Wesleyan University

COMPOSING WITH INTERACTIVE COMPUTER MUSIC SYSTEMS

By

Héctor Manuel González Orozco

Faculty Advisor: Ronald J. Kuivila

Readers: Paula Matthusen and Andrew Greenwald

A Thesis submitted to the Faculty of Wesleyan University in partial fulfillment of the requirements for the degree of Master of Arts in Music

Middletown,Connecticut May 2022

Contents

Aknowledgments

asdfasdfas

Introduction

Refactor.

Help bridge interactivity and ML.

Pointing to Heidegger!

In 1984 Joel Chadabe defined interactive composition as "… a method for using performable, real-time computer music systems in composing and performing music". For interactivity to emerge we require at least two elements that influence each other's behavior, in the case of computer music systems, a performer and some kind of software. The later receives some input by the performer but also reacts to it in not entirely predictable ways. Those reactions can be determined via, for example, a generative layer or a dynamic mapping scheme. Its sonic result then has some impact in the performer's actions, who isn't given absolute control and must make those responses at the time of the performance – in 'real time'. Reciprocal reaction is what distinguishes these systems from those involving digital musical instruments.

Composing, in this sense involves not only "the software that is written, the controllers that are used and the interaction that is defined" (Momeni, 1997, p. 2) but also the act of interacting with the system or playing the instrument. Thus, some of the traditional dividing lines between roles in music are blurred, such as composer/performer and instrument/score. The responsibility for the music composition and performance process is shared by the human performer and the software, while the latter's behavior functions both as the instrument and the score.

So, the creative process involves multiple steps: Creating the system and interacting with that system as part of both a compositional process and a performance practice. Thus, composition, performance and the 'meta-composition' of system design are inescapably entangled. Creating the system involves a feedback process of continuously testing and adjusting, and interacting with the system in performance situations inevitably leads to ideas for further refinements of both the performance and the composition itself.

Although I'll describe some historical interactive systems that employ purely analog media, I'll focus this discussion on systems for computer music. Therefore, whenever I talk about interactive systems I am really referring to interactive *computer music* systems.

In the first chapter I'll examine what constitutes an interactive computer music system. I'll discuss what makes them different to both acoustic and electronic instruments, and then explore some of the earliest examples of interaction in music using digital media. Then the focus turns to two of their constituent elements, namely controllers and the mapping scheme. These two will serve as framework to discuss some aspects that arise in the work of Laetitia Sonami and Michel Waisvisz that I consider relevant for the way I approach my compositional practice.

The second chapter will be discussion about machine learning, a branch of artificial intelligence that helps computers identify patterns on input data, and thus opens new kinds of meaningful human-computer interaction. I'll discuss its

usefulness to create both arbitrarily complex mappings and novel sounds. The discussion will center on the uses of supervised learning for the first, using the software *Wekinator*, and on unsupervised learning for the later, using *NSynth*. I'll explain the difference between them and describe in detail an interactive system I developed to explore a corpus of sounds created using machine learning assisted timbre interpolation.

In the third I will discuss 5 pieces I created during the last two years, all of which employ interaction in different ways and some of which use machine learning to create mappings that would be next to impossible using a rule-based style of coding. I'll start by describing my piece *Dasein* (2020), where I'll discuss *modes of engagement* and *authenticity*, two concepts found in Martin Heidegger's work that guided the way I approach most of my interactive work.

Chapter 1

Interactive Computer Music Systems

An interactive computer music system involves one or many performers and a means of conveying information about their actions to a piece of software that is ultimately responsible for the production of sound. This information is usually transduced via a physical device, which we'll call the "controller". The controller can be anything capable of producing data, examples are a couple of sensors attached to

an acoustic instrument[1] (the hyper-flute (Quintin, 2003) or overtone violin (Overholt, 2011)), mechanisms resembling an existing instrument (Piano MIDI controllers, the EWI), graphic interfaces on screens (the reacTable (Jordá, 2005)) or videogame controllers (Kinnect or Wiimotes).

Unlike acoustic instruments, the sound producing mechanism is decoupled from the physical gesture, the latter producing data that is applied in some way to the shaping of sound produced by mapping it to parameters of a sound producing algorithm. This relationship between gesture and sound created by the systems is defined as part of the composition process, while the composition itself may more or less define the instruments needed to be performed. The mapping scheme can involve unpredictable elements, such as parameters controlled by random number generators, patterns composing music algorithmically in real-time or independent agents, such as statistical models used to classify gestures or elements triggered via machine listening. These can either be hard coded, or inferred from a corpus of examples using machine learning. It's when involving such unpredictability that the system truly becomes interactive, as it involves real-time decisions being taken by at least two agents in response to each other.

For clarity and focus, I will distinguish interactive systems from electronic instruments. One of the distinguishing features of the first is that the mapping layer involves some kind of generative approach. The system doesn't simply allow a one-direction passive information flow, but takes the role of a musician in its own right,

[1] Also known as hyper, extended or augmented instruments.

becoming a co-creator of the piece. The performer's gestures can be made to control sound at different levels, from determining individual sounds, shaping a flow of events created by the system (akin to a conductor), or the simple triggering of events. The system itself can take various roles, including but not limited to those traditionally assigned to human musicians, such as performer, composer and conductor.

A case could be made to consider some acoustic instruments as interactive systems, as they tend to respond non-linearly or in a chaotic manner to energy input provided by a human. This can be attested by anyone that tried to learn a bowed string or wind instrument in a traditional western art music setting. The instrument appears to have a life of its own, creating sound in response more to the requirements of its physicality than to the urges of the novice performer. The instrument needs to be "tamed", that is, the performer is required to be able to exert control over its sonic output. However, differences with acoustic musical instruments are many, the main one being that in interactive systems the performers rarely have absolute control over the sonic result and are, instead, constantly in a kind of conversation with the system. The same input by the performer can generate radically different kinds of sonic behavior depending of the way the data is mapped. Furthermore, the unpredictable elements of the system are themselves explicitly composed and, ungoverned by any inherent material constraints, can be shaped in time or change from performance to performance.

In this chapter we'll first explore the development of some of the first interactive music systems: Chadabe's *CEMS* and Martirano *SalMAr Construction.* Then, we'll explore some of the elements that help us differentiate them from traditional instruments: the controller, mapping schemes and decision-making algorithms.

History: earliest interactive computer music systems

Algorithmic thought and generative techniques in western art music have a long and fruitful history. One of the earliest known examples is Guido d'Arezzo's combinatorial algorithm, used to set text to music by assigning two or three sets of notes in the 12-tone scale to a particular vowel, in a very similar way to how the syllables used in the solfège system were born. This is characteristic of abstract thought in music, where sounds are conceived not only as perceptual experiences but also as elements of a grammar. The modular nature of 12-tone equal temperament allowed for combinatorial practices to be commonplace in western music, with pitch classes maintaining identity even with variations of register. Some examples include the 18-th century practice of musical dice game and the 20th century fascination with serialism. All kinds of algorithmic approaches have been explored, ranging from the unpredictable to the deterministic.

However, it wasn't until the 1970's that technology allowed for algorithms to run independently of human agency and respond to real-time changes. Early interactive music can be traced to the work of Joel Chadabe and Salvatore Martinaro.

At the State University of New York in 1969, Joel Chadabe installed the Coordinated Electronic Music Studio (*CEMS*) System, an automated synthesizer system designed by himself and built by Robert Moog. It consisted of three modular systems: Audio (oscillators, filters, amplifiers, noise generators, etc), Control (sequencers, envelope generators, mixers, etc) and Timing (a four-digit clock and 10 decoder/delays). Some of the modules were custom built, and the studio had the largest concentration of Moog sequencers at the time. The idea was to build a programable system that allowed control of independent but related parameters of sound synthesis by a single source. It was one of the first systems that allowed for real-time algorithmic composition.

Soon after, he started sharing control of the sonic output by using joysticks as input device for his piece *Ideas of movement at Bolton landing* (1971). Any of the audio or control modules' output could be shaped by voltage coming from the controllers. The result ended up being interactive: the system reacted to the joystick movements in ways not entirely predictable, while the performer reacted to the system's output and tried to shape its behavior.

Over the next decades he continued building and performing with interactive systems, starting to use digital media with his piece *solo* (1978), where he could effectively conduct an improvisation of an orchestra of electronic sounds. The system involved using a pair of antennae to sense proximity, somewhat similar to Theremin's *Thereminvox,* but using them as control sources to schedule sounds on a Synclavier instead of controlling low-level sound parameters such as pitch and amplitude. The

performer then takes a role closer to that of a conductor than an instrumentalist. Instead of shaping individual sounds by controlling pitch and amplitude with left and right hand respectively, Chadabe shaped the whole piece by controlling tempo and timbre on real-time. Pitch and amplitude of every individual sound were left to be decided algorithmically by the software, which in this case takes a role like that of an orchestra improvising.

Simultaneously to the development of *CEMS*, composer Salvatore Martirano built an instrument called *Marvil Construction* with the help of engineer James Divilbiss. This proved to be a steppingstone in the development of a more ambitious interactive music system called *SalMar Construction*, which was finished in 1972 with the help of a group of engineers and graduate students from the University of Illinois, where Martinaro was a professor. The result was a 180-kg instrument and a configuration of twenty-four loudspeakers and four subwoofers required for audio playback and spatialization. Its interface consisted of two sections. The lower was the main panel for live performance, consisting of an array of 291 touch-sensitive switches and lights to indicate their current state. The top consisted of a patching matrix to connect those digital control circuits to analog sound synthesis and note generation modules.

SalMar Construction could play 73 sound sources that were divided in four "orchestras", basically interconnected sets of sounds patched in a way that they could share information coming from the performer via the state of the touch-sensitive switches. The way such information was modified by each orchestra could also be

determined by such switches, so the logic of event scheduling by the instrument was almost completely unpredictable. The performer could loosely determine the overall texture of the piece and its general timbral distribution, switching anywhere from controlling all the orchestras to changing the evolution of a single processes, but they always shared control of the resultant sounds with the instrument. The interaction devised for the instrument was analogous to conducting four different orchestras, each one improvising a concerto-style piece with its own soloist and ensemble.

The composer himself became a devoted and virtuoso performer of the *SalMar Construction*. However, he clearly wasn't the only agent responsible for the piece, he could only make educated guesses as to what sound would result. According to himself, "Control was an illusion. But I was in the loop. I was trading swaps with the logic. I enabled paths. Or better, I steered." (Chadabe, 1997). Over the years he continued refining the *SalMar,* as well as composing and performing interactive music systems, such as the *YahaSALMaMAC Orchestra*, involving a Machintosh II computer running his SAL (Sound and Logic) software, a Yamaha DX7, multiple digital synthesis modules and Zeta MIDI violin, performed by Dorothy Martirano.

When working on the *SalMar,* Martirano wrote Progress *Report #1* (1971), a text describing the state the inner workings of the system at the moment. It ends with a short chapter consisting mostly of a series of questions about the concept of "real time", sometimes of a puzzling nature:

WHAT IS REAL TIME?

> Those two four letter words have been used in this proposed report [many] times.
>
> Does real time only exist when you think of it? Have you, who have skimmed through, thought of a better way to say it? Are you aware that the process that allows a real musical time to happen is a real musical? Where's the trance? Can you sing and dance? Where's the reflex? Is Wagner's idea to put all the melodies together at the end of the overture less of an inspiration than the melodies themselves?
>
> The best is A HEAD. (p. 84)

The emergence of a system seems paradoxical if we consider only its constituent parts. How does the organization of inert parts give rise to life? How is consciousness born from electro-chemical signals? When do discrete data points generate the illusion of continuity needed for computer music interaction? When does a thematic material become music? How are historically disjointed practices brought together to create a new tradition?

The meaning of the last (and rather cryptically typed) sentence in the aforementioned chapter seems to be open for interpretation. It could be refering to a "head" in jazz terms, suggesting the intertwining of real-time and musical structure. But reading it literally, the best indeed was "ahead", with pieces like Lewis' *Voyager* (1987) and Rowe's *Maritime* (1992) (to name just a couple of immediate successors) continuing with such developments. The next half century oversaw an exponential

increase in the creation of interactive music and real-time composition/improvisation, aided by technological breakthroughs, the development of computer programming languages and sound synthesis software, and research on algorithms for machine listening, real-time digital signal processing and audio synthesis. Furthermore, the "entry fee" has been steadily decreasing. While the first experiments required institutional backing to see the light, powerful open-source software and cheap microcontrollers are commonplace now. Few are the prerequisites nowadays beyond a certain patience and frustration tolerance: while technology can be unwieldy at times it's still within arm's reach. In consequence, a world of possibilities has been open, with a myriad of artists exploring anything from software for collaborative improvisation to interactive sound art installations.

Controllers

When using the word "controller" in a musical context, I am referring to any kind of input device used for musical purposes. It's the interface "mediating gesture and sound" (Roads, 1996), transforming information about physical actions of the performer to a signal suitable to be sent to a playback device, usually with an intermediate mapping layer that shapes it in some way. Such signals consist of discrete data points extracted from attributes of other signals (dial position, key presses, etc) that convey information about the performer's gestures. The key feature here is the transduction of physical gestures into digital signals. The differences between traditional acoustic instruments and controllers are manifold, and it could be

argued that they are part of different categories: the first are integrated sound producing devices, while the latter form only the first step in the chain.

Acoustic instruments are easier to be perceived as a whole unit, each one forming an essence of sorts from where all kinds of sonic events can be brought forth into the world without them losing a fundamental identity. Even when we can split them in their constituent parts, these have roles that are interconnected, each one contributing in some measurable way to the overall sound. Particular configurations of material produce particular results, for example, it's always possible to trace a sound produced by a piano to its original source. Even when considering instrumental extended techniques their timbre profiles tend to be limited to a vast but finite space of possibilities, where the limit is not only determined by physical and mechanical constraints on the material or the arrangement of elements, but by the physicality and the vocabulary of techniques available to the performer. Moreover, the sound producing mechanism is the same as the instrument, with some of the energy provided by the performer's physical gestures being transformed to sound.

Of course, there is also a whole spectrum of possible designs between acoustic and electronic musical instruments, and multiple hybrid approaches exist. Everyone is familiar with electric instruments (for example, the Electric Guitar), they are basically acoustic ones that require external amplification to be heard at loud volumes, thus becoming an entirely different instrument. Instead of sending information about a human gesture, soft sounds are converted to electrical signals that can be subjected to multiple kinds of processes, resulting in a wide array of possible transformations.

Furthermore, acoustic instruments can be extended with sensors that can send data to control sound synthesis or processing parameters in real time (some examples include the hyper-flute (Quintin, 2003) or overtone violin (Overholt, 2011)). Even though a case could be made to consider both as controllers, we'll focus our discussion on systems where the controllers only generate data (usually via switches and voltage control) rather than audio waveforms.

Electronic musical instruments consist of at least two parts: a controller and a sound producing mechanism. They're decoupled from each other and thus can be shared or exchanged by other instruments, as anyone with a cheap commercial digital synthesizer is able to experience by a simple change of patch. Furthermore, a one-to-one relation need not be maintained, multiple controllers could be shaping sounds on a single synthesis mechanism or the other way around. They form entirely contingent systems, with no necessity shaping them and the specific configuration depending on the whims of the musician using them. In acoustic instruments the sound can always attributed to actions of the performer, even when repeatedly hearing an unfamiliar sound from a familiar instrument we tend to integrate it into our understanding of the range of sounds possible by it. A similar situation exists with synthesis techniques, for example, modulation or granular synthesis usually have very defined sonic profiles. However, this need not be the case for electronic instruments, where a single action can shape sound on multiple levels and produce different sonic behaviors depending on the mapping used, an issue we will discuss in the next section of this chapter.

Many controllers are built with shapes that imitate those of acoustic instruments, and techniques like physical modelling synthesis and sampling recreating their sonic counterpart. This can be attributed to their relatively new emergence and to a very human inclination for familiarity. It's easier for an explicitly musical controller to be commercially viable if it has a smooth learning curve, therefore ensuring its adoption by performers and guaranteeing further refinements. Also, the functioning of such controllers is easier to grasp for the average concert attendee, ostensibly making the music more engaging.

On the other end there are novel or custom designs, or other kinds of controllers being adopted for musical use, such as videogame controllers like Kinect, a motion sensor device originally built as a peripheral for the Xbox 360 but that has evolved to become a commonplace device for many artists working on motion tracking. We can even find idiosyncratic designs in some of the early commercial examples, such as the Buchla Thunder. Designed in 1990, its interface consists of an array of pressure and position sensitive plates, distributed in a way to be accessible to the fingers while allowing the hands to stay in virtually the same position. While the interface is a novel design, it was built with performances using a hybrid of hand drumming and keyboard techniques in mind.

Nowadays, computers and tablets offer the possibilities of creating graphical user interfaces that allows us to employ them as musical controllers on their own right, harnessing affordances such as their multi-touch capabilities. The advantages are that novel designs tend to help generate new ways to engage with musical

material, or sometimes a specific kind of controller is required for the way a composer envisions a piece. The reasons to choose between designs are numerous, and I've simplified the diversity of advantages of designs available. In chapter 3, I will discuss how I have used traditionally inspired designs (like the EWI[2]) to shape the overall evolution of a piece in my *Dasein* (2019) instead of playing individual sounds. Given enough time, even some original designs become commonplace, as exemplified by the Thereminvox pair of antennae. It ultimately depends on the piece or genre being played, as well as personal choices of the performers.

A protocol is a set of procedures and rules to process data that allows its transmission between devices. Some controllers come with predefined ones allowing for greater connectivity, as it can potentially control anything that follows that protocol. The most famous of these is MIDI (Musical Instrument Digital Interface). Created in the 1980's by an effort of multiple instrument manufacturers to standardize a communication protocol for commercial digital synthesizers to communicate with each other. It encodes control data for musical performance, such as patch changes, pitch, and volume, but a flexible channel system allows routing any MIDI value to any parameter desired. The original protocol allows the passing of 7-bit control values, therefore its resolution is usually limited to 128 steps, although it can be expanded, such as it's usually done for pitch bend messages by employing two 7-bit values to get 14-bit resolution. MIDI 1.0 was so successful that it became the de facto

[2] Electronic Wind Instrument, a controller shaped like a woodwind.

protocol for communication between commercial controllers and music software, and it took around 40 years for it to be extended into MIDI 2.0.

Another protocol explicitly created for music applications is OSC (OpenSound Control), developed at CNMAT, in Berkeley, California, and released in 1997. While MIDI was designed to interconnect synthesizers using serial ports, OSC was designed to employ the expanded data rates of ethernet protocols. It has a higher resolution and flexibility than MIDI, allowing control data to be organized and routed in almost any desirable way. Each message can arbitrarily large and contain multiple data types: integers, 32-bit floating point numbers and strings[3]. It has been employed for client-server software architectures (like SuperCollider) and adopted as a control protocol by most DAWs, and for real-time interactive applications due to its low latency and ease of use. It is now employed for uses other than music, with fields such as robotics and visual art performance finding it useful.

On top of the obvious layer of interaction in real time performance of the system, continually developing and engaging with an interactive system over a long span of time requires a kind of interaction itself. It's a process that involves two-way communication, allowing the possibility of feedback loops. Performing with them often suggest ideas for alterations, which themselves suggest new ways to perform, and so on. As described by Jorda (2005), "Music instruments are not only in charge of transmitting human expressiveness like passive channels. They are, with their

[3] Basically integer numbers, real numbers and text.

feedback, responsible for provoking and instigating the performer through their own interfaces". Even though alterations can be made at the mapping or synthesis algorithm layer, it's influence nowhere more noticeable than at the controller design level.

Therefore, by trying to expand on existing models one is usually witness to the emergence of idiosyncratic approaches not only to controller design, but to music performance practices. Two paradigmatic examples are Michel Waisvisz *The Hands* and Laetitia Sonami's *Lady's Glove*. Both used controllers built from scratch by themselves and developed with the assistance of engineers at STEIM, a center for research on electronic performance located in Amsterdam, Netherlands. By employing different approaches to harness hand and arm movements as musical gestures they both managed to develop decades long performance practices that involved multiple iterations of the controller, numerous pieces, and a multitude of approaches to live performance.

The *Lady's Glove* had 5 versions, constructed by Sonami from 1991 to 2003. It started as a humorous commentary on male-centered apparel in the design of controllers, placing some hall effect sensors in each finger and a magnet in the palm of a pair of rubber kitchen gloves. It evolved to be an arm-long thin lycra glove equipped with all kinds of sensors: accelerometers, ultrasonic receivers, resistive strips, to name only a few. The analog signal is converted to MIDI, which is used to control anything from sonic material to motors and live video. Furthermore, she strived to control the music on multiple levels, from the individual sound to the

structural elements of the piece. Being able to switch the focus on level and changing the degrees of freedom available to her, surrendering some control to the generative part of the system. This unlike Waisvisz's *The Hands*, whose mapping scheme, as discussed below, was usually focused on more direct control of the sound

Even if the controller is built or approached with a set of ideas, the feedback process of design-perform engenders new sets of them. What started with an idea of feminist interaction design came through the years to incorporate issues of communication and embodiment. According to the composer herself, "… I realize that my imagination is pretty much molded by the system I use. I don't think as much how will I adapt my ideas to the instrument, but I realize that the instrument has already influenced what I envision." (Rogers, 2010). In such cases the evolution of the system is somewhat paradoxical, what can in retrospect be considered almost a teleological development into the current form is entirely contingent on the whims of the composer and their relationship with it, as shaped by the experience of working within the sets of relationships afforded by the system. It requires agency but also kind of surrendering and close attention to the requirements and issues suggested by the system itself. The controller, being the most visible and tangible part but the least flexible, is usually the clearest path of communication where new ideas are suggested.

Of course, in live-electronics music performance there's a third element involved in the communication: the audience. Sonami became concerned in creating a channel of communication that bypassed the opacity of many electronic music practices. It does this primarily by an evident concern with a directly embodied

experience, with the attention paid to each gesture being the most obvious evidence. *Lady's Glove* is a very clear example of a system that employs what Harrison et al. (2007) call the "third paradigm of human-computer interaction". Focused not on human-computer coupling, and information processing and flow, but on "interaction as phenomenologically situated" (Harrison et al., 2007). This means that the system is focused on action as an activity that is charged with meaning depending on the context surrounding it.

Sonami's performances with the *Lady's Glove* can never be divorced from a multitude of elements that give meaning to it, including the interplay between the sonic output and the culturally dependent understanding of certain hand gestures. Therefore, pieces like this have a fluid sense of identity that cannot be ascribed to a fundamental element or idea. They have the potential to truly become integrated into the moment-to-moment fabric of reality, meaning provided only (if at all) by a collective sense of interaction and temporal evolution. Embodiment itself then becomes the central focus, with the controller itself simply becoming a vehicle for the performance and potentially becoming translucent. Situatedness takes a central role, and the controller turns into one thread in a truly tightly woven web. In Heidegger's terminology, the controller turns from being *present-at-hand* (subject of enquiry, the focus of the performance itself) to being *ready-to-hand* (inconspicuous, a vehicle for the performance). How these two different relationships create different kinds of performing subjects will be apparent when discussing the work of Michel Waisvisz, where the instrument as *present-at-hand* becomes a vehicle to explore physical effort

as an important dimension of musical practice. However, these parallel the development of systems in general and *Lady's Glove* in particular, with successive cycles of innovation engendering established practices and providing the scaffolding for new variations to arise.

Mapping

If we view interactive systems as a kind of information system, controllers represent the input: they are the way the performer uses their physicality to convey information to the system, transmitting their energy and starting the information flow. The output layer is usually some kind of sound generator, although it can take the form of other modalities, such as video. However, this model requires at least an intermediate layer between them, namely one that processes and transforms the incoming information in a way suitable to be used in some way by the output. This is the crucial component that takes the role of what we refer to when using the word *mapping*. A concept borrowed from mathematics; it describes the way elements of one set are assigned to elements of another.

The mapping layer converts the flow of data, allowing the performer to control the sound. Assigning values from one level to another is not an inconsequential task, as it determines the character of the resulting piece. Mapping can be considered stage in the creation of interactive systems where the developer of the system takes a role that is most like the traditional role ascribed to a composer. It's the arena where artistic freedom can be most easily exercised, the only limitation

being those inherent in the media chosen for the output. Any kind of mapping schema can be adopted and usually easily tested in real-time, thus allowing minute control over the characteristics of the system. Furthermore, the mapping layer usually works in a way analogous to that of a composer, whose role usually demands determining parameters of control (pitch, duration, timbre, etc) that will be performed by the output element in the system (the performer(s)).

Mappings can be implemented in a multitude of ways, most of which fall in two categories: explicit and generative (Hunt & Wanderley, 2002). The first involves the composer of the system directly determining a set of rules that the mapping will follow. In contrast, the latter involves training a model that learns its own rules to associate controller inputs to sound synthesis parameters by providing paired examples of both. Machine learning approaches, such as artificial neural networks, are useful for such purpose and will be discussed in the next section of this chapter.

Explicit mappings are usually further categorized based on how each performance parameter is connected to each synthesis parameter, and therefore their correspondence. These are described by Hunt & Wanderley (2002) as one-to-one, one-to-many, many-to-one, and many-to-many. Figure 1 depicts one-to-one and many-to-many mappings (a and b). The middle rectangle represents the mapping layer, processing controller parameters and routing them to suitable synthesizer parameters.

One-to-one mappings are the most straightforward to implement, it's as simple as mapping (in the mathematical sense) each input value to values in a range

suitable for a particular synthesis parameter. For example, converting MIDI CC values (0-127 on a linear scale) from a controller to oscillator frequency (0-20Khz, on an exponential scale) values. This makes a lot of sense when dealing with parametric control of synthesizers, but such simplicity paradoxically offers less control of the sound from a performer's perspective, where dealing with multiple dimensions of movement at the same time is next to impossible. Only a few faders can be consciously moved at the same time and accurately controlling movement along 3 different axes for each body part in motion tracking controllers is next to impossible.

A study conducted by Paradis (described in Hunt, Wanderley & Paradis (2001)) explored user reported reactions when employing three different kinds of mapping, while keeping the rest of the system (a MIDI fader box and FM synthesis)

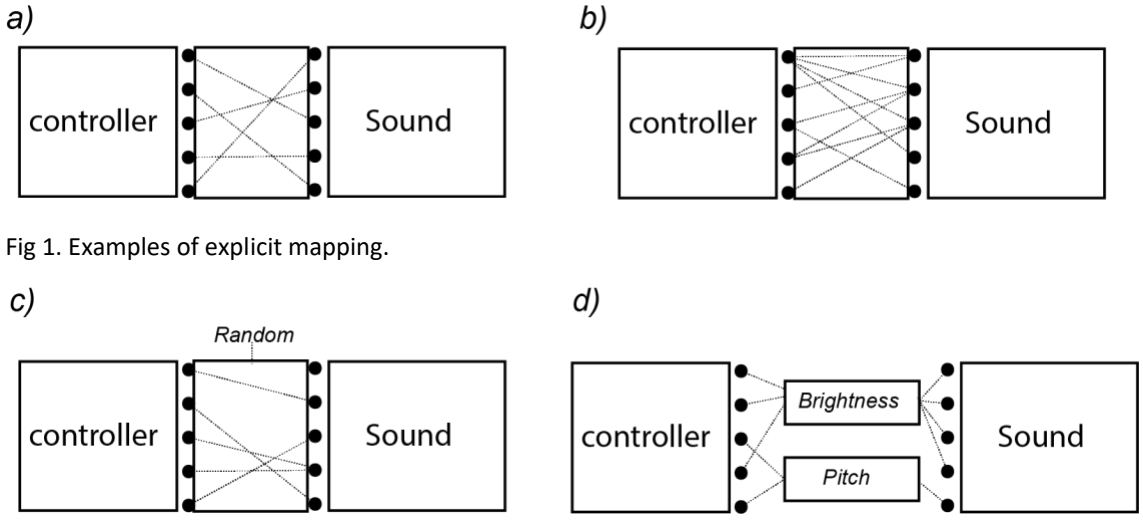


Fig 1. Examples of explicit mapping.

Note: a) one-to-one mapping. b) many-to-many mapping. c) dynamic mapping. d) multiple layers.

unchanged. According to the researchers, when using one-to-one mappings "… many users noted that the simple division of parameters was not very stimulating", while the most satisfactory was a many-to-many mapping scheme. Sound production is an inherently multidimensional endeavor, so when users are given multidimensional

control by manipulating few parameters it makes synthesis more accessible to control. Concessions should be made by the system to accommodate the performer's embodiment, and rarely the other way around.

However, an even more interesting result of such study was that even if constant energy input wasn't required for sound production, its introduction into the system made it feel "more natural.". Such extra input is analogous to bowing or blowing on string and wind instruments respectively. By giving an extra measure of control that required constant movement and some time to learn, users were encouraged to explore with the system. Up to a point, we are interested by challenging activities, exploring levels of mastery that we can achieve over them. Exploring different mappings allows to find a middle way between interactive systems requiring a steep learning curve and being uninteresting, where user interaction becomes both playful and challenging.

Multiple and/or parallel mapping layers can be used which receive and control only subsets of inputs and output parameters (letter d in fig. 1). Called mapping chain by Arfib et al. (2003), this approach has the potential to make the mapping even more organic to human users by employing intermediate layers that map to psychoacoustical parameters of sound. For example, using "brightness" as a sound feature that can be determined by multiple controller parameters, such as lip and breath pressure in wind controllers (Hunt & Wanderley, 2002), and respectively determines multiple synthesis parameters, such as frequency and resonance in a low-pass filter or formant frequency and width in a formant oscillator.

Any kind of multilayer approach can be explored, for instance defining a space of features related to the controller as an intermediate layer before brightness in our example. This can be useful to simulate the interconnected nature of parameters in physical sound, for example the envelope of a sound could be determined by energy input (velocity) and note number using a MIDI keyboard. We can thus give shorter duration and shaper envelopes to higher notes, in a similar fashion to how plucked string instruments work. Furthermore, the modular nature of such approaches makes the mapping more flexible, allowing a single scheme to be easily adapted for different controller/sound generator combinations. It's a way to allow more intuitive control and transparency in the performance by mapping using meaningful perceptual categories, instead of sound synthesis parameters, which are usually more related to the way the algorithm that produces them is implemented.

However, mappings don't have to be static. A multitude of approaches exist that can change the way the mapping work in what Murray-Browne (2012) calls *Dynamic* mapping. The simplest is to introduce randomness (letter c in fig. 1), either to shape the behavior of a mapping function or to control any combination of output parameters. This would involve the most basic requirement for a system to be interactive, responding to the performers' input data and shaping the output based on its own deterministic state. Implementations of randomness usually employ pseudorandom number generators, creating determinate sequences of numbers that simulate random numbers.

Dynamic mapping schemes can open new levels of affordances to the performer or limit their feeling of being in control. On one end, switching between mappings could be another dimension of possibilities accessible to the performer, such as by changing the state of the controller, for example, pressing a button in a controller can allow to switch between "patches". In contrast, schemes could be designed that constantly interpolate or suddenly change between different mappings, requiring the performer either to constantly adjust or give in to the inherently unpredictable character of the system.

Parameter mapping is not only musically useful for real-time digital instruments but is also a technique commonly used in data sonification. Sonification was born out of a need to create tools that exploit the auditory perceptual abilities for uses such as data analysis and information retrieval. The term "sonification" is understood as the derived techniques that deal more specifically with "…the data-dependent generation of sound" (Hermann, 2011). It is the sonic equivalent to data visualization.

While sonification has been used for scientific purposes, it has also enriched and engendered new approaches to artistic sound creation and music composition. It is rarely used as an all-or-nothing technique by composers, but as one in a set of tools used to define parameters of the piece and relate the music to some extra-musical phenomena. Some examples of music built in part by sonification are Iannis Xenakis'

Metastaseis, Alvin Lucier's *Music for Solo Performer*, and Charles Dodge's *Earth's Magnetic Field*.

Earth's Magnetic Field provides a great example of a composer using parameter mapping to determine the pitch content of the piece. Dodge used Bartel's "musical" diagram, a common method to display visually (in a way similar to the use of staves and notes) the average global geomagnetic activity, known as Kp-index[4]. He used the measurements for 1961 and mapped the 28 possible values of each "note" to a four-octave diatonic scale. This information was fed to a computer employing the Music IV software at Columbia-Princeton Electronic Music Center. This kind straightforward parameter-mapping sonification makes it possible to follow the melodic contour or sometimes individual notes throughout the whole piece, with Bartel's diagram providing the score for the piece. He maintained freedom to choose the actual character of the piece, as the data (while informing other aspects of the work) was used exclusively to provide the pitch parameters, while he intuitively decided such defining aspects as the length and form. The timbre palette was freely chosen, loosely related to the phenomena that inspired the piece, thinking of "radiant" characteristics and "the feeling of the human response… [to] the radiation from the sun that is essential to life" (Thieberger & Dodge, 1995).

This is an example of mapping as a "compositional process that engenders a structure of constraints" (Magnusson, 2010), namely that the relationship between pitch and solar radiation was fixed. The duality between affordances and constraints

[4] Index that measures the average global geomagnetic activity.

is an important mark in interactive music systems, as they determine the freedom of action given to the performer and therefore the way they interact with it. Even if we'll untangle such duality for analysis purposes, they are interdependent, with constraints allowing for the emergence of affordances and affordances entailing a set of constraints.

Affordances is a concept borrowed from the field of ecological psychology, defined as "a property of the environment that [allows] actions to appropriately equipped organisms" (Dourish, 2001). It's the way an individual as situated in a particular context maps their potential actions to what is possible. However, it should not be understood as analogous to the controller/sound relationship that we have been discussing, with an intermediate mapping layer. Affordances don't depend on intermediate "mappings" as they are more integrated, related to the way we already exist as being-in-the-world and not to a duality between subject and object (with mapping provided by perception). The world reveals itself and is embedded in our embodied participation with it. In interactive systems, affordances are determined in the most obvious way by controllers. Depending on previous experience, a knob suggests the action of turning it, while a motion tracking device suggest a greater range of possibilities.

On the other hand, constraints determine a range of possible variations within which the performer can explore. If affordances affect the way we act with the world, constraints give shape to the extent of our actions. Inherent to all musical practice are a series of cultural and physical constraints, determining the limits allowed for a

performer and defining the sonic space. While physical constraints are defined by the limits of human motion and the materiality of the instrument, and cultural constraints by what's expected in an environment defined by social relationships, interactive music systems allow for arbitrarily complex mappings to define what is *sonically* possible. Even if there are inherent limits in human perception, the potential to explore within those limits is to vast and ever more accessible via software. It's a self-imposed limitation, but it's what defines the character of the work.

If we conceive a piece of music as existing within a set of boundaries, and composition as carving a home for this entity in the multi-dimensional space of sound, then setting a series of constraints is where the compositional process is more explicit in the development of interactive systems. And mapping is what defines this set of constrains. As explained by Murray-Browne (2011) "In a time when musical programming languages have unleashed a bewildering amount of sonic potential, it is the constraints rather than the affordances of an instrument that characterize it" (p. 3). Mapping, as setting constraints, determines the state of the system and the kind of interaction the performer is most likely to engage in.

Interactive music systems emerge from affordances and constraints in a process of double articulation: affordances determine the elements to be used, while constraints select and organizes a subset of these in particular configurations. While the affordances can allow a system to potentially accrue layers of meaning by linking actions or controllers to a social understanding of them, it's the constraints that carve a figure in the 3-dimensional space built by those layers, shaped by the idea behind

the piece. For example, while using a Kinect as a controller could link the piece to a plethora of interrelated ideas in videogame culture, it's the set of movements considered as meaningful by the mapping that defines the piece. We'll explore this type of dual emergence as evident in performances of *The Hands*, a custom-made MIDI-based digital musical instrument built and performed by Dutch composer Michel Waisvisz.

Waisvisz had a life-long interest in human touch as mediator of electronic sound, evidenced by his work with previous instruments, such as his *Crackle* series. Built in the late sixties and seventies, they consisted of devices built using analog circuitry in the form of oscillators, sensitive enough to finger pressure to elicit chaotic behavior with the slightest contact. With the performer sharing some measure of control with the unpredictable nature of the circuit, these stand together with *SalMar* and *CEMS* as one of the earliest interactive analog music systems.

He entered the realm of real-time computer music with *The Hands*, a device used to produce MIDI messages out of hand and finger movements which were used to control a Yamaha TX7 synthesizer. A first version of the instrument was built in 1984, with two iterations following in the next decades. They consisted of a controller strapped on each hand, with 12 buttons at fingers' distance, mercury switches detecting inclination of each hand, a potentiometer on the thumb, a set of ultrasonic transmitter/receiver to measure the distance between each hand, and a microphone

usually employed to record and loop in real-time. Further technical details of its construction can be found in Torre et al. (2016).

Distance between hands was straightforwardly mapped to amplitude in most cases, a one-to-one mapping. MIDI note values, however, effectively formed its own mapping layer, with 12 buttons used to select pitch class, and hand inclination to determine the octave. This can be conceptualized as a many-to-one mapping using an intermediate layer. A "scratch" function (Waisvisz, 1985) could also be toggled on/off using the thumb potentiometer. Consisting of repeating a set of note-on messages (almost at audio rate) for each MIDI note being played when changing the distance between hands, allowing a kind of granular control of timbre using a gesture similar to bowing in string instruments. In such cases, this movement was non-linearly but directly mapped the overtonal content of the sound, using a deterministic dynamic mapping scheme.

In contrast to Sonami's *Lady's Glove, The Hands* were constructed to reflect Waisvisz interests in music performance as a display of physical effort and tension, a dimension lacking in most of live-electronics practices at that time. This resulted in the mapping described before, affording command in a way akin to how most traditional instruments are controlled by shaping sounds using direct energy input from the performer. Even the distribution of independent MIDI note messages to each hand, with the possibility of polyphony, suggests an expansion of control possibilities of keyboard instruments to include more dimensions related to timbre. Sonami's performances vary in the level of direct control she has over the sonic gestures, with

that dimension being somewhat subservient to an interest in communication and the unfolding of meaning from hand gestures. Waisvisz's brought such control to the forefront of the musical discourse, shaping his whole aesthetics on a virtuoso display using physical effort as an expressive medium. Physical effort wasn't just an end on itself, but was a way for him to also show the "spiritual efforts made by the composer/performer" (Waisvisz, 1985). His performances usually went on anywhere between 30 minutes and an hour, resulting in him usually being drenched in sweat afterwards (Bellona, 2017). While both strived to generate a clear channel of communication with the public employing their extremities as main tools, Waisvisz's style focuses a more concrete and almost vicarious experience of somatic tension and release, while Sonami's is more concerned with abstract, semantic relations between gesture and sound. *The Hands* is not meant to be as transparent for the performer as the *Lady's Glove* is, but is intended to almost stay perpetually as *present-at-hand,* in Heidegerian terms.

Both the controller and mapping of *The Hands* set up a system of affordances. The first allowing for multiple dimensions of physical engagement, ranging from the intimate (buttons that require little movement) to the extensive (distance between hands), and the latter defining the almost instrumental-style control of sounds. This is the ecosystem where the piece was allowed to evolve.

However, he also set a series of constraints that helped shape such evolution. On the controller side, he consciously froze the development of the instrument for years at a time to focus on mastering its performance and discovering its underlying

affordances. The previously mentioned “scratch” function was an affordance limited by an inherent physical constraint, namely the extent of the performer’s arms. The mapping, although somewhat dynamic, usually revolved around few parameters at a time, controlling individual sounds or the evolution of patterns. In contrast to the *Lady’s* Glove, the computer rarely took generative roles and physical gesture was meant to be more tightly coupled to sound.

It’s perhaps paradoxical that tighter constraints allow for more direct control, but when a limited space of possibilities exists the performer has more chances to map it (in a cartographical sense) and internalize its functioning.

This leads us to Chadabe’s criticism of the concept of mapping itself to describe the coupling between the controller and sound producing mechanism in the context of interactive music systems (Chadabe, 2002). In short, he makes a distinction between two different approaches to control of a system, both related to different compositional approaches, similar to the distinction between explicit/generative mapping (Hunt & Wanderley, 2002).

First is a more direct style of control, similar to Waisvisz’s close relation between physical movement and sound. It consists of a hierarchical (arborescent) structure, where control is mostly top-down. Even if the system shapes some of the resultant sound it is subservient to the emphasis on performance. It’s here that we notice the prevalent tendency of digital instrument design to strive to be taken on equal footing as traditional acoustic instruments, trying to make its position

unassailable by harnessing the prestige given to the "virtuoso" archetype in Western Art Music[5].The importance of the human agent in the system is shown by giving the organic elements mastery over its silicon-based counterparts. It's analogous to an evolutionary adaptation of new forms of life trying to draw out of an already established ecological environment, looking for coexistence on equal grounds. It's in such cases when explicit mapping is useful, trying to keep tight constraints to allow more control for the performer.

In contrast, he talks about networks structures to explain what we called "generative mapping". This consists of a distributed network of control, where multiple agents share responsibility for the sound being generated, more akin to Sonami's approach. Instead of a hierarchical we get something similar to a flat structure. The human agent stops being the sole focus and becomes just a part of the system, its most visible and the one gifted with conscious decision-making, but a part, nonetheless. Arborescence turns to rhizome, with multiplicity of elements engaging in a conversation instead of a monologue. Emphasis is sometimes given to the performer, but more to their embodied experience as a node in an ever-mutating web than the center of it. This allows for non-virtuoso engagement with the system, and potentially stimulates the exploration of new modes of engaging with sound by setting an ever-changing set of constraints. The organism evolves by finding its own ecological niche but striving to eventually become a generalist.

[5] This tendency is very prevalent in some academic circles. For example, the theme of the International Computer Music Conference 2021 was "The Virtuoso Computer".

This is where machine learning becomes useful. If we really intend for computers to take their own decisions in interactive systems, we can't limit to setting up a set of if-then rules. Teaching them to make inferences by providing hand-picked data is a way to steer their functioning without directly determining the constraints. Distributed structures can be achieved by employing literal "network" architectures for each node, such as artificial neural networks, but many approaches to machine learning exist, even allowing for deterministic styles of control. In such way, arbitrarily complex and generative mappings can be created, and the idea of sharing control of the system with digital intelligent agents is brought one step closer to fruition.

In the next chapter we'll explore two ways I've employed machine learning for music composition: as a generative mapping layer for interactive music systems, and to create sound material in non-real-time.

Chapter 2

Machine Learning

In the years after its inception as a discipline in 1956 at a workshop in Dartmouth College, the field of artificial intelligence has suffered multiple false starts, owing to cycles of overpromising followed by the inevitable under-deliveries. Most of the early developments and breakthroughs focused on "symbolic" artificial intelligence, that is, trying to emulate intelligent behavior by manipulating high-level (symbolic) concepts, in a way that each step taken by the computer would be absolutely understandable to human agents. Expert systems basically consisted of a series of if-then computations that were supposed to mimic the way human decisions are made.

Even if some strides were made with this approach, the underwhelming results led to research on alternatives. They were further spurred by criticism of symbolic AI by philosophers such as Dreyfus (1986), focusing on the reductionist nature of symbolic approaches due to the assumptions made about the human mind working in such a mechanistic fashion. He positioned symbolic AI as rooted in a worldview biased by a Cartesian subject-object duality, where the ontological assumption was that elements of the world can be isolated from their context and manipulated independently. An intelligence was supposed to make inferences born of an ever-complex set of rules dictated from above, and not as a situated, embodied agent that engages with the world in ways learnt from previous experiences with its environment. The idea that objects can be abstracted from their spatiotemporal features, and thus generalized and made independent of particular situations, is part of a deep philosophical underpinning in western culture, going back to platonic realism

and its adoption by the Roman Catholic Church. To escape from such constraints, we had to find a way to model consciousness as already embedded in the world and inseparable from its environment, allowing for a sort of subconscious thought. Nothing short from modeling intuition.

Thus, sub-symbolic approaches that employed statistics-based methods were developed in the early 21st century, giving rise to the field of machine learning. It's basically a series of algorithms that recursively improve their own functioning by learning from data, which is usually fed to it by a human agent.

It's impossible with our current means to model a completely embodied learning experience analog to human learning, but providing an algorithm with data is the closest we can do at the moment to provide context into particular phenomena, without trying to mimic what is assumed to be the rules of human reasoning. Even if the human agent is supposed to carefully select the data, choose the model, and tune a series of meta-parameters, computers makes their own inferences at a level that makes sense for them, as proved by the "naivete" evident in adversarial attacks[6]. Even with these disadvantages, machine learning has been found useful for many daily life applications, anything from self-driving cars to music recommendation on digital platforms.

Machine learning approaches are usually classified as supervised or unsupervised learning, although there exist multiple techniques that don't fit neatly

[6] For, example, where changes in some pixels in an image (undistinguishable to humans) can make a neural network classify it as something completely different.

into either category. In supervised learning the computer is trained by giving examples of paired inputs and outputs, expecting it to learn to relate them in some way and produce its own outputs when given new inputs.

The most common use of supervised learning are for classification or regression problems, the first giving discrete and the later continuous results. For example, a machine learning model can be to classify instruments in audio recordings. Training it would require feeding it with a dataset of features (power spectrum, MFCCs, zero-crossing rate, etc) extracted from recordings of a single instrument associated with the corresponding label (Oboe, Violin, etc). If properly trained, when presented with instrumental recordings or real-time feed from a microphone it will output the name of the instrument being played. Classification always distils the result to a single category, determined by an integer. In contrast, the results of regression can be almost any numerical value, including rational numbers. These are useful for cases such as creating a complex non-linear generative mapping between hand position and a continuous synthesis parameter, such as frequency. They are both basically complex functions, where the inner working is arbitrarily complex and learned by training using data.

On the other hand, unsupervised learning requires no previous labeling or steering from a human expert, it learns underlying patterns in the data on its own and thus help give structure to seemingly disparate information. These structures can be represented in a way that allows the model to consistently map the input data and to modify it or even create new instances of it, common uses include text translation and

denoising of images. This underlying representation of structures also makes them ideal for generative purposes, as it can create new instances with a similar data distribution to the original. It's useful when the data shows no apparent correlation or when labeling is impossible due to the size or type of the database, such as when using unlabeled audio on a sample-by-sample basis.

While there are many machine learning models (decision trees, support vector machines, etc), artificial neural networks approaches have become particularly prevalent during the last decade, even giving shape to its own sub-field withing machine learning, that of "deep learning". Its architecture is an attempt to loosely emulate the way neurons in the brain works as nodes in a network. Although not claiming to be an analogous representation of a biological process, it's still a useful model based on developments on neuroscience. They consist of interconnected layers of artificial neurons, with an input layer receiving the data and output one calculating the result. Each artificial neuron is a unit of computation that receives numeric values from every neuron in the previous layer, usually weighting every connection differently, summing the result and passing it through a function to normalize its value before sending it to every neuron in the next layer.

Although at first this seems to be a one-way flow of information (with the feedforward approach previously described from input to output needed to generate the result of the calculation) this is far from the case, with result data flowing in the opposite direction for training purposes, in a process called "backpropagation". This recalibrates the network to make more accurate result given the previous batch of

computations, it's one of the processes with which the network "learns". With this in mind, neural network architectures are analogous to the feedback model we've used to describe interactive music systems, with a multiplicity of agents giving rise to it and the output influencing to the performance and the behavior of the system (fig 2). We can therefore consider generative mapping using neural networks for interactive systems as an example of self-similarity.

However, even though feedback and backpropagation share a similar structure, there are some important differences. First, backpropagation doesn't influence the input data, only the weights of the network and therefore the output. In contrast to feedback in truly interactive systems, where we expect the performer(s) to modify their gestures depending on the way the system responds to them. Second, backpropagation is intended to improve the accuracy of the system by measuring the

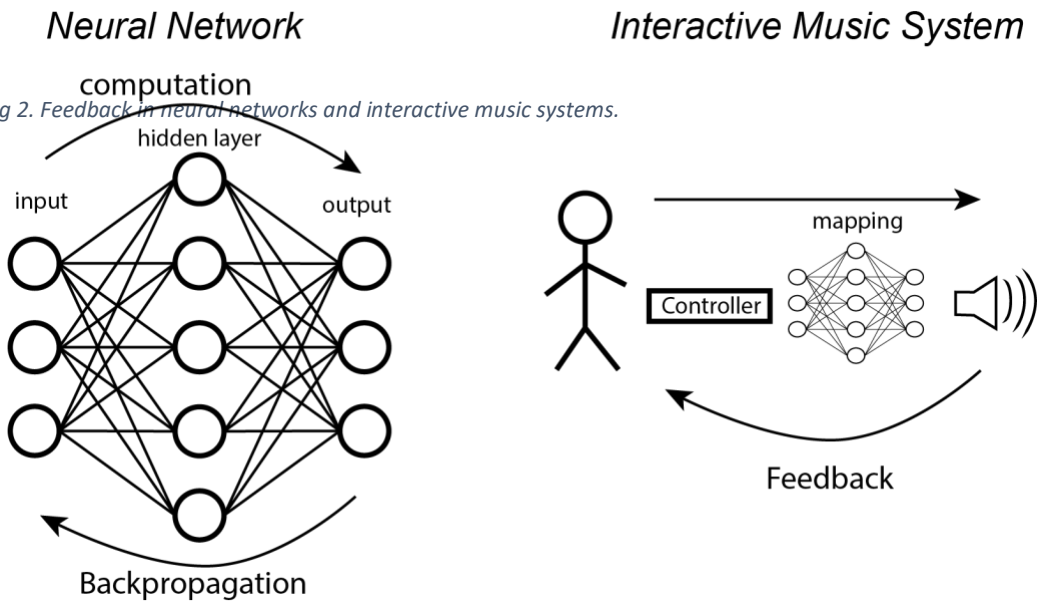


Fig 2. Feedback in neural networks and interactive music systems.

loss function between input and output, while its counterpart in music systems is much more open ended and depends on the judgment and situatedness of human

agents. Finally, if we intend for a neural network to be able to infer results from new data, a limit must be set to the influence of backpropagation to prevent the phenomena of overfitting[7]. This is particularly important if we intend to use machine learning not only to create mappings, but as a generative tool for audio, where we don't want to reconstruct the original audio but create a new one that shares some similarities to it. Examples of such uses will be explored further down.

Admittedly, the way we dissected interactive music systems in the previous chapter into a set of constituents (controller/mapping/sound) can be seen as reductive, as it is only a way to describe previously constructed systems. Even if they can be unthreaded for analysis purposes, music creation rarely engages consciously with them as independent of each other. A change in the mapping might be required by a modification of the controller, possibly changing entirely the affordance/constraints balance, and thus opening new dimensions of sound. Furthermore, when the relation between performance as a practice of embodied discovery and the shifting roles and relations with the system takes central focus, there's a need to explore alternative architectures to explore clearer relations between gesture/interaction/sound.

In Van Nort's article *Sound, Senses, Musical Meaning and Digital Performance: Epistemological Refamings*, the author suggests machine learning is a way to engage with "phenomenological design work, allowing a more holistic

[7] This happens when a machine learning model learns the patterns of a particular dataset too closely. For example, this can lead the model to mistakenly classify a recording by using features we didn't intend to use, such as noise floor.

approach to associating experienced/heard sounds to embodied and enacted gestures, collapsing listening and acting in the design loop" (Van Nort, 2020). It would potentially allow us to bypass not just mapping as an explicitly defined layer, but even parametric control of sound synthesis itself by using methods such as Neural Synthesis (Engel et al. 2017), constructing audio directly on a sample-by-sample level.

Thus, the last decade has seen the evolution of machine learning tools for interactive art, such as the *Wekinator*, a piece of software developed by Fiebrink (2011) that makes multiple machine learning models and training accessible with a very simple graphical user interface. It receives data and outputs the results as OSC messages, thus flexible enough to be used with a plethora of controllers and audio synthesis software. With it, training itself becomes interactive, with real-time feedback in the system design process allowing for the creation of "complex relationships between performer actions and computer responses" (Fiebrink, 2011). Multiple artists have employed it as a convenient way to explore machine learning approaches without having to throw themselves at the deep end, such as Sonami herself on her latest instrument, the *Spring Spyre.*

Most of the work with interactive systems requires supervised learning, as human labeling of some kind of data is desirable for mapping purposes. However, if we're interested in synthesizing sound at the sample level to create new sonic material, labeling each sample by hand becomes too time consuming to be practical. It's in those cases where unsupervised learning approaches become useful. In the rest

of this chapter we'll describe our experiments using unsupervised learning to generate material for electroacoustic music composition by interpolating between previously analyzed sounds, as well as employing a very simple interactive system to explore the resultant corpus of sounds.

Timbre Interpolation Explorer

Sound synthesis consists in generating audio signals from scratch using electronic hardware or software. Analog synthesis employs continuous signals usually generated and modified by electrical circuits such as oscillators, noise generators, filters, and envelope generators. Digital synthesis, on the other side, involves generating or modifying via software discrete signals consisting of individual and closely spaced samples that usually determine the position of a speaker cone in time. What most of the usual techniques for sound synthesis (such as additive, subtractive and modulation synthesis) have in common is that they involve shaping sound at a high level. Few are concerned in dealing with audio on a sample-to-sample level, given the overload of data required to do so.

The technique I'm employing is called NSynth (Engel et al., 2017), which generates audio signals on a sample-by-sample level using a neural network model. The functions required to implement it are part of Magenta[8], an open-source machine learning Python library geared towards generating music and visual art. The model I'm using analyses sounds and creates a series of hidden embeddings that can be combined

[8] https://magenta.tensorflow.org/

with others to effectively be resynthesized as sounds whose timbre consists of a linear interpolation between their respective timbres. This can be expanded to include interpolation between more than two sounds and adding weights to each embedding to determine the percentage of each sound present in the output.

Trying to find new sonic possibilities to explore new sonic materials, in this case, involves looking for strategies to explore the multitude of sounds created and easily compare them with one another. I decided to assemble sounds on a 3-dimensional space (fig 4.), where moving along each axis involves adding more of that timbre to the resultant sound in ¼ increases (that is, quantized to 4 steps). Furthermore, I created a system to interactively explore such space with hand gestures using a Leap Motion Controller. The right hand selects previously resynthesized sounds by moving through space, while the left can select sounds by making a grabbing gesture that's recognized as such by a classifier using the previously described Wekinator software (Fiebrink et al., 2011).

Autoencoder neural networks are unsupervised learning algorithms that can create compressed (latent-space) representations of the input data, and then reconstruct a meaningful representation of the original data using such representation. These two steps are respectively called the "Encoder" and "Decoder". The model effectively creates a reconstruction of the input data. This would be a trivial pursuit were it not for the case that it can learn to detect some structures in the data that can be further reconstructed in a different way. Therefore, they have been employed for uses such as denoising images and translating text. Its usefulness lays in the latent-space

representation, which can be shaped in any desirable way or fed into a different kind of decoder. Autoencoders are also generative models, which mean they can generate new data that is similar to the input.

Model & Data

The model I employed is based on WaveNet, an audio generative model that operates "directly on the raw audio waveform" (van den Oort et al., 2016). However, the need for external conditioning inherent in the original model is bypassed by using the hidden embedding as conditioning and therefore recreating the original signal. It is called "WaveNet Autoencoder" (Engel et al., 2017).

This model is pretrained using a large dataset of raw audio, in this case 4 second musical notes found in the "NSynth Dataset" (explained below), allowing it to infer a useful representation for the embeddings. Training a different model would have been next to impossible given the resources I have at hand, as it "takes around 10 days on 32 K40 gpus (synchronous) to converge at ~200k iterations"[9].

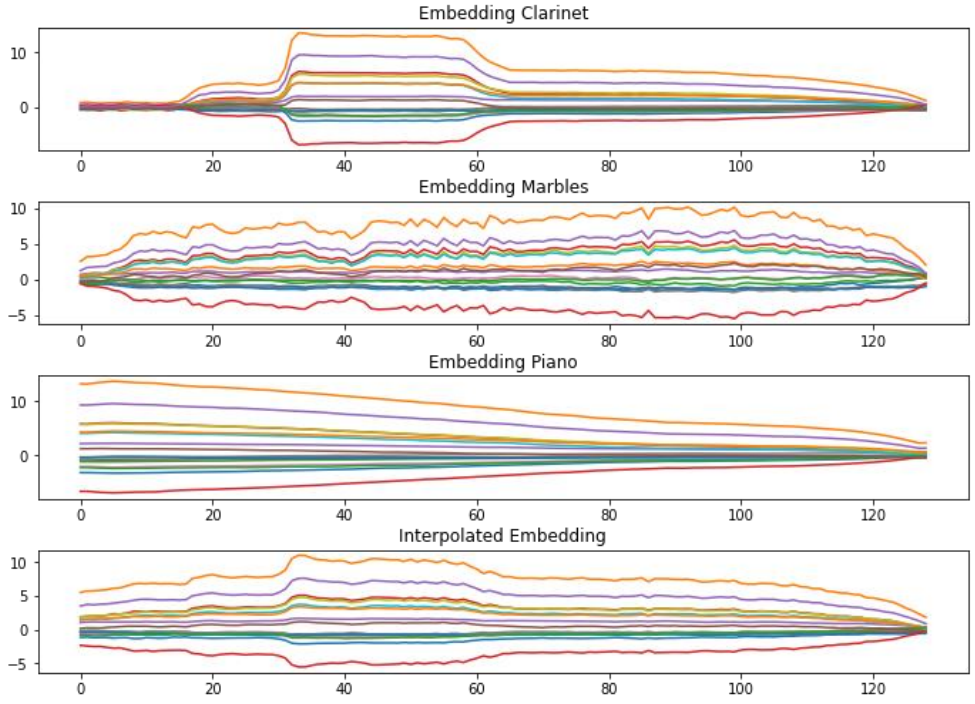
The encoder is a deep neural network that effectively creates a downsampled representation of the sound, called the "embedding". A single 16-dimensional point is inferred every 512 samples, therefore getting a 125x16 embedding for 4 seconds of audio sampled at 16kHz (the only available sample rate). The decoder takes the embedding and upsamples it to the original rate, thus reconstructing the original sound with some sonic artifacts being added by the statistical nature of reconstruction. Each

[9] https://github.com/magenta/magenta/tree/main/magenta/models/nsynth

sample of the resultant audio is chosen based on its probability given every sample that came before, conditioned by the embedding. Both steps are influenced by the pretrained model, available as a TensorFlow checkpoints on the Magenta GitHub repository (link on footnote 7).

Embeddings of multiple sounds can be combined in any desirable way before decoding. Fig 3 shows an example where each embedding is divided by three, leading

Figure 3. Embeddings of three sounds and the resultant 1:1:1 interpolated embedding



Note: Equal weights. Sounds correspond respectively to: A clarinet multiphonic, multiple marbles being rolled together and a single low piano note.

to each one having an equal weight in the resultant sound (1:1:1 ratio). Each one of the 16 channels is shown in a different color. The influence of every sound in the interpolated embedding is clear by simply looking at the way the embedding is shaped,

with jagged edges like the second embedding and a combination of the envelopes of the second and third.

The NSynth Dataset (Engel et al., 2017) consists of over 300k musical notes. Each one consists of a four second, monophonic sound sampled at 16kHz, at pitches corresponding to every one of the 88 keys of the piano (if available) and 5 different volumes taken from over 1000 instruments found in commercial sample libraries. Each note has an envelope with 3 seconds sustain and 1 second decay. Thus, the model trained on it is expected to learn to identify structures found in pitched sounds shaped in a very particular way.

Being a generative model means that it has certain advantages over simple convolution of signals or cross synthesis when it comes to creating timbre interpolations. The most important in my case being the tendency to add additional harmonics (even sub-harmonics) or dynamically mix the overtones in time. This creates a richer sound that creates subtle changes with every different weighting of the embeddings. I also became interested in experimenting with noisy sounds with very little pitch content (such as recordings of water and whispers) and attempting to interpolate between sounds with different pitches.

I wrote a Python script[10] that takes 3 sounds as input, determines their embeddings, makes them all the same length, computes 4^3 different embeddings consisting of every combination of sounds and weights (0, 33, 66 and 100%), and then

[10] Which can be accessed as a Colab Notebook here: https://colab.research.google.com/drive/1B1TikSrq0XeXJ0EvMzDXuQe4UqVFXsj7?usp=sharing

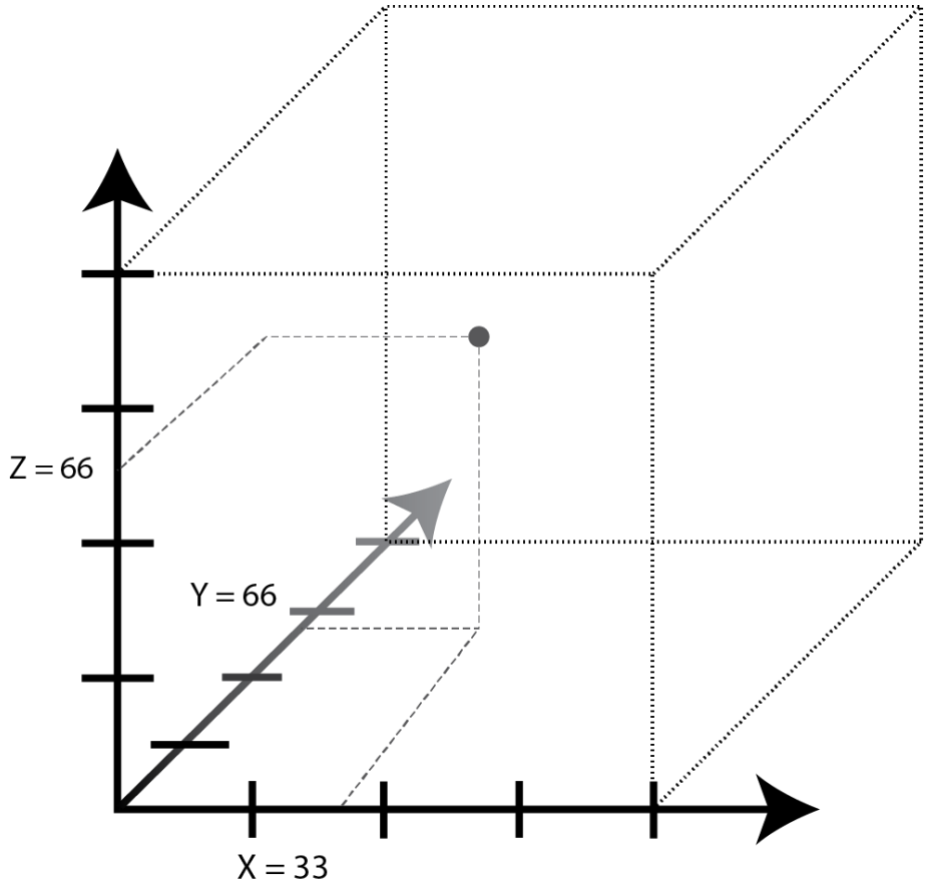
resynthesizes them. The process is time consuming: the later alone takes around 30 minutes for 4 seconds of audio with the hardware I had at my disposal, a long endeavor even when employing multithreading.

Interactive System

The system for interactive exploration of the resultant 64 sounds was built to fill out a virtual 3-Dimensional cubic space with smaller cubic areas corresponding to each sound. It involves tracking the position of the right-hand index finger and playing the sound corresponding to that point along the x, y and z axis in the virtual space, as can be seen in fig 4. If an interesting sound is found, a simple hand closing gesture with the left hand indicates the system to "grab" the sound so that it can be listened to and manipulated without having to keep the right hand still. Opening the left-hand allows the exploration to continue.

A Leap Motion controller is used as a hardware sensor to extract data about fingers motion and position. It employs infrared LEDs and cameras to observe a hemispherical area directly above itself, "it is based on binocular visual depth and provides data on fine-grained locations such as hands and knuckles" (Youchen et al., 2017). The data consists of positions of every joint and finger bone using a cartesian

Figure 4. 3-D timbral space



Note: The sound at this point will be equally influenced by sounds #2 and 3(y=66 and z=66) but half as much by sound #1 (x = 33). A ratio of (2:3:3).

coordinate system in the hand can be extracted via the device's API using Python. The coordinates of the very tip of the right-hand index finger distal phalange are sent to the playback system built on SuperCollider via OSC. The extension of the left-hand fingers is calculated by taking the difference between the position of each fingertip and the center of the palm, then it is sent via the same protocol to Wekinator to be classified.

With Wekinator's k-nearest neighbors' classifier I trained a model to identify a "grab" gesture (closing left hand) using the previously mentioned finger extension

data as input. It then sends the class inferred (0 or 1, gesture absent or present) in real-time to SuperCollider.

The patch programmed in SuperCollider is then responsible for the playback of the audio files. The 64 sound files in every directory created by NSynth are loaded into buffers and placed according to their names into the 3-D space, different batches of interpolation spaces can be chosen by selecting them from a list in a drop-down menu. A graphical user interface gives feedback to the user about the finger position using a 2-D and one 1-D slider mapped to the coordinates. It also displays information about which axis corresponds to which original sound, and the weights on the sound currently being played. If the Wekinator classifier sends 1 then the whole system stops and the playback continues the current sound, which can then be listened to for as long as the user wishes or manipulated by using live-coding techniques. Furthermore, it is possible to explore the space by manipulating the percentages of each sound by incrementally adding it using a MIDI controller, such as using faders found on many commercial models.

Findings

As mentioned before, the model I used was trained on instrument notes, with semi-stable harmonics and clearly defined pitch content. Therefore, it isn't surprising to find that the resynthesis doesn't work well with sounds with strong noise components (like the marbles example) or with sounds consisting of inharmonic overtones (like the

clarinet multiphonic). Therefore, timbre interpolation with such sounds isn't very perceptually convincing. The piano note was the only sound from the first set (fig. 3) whose structure the model captured properly, having the same 1 second decay time as the sounds in the NSynth dataset.

Similar unsuccessful results were achieved while interpolating between a sound of water flowing or whispers: the resynthesis was like the original only in that it contained a similar noise profile and articulations. Good enough to interpolate with another sound, but not to resemble the original sound in any perceptually meaningful way. Better results were achieved when using instruments playing individual notes, even when pitches differed from one another. The latter case sometimes created hybrid sounds with a complex evolution of overtones. Unsurprisingly, the most convincing (but not necessarily more interesting) interpolations are achieved when the three instruments play the same pitch, which is the way the model was trained.

The output sounds are usually low pass filtered (due to the 16kHz sample rate required) and share a crackly, grainy, and distorted quality at times, which might or might not be useful depending on the musical context. The probabilistic nature of the algorithm adds some noise to the signal, but also creates structures not existing in the original sound, in a way akin to Xenakis' dynamic stochastic synthesis (Xenakis, 1971). Thus, the result is mostly unpredictable and nonlinear, with the 3-D exploration helping discover areas where new and sonically interesting structures emerge among unremarkable sounds. Many interesting sounds that would be hard to unearth by

individual trial and error alone can be discovered by interactively exploring batches of interpolations.

The interactive system developed helps in the navigation of a corpus of sounds created by incrementally interpolating between three sounds using the NSynth Wavenet autoencoder. This helped discover the possibility of, under certain conditions, creating 3-D spaces that convincingly interpolate between timbre using such a model. Wherever this wasn't achieved, the model still tried to fit known structures to the sounds, which sometimes resulted in interesting machine learning assisted sound synthesis.

Even though the success of the model depends on generating audio signals serially, the long time it takes for each audio to be synthesized is a drawback of the overall approach. Thus, employing it in real-time applications remains a very distant possibility. However, some research has been made on alternative was to synthesize audio using machine learning, such as Adversarial Neural Audio Synthesis (Engel et al., 2019), that generates audio sequences in parallel.

I personally enjoy the sound quality of the resultant sounds and find them inherently interesting and useful as material for electroacoustic music. Furthermore, even though there is room to grow in the development of the interactive system, I consider the exploration of the resultant space using hand position and gestures not only to be useful to discover sounds, but also an inherently performative activity.

Chapter 3

My Pieces

Dasein

This piece deals with two main ideas found in division I and II of Heiddeger's "Being and Time", namely how human experience is shaped by *modes of encounter* of phenomena and the *authenticity* that shapes such experience. The concept of *modes of encounter* describes how our usual unreflective engagement with objects (as *ready-to-hand*) depends on them being already embedded within a web of significance that precedes our experience of them as objects (*present-at-hand*). The latter usually comes when they are taken out of the usual context, either by malfunctioning or being used in an unfamiliar way. *Authenticity* concerns an engagement consciously guided by envisioning its own mortality and reinterpreting and adapting its own past.

Both are key concepts for his attempts to sketch what a fundamental ontology could be, which is centered around *Dasein* (being-there): Heidegger's term for the entities for whom *being* itself is an issue and who exist as already immersed in their surrounding world, (including, but not limited to, human beings). When *Dasein* exists *authentically*, it is aware of the finite terms of its own being even when immersed and inseparable of its surrounding world

According to Heidegger, Dasein is inherently social, but even if equipment is inseparable from a constellation of shared significance, the experience of engaging

with it is a deeply personal one. Can the difference between modes of encounter be made explicit to the public by relating authentically to a system mediated by sound? To investigate this, I made a piece employing a controller inspired in an acoustic instrument which I could endow with different rules of behavior. As a controller for this piece, I used an EWI[11], a MIDI controller built to resemble and be played like woodwind, that controls a patch programed in SuperCollider. The controller's shape and mode of operation can give rise to a constellation of associations, such as the roles similar acoustic instruments take and what to expect from particular gestures in music performance, that allow for the instrument to become ready-to-hand. In contrast, the same controller can be made to be intractable by giving it unexpected roles or only allowing for limited and dynamic measures of control. This latter case requires conscious engagement with the physicality of the instrument as present-at-hand.

I used the Akai EWI USB model for the performances, which provided continuous MIDI data coming from breath and bite pressure sensors on the mouthpiece, as well as a couple of touch sensors for the right thumb, and MIDI note information from the fingering and a series of rollers on the left thumb to choose between 5 octaves. This delimited the controller's set of affordances to be somewhat similar to those of playing an Oboe, with which I'm deeply familiar and thus could easily experience as ready-to-hand. Within these I determined a mapping scheme with different sets of constraints depending on the octave used, thus allowing the

[11] Electronic Wind Instrument.

performer to effortlessly switch between different levels of control and sonic environments.

The five octaves are mapped as follows:

1. Used exclusively to cue the main 5 sections of the piece.
2. C to A flat either triggers or controls high-level parameters of a series of long events.
3. Allows for playback and minute control of an iterative gesture.
4. Playback and loop of a series of recordings, either of percussive sounds or chants.
5. Accordion-like granular synthesizer, timbre can be changed via MIDI continuous controllers provided by the wind and bite sensors.

Some mappings (like in octave 3 and 5) were explicitly devised to make the controller ready-to-hand, maintaining the usual links such as that between fingering and pitch and making the device almost transparent, focusing on the relation between performer and sound. In contrast, the mapping of some other elements bears little resemblance to the role usually allotted to woodwinds, and is strongly non-intuitive, thus allowing for a present-at-hand engagement. Furthermore, the mapping tends to be more dynamic and thus unpredictable. In such cases the controller itself becomes somewhat inflexible as a tool in the performance, emerging as present-at-hand and switching the focus to that between performer and controller. This is an example of how setting a series of constraints via mapping allows for the emergence of

phenomenologically differentiated experiences in a performer for interactive computer music.

Of course, after performing with it for a while, even the dynamic mapping gets internalized, and the instrument becomes ready-to-hand all over again. So, a successful performance requires the performer to always be pushing the limits of the system. This involves looking from time to time for unexpected behaviors from the system that makes physical engagement with the controller a conscious activity.

The idea is that both modes of encounter afford different kind of control, which result in different relations with the instrument that can be made evident to the public via the physical effort of the performer. This is where the concept of *authenticity* becomes important, describing a way to consciously engage with others without losing ourselves into the public discourse. Authenticity requires an understanding of temporality, as within it is where envisioning outcomes and reinterpreting past experiences becomes possible. Only "authentically" can such a piece of music hope to make engagement visible. Real-time interaction becomes the ideal vehicle, one where the performer can choose at every moment how to act based on previous events and looking for particular results.

Authenticity involves considering Dasein's own past and reinterpreting it in light of their current goals. So, as the composer and planned performer of the piece, I decided to use material with deep personal significance at the moment of working on it. My main struggle at that time was to find a way to move from Chiapas, Mexico to Connecticut, US in at the beginning of the COVID-19 pandemic. Thus, most of the

material comes from field recordings I took during a period of 6 months around that time. The recordings are varied in character, anything from people chanting in Tzotzil language, rubbing of rocks in ancient maya temples, improvisations with Mesoamerican instruments, and environmental recordings of various out-doors environments.

The three first sections of the piece are a series of evolving textures made by processing the recordings, mostly by using granular synthesis. The fourth section slowly fades out the previous textures and leaves a kind of blank slate, allowing for improvisation with live processing and looping of various recordings. The fifth and last section employs as sonic material a reading of the last sentence in Macquarrie & Robinson's translation of *Being and Time* (Heidegger, 1962): "Does time itself manifest itself as the horizon of being?". The sentence is sliced into its constituent words, which are scrambled and treated in a granular fashion like the earlier material. The piece slowly fades without giving any definitive answer to the question I posed earlier about making engagement explicit, just like Heidegger leaves open the question about the nature of time.

An important concept in the sonic environment of the piece is the distinction made by Smalley (1986) between *texture* and *gesture*. The first correspond to sound events whose evolution tends to be slower and to follow an immanent logic, for example the drone that starts the piece or the dense choral textures starting at around

1:20[12]. In contrast, gesture involves sudden changes in the profile of the sound, usually with sharper attacks and a spectral evolution where the energy input is evident. Gesture "it is synonymous with intervention, growth and progress, and is married to causality" (Smalley, 1986). For example, compare the previous excerpts with the more dynamic gestures at 1:30 and 6:20.

The roles taken by elements in the system are usually distributed according to this distinction. The textural elements are triggered and controlled by autonomous processes defined in the code, while gestures are the performer and controller's domain. All the examples previously mentioned follow this distinction, although there are a few elements where the roles are reversed, such as the percussive gestures at 7:59 and the vibrato texture at 12:55. Furthermore, the distinction is purposefully blurred in cases such as the looping of percussion instruments, where constant repetition of short percussive gestures eventually becomes textural in character.

Gesture and texture are used as way to make modes of encounter between performer and controller more explicit, with the first usually being shaped by the performer's actions as mediated by the mapping. Not only that, but the contrast between gesture and texture also mirrors that of present-at-hand and ready-to-hand, with intentionality being the key distinction in both cases.

[12] https://soundcloud.com/hector-gonzalez-orozco/dasein-for-wind-midi-controller

Aurora

This piece explores the idea of musical creation as a middle ground between two extremes: freedom of action and deterministic systems. It was born by reflecting during a period of feeling constrained by natural forces beyond my control and trying to carve enough wiggle room to further my interests, namely the COVID-19 pandemic once again. The idea behind the title was born after trying to watch the Aurora Borealis during a period of intense geomagnetic storms, and how this led to having to make choices (from driving to locations to whether to even stay up to watch for it) based on the current state of astronomical phenomena.

Auroras are caused by disturbances in Earth's magnetic field caused by streams of charged particles released by the sun. When a geomagnetic storm releases a big enough number of them, they interact with Earth's magnetic poles creating a visual display of colors (with some sound component to them, according to some sources) in high-latitude regions.

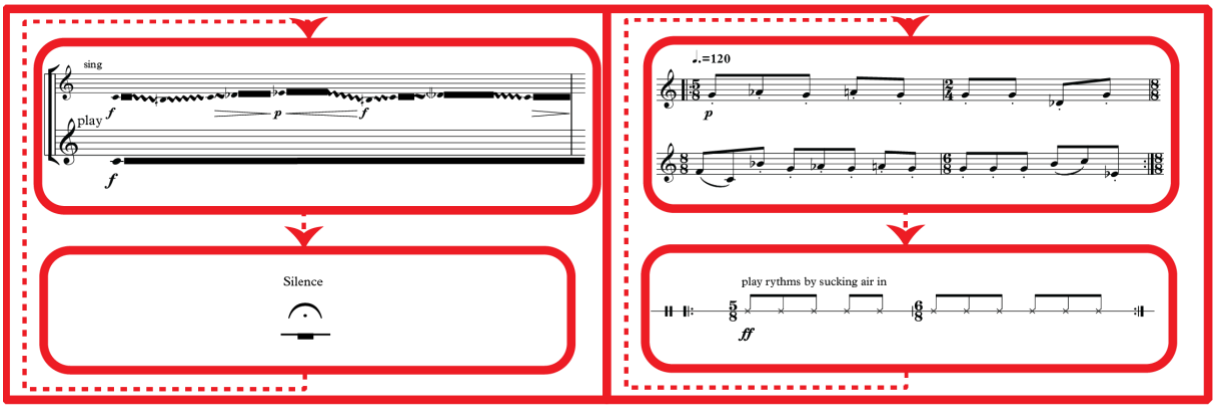
The area where Auroras are visible is expanded during events such as coronal mass ejections, and the simplest way to determine the probability of watching them at a specific latitude is to monitor the K-index. This index is determined by averaging several readings of fluctuations in the magnetosphere taken at 3-hour intervals around the globe It is measured from 0 to 9 with three intervals in between each digit, amounting to 28 different possible levels.

The piece involves a score and a piece of software that functions as a conductor. The score consists of 6 sections, like the one shown in fig. 5. The first one

is through-composed and the rest consist on two columns (A and B) with diverse musical material and instructions. Some sections have material that is more determinate, like specific rhythmic patterns, while others just suggest via graphical notation musical gestures to be performed. Most give at least a couple of choices of material to be played and connects them via edges in a flowchart manner, giving enough freedom to the performer in the way they want to structure each section.

The software looks online for the latest K-index and uses it to determine the length of the whole piece and each of the 6 sections. The index is mapped in an inverse relation to the length of the piece. So, the more magnetic activity leads to a shorter piece, with greater change of rate between material. A graphical interface displays a counter with the time remaining for each section. This is only intended to give a general idea of the location within the whole piece, and performers are encouraged to switch sections independently and even some time before or after the counter reaches 0. The use of Independent, asymmetrical, and even fluid tempos is encouraged, and strict synchronization between both instruments isn't intended.

Figure 5. Fragment of Aurora, oboe part.



From section 2 on, the software also uses the aforementioned index to make a weighted choice for each instrument between the "A" and "B" columns, displaying the result in a graphical interface. Using spectromorphological definitions, the piece can fluctuate between being predominantly textural or gestural, depending on low and high K-index values respectively. Musical material found in "B" columns tend to have more active and rhythmic characteristics, while those found in "A" tend to be more continuous and textural. Therefore, lower geomagnetic activity would on average reflect on a performance with more sustained elements and slower timbral evolution, while the opposite would involve more active and dynamic gestures.[13]

The piece is a sort of very loose sonification of geomagnetic data, following in the footsteps of Charles Dodge's *Earth Magnetic Field*. However, while in Dodge's piece the pitched material is chosen based on successive reading of the k-index, in this case only the form and overall orientation toward the gesture/texture poles are defined by a few readings constrained in time by the length of the performance. The score defines a constrained sonic space within which the instrumentalists choose in real-time the material.

[13] A performance of the piece can be found on:
https://www.youtube.com/watch?v=EKQTLL-AJWo
In this instance the solar activity was on the low side, therefore the predominance of textural passages and slightly lengthy performance.

Leap Studies

The Leap Motion Controller is a hardware sensor capable of contactless tracking of hand and finger positions, described in more detail in the previous chapter. By doing some light editing to a script provided by its software development kit, I managed to transmit data coming from the sensor via OSC, to be subsequently used by audio synthesis software such as SuperCollider. The issue then became how to classify a gesture and ascribe different meanings to its variations.

My experience as an orchestral musician made me interested in how hand gestures can contribute to give shape to musical material. Although I've come to question the overly hegemonic role played by conductors in symphonic orchestras (and all the social and political assumptions this entails), I'm still amazed by how variations on gestures can have some impact in the resultant sound. This can be easily accomplished in systems involving human beings as mediators by harnessing our finely tuned communicational skills, as we attach meaning to subtle variations in verbal and non-verbal communication. Even when there's no clear one to one relation between intention and movement, we tend to ascribe one by relating to it vicariously.

In my past work with motion and gesture tracking I've focused on a very straightforward parameter mapping paradigm: one parameter of movement (be it location, speed, rate of change, etc) gets translated into one or multiple parameters for a sound synthesis engine. However, in order to consider gestures as they evolve in time as meaningful elements in our being-in-the-world and not only as positions or lengths in a three-dimensional space, I had to provide the mapping layer in the system

with a certain kind of agency. In a way this is analogous to an orchestral performer being constantly on the lookout for the conductors' gestures. Machine learning techniques such as classification helped me approach this goal, as well some basic gesture recognition natively provided by the Leap Motion.

The first two studies I worked on utilize only a single gesture: circles drawn by the index finger. In the first study involves rhythmic values that can be added or subtracted to a sequencer, their length determined by the diameter of the gesture. The patch reacts to four types of gestures: Clockwise circles adds and counterclockwise subtracts, while the hand used determines whether the rhythm will be added or subtracted from the beginning (left hand) or the end (right hand) of the list of values. An intuitive interactive system is thus created, with very clear rules and a transparent mechanism.

The second study involves the same set of gestures. However, instead of adding and subtracting single rhythmic values, each gesture creates and loops more complex musical phrases using the HenonC Ugen on SuperCollider, a chaotic generator. Some parameters are determined randomly, but the range used by the pitched material and average amplitude are determined respectively by the diameter and the number of times the circumference is drawn. The resultant sounds are unpredictable but stay within some constraints determined by the gesture.

The third study consists of a 3-track live looping system. Each track can be selected by raising the corresponding number of fingers in the left hand using the index, middle and ring fingers. Each track can be controlled by three right hand gestures: closing the hand to start recording audio coming from the microphone, opening it to start looping the recorded sound and waving it to stop the current loop. Once stopped it can't be played back again and a new loop should be recorded from scratch. This creates a very intuitive system, with hand gestures whose meaning tends to be transparent to both the performer and the public.

To train a system with such gesture recognition skills I once again employed Fiebrink's open-source software called Wekinator. For the third study I employed one of its pre-coded algorithms known as "dynamic time warping", as it can be trained to perceive time-based gestures independently of the speed they are performed.

However, the training is not flawless. The system can classify gestures incorrectly depending on a long series of variables. It can be influenced by the length of the training data, complexity of gesture, slight variations in the position of the sensor, unforeseen secondary gestures and the threshold chosen to confidently choose out of multiple options. Successful training can be a time consuming and frustrating endeavor, and even after achieving satisfying results during a training session I often had to retrain it after picking it up the next day. Furthermore, every person using the instrument would have to train it first, which involves some troubleshooting.
This kind of training is useful when the expected result is an instrument for musical performance adapted to the idiosyncrasies of a particular individual and not that much

as a general-purpose controller for several users for interactive computer music or as part of an installation. That being said, the gestures I chose for this study are simple enough and can be easily trained, and the low number of them also helps with accuracy. I consciously sent a different floating-point number for each hand as part of the training data, even though it could infer this from the position of my fingers I found that some redundancy in the system helps it make better decisions, helping avoid the dreaded issue of overfitting. This makes the role of both hands clear for everyone involved, including the software.

The fourth study is a longer piece that involves a more formal interactive music system than the previous studies. The idea was to explore if a middle ground between Heidegger's two modes of encounter[14] could be approached without employing instrumentality as a medium. Can hand gestures be made to resemble conscious engagement with an object while retaining freedom of action characteristic of the present-at-hand? A formal phenomenological exploration along such lines could lead to the discovery of differentiated nodes along the continuum between the poles of both modes of encounter. Although such ventures are far beyond my means, an informal kind of exploration can still be attempted by employing sound as the object to be manipulated.

I used wekinator once again, but this time to classify hand gestures and assign functions to those classes. The roles of both hands are divided along the previously

[14] Described in the piece *Dasein,* at the beginning of this chapter

described gesture/texture divide, the left hand tends to move at a slower pace and controls the evolution of textural material, while the right one controls dynamic gestures (both in the spectromorphological and literal sense), such as the circular motion described in studies 1 and 2. I choose these roles to imply a relation with the archetypal divide of both hands in homophonic keyboard playing style: melody and accompaniment. The purpose of this is twofold. First, to make the interaction mode and control style of the system more transparent to the public, assuming their familiarity with the keyboard performance. And second, to make a clear connection to instrumentality and anchor the gestures to roles my hands are accustomed to. While the affordances of the leap motion controller can be widely expanded by employing machine learning, a space of constraints can thus be carved to allow for exploration within the piece to occur.

The left hand employed sound material similar to that found on my piece *Dasein*. That is, mostly granulated field recordings taken once again from my surroundings, this time a hiking stop right across the street from my apartment in Middletown, Connecticut. Field recording was employed, because it is itself an activity that turns what is ready-to-hand, the ever-present sonic environment which we are usually unaware of, into a present-at-hand entity, a digital file that can be used for reproduction and manipulation.

The granular texture ranges from short clicks at random intervals to slow changes in the overtonal outline of the recordings. The changes depend on the hand

being closed into a fist, slightly opened, and as open as possible, as well as the flexing of each finger individually.

On the right hand, the mapping is devised so that the hand gestures require more energy input, which is then reflected in the sound. Three types of movement are used to control sound: a swipe motion, clockwise and counterclockwise circles. The first is used sparsely, and simply triggers a kind of percussive hit with a long envelope. The latter two play, respectively, a heavily reverberated tone rising in pitch following continuous circle drawing by the performer, and an individual harpsichord-like tones with pitch inversely mapped to the circumference.

Shoshin

The title is a concept borrowed from Japanese Zen Buddhism, meaning literally “beginners mind”. It outlines a way of approaching familiar practices with new eyes to find possibilities previously unthought of. This is usually meant to apply to the way we approach phenomena in the world, allowing them to radiate their inherent series of affordances without being limited to the constraints we impose to them after encountering repeatedly. According to Shunryu Suzuki’s (1970) famous remark: “In the beginner’s mind there are many possibilities, but in the expert’s there are few”.

This is related to what’s known in the western world as the “Einstellung effect”: a distinguishable tendency to mechanize a set of operations and abstract problems to solve them faster, as described by Luchins & Luchings (1942) famous

water jar experiment. In it, subjects were conditioned to discover an algorithm to solve a series of problems involving measuring a predefined amount of water using 3 jars with different sizes. After that, it was observed that they continue using it even for problems where easier answers were possible. They had generalized a set of rules and extrapolated them to every problem that shared only superficial qualities.

Even if approaching problems with a beginner's mind requires more thought, and is therefore not the usual tendency, it can help bring forth new possibilities by employing the same material. It even has the potential to save lives, for example when employed for disease diagnostics, where every case may be different in some meaningful way.

I wrote this piece for Koto and EWI, with the encouragement and help of Koto player, Japanese Music scholar, and fellow graduate student Garret Groesbeck. "Shoshin" was the approach I employed while working on this piece in two ways. First, I am literally a beginner with respect to writing for Koto and Japanese Music in general, so I could approach the instrument with fresh eyes and find a few sonic possibilities that wouldn't occur to a professional performer, while at the same time learning the conventions of writing for Koto. I spent some time exploring the instrument, trying to find interesting ways for it to produce sound, either acoustically or employing real-time signal processing.

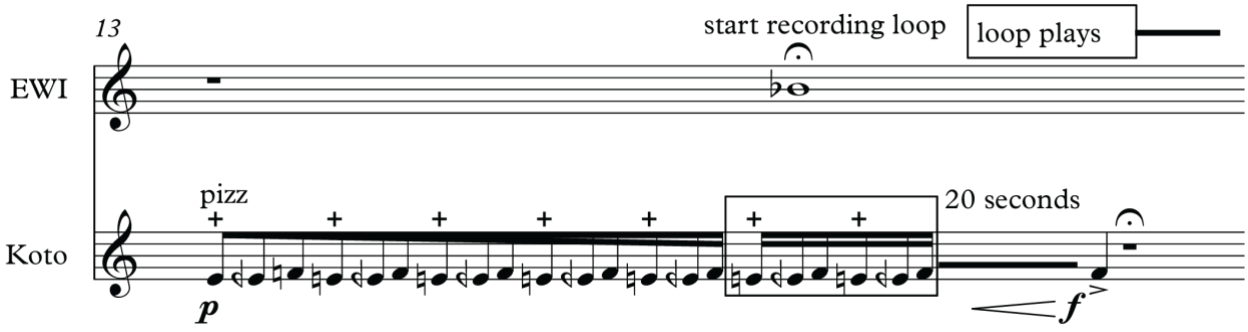
Furthermore, I tried to consciously engage with the concept of the piece by limiting the amount of material I would employ, carving a timbral space within which the piece could exist. This involved once again engaging with the related

Heideggerian concept of *authenticity*, stripping my compositional practice of a lot of the conventions I've been employing the last few years and switching some of the focus to practices that highlight my origins as a conservatoire-trained musician. In short, I decided to employ a more traditional western ABA form, with the material being strongly melodic.

So, the piece evolved as a middle way between both extremes, both of which I considered integral parts of my personal approach to the piece with a *beginner's mind*. Approaching the instrument as an unknown sonic device gave shape to the middle section, employing the variable gestures created by the release of different overtones when striking the string in different areas in relation to the bridge. But also, approaching the A section as a conventional exploration of austere melodic material and silence, leading to more rhythmic material. Both materials were simply juxtaposed, allowing them to coexist and shape the piece.

Furthermore, the Koto and EWI are conceived as shaping an extended instrument. A single contact microphone is placed in the body of the Koto, and the sound coming from it can be processed in real time by the EWI, with every note mapped to a different function independent of octave. Further control is achieved by

Figure 6. Fragment of Shoshin.



mapping air flow to volume, while bite and both thumb sensors provide control of high-level parameters, with intermediate mapping layers as described in the first chapter. This allows for more intuitive and direct control by linking multiple parameters of signal-processing algorithms. Notes C to F# are set to a kind of harmonizer, leaving the original pitch and simultaneously playing two copies of it with the pitch shifted according to the first trichord of a series of scales found in Japanese Music[15], with octave above or below chosen randomly. This can be heard at 3:47 in the recording made for the 8th installment of the Wesleyan Graduate Music Series[16]. G to A allow for control of a usual set of effects: distortion, feedback reverb and delay respectively. B flat and B are used to record, play and stop two different loops. This is used, for example, to extend the gesture shown in figure 6 and heard at 6:20 until it is repeated at 9:50.

The mapping scheme for this piece is less dynamic than the rest of the pieces being discussed here. However, some unpredictability in the interaction with the system is maintained by involving two performers. Both react to each other's actions, with the Koto-EWI paring emerging as a single stream of sound to which both contribute equally.

[15] Yo, Ro, Ritsu, In and Ryukyu scales. F and F# are harmonized with fourths and augmented fourths respectively

[16] Available at: https://www.youtube.com/channel/UCLB2eWWh3MBAH1KTHXxqeWg

Sunyata

The current iteration of the piece is conceived for Oboe and Kinect. I started working on this piece as my final project for the class "Composition in the Arts", co-taught by Professors Ronald Kuivila and Jefrey Schiff. In it, we created a piece every week following a series of prompts, with a final project based on any one of the prompts from the semester. The prompt for was to create a piece in which the material used comes from our own bodies.

I chose to use my body with relation to the interactive system in two fundamentally opposite ways throughout the piece: as provider of sound material being processed by the computer, and as movement shaping the sound generated by it. The body then feeds both audio and control information to the system at different times in the piece.

Sunyata is a concept in Mahayana[17] Buddhist philosophy, highlighted by 2nd century philosopher Nagarjuna as part of the "middle way" (*Madhyamaka*) school of thought. The middle way is an attempt to find a philosophical position that avoided the poles of "eternalism" and "nihilism" (in western terminology). The first posits that things exist because they are sustained by an eternal essence, while the latter ascribed no existence at all to things beyond the mind. What is sought by the *Madhyamaka* is not a synthesis in the Hegelian sense, but a third point of view that explicitly avoids the perceived pitfalls of both.

[17] "Great Vehicle", one of the two major extant Buddhist traditions. It's the main tradition in Tibet, China, Japan, Korea and most of southeast asia.

Thus, Nagarjuna emphasized such distinction with the Sanskrit word *Sunyata*, usually translated as "emptiness". Every phenomenon is said to be empty, but not on the nihilist sense of things being ultimately devoid of existence, but just of the fixed intrinsic existence we usually ascribe to them. Things have no eternal essence, but they do exist by virtue of a process of dependent origination, that is, arising out of a complex web of interrelated phenomena. No thread in the tapestry can be pulled individually. This applies to the self, and goes to explain it as a sort of body/mind/environment system, similar in many ways to *Dasein*. Being is always a *being-in-the-world*.

In Buddhist practice it's not sufficient to understand *Sunyata* conceptually, it has to be experienced repeatedly. Traditionally, visualizations of impermanence and dependent arising are employed for this end, but some sonic practices like Mantras and recitations have been employed by different cultures for the same purpose. In this piece I try to find a way to not only embody the concept of emptiness as a performer, but to create a practice mediated by sound that would help me experience and understand it phenomenologically.

I employed a Kinect as a way to use movement as material for the piece. The device was initially developed as a video-game controller, as it can sense the position and motion of body parts in a space directly in front of it via a series of cameras and infrared projectors. By using a computer program called "NI MATE"[18], I managed to

[18] https://ni-mate.com/

extract coordinates of various parts of my body in a cartesian space (x, y and z) and send them to SuperCollider via OSC.

The piece itself involves no score, and is developed as an interactive computer music system for improvisation. It has two main sections that can be repeated as many times as the performer wishes, before pressing a button in a graphical user interface that stops the piece altogether and fades out the sound.

The first one is triggered if the performer moves out of the sensing area of the Kinect, triggering a dense series of delays that effectively lengthens and harmonizes every sound gesture into a dense cluster with a long decay time. In this section of the piece, I usually explore sounds coming from my body, mainly from my mouth cavity and fingers, as well as the interaction between them and the reed-less Oboe. I continue exploring such sounds, incrementally raising my volume until the output sound feeds back into the microphone and is sustained indefinitely.

The second section starts as soon as the Kinect senses at least one person in front of it, the feedback from the previous section is slowly faded out. For a random amount of time between 2 and 3 minutes, I'll improvise a series of gestures that are recorded onto a buffer. After said time, the recording is separated into harmonic and percussive components, and then sliced into individual elements using automatic audio segmentation as implemented by the Fluid Corpus Manipulation Toolkit[19] on SuperCollider. Such slices are then played and looped in random order thought the

[19] https://www.flucoma.org/

rest of the section, with the playback rate and various effects being controlled by the position of body parts in space.

Even if the mapping between coordinates and effects parameters is explicit and one-to-one, the sheer quantity of them and the way they are distributed through body parts not neighboring each other makes them difficult to control in isolation. There are very few exceptions, such as room size and wetness on reverb growing when moving further back from the sensor, and even then, unexpected results tend to happen depending on the position between body parts. Movement is thus expected to be conducted as meaningful to the performer and/or audience and not only as coordinates in space. The self is thus created as a complex web of relations, embedded and forming part of a bigger interactive system within which agency is distributed but individual actions result in changes.

INCLUDE SOUND EXAMPLES, not here but in text.

"On the right hand, the mapping is devised so that the hand gestures require more energy input, which is then reflected in the sound."

It would be helpful to include sound examples for the approach you take to each section.

Conclusion

During my two years at Wesleyan I've focused on composing and performing with interactive computer music systems, and through doing so I've come to get a glimpse to their potential as tools to explore and deepen my understanding phenomena that might be otherwise elusive. I've tried to make tangible my own understanding of reality as interdependent, in a way that I could manifest it through performing and in my relating with others.

Trying to understand the fundamental unity between object/object and body/environment through manipulating sound has been extremely enriching personally, as now I understand it started as a way to cope with feeling extremely detachment from my environment, not only due to moving to a foreign country and all the expected cultural shocks but arriving during the worst of the COVID-19 restrictions and thus being extremely isolated. Just like objects turning from being *ready-to-hand* to *present-at-hand* is a bit unnerving for *Dasein*, but in this process they reveal previously undisclosed features and uses that can eventually be incorporated back into the *ready-to-hand.*

My attempts have been to use make such non-duality clear both to me personally as a contemplative practice as well as to an audience as a performative action, through exploring analog relations found in sonic material and the architecture of the systems themselves. I've employed them to explore different modalities that objects appear, to investigate using the body as generator of meaning as embedded in

an environment, and to try to approach what dependent origination could mean through an embodied practice. Distributing agency has been instrumental for such purposes, therefore my emphasis and plans to further engage with artistic uses of machine learning.

All this has I've tried to make in an *authentical* way, in the Heidegerian sense previously described, by employing all the tools at my disposal and my situatedness as a starting point. My use of real-time material (sound and movement) emerged out of that urge, and I intend to continue pursuing ways to explore issues emerging at the time of performance that might be impossible to account for in advance. Furthermore, I've used sonic practices and materials that truly speak to musical experiences that have been meaningful or formative for me, balancing and allowing them to coexist in what I envision as a middle way. Just like switching modes of engagement reveals more about *Dasein*, I find that it's in the in between states where the most interesting sounds exist, and practices occur. Petrification is the death of the magic that is life.

Bibliography

Arfib, Daniel, Couturier, Jean-Michel, Kessous, Loic, & Verfaille, Vincent. (2003). Strategies of mapping between gesture data and synthesis model parameters using perceptual spaces. *Organised sound : an international journal of music technology, 7(2)*, 127-144.

Bellona, Jon. (2017). Physical Intentions: Exploring Michel Waisvisz's The Hands (Movement 1). *Organised sound : an international journal of music technology, 22*(3), 406-417.

Chadabe, Joel. (1984). Interactive Composing: An Overview. *Computer music journal, 8*(1), 22-27.

Chadabe, Joel. (1997). *Electric sound : the past and promise of electronic music*. New Jersey: Prentice-Hall.

Chadabe, Joel. (2002). *The Limitations of Mapping and a Structural Descriptive in Electronic Instruments*. [Paper presentation]. Conference on New Instruments for Musical Expression, Dublin, Ireland.

Du, Youchen, Liu, Shenglan, Feng, Lin, Chen, Menghui, & Wu, Jie. (2017). Hand gesture recognition with leap motion. *arXiv preprint arXiv:1711.04293*.

Dreyfus, Hubert L., Anthanasiou, Tom, & Dreyfus, Stuart E. (2000). *Mind over Machine: The Power of Human Intuition and Expertise in the Era of the Computer*. Simon and Schuster, Inc.

Engel, Jesse, Agrawal, Kumar Krishna, Chen, Shuo, Gulrajani, Ishaan, Donahue, Chris, & Roberts, Adam. (2019). Gansynth: Adversarial neural audio synthesis. *arXiv preprint arXiv:1902.08710*.

Engel, Jesse, Resnick, Cinjon, Roberts, Adam, Dieleman, Sander, Norouzi, Mohammad, Eck, Douglas, & Simonyan, Karen. (2017). *Neural Audio Synthesis of Musical Notes with Wavenet Autoencoders.* [Paper presentation]. International Conference on Machine Learning, Sydney, Australia.

Fiebrink, Rebecca, & Cook, Perry. (2010). *The Wekinator: A System for Real-time, Interactive Machine Learning in Music*. [Paper presentation]. Proceedings of The

Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010), Utrecht, Netherlands.

Fiebrink, Rebecca Anne. (2011). *Real-time Human Interaction with Supervised Learning Algorithms for Music Composition and Performance* [Doctoral dissertation, Princeton University]. Princeton, NJ, USA.

Goudeseune, Camille. (2002). Interpolated mappings for musical instruments. *Organised sound : an international journal of music technology, 7*(2), 85-96.

Harrison, Steve R., & Sengers, Phoebe. (2007). *The Three Paradigms of HCI*. [Paper presentation]. SIGCHI Conference on Human Factors in Computing Systems San Jose, California, USA

Heidegger, M., Macquarrie, J., & Robinson, E. (1962). *Being and time*. Malden, MA: Blackwell.

Hermann, Thomas, Hunt, Andy, & Neuhoff, John. (2011). *The Sonification Handbook*. Berlin: Logos Publishing House.

Hunt, Andy, & Wanderley, Marcelo M. (2002). Mapping performer parameters to synthesis engines. *Organised sound : an international journal of music technology, 7*(2), 97-108.

Hunt, Andy, Wanderley, Marcelo M., & Paradis, Matthew. (2003). The Importance of Parameter Mapping in Electronic Instrument Design. *Journal of new music research, 32*(4), 429-440.

Jordà, Sergi. (2005). *Digital Lutherie : Crafting musical computers for new musics' performance and improvisation*. [Doctoral dissertation, Universitat Pompeu Fabra]. Barcelona, Spain.

Jordá, Sergi, Kaltenbrunner, Martin, Geiger, Gunter, & Bencina, Ross. (2005). *The reacTable*. [Paper presentation]. International Computer Music Conference, Barcelona, Spain.

Krefeld, Volker, & Waisvisz, Michel. (1990). The Hand in the Web: An Interview with Michel Waisvisz. *Computer music journal, 14*(2), 28-33.

Lansky, Paul. (1990). A View from the Bus: When Machines Make Music. *Perspectives of new music, 28*(2), 102-110.

Luchins, Abraham S., & Luchins, Edith H. (1942). Mechanization in problem solving: The effect of Einstellung. *The Psychological Monographs, 54(6)*, i-95.

Magnusson, Thor. (2010). Designing Constraints: Composing and Performing with Digital Musical Systems. *Computer music journal, 34*(4), 62-73.

Martirano, Salvatore. (1971). *Progress Report #1, An electronic music instrument which combines the composing process with performance in real time*. Accessed on http://www.jaimeoliver.pe/courses/ci/pdf/martirano-1971.pdf

Minsky, Marvin. (1981). Music, Mind, and Meaning. *Computer music journal, 5*(3), 28-44.

Miranda, Eduardo Reck, & Wanderley, Marcelo M. (2006). *New digital musical instruments : control and interaction beyond the keyboard*. Middleton, Wisconsin, USA: A-R Editions.

Morales-Manzanares, Roberto, Morales, Eduardo F., Dannenberg, Roger, & Berger, Jonathan. (2001). SICIB: An Interactive Music Composition System Using Body Movements. *Computer music journal, 25*(2), 25-36.

Murray-Browne, Tim. (2012). *Interactive music: balancing creative freedom with musical development* [Doctoral thesis, Queen Mary University of London.] London, UK.

Murray-Browne, Tim, Mainstone, Di, Bryan-Kinns, Nick, & Plumbley, Mark D. (2011). *The Medium is the Message: Composing Instruments and Performing Mappings*. [Paper presentation]. Conference on New Instruments for Musical Expression, Oslo, Norway.

Overholt, Dan. (2011). *The Overtone Fiddle: an Actuated Acoustic Instrument*. [Paper presentation]. Conference on New Instruments for Musical Expression, Oslo, Norway.

Palacio-Quintin, Cléo. (2003). *The Hyper-Flute*. [Paper presentation]. Conference on New Instruments for Musical Expression, Montreal, Canada.

Roads, Curtis. (1995). The computer music tutorial. Cambridge, MA, USA: MIT Press

Rodgers, Tara S. (2010). *Pink Noises: Women on Electronic Music and Sound.* [Paper presentation]. Durham and London: Duke University Press

Ryan, Joel. (1991). Some remarks on musical instrument design at STEIM. *Contemporary music review, 6*(1), 3-17.

Schnell, Norbert, & Battier, Marc. (2002). Introducing Composed Instruments: Technical and Musicological Implications. [Paper presentation]. Conference on New Instruments for Musical Expression, Dublin, Ireland.

Smalley, Denis. (1986). Spectro-morphology and Structuring Processes. In Simon Emerson (Ed.), *The Language of Electroacoustic Music* (pp. 61-93). The MacMillan Press LTD.

Sonami, Laetitia. (2006). On my work. *Contemporary music review, 25*(5-6), 613-614.

Sonami, Laetitia. (2021). *Lady's glove. Retrieved January 20*, 2022, from https://sonami.net/portfolio/items/ladys-glove/

Suzuki, Shunryū, & Dixon, Trudy. (1970). *Zen mind, beginner's mind*. New York: Walker/Weatherhill.

Torre, Giuseppe, Andersen, Kristina, & Baldé, Frank. (2016). The Hands: The Making of a Digital Musical Instrument. *Computer music journal, 40*(2), 22-34.

van den Oord, Aaron, Dieleman, Sander, Zen, Heiga, Simonyan, Karen, Vinyals, Oriol, Graves, Alex, Kalchbrenner, Nal, Senior, Andrew, & Kavukcuoglu, Koray. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Van Nort, D. (2020). Sound, Senses, Musical Meaning, and Digital Performance: Epistemological Reframings. *Canadian Theatre Review 184*(1), 57-61.

Waisvisz, Michel. (1985). *The Hands: A Set of Remote MIDI-Controllers*. [Paper presentation]. ICMC, Burnaby, BC, Canda.

Xenakis, Iannis. (1992). *Formalized music : Thought and mathematics in composition* (Rev. ed.. ed.). Stuyvesant, NY : Pendragon Press.

Appendix A: Scores

Aurora score

Shoshin score.

Notations.