter.

While metric modulation is a technique for changing meter, and Stochastic Tempo Modulation is a technique for changing tempo, the former led to the later in a surprising way. Carter's reputation for complexity in music attracted the attention of composer and cellist Tod Machover. While Machover was studying with Carter, he wrote a trio for violin, viola, and cello, in which each instrument would accelerate or decelerate relative to the others. The piece turned out to be so difficult that it was impossible to find anyone who could play it correctly. Faced with this challenge, Machover saw opportunity:

> "A sort of lightbulb went off... computers are out there, and if you have an idea and can learn how to program, you should be able to model it."[3]

If the music was too complex for a human to process, but we can define it formulaically, we can teach a computer to play sounds that a human cannot. Stochastic Tempo Modulation builds on this idea with inspiration from Xenakis.

3.1 Stochos

In chapter 1 (see figure 1.1) we saw how Xenakis used ruled surfaces in his composition to compose swarms of notes that move together, creating stochastic sonorities. The goal of Stochastic Tempo Modulation

[1] Works by Italian composer Luciano Berio illustrate the complexity of postwar contemporary music. Beginning in 1958, Berio wrote a series of works he called *Sequenza*. Each was a highly of highly technical composition written for a virtuosic soloist. Each was for a different instrument ranging from flute to guitar to accordion. In Sequenza IV, for piano, Berio juxtaposes thirty-second note quintuplets, sextuplets, and septuplets (each with a different dynamic), over just a few measures.

[2] D. J. Grout, J. P. Burkholder, and C. V. Palisca. *A History of Western Music*. W.W. Norton, 7th edition, 2006. ISBN 0-393-97991-1

[3] Evan Fein. Q&A With Tod Machover, 2014. URL `http://www.juilliard.edu/journal/1402/hyperinstruments-crowd-sourced-symphonies`

is to enable composition with swarms of tempo modulations that move in correlated, cohesive patterns. Music with two or more simultaneous tempos (polytempic music) is itself not a new concept, and many examples of polytempic music exist.[4] Slightly less common is polytempic music where continuous tempo accelerations or decelerations are defined relative to each other. This style of music is well suited to tape music, because tape machines can play recordings back at variable rates. However, it is difficult to control the exact point (or phase) when de-synchronized tape becomes re-aligned. Performative music with simultaneous tempi that accelerate and decelerate relative to each other is unusual, but does exist. In a 1971 interview composer Steve Reich described how he made the transition to performative polytempic music after working on his tape music composition, *Come Out*:

[4] John Greschak. Polytempo Music, An Annotated Bibliography, 2003. URL http://www.greschak.com/polytempo/ptbib.htm

> "1966 was a very depressing year. I began to feel like a mad scientist trapped in a lab: I had discovered the phasing process of Come Out and didn't want to turn my back on it, yet I didn't know how to do it live, and I was aching to do some instrumental music. The way out of the impasse came by just running a tape loop of a piano figure and playing the piano against it to see if in fact I could do it. I found that I could, not with the perfection of the tape recorder, but the imperfections seemed to me to be interesting and I sensed that they might be interesting to listen to."[5]

[5] Michael Nyman. Steve Reich. *The Musical Times*, 112:229–231, 1971

Reich's experience illustrates what other composers and performers have also encountered: It is quite difficult to perform polytempic music accurately. In *Piano Phase* Reich has two performers playing the same 12 tone series on the piano. After a set number of repetitions through the pattern, one performer begins to play slightly faster until she is exactly one note ahead of the other performer, at which point both performers play at the same rate for a time. This process is repeated and iterated on, creating a live *phasing* effect without the pitch shifting that would occur when phasing analog tape. If we compare a live performance[6] with a programatic rendering[7] of *Piano Phase*, we can hear how the programatic rendering is able to accelerate more smoothly. The programatic example spends longer on the transitions where the two parts are out of phase.

[6] Tine Allegaert and Lukas Huisman. Reich: Piano Phase, 1989. URL www.youtube.com/watch?v=i0345c6zNfM

[7] Alexander Chen. Pianophase.com, 2014. URL https://vimeo.com/98823512

3.2 *Objective*

Steve Reich composed *Piano Phase* for two performers. Through experimentation, he found that if the music is reasonably simple, two performers can make synchronized tempo adjustments relative to each other well enough to yield compelling results. Stochastic Tempo Modulation allows us to write music with many more simultaneous tempi. However, the requirements are probably too demanding for unassisted performers. Our goal is to compose and audition music where:

1. Swarms of an arbitrary number of simultaneous tempi coexist.

2. Each individual player within the swarm can continuously accelerate or decelerate individually, but also as a member of a cohesive whole.

3. Each musical line can converge and diverge at explicit points. At each point of convergence the phase of the meter within the tempo can be set.

We start by defining a single tempo transition. Consider the following example (shown in figure 3.1):

- Assume we have 2 snare drum players. Both begin playing the same beat at 90 BPM in common time.

- One performer gradually accelerates relative to the other. We want to define a continuous tempo curve such that one drummer accelerates to 120 BPM.

- So far, we can easily accomplish this with a simple linear tempo acceleration. However, we want the tempo transition to complete exactly when *both* drummers are on a down-beat, so the the combined effect is a 3 over 4 rhythmic pattern. **Linear acceleration results in the transition completing at an arbitrary phase.**

- We want the accelerating drummer to reach the new tempo after exactly 20 beats.

- We also want the acceleration to complete in exactly 16 beats of the original tempo, so the drummer playing a constant tempo, and the the accelerating drummer are playing together.

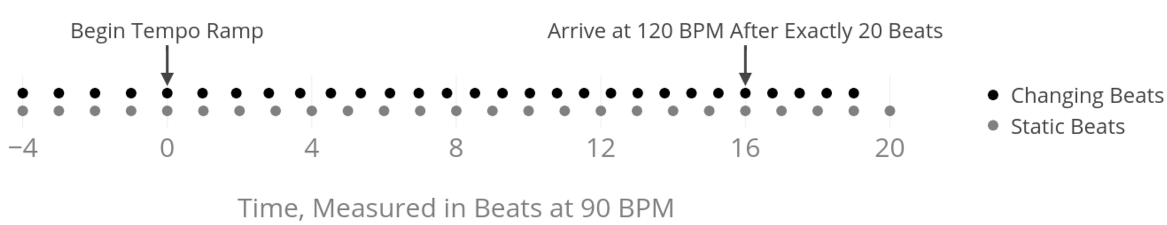

Figure 3.1: Tempo Transition from 90 BPM to 120 BPM

3.3 Solution

We are interested in both the number of beats elapsed in the static tempo *and* in the changing tempo, and the absolute tempo. If we think of the number of beats elapsed as our *position*, and the tempo as our *rate*, we see how this resembles a physics problem. If we have a function that describes our tempo (or rate), we can integrate that function, and the result will tell us our number of beats elapsed (or position). Given the above considerations, our tempo curve is defined in terms of 5 constants:

- Time $t_0 = 0$, when the tempo transition begins
- A known time, t_1, when the tempo transition ends
- A known starting tempo, $\dot{x}_0$
- A known finishing tempo, $\dot{x}_1$
- The number of beats elapsed in the changing tempo between t_0 and t_1, x_1

The tension of the tempo curve determines how many beats elapse during the transition period. The curve is well-defined for some starting acceleration a_0 and finishing acceleration a_1, so we define the curve in terms of linear acceleration. Using Newtonian notation we can describe our tempo acceleration as:

$$\ddot{x}_1 = a_0 + a_1 t_1 \tag{3.1}$$

Integrating linear acceleration (3.1) yields a quadratic velocity curve (3.2). The velocity curve describes the tempo (in beats per minute) with respect to time.

We must specify the same time units for input variables like t_1 and $\dot{x}_1$. I prefer *minutes* for t_1 and *beats per minute* for $\dot{x}_1$ over *seconds* and *beats per second*.

$$\dot{x}_1 = \dot{x}_0 + a_0 t_1 + \frac{a_1 t_1^2}{2} \tag{3.2}$$

Integrating velocity (3.2) gives us a function describing position (the number of beats elapsed with respect to time).

$$x_1 = x_0 + \dot{x}_0 t_1 + \frac{a_0 t_1^2}{2} + \frac{a_1 t_1^3}{6} \tag{3.3}$$

With equations (3.2) and (3.3), we can solve for our two unknowns, a_0 and a_1. First we solve both equations for a_1:

$$a_1 = \frac{-2}{t_1^2}(\dot{x}_0 - \dot{x}_1 + a_0 t_1) = \frac{-6}{t_1^3}(\dot{x}_0 - x_1 + \frac{a_0 t_1^2}{2})$$

Assuming $t_1 \neq 0$, we solve this system of equations for a_0:

$$a_0 = \frac{6x_1 - 2t_1(\dot{x}_1 + 2\dot{x}_0)}{t_1^2} \tag{3.4}$$

Evaluating (3.4) with our constants gives us our starting acceleration. Once we have a_0 we can solve (3.2) for a_1, and evaluate (3.2) with a_1 and a_0 to describe our changing tempo with respect to time.

3.4 *Stochastic Transitions*

Equipped with out equations from the previous section, it is quite simple to create swarms of parallel tempos that are correlated and complex. In figure 3.2 we build on the previous example. Here, each additional tempo curve is calculated the same way, except x_1 (number of beats in our accelerating tempo during the transition) is incremented for each additional tempo line.

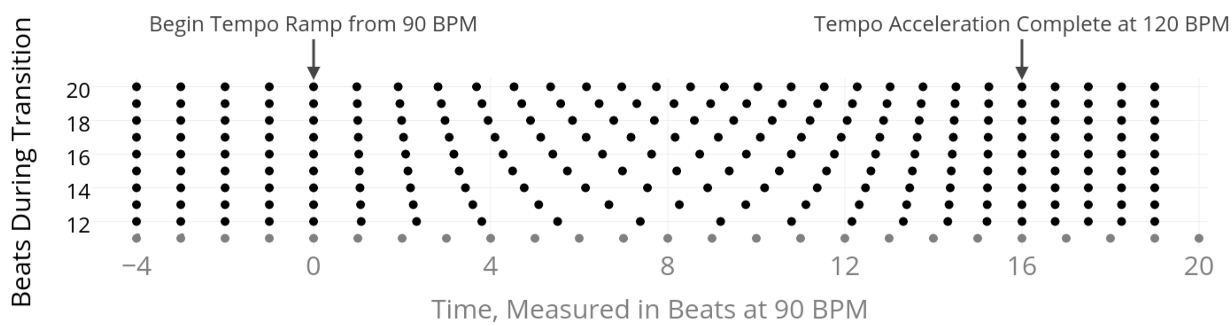

Figure 3.2: Stochastic Tempo Transition from 90 BPM to 120 BPM. Black dots are beats in our changing tempi. Grey dots show a continuation of beats at the initial tempo. $12 \leq x_1 \leq 20$

This pattern clearly exhibits controlled chance that Xenakis would describe as *stochastic*. On the very first beat at $t = 0$, all parallel parts are aligned. Beats 2 and 3 can be heard as discrete rhythmic events, but become increasingly indistinct. The end of beat 4 then overlaps with the start of beat 5, before articulated beats transition to pseudo random noise. By beat 13 of the static tempo, the chaos of the many accelerating tempi begin to settle back into order before returning to complete syncronicity at $t = 16$.

3.5 *Tide: Composition with Stochastic Tempo Modulation*

An earlier version of the equation derived here was packaged as a patch for the Max[8] graphical programming language. Composer Bryn Bliska developed a user interface and used it in the composition of *Tide*.[9] While this ealier version of the equation did not support a variable x_1 parameter, *Tide* uses Stochastic Tempo Modulation to drive the phasing tempos of the bell-like synthesizers throughout the piece for two to three simultaneous tempi.

[8] https://cycling74.com/products/max/

[9] Available online: `http://web.media.mit.edu/~holbrow/mas/Tide_Bliska_Holbrow.wav`

3.6 Prior Work

Many commercial and research projects deal with different ways to manipulate rhythm and tempo. Flexible digital audio workstations (DAWs) like Cockos Reaper[10] and MOTU Digital Performer[11] include features for auditioning tracks or music-objects with unique simultaneous tempi, and individual tempos can even be automated relative to each other. However, the precise non-linear tempo curves that are required for the syncopated musical content to synchronize correctly after a transition completes are not possible in any DAW we tried. Audio programming languages like Max and SuperCollider[12] could be used to create tempo swarms, but require equations like the ones defined in section 3.3. One project, *Realtime Representation and Gestural Control of Musical Polytempi*[13] demonstrates an interface for generating Polytempic music, but is not intended or capable of generating coordinated or stochastic tempi swarms. *The Beatbug Network*[14] is described as a multi-user interface for creating stochastic music, but is focused on "beats," or musical rhythmic patterns, and timbres, rather than tempi. *Stochos*[15] is a software synthesizer for generating sound using random mathematical distributions, but is also not designed to work with simultaneous tempos or even as a rhythm generator. Finally, *Polytempo Network*[16] is a project that facilitates the performance of polytempic music, but does not aid the composition thereof.

[10] http://www.reaper.fm

[11] http://www.motu.com/products/software/dp

[12] http://supercollider.github.io/

[13] Chris Nash and Alan Blac. Realtime Representation and Gestural Control of Musical Polytempi. In *New Interfaces for Msical Expression*, pages 28–33, Genova, Italy, 2008

[14] Gil Weinberg, Roberto Aimi, and Kevin Jennings. The Beatbug Network - A Rhythmic System for Interdependent Group Collaboration. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 186–191, Dublin, Ireland, 2002. URL http://www.nime.org/proceedings/2002/nime2002_186.pdf

[15] Sinan Bokesoy and Gerard Pape. Stochos: Software for Real-Time Synthesis of Stochastic Music. *Computer Music Journal*, 27(3):33–43, 2003. ISSN 0148-9267. DOI: 10.1162/014892603322482510

[16] Philippe Kocher. Polytempo Network: A System for Technology-Assisted Conducting. *Proceedings of the International Computer Music Conference*, 163 (September):532–535, 2014

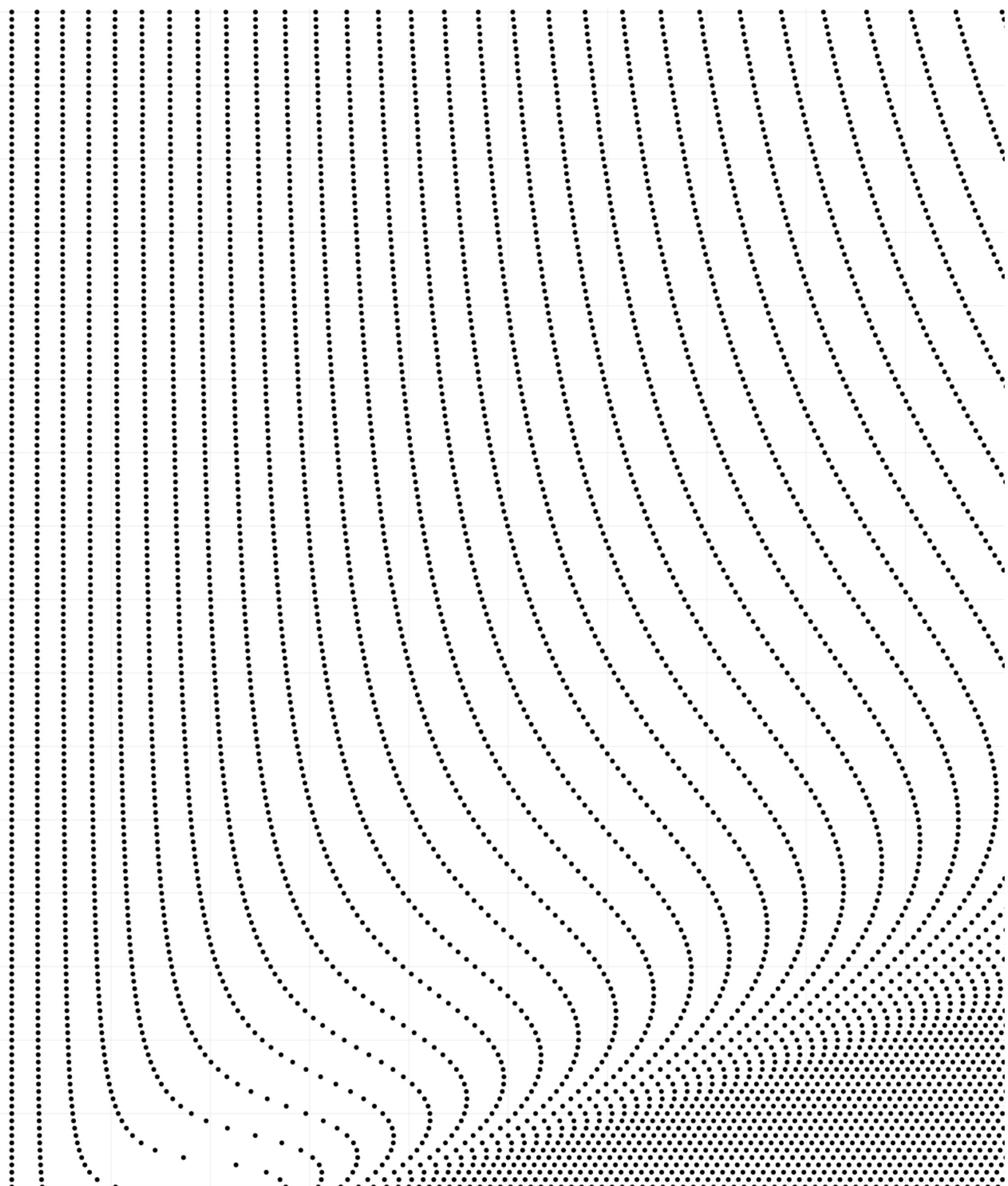

Figure 3.3: Stochastic Tempo Modulation with variable t_1, variable x_1, and 161 simultaneous tempi.

4
The Hypercompressor

The motivation for Hypercompression came during the development of Vocal Vibrations, an interactive music installation about the human voice and about engaging the public in singing.[1] The project featured a Music Concrète composition, *The Chapel* by Tod Machover, which was mixed in a 10 channel surround sound format and played throughout the installation. During the mixing process, I noticed an important surround sound tool missing from my mixing workflow. When mixing in mono or stereo, audio compression lets us meticulously shape and balance sounds in time. I found myself wishing I could shape and position sounds in space just as easily.

[1] Charles Holbrow, Elena Jessop, and Rebecca Kleinberger. Vocal Vibrations: A Multisensory Experience of the Voice. *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 431–434, 2014. URL http://www.nime.org/proceedings/2014/nime2014_378.pdf

Unless noted otherwise, "compression" is used in this thesis to describe dynamic range compression, as opposed to data compression.

4.1 Building on the Compression Paradigm

The design, implementation, and use of traditional dynamic range compression is well documented in the literature,[2] so we will describe dynamic range compression only as much as is needed to explain the foundation for Hypercompression. Imagine we are mixing a vocal pop performance, and during the verse our vocalist is singing moderately loud, or *mezzo-forte*. At the beginning of the chorus, our singer wants a full and powerful sound, so she adjusts the dynamic to very loud, or *fortissimo*. However, the new louder dynamic interrupts the balance between the vocals and the other instruments in our mix. We like the powerful sound of our singer's *fortissimo* performance, but our balance would be improved if we had the volume of a *forte* performance instead. One option is to manually turn down the vocalist during the chorus, which in some cases this is the best solution. When we want more precise control, we can use a compressor.

[2] Dimitrios Giannoulis, Michael Massberg, and Joshua D Reiss. Digital Dynamic Range Compressor Design - A Tutorial and Analysis. *Journal of the Audio Engineering Society*, 60 (6):399–408, 2012; Alex Case. *Sound FX: Unlocking the Creative Potential of Recording Studio Effects*. Focal Press, 2007. ISBN 978-0240520322; and Emmanuel Deruty, Francois Pachet, and Pierre Roy. Human-Made Rock Mixes Feature Tight Relations. *Journal of the Audio Engineering Society*, 62(10), 2014

Traditional Compression

A compressor is essentially an automated dynamic volume control. Most compressors include at least four basic parameters in the user interface that allow us to customize its behavior: *threshold*, *ratio*, *attack time*, and *release time*. We can send our vocalist's audio signal through a compressor, and whenever her voice exceeds the gain level set by our threshold parameter, the signal is automatically attenuated. As the input signal further exceeds the threshold level, the output is further attenuated relative to the input signal. The ratio parameter determines the relationship between the input level and output level as shown in figure 4.1.

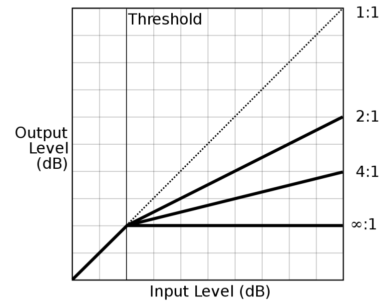

Figure 4.1: "Compression ratio" by Iain Fergusson. Licensed under Public Domain via Wikimedia Commons https://commons.wikimedia.org/wiki/File:Compression_ratio.svg#/media/File:Compression_ratio.svg

Threshold and ratio settings are essential for controlling dynamic range, but the power and creative flexibility of the compressor comes with the attack time and release time parameters. These parameters determine the speed at which the compressor attenuates (attack time) and disengages (release time) when the input signal exceeds the threshold. By adjusting the attack and release times, we can change the temporal focus of the compressor.

- Perhaps we want the compressor to engage or disengage at the time scale of a musical phrase. We could set our attack time long enough to let transients through without engaging the compressor significantly (try 20 milliseconds). If our release time is quite long (try 300 milliseconds), and we set our threshold and ratio carefully, we might be able to convince the compressor to smooth musical phrases.

- If we want our compressor to focus on syllables instead of phrases, we can shorten our attack and release times (try 10 milliseconds and 40 milliseconds respectively). When the compressor engages and disengages at each syllable, it imparts a different quality (sometimes described as "punch").

- If we reduce our attack and release parameters enough, we can instruct our compressor to engage and disengage at the time scale of an audio waveform, compressing individual cycles. This will distort an audio signal, adding odd order harmonics,[3] and imparting an entirely different quality.

[3] Not every compressor model can react quickly enough to distort a waveform. The Dbx 160 and Teletronix LA2A are known to be fast enough to distort.

The attack and release times listed here are a rough guide only. The exact function of these parameters varies from one model of compressor to another, and results also depend on the audio input material and on the threshold and ratio settings. The results of audio compression can sometimes be characterized better by a feeling than a formula.

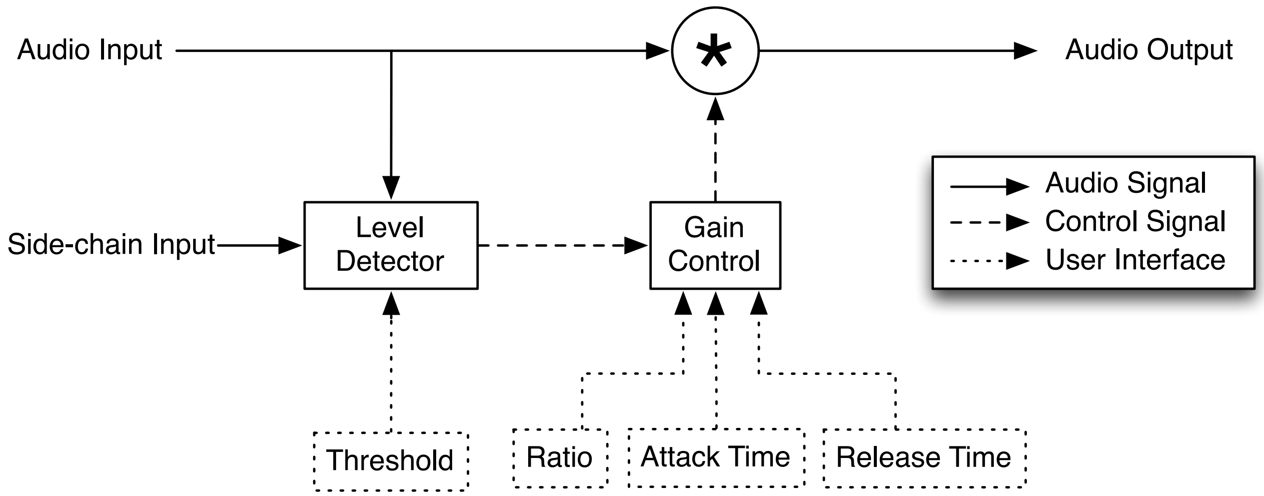

Figure 4.2: Block diagram of a simple traditional dynamic range compressor.

Side-Chain Compression

Compressors often have an additional operational mode that is the primary inspiration for Hypercompression. We know that compressors automatically reduce the gain of a signal that exceeds a given threshold. Some compressors allow us to attenuate the level of a signal when a *different* signal exceeds the threshold level. Models that suports side-chain compression have a second audio input. When we switch the compressor into side-chain mode, the compressor attenuates the first signal only when the second signal exceeds the threshold.

Side-chain compression is often used to moderate the balance of kick drum and bass guitar. If the bass guitar is briefly attenuated just enough just the right amount each time the kick drum hits, we can set the kick and bass guitar at exactly the gain levels we want without one masking the other. Because the bass guitar is only briefly attenuated, it will not be perceived as any quieter.

In this example we use the kick drum to create a gain envelope for our bass guitar. The kick *pushes* the bass to make room for itself. The attack time and release time parameters give control over this behavior in the temporal domain. The next step is to expand this model to add control in the spatial domain.

4.2 *Ambisonics*

Ambisonics is a technique for encoding and decoding three-dimensional surround sound audio.[4] Ambisonic audio differs from other surround sound formats like 5.1 and 7.1 in that it does not depend on a particular speaker configuration. An ambisonic recording can be decoded on

[4] Michael Gerzon. Periphony: With-Height Sound Reproduction. *Journal of the Audio Engineering Society*, 21(1):2–10, 1973; and Michael Gerzon. Ambisonics in Multichannel Broadcasting and Video. *Journal of the Audio Engineering Society*, 33:859–871, 1985. ISSN 00047554

any surround sound speaker configuration without disarranging the spatial contents of the audio recording.

Imagine we use an omnidirectional microphone to record an acoustic instrument at a sample rate of 44.1 kHz. We sample and record 44100 samples every second that represent the air pressure at the microphone capsule during the recording. Our omni-directional microphone is designed to treat sound arriving from all angles equally. The omnidirectional microphone sums together sounds arriving from all angles and the acoustic directional information is lost.

If we want to encode, decode, transmit, or play audio that preserves full sphere 360 degree information, ambisonics offers a solution. Ambisonic audio uses *spherical harmonics* to encode surround sound audio that preserves the direction-of-arrival information that discrete channel recordings (such as mono and stereo) cannot fully capture.

Spherical Harmonics

We know that we can construct any monophonic audio waveform by summing a (possibly infinite) number of harmonic sine waves (Fourier series).[5] For example, by summing odd *order* sine harmonics at a given frequency f, $(1f, 3f, 5f, 7f, \dots)$, we generate a square wave with fundamental frequency f. As the order increases, so does the temporal resolution of our square wave.

[5] An excellent description of the transformation between the time domain and frequency domain can be found at http://betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/

By summing sinusoidal harmonics, we can generate any continuous waveform defined in two dimensions (one input parameter and one output). Similarly, by summing *spherical harmonics*, we can generate any continuous shape defined over the surface of a three-dimensional sphere (two input parameters, or polar angles, one output). Where a traditional monophonic audio encoding might save one sample 44100 times per second, an ambisonic encoding would save one sample *for each spherical harmonic* 44100 times per second. This way we capture a three-dimensional sound image at each audio sample. The number of spherical harmonics we encode is determined by our *ambisonic order*. As our ambisonic order increases, so does the angular resolution of our result on the surface of the sphere.

Spherical Harmonic Definition

For encoding and decoding ambisonics, the convention is to use the real portion of spherical harmonics as defined in equation 4.1, where:

- $Y_n^m(\varphi, \vartheta)$ is a spherical harmonic that is:
  - of order, n
  - of degree, m

Some literature on spherical harmonics swaps the names of *order* and *degree*. In this thesis we use Y_{order}^{degree}. In literature where Y_{degree}^{order} is used, the function of the subscript and superscript remain unchanged; only the names are inconsistent.

  - defined over polar angles (φ, ϑ)

- $N_n^{|m|}$ is a normalization factor.[6]
- $P_n^{|m|}$ is the associated Legendre function of order n and degree m.

$$Y_n^m(\varphi, \vartheta) = N_n^{|m|} P_n^{|m|}(\sin\vartheta) \begin{cases} \sin|m|\varphi, & \text{for } m < 0 \\ \cos|m|\varphi, & \text{for } m \geq 0 \end{cases} \tag{4.1}$$

[6] In ambisonic literature (and software), there are multiple incompatible conventions for the normalization of spherical harmonics. The Hypercompressor uses the *Furse-Malham* (FuMa) normalization convention.

Given equation 4.1, we can define an ambisonic audio recording as:

$$f(\varphi, \vartheta, t) = \sum_{n=0}^{N} \sum_{m=-n}^{n} Y_n^m(\varphi, \vartheta)\phi_{nm}(t) \tag{4.2}$$

Where:

- φ and ϑ describe the polar angle of sound arrival in two dimensions.[7]
- t is time
- $\phi_{nm}(t)$ are our *expansion coefficients*, described below.

[7] Note that ambisonics uses polar angles to describe the angle of arrival of sound. These are similar to spherical coordinates, minus the inclusion of *radial distance*. Distance is not part of the ambisonic specification.

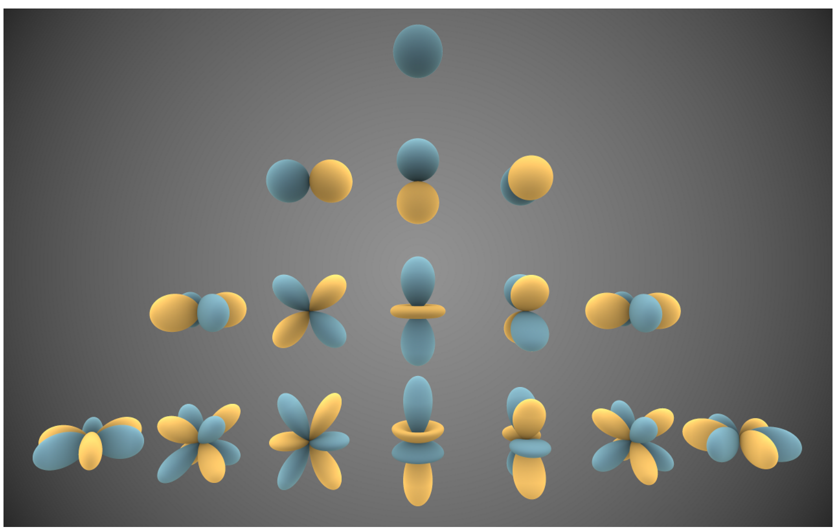

Figure 4.3: Spherical harmonics 0th order (top row) through 3rd order (bottom row). This for image shows the output of $Y_n^m(\varphi, \vartheta)$ for $n = 0, n = 1, n = 2,$and $n = 3$. The distance of the surface from the origin shows the value at that angle. Darker blue regions are positive, while lighter yellow regions are negative. Image credit: Ingo Quilez, licensed under *Creative Commons Attribution-Share Alike 3.0 Unported*.

Spherical Harmonic Expansion Coefficients

In our monophonic recording example, we save just one digital sample 44100 times per second, with each saved value representing the air pressure at a point in time. We know that by summing the correct combination of spherical harmonics, we can describe any continuous function over the surface of a sphere. Instead of sampling air pressure directly, we sample a coefficient describing the weighting of each spherical harmonic 44100 times per second. The resulting sphere encodes the pressure including the direction of arrival information. The weighting coefficients or *expansion coefficients* are recorded in our audio file instead of values representing air pressure directly. Now, by summing together our weighted spherical harmonics, we can reconstruct the fluctuations in pressure including the angle of arrival information. We can recall this snapshot of information at our 44.1 kHz audio sample rate.

Ambisonic Encoding

There are two ways to create an ambisonic recording. First, we can use a soundfield microphone to record an acoustic soundfield. Soundfield microphones like the one developed by Calrec Audio can capture angle of arrival information with the spatial resolution of first order ambisonics.[8] Alternatively, we can algorithmically encode pre-recorded sources, creating virtual sources in an ambisonic bus.[9]

[8] Ken Ferrar. Soundfield Microphone: Design and development of microphone and control unit, 1979. URL http://www.ai.sri.com/ajh/ambisonics/wireless-world-farrar-10-1979.pdf

[9] D G Malham and A Myatt. 3-D sound spatialization using ambisonic techniques. *Computer music journal*, 19 (4):58–70, 1995. ISSN 0148-9267. DOI: 10.2307/3680991

4.3 *Ambisonic Conventions used for Hypercompression*

This thesis follows ambisonic convention for describing axis of rotation. The x-axis points forward, the y-axis point left, and the z-axis points up. Polar angles are used to describe orientation with 0° azimuth being forward, and increasing as we move to the right. 0° elevation also points forward, and increases as we move upward, with 90° being strait up along the z-axis. When working with ambisonics, multiple incompatible conventions exist for ordering and normalizing spherical harmonics.[10] The Hypercompressor uses *Furse-Malham* normalization (FuMa)[11], and first order ambisonics with *B-format*[12] channel ordering. B-format ordering labels the four first-order ambisonic channels as W, X, Y, and Z, with W being the spherical harmonic of order zero and degree zero, and X, Y, and Z being the pressure gradient components along their respective axes.

[10] Christian Nachbar, Franz Zotter, Etienne Deleflie, and Alois Sontacchi. AMBIX - A Suggested Ambisonics Format. In *Ambisonics Symposium 2011*, 2011

[11] D G Malham. Higher order Ambisonic systems, 2003. URL http://www.york.ac.uk/inst/mustech/3d_audio/higher_order_ambisonics.pdf

[12] Florian Hollerweger. An Introduction to Higher Order Ambisonic, 2008. URL http://flo.mur.at/writings/HOA-intro.pdf

4.4 *Hypercompressor Design*

The Hypercompressor (or ambisonic compressor) combines the traditional model of compression with the surround sound capability of ambisonics. Given ambisonic input, and an optional ambisonic side-chain input, the ambisonic compressor is intended to process our input material in one of two modes:

1. Standard mode: We set a compression threshold, similar to on a traditional compressor. When a region in our surround sound

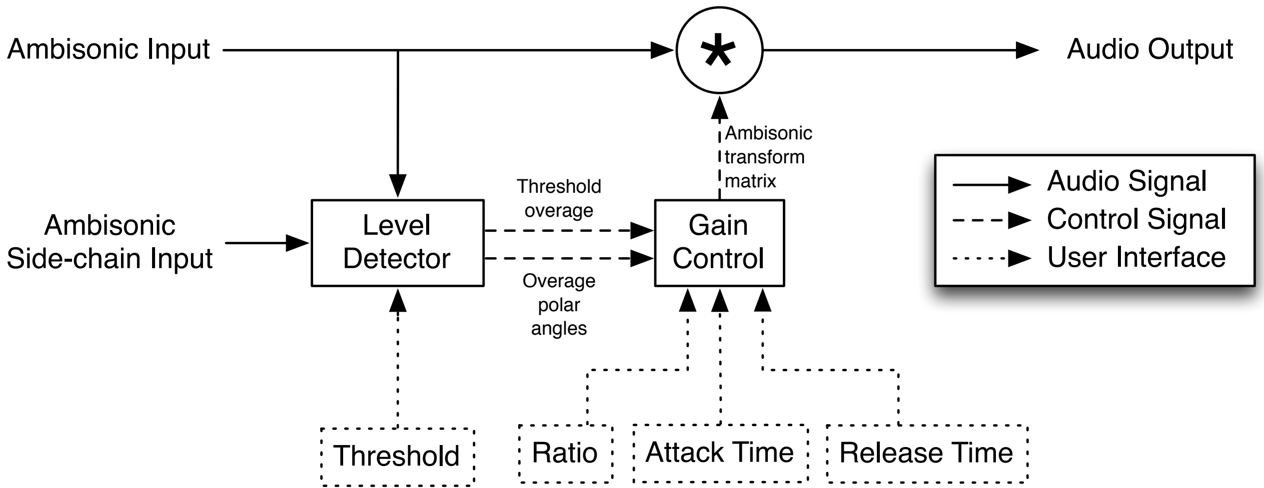

Figure 4.4: Hypercompressor block diagram

input material exceeds the set threshold, the compressor engages and attenuates only that region.

2. Side-chain mode: This mode takes advantage of a second ambisonic input to our signal processor. When the gain of spatial region in our secondary input exceeds our threshold, we attenuate that same region in the the main input, and output the results.

In both modes, our ambisonic compressor must attenuate and release attenuation according to the attack time and release time parameters. The block diagram for our new hypercompressor (figure 4.4) can remain largely unchanged from the the block diagram for our traditional compressor in figure 4.2. The most important changes are:

- Our audio signals must be updated to handle encoded ambisonics. This is as simple as increasing the number of channels on each solid black connection in figure 4.2. The hypercompressor works with first order ambisonics, so every audio path must carry four audio channels.

- On a traditional compressor, the level detector only needs to detect the difference between the gain of the input signal and the gain specified by the threshold parameter. Our ambisonic level detector needs to decode the incoming signals and identify both a threshold overage and the region where the overage occurred.

- Our gain control module needs to listen to the input coming from the level detector module and be able to attenuate the specific regions that exceed our threshold parameter.

Level Detection Module

In *Spatial Transformations for the Alteration of Ambisonic Recordings*, Matthias Kronlachner describes one approach for making a visual ambisonic level meter:[13]

[13] Matthias Kronlachner. *Spatial Transformations for the Alteration of Ambisonic Recordings*. Master's thesis, Graz University of Technology, 2014a

1. Choose a series of discrete points distributed on the surface of a sphere. Ideally the points are equally distributed, so the vertices of platonic solid shapes like the dodecahedron (12-sided polyhedron) and icosahedron (20-sided polyhedron, figure 4.5) work well. For spatial accuracy, Kronlachner recommends a spherical *t*-design with 240 points described by Hardin and Sloane.[14]

[14] R. Hardin and N. Sloane. McLaren's Improved Snub Cube and Other New Spherical Designs in Three Dimensions. *Discrete Computational Geometry*, 15: 429–441, 1996

2. Evaluate each spherical harmonic at every point chosen. Cache the results in a matrix.
3. With the cached spherical harmonics, it is then possible to calculate the RMS and peak values more efficiently at the audio rate.
4. A level meter does not need to refresh the display at the audio sample rate, so it is acceptable to interpolate between the points on the sphere and update the graphical representation at the control rate, which could be as slow as 30 Hz (approximately every 33 milliseconds).

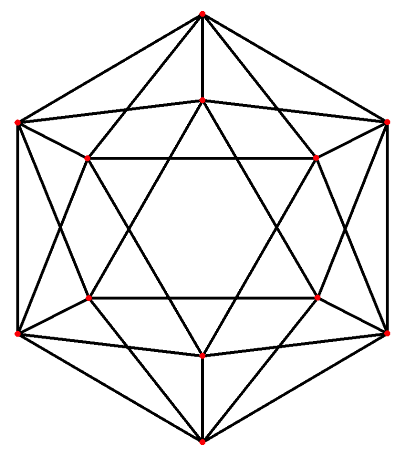

Figure 4.5: An icosahedron.

A similar approach can be used to make an ambisonic level detector. However, a compressor needs to react much quicker than a level meter. The compressor cannot even *begin* to engage until the level meter has responded, and attack times faster than 33 milliseconds are common in conventional compression. Every point on the sphere requires a buffer to calculate the RMS. We also need to decode ambisonics at the audio sample rate and keep track of peak values. Ideally we would also interpolate between the points.

An Efficient Level Detection Module

The Hypercompressor needs to detect the level of our ambisonic input material and identify (as quickly as possible) when and where the signal exceeds the compressor threshold. In the interest of computational efficiency, the first level detector I wrote attempted to extract overage information with minimal ambisonic decoding and signal processing.

1. To accurately play a first order ambisonic encoding, we need a minimum of 6 speakers placed around the listener. In this level detector, we calculate the root mean square (RMS) average at the center of 6 lobes corresponding to the first order spherical harmonics: front, rear, left, right, top, and bottom.

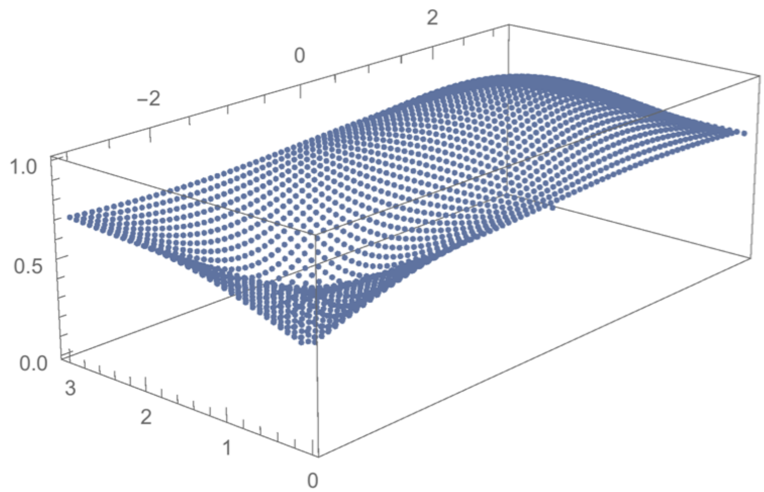

Figure 4.6: Calculating the cylindrical projection of single ambisonic panned source in the Wolfram Mathematica software package

Figure 4.7: Influence maps of 3 first-order spherical harmonics: left, top, and front. Pure white is −0 dBFS black is -inf dBFS. Cylindrical projection.

2. Calculate a map of the influence of each lobe on the surround image**TODO: Clean** (figures 4.6, 4.7). For example, pan a monophonic sound directly forward in an ambisonic mix, cache an image of the resulting sound sphere. Save one image for each of the 6 lobes.

3. We have 6 images, each representing one of the 6 lobes of our first order ambisonic spherical harmonics. In step 1, we calculated the RMS level at each of the corresponding points on our surround sphere. Use the 6 RMS levels to weight each of our 6 maps. The sum of the weighted maps shows the gain distributed across our ambisonic sphere.

Ambisonic Efficient Level Detection Module Results If the input to the level detector is encoded as an ambisonic plane wave, this level detector does yield accurate results. In the more common case, when our ambisonic input material contains multiple sources that are each ambisonically panned to different positions, this interpolation technique does not accurately calculate the RMS at any angle. In simple cases, where we can be sure our input material is appropriate, the technique described here might be useful, but in most cases, a different approach will be more effective.

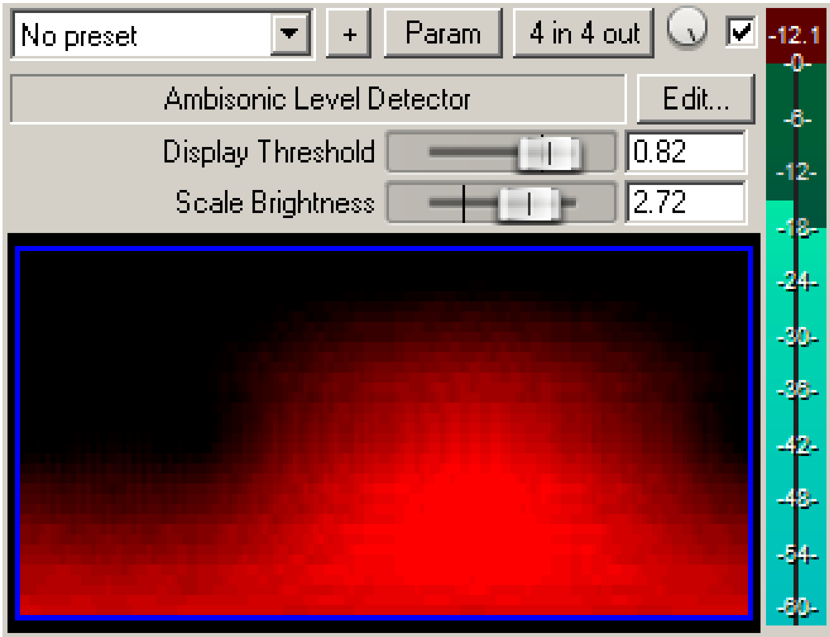

Figure 4.8: The Hypercompressor visualizer written for the efficient ambisonic level detector. The surround sphere is projected to a cylinder and unwrapped on the flat surface. In this image, a monophonic source is panned slightly down and to the right (45° azimuth, −45° elevation).

Ambisonic Gain Control Module

The spherical harmonics defined in equation 4.2 form a set of orthogonal basis functions. If we define a sequence for our spherical harmonics and spherical harmonic expansion coefficients, we can treat a set of expansion coefficients as a vector, and perform matrix operations on them that rotate, warp, and re-orient our three-dimensional surround sound image.[15] The ability to mathematically warp and manipulate our surround sound image makes ambisonics the perfect choice for implementing a surround sound compressor.

[15] Hannes Pomberger and Franz Zotter. Warping of 3D Ambisonic Recordings. *International Symposium on Ambisonics and Spherical Acoustics*, 3, 2011

The Focus Transform One transform that lets us attenuate a region of the surround sound sphere is the *focus* transform distributed as part of the open source Ambisonic Toolkit (ATK).[16,17]

[16] http://www.ambisonictoolkit.net/

[17] Joseph Anderson. Introducing... the Ambisonic Toolkit. In *Ambisonics Symposium*, 2009

$$F(w) = \begin{pmatrix} \frac{1}{1+\sin|w|} & \frac{1}{\sqrt{2}}\frac{\sin(w)}{1+\sin|w|} & 0 & 0 \\ \sqrt{2}\frac{\sin(w)}{1+\sin|w|} & \frac{1}{1+\sin|w|} & 0 & 0 \\ 0 & 0 & \frac{\cos(w)}{1+\sin|w|} & 0 \\ 0 & 0 & 0 & \frac{\cos(w)}{1+\sin|w|} \end{pmatrix} \tag{4.3}$$

This transform is intended to focus attention on the region directly in front of the listener (0° azimuth, 0° elevation), by attenuating the region in the opposite direction, and gently warping the surround image toward the front. w is a value between 0 and $\frac{\pi}{2}$ radians, and specifies the intensity of the transformation. When $w = 0$, the surround field is unchanged. When $w = \frac{\pi}{2}$ sounds panned hard to the rear are muted, sounds panned to the left and right will be attenuated by 6dB, the entire surround image is warped to the front, and the gain facing forward is unchanged. This enables us to push one sound out of the way to make room for another sound as described in section 4.4.

Equation 4.3 attenuates the region behind the listener. If we want to attenuate a region other than the rear, we can rotate F using a rotation matrix like the one below.

$$R_z(\varphi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\varphi) & \sin(\varphi) & 0 \\ 0 & -sin(\varphi) & cos(\varphi) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.4}$$

Equation 4.4 (from the ATK) describes a rotation around the z-axis, by φ radians. To rotate the focus transform to the right instead of to the front, we first apply the focus transform to the inverse of a 90° right rotation. Then we apply the 90° matrix to the result. This example is generalized by:

$$X(w, \varphi, \vartheta) = R_z(\varphi)R_y(\vartheta)F(w)R_y^{-1}(\vartheta)R_z^{-1}(\varphi) \tag{4.5}$$

Equation 4.5 lets us programmatically generate an ambisonic focus transform matrix that targets a specified region of the surround field, fulfilling the objectives for our ambisonic gain control module in the Hypercompressor.

Ambisonic Gain Control Module Results The focus transform lets us warp the surround field, pushing the field to make room for new sounds. In some cases (for example, when mastering an ambisonic recording) warping the surround image is undesirable, and a simple directional gain transform should be used instead (an appropriate transform is defined elsewhere[18]). However, the goal of the Hypercompressor is not to compress dynamic range like a traditional compressor. The goal is to compress *space*. The focus transform is a compromise: We partly attenuate a region, but we also bend the surround image so that important parts of our surround texture are panned to a position with fewer competing sounds. This is an effect that is not possible with a traditional compressor.

[18] Matthias Kronlachner. Warping and Directional Loudness Manipulation Tools for Ambisonics, 2014b. URL http://www.matthiaskronlachner.com/wp-content/uploads/2013/01/EAA_2014_Kronlachner_Zotter.pdf

The focus transform also ties attenuation amount to attenuation radius. If we use only a single focus transform, is not possible to only slightly attenuate a large region of the surround sound field. The following section describes how we used this to our advantage during the live performance of *De L'Expérience*.

4.5 Of Experience

TODO: To Be Continued...

Bibliography

Tine Allegaert and Lukas Huisman. Reich: Piano Phase, 1989. URL www.youtube.com/watch?v=i0345c6zNfM.

Joseph Anderson. Introducing... the Ambisonic Toolkit. In *Ambisonics Symposium*, 2009.

Sinan Bokesoy and Gerard Pape. Stochos: Software for Real-Time Synthesis of Stochastic Music. *Computer Music Journal*, 27(3):33–43, 2003. ISSN 0148-9267. DOI: 10.1162/014892603322482510.

Alex Case. *Sound FX: Unlocking the Creative Potential of Recording Studio Effects*. Focal Press, 2007. ISBN 978-0240520322.

Alexander Chen. Pianophase.com, 2014. URL https://vimeo.com/98823512.

Zhixin Chen and Robert C. Maher. Parabolic Dish Microphone System, 2005. URL http://www.coe.montana.edu/ee/rmaher/publications/maher_aac_0805.pdf.

Joseph Clarke. Iannis Xenakis and the Philips Pavilion. *The Journal of Architecture*, 17(2):213–229, 2012. ISSN 1360-2365. DOI: 10.1080/13602365.2012.678641.

T.J. Cox. Acoustic Diffusers: The Good, the Bad and the Ugly. *Proceeding of the Institute of Acoustics*, 2006.

Peter D'Antonio. Performance Acoustics: The Importance of Diffusing Surfaces and the Variable Acoustics Modular Performance ShelI–VAMPS (TM). In *91st Audio Engineering Society Convention*, New York, 1991.

G Davis and R Jones. *The Sound Reinforcement Handbook*. Recording and Audio Technology Series. Hal Leonard, 1989. ISBN 9780881889000.

Emmanuel Deruty and Damien Tardieu. About Dynamic Processing in Mainstream Music. *AES: Journal of the Audio Engineering Society*, 62(1-2):42–56, 2014. ISSN 15494950.

Emmanuel Deruty, Francois Pachet, and Pierre Roy. Human-Made Rock Mixes Feature Tight Relations. *Journal of the Audio Engineering Society*, 62(10), 2014.

Evan Fein. Q&A With Tod Machover, 2014. URL http://www.juilliard.edu/journal/1402/hyperinstruments-crowd-sourced-symphonies.

Ken Ferrar. Soundfield Microphone: Design and development of microphone and control unit, 1979. URL http://www.ai.sri.com/ajh/ambisonics/wireless-world-farrar-10-1979.pdf.

Michael Gerzon. Periphony: With-Height Sound Reproduction. *Journal of the Audio Engineering Society*, 21(1):2–10, 1973.

Michael Gerzon. Ambisonics in Multichannel Broadcasting and Video. *Journal of the Audio Engineering Society*, 33:859–871, 1985. ISSN 00047554.

Dimitrios Giannoulis, Michael Massberg, and Joshua D Reiss. Digital Dynamic Range Compressor Design - A Tutorial and Analysis. *Journal of the Audio Engineering Society*, 60(6):399–408, 2012.

John Greschak. Polytempo Music, An Annotated Bibliography, 2003. URL http://www.greschak.com/polytempo/ptbib.htm.

D. J. Grout, J. P. Burkholder, and C. V. Palisca. *A History of Western Music*. W.W. Norton, 7th edition, 2006. ISBN 0-393-97991-1.

R. Hardin and N. Sloane. McLaren's Improved Snub Cube and Other New Spherical Designs in Three Dimensions. *Discrete Computational Geometry*, 15:429–441, 1996.

Peter Hoffmann. Xenakis, Iannis. *Grove Music Online, Oxford Music Online, Oxford University Press*, 2015. URL http://www.oxfordmusiconline.com/subscriber/article/grove/music/30654.

Charles Holbrow, Elena Jessop, and Rebecca Kleinberger. Vocal Vibrations: A Multisensory Experience of the Voice. *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 431–434, 2014. URL http://www.nime.org/proceedings/2014/nime2014_378.pdf.

Florian Hollerweger. An Introduction to Higher Order Ambisonic, 2008. URL http://flo.mur.at/writings/HOA-intro.pdf.

Bob Katz. *Mastering Audio: The Art and Science*. Focal Press, 2nd edition, 2007. ISBN 978-0240808376.

Philippe Kocher. Polytempo Network: A System for Technology-Assisted Conducting. *Proceedings of the International Computer Music Conference*, 163(September):532–535, 2014.

Matthias Kronlachner. *Spatial Transformations for the Alteration of Ambisonic Recordings*. Master's thesis, Graz University of Technology, 2014a.

Matthias Kronlachner. Warping and Directional Loudness Manipulation Tools for Ambisonics, 2014b. URL http://www.matthiaskronlachner.com/wp-content/uploads/2013/01/EAA_2014_Kronlachner_Zotter.pdf.

Philips Natuurkundig Laboratorium. *Philips Technical Review*. Number v. 20. Philips Research Laboratory, 1959.

Vincenzo Lombardo, Andrea Valle, John Fitch, Kees Tazelaar, and Stefan Weinzierl. A Virtual-Reality Reconstruction of Poeme Based on Electronique Philological Research. *Computer Music Journal*, 33 (2):24–47, 2009.

Oscar Lopez. AD Classics: Expo '58 + Philips Pavilion / Le Corbusier and Iannis Xenakis, 2011. URL http://www.archdaily.com/157658/ad-classics-expo-58-philips-pavilion-le-corbusier-and-iannis-xenakis/.

D G Malham. Higher order Ambisonic systems, 2003. URL http://www.york.ac.uk/inst/mustech/3d_audio/higher_order_ambisonics.pdf.

D G Malham and A Myatt. 3-D sound spatialization using ambisonic techniques. *Computer music journal*, 19(4):58–70, 1995. ISSN 0148-9267. DOI: 10.2307/3680991.

Christian Nachbar, Franz Zotter, Etienne Deleflie, and Alois Sontacchi. AMBIX - A Suggested Ambisonics Format. In *Ambisonics Symposium 2011*, 2011.

Chris Nash and Alan Blac. Realtime Representation and Gestural Control of Musical Polytempi. In *New Interfaces for Msical Expression*, pages 28–33, Genova, Italy, 2008.

Michael Nyman. Steve Reich. *The Musical Times*, 112:229–231, 1971.

Hannes Pomberger and Franz Zotter. Warping of 3D Ambisonic Recordings. *International Symposium on Ambisonics and Spherical Acoustics*, 3, 2011.

L Russolo. *The Art of Noises*. Monographs in musicology. Pendragon Press, 1986. ISBN 9780918728227.

Tom Service. A Guide to Iannis Xenakis's Music, 2013. URL http://www.theguardian.com/music/tomserviceblog/2013/apr/23/contemporary-music-guide-xenakis.

Martijn Vercammen. The reflected sound field by curved surfaces. *The Journal of the Acoustical Society of America*, 123(5):3442, 2008. ISSN 00014966. DOI: 10.1121/1.2934246.

Gil Weinberg, Roberto Aimi, and Kevin Jennings. The Beatbug Network - A Rhythmic System for Interdependent Group Collaboration. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 186–191, Dublin, Ireland, 2002. URL http://www.nime.org/proceedings/2002/nime2002_186.pdf.

Trevor Wishart. Audible Design, 1994.

Iannis Xenakis. *Formalized Music*. Pendragon Press, 1992. ISBN 0-945193-24-6.