Iterative design in DMIs and AMIs: expanding and embedding a high-level gesture vocabulary for T-Stick and GuitarAMI

Eduardo A. L. Meneses

Department of Music Technology
Input Devices and Music Interaction Laboratory
McGill University
Montreal, Canada

April 2022

A dissertation submitted to McGill University in partial fulfillment of the requirements for the degree of Ph.D.

2022-04-21

Abstract

Digital and Augmented Musical Instruments (DMIs/AMIs) are created using sensors, actuators, and sound production (synthesis) components. Mastering instrumental techniques requires developing expertise in any instrument, including DMIs/AMIs.

A critical difference between DMIs/AMIs and conventional instruments is that the user controls and sound production are not acoustically coupled. Because of this, DMI/AMI mappings between instrumental gestures and synthesis units are arbitrary, and performers cannot rely on sound output to create and practice instrumental techniques transferable between performers or pieces. One possible solution is to create a set of techniques based on high-level gestural descriptors, effectively building intermediate mapping layers to expose the DMI/AMI's gestural vocabulary. This high-level gestural vocabulary is created from processed sensor data and organized by movement rather than sound output.

This dissertation presents an investigation of two interrelated research questions about the performance and development of expertise with DMIs/AMIs. The first question is how do instrument designers, composers, and performers develop and expand gestural vocabularies for AMIs and DMIs? The second question is how do these vocabularies impact performance and pedagogy with DMIs/AMIs?

We employed iterative design to investigate aspects of instrument exploration directly related to the reliability, controllability, playability, and longevity of DMIs/AMIs to answer the first question. A high-level gestural vocabulary emerged from the identified parameters and gestural exploration and was embedded into the employed DMIs/AMIs.

We proposed and carried out two research-creation projects to explore how composers and performers learn to play specific DMIs/AMIs and create instrumental techniques based on gestures to answer the second question. We also explored how high-level gestural vocabulary accessibility impacts the engagement of the performers.

Through observations, we verified the role of an already established gestural vocabulary for the engagement of performers and this vocabulary's impact on the learning process. We found that embedding high-level gestural descriptors into the controller firmware and making them as available mapping parameters facilitated access to the instrumental techniques and communication during ensemble performances.

This dissertation contributes to performers, composers, designers, and music technology researchers' understanding of how users engage with DMIs/AMIs. The dissertation contributions extend to how designers can facilitate the use of these instruments by providing and promoting the creation of a high-level gestural vocabulary.

Résumé

Les instruments de musique numériques et augmentés (IMN/IMA) sont créés à l'aide de capteurs, d'actionneurs et d'unités de production sonore (synthèse). Maîtriser les techniques instrumentales est l'une des exigences pour développer une expertise pour tout instrument, y compris les IMN/IMA.

Une différence critique entre les IMN/IMA et les instruments conventionnels est que les commandes de l'utilisateur et la production sonore ne sont pas couplées acoustiquement. Pour cette raison, les mappages IMN/IMA entre les gestes instrumentaux et les unités de synthèse sont arbitraires, et les interprètes ne peuvent pas s'appuyer sur le son pour créer et pratiquer des techniques instrumentales IMN/IMA transférables entre interprètes ou pièces. Une solution possible consiste à créer un ensemble de techniques basées sur des descripteurs gestuels de haut niveau, ce qui mène à construire des couches de mappage intermédiaires pour exposer le vocabulaire gestuel des IMN/IMA. Ce vocabulaire gestuel de haut niveau est créé à partir des données de capteurs traitées et organisées par mouvement plutôt que par sortie sonore.

Cette thèse se focalise sur deux questions de recherche liées à la performance et au développement de l'expertise avec les IMN/IMA : 1) comment les concepteurs d'instruments, les compositeurs et les interprètes développent et étendent les vocabulaires gestuels pour IMN/IMA ; 2) comment ce vocabulaire impacte-t-il la performance et l'enseignement avec les IMN/IMA ?

Nous avons étudié par conception itérative les aspects de l'exploration des instruments directement liés à la conception, la fiabilité, la contrôlabilité et la jouabilité des IMN/IMA. Un vocabulaire gestuel de haut niveau a émergé des paramètres identifiés et de l'exploration gestuelle et a été intégré dans nos IMN/IMA.

Nous avons proposé deux projets de recherche-création pour explorer comment les compositeurs et les interprètes apprennent à jouer des IMN/IMA spécifiques, créent des techniques instrumentales basées sur le geste et, comment l'accessibilité du vocabulaire gestuel de haut niveau impacte l'engagement des interprètes.

A travers des observations, nous avons vérifié le rôle d'un vocabulaire gestuel déjà établi pour l'engagement des interprètes et l'impact du vocabulaire sur le processus d'apprentissage. Nous avons constaté qu'intégrer des descripteurs gestuels de haut niveau dans le micrologiciel du contrôleur et exposer leurs paramètres de mappage facilite l'accès aux techniques instrumentales et la communication pendant les performances d'ensemble.

Cette thèse aide les interprètes, les compositeurs, les concepteurs et les chercheurs en technologie musicale à comprendre comment les utilisateurs interagissent avec les IMN/IMA. Les contributions de la thèse s'étendent à la façon dont les concepteurs peuvent faciliter l'utilisation de ces instruments en fournissant et en favorisant la création d'un vocabulaire gestuel de haut niveau.

Acknowledgements

I want to thank my advisors Marcelo Wanderley and Sérgio Freire. Marcelo, thank you for your enthusiasm, support, tutorship, and belief in your students. You were pivotal in my journey. Sérgio, thank you for sharing your knowledge and being a teacher in the truest sense of the word.

I consider myself very fortunate of studying in the Input Devices and Music Interaction Laboratory (IDMIL). Thank you to all of my IDMIL colleagues for the great interactions, support, exchange of knowledge, and philosophical conversations at the Music Tech kitchen.

A special thanks to the friends that collaborated with me in several projects and somehow contributed to this research: Johnty Wang, Takuto Fukuda, John Sullivan, Jeronimo Barbosa, Alex Nieva, Ivan Franco, Anésio Azevedo, Travis West, Christian Frisson, and Darryl Cameron.

I would like to specifically thank Takuto Fukuda as one of the T-Stick Music Creation Project organizers. Thank you Takuto, for the hard work, commitment, patience, and friendship. I also extend my thanks to all collaborating composers and performers on the project: Diego Bermudez Chamberland, Erich Barganier, Kasey Pocius, Michał Seta, Vânia Pontes, Vincent Cusson, and Yanik Tremblay-Simard.

Extra thanks to Adam Cicchillitti and Steve Cowan for their enthusiasm, commitment, and professionalism during the GuitarAMI Research-Creation Project. We started the project as collaborators and ended as friends.

I also want to thank Joseph Malloch and D. Andrew Stewart. Thank you for the insights and engaging discussions about mapping, instrument design, gesture, and T-Stick.

Thank you all CIRMMT researchers and staff, but especially to Yves Méthot, Sylvain Pohu, and Julien Boissinot for the incredible support.

Extra thanks to Johnty Wang for the partnership, sharing your knowledge, friendship, barbecues, and making my family and me welcome in Canada.

I also like to thank my mother, Antonia de Fatima, and my brother Leandro. I have never heard anything but encouraging words from you. You make me feel capable of anything.

Finally, I could not have done this dissertation without my family: Tati, Bernardo, and Luna. Thank you for sharing this challenging journey with me. Our love is the greatest support of all.

Contents

List of Figures

List of Tables

List of Acronyms

21CGUITAR	21st Century Guitar Conference
AR	action research
AMI	Augmented Musical Instrument
B.E.A.T.	Brazilian Electronic Aleatorium Trio
BLE	Bluetooth Low Energy
CIRMMT	Centre for Interdisciplinary Research in Music Media and Technology
CLOrk	Concordia Laptop Orchestra
DAW	Digital Audio Workstation
DCS	Digital Composition Studios
DMI/AMI	Digital and Augmented Musical Instrument
DMI	Digital Musical Instrument
DSP	Digital Signal Processing
DSR	design science research
FSR	force-sensing resistor
GRCP	GuitarAMI Research-Creation Project
HCI	Human-Computer Interaction
HSI	Human-Sound Interaction
IDMIL	Input Devices and Music Interaction Laboratory
IMU	inertial measurement unit
IoMusT	Internet of Musical Things

JACK JACK Audio Connection Kit

LCD Liquid-crystal display

LP long play

LV2 Linux Audio Developer's Simple Plugin API Version 2

MIDI Musical Instrument Digital Interface

MMR Music Multimedia Room

NIME New Interfaces for Musical Expression

OS operating system

OSC Open Sound Control

PBR practice-based research

PR practical research

RPM revolutions per minute

SPU Sound Processing Unit

SSH Secure Shell Protocol

TMCP T-Stick Music Creation Project

UART universal asynchronous receiver-transmitter

UDP User Datagram Protocol

UNICAMP Campinas State University

USB Universal Serial Bus

Chapter 1

Introduction

1.1 DMIs, AMIs, and digital lutherie

Digital and Augmented Musical Instruments (DMIs/AMIs) are musical instruments created using sensors, actuators, and sound production (synthesis) units. Those instruments are usually divided into two parts: a gestural controller and the already mentioned sound production unit (Wanderley and Battier 2000). Gestural controllers use sensors to acquire and interpret human body movements. This information (sensor data) is connected (mapped) to the sound production unit's sound synthesis parameters. These mappings define the DMI/AMI behaviour, i.e., the relationship between gesture and sound. Digital Musical Instruments (DMIs) are built with gestural controllers and synthesis units whilst Augmented Musical Instruments (AMIs) are built around an acoustic/electric instrument, in addition to the gestural controller and synthesis unit (Miranda and Wanderley 2006, p. 3, p. 21). The acoustic counterpart in AMIs is often used to create sonic material manipulated for the synthesis unit. Miranda and Wanderley define AMIs as a subcategory of DMIs.

The design of DMIs/AMIs, often referred to as digital lutherie,[1] is mainly dedicated to controlling sound synthesis and Digital Signal Processing (DSP) in real time. Jordà (2005, p. 4) refers to digital lutherie as "crafting musical computers" or "constructing the tools that will allow playing and improvising with them [computers]." Digital lutherie most likely evolved from electroacoustic exploration (see Chapter 2) and the desire to: 1) Create devices to make new sounds (DMIs/AMIs); and 2) Control other devices that make new sounds (gestural controllers).

As gestural controllers are employed to control sonic characteristics, one can assume that similarly to acoustic performers, DMI/AMI performers also develop instrumental techniques for their digital instruments, eventually leading to standard gestural vocabularies and techniques. The Hands and the T-Stick are two examples of gestural controllers with established instrumental techniques.

Michel Waisvisz developed and performed with The Hands (Waisvisz 1985) from 1984 until his death in 2008. The Hands is a gestural controller designed in three parts: two hand controllers and one analog to Musical Instrument Digital Interface (MIDI) converter. The controller was designed to output the distance between the performer's hands, one degree of freedom (DoF) orientation (tilt) per hand, and extra user control through switches and potentiometers. It took 32 years from the first time Waisvisz performed using The Hands to Torre, Andersen, and Baldé (2016) provide a more detailed overview of the instrument and its different setups to the New Interfaces for Musical Expression (NIME) community.[2] One year later, Bellona (2017) presented an analysis of the piece *The Hands (movement 1)*, composed by Waisvisz in 1986. This analysis presents the instrumental technique in the

[1] The term lutherie acquired, over time, the meaning of instrument making in general, as opposed to the original meaning of lute maker (*Luthier. In Grove Music Online* n.d.).

[2] The New Interfaces for Musical Expression (NIME) community is built around a conference of the same name. It gathers researchers and musicians interested in sharing and publishing late-breaking work on new musical interface designs. More information can be found at `https://www.nime.org/`.

form of a gestural taxonomy involving hand postures to control the *mercury switches* and arm postures to set the distance between the ultrasonic sensor transmitter and the receiver.

Joseph Malloch and D. Andrew Stewart conceived the first *T-Sticks* in 2006 (Malloch and Wanderley 2007). The T-Stick is a gestural controller created at the Input Devices and Music Interaction Laboratory (IDMIL) in 2006 by Joseph Malloch. More information on the T-Stick can be found in Section 3.3. Performers and composers include Aaron Lindh, Ana Dall'Ara-Majek, D. Andrew Stewart, Diego Bermudez Chamberland, Erich Barganier, Fernando Rocha, Kasey Pocius, Michał Seta, Takuto Fukuda, and Xenia Pestova. The gestural taxonomy created for Stewart's compositions evolved into one of the T-Stick's standard instrumental set of techniques (Stewart 2010; Stewart and Malloch 2010), similar to standard acoustic instrumental techniques. The techniques developed for the pieces *Everybody to the power of one* and *Catching Air and the Superman* constitute the basis of what can be called the T-Stick instrumental techniques.

Although both The Hands and the T-Stick are instruments with an established gestural vocabulary, they present a clear difference: The Hands was performed mainly by Waisvisz, while the T-Stick, as stated before, has been performed by several composers and performers during its 15 years of existence. However, longevity, extensive repertoire, and promoting a community around a DMI/AMI may not be part of the performer's or instrument developer's goals (these elements are discussed in Section 2.4). Several NIMEs have been designed exclusively by and for a single performer, e.g., Cléo Palacio-Quintin's *Hyperflutes* (Palacio-Quintin 2011), Laetitia Sonami's Lady's Glove (Chadabe 1997, p. 249), and The Hands, mentioned above.

Longevity, repertoire, and community become common concerns when designers create or share their instruments with multiple performers and composers. These elements depend on all involved agents being in a constant feedback loop (Meneses, Fukuda, and Wanderley

2020). Composers and performers need designers/builders to make instruments to compose and play, while designers/builders need composers and performers to play and keep their instruments relevant.

Along with the social elements that influence DMI/AMI adoption, technical aspects impact the use of digital instruments. As DMIs, AMIs, and gestural controllers separate control and sound generation. They allow the flexibility to use any gestures to control digitally crafted sounds. However, connections between control and sound synthesis usually introduce delay (latency) and variability (jitter), a recurrent issue when designing gestural controllers and DMIs/AMIs (McPherson, Jack, and Moro 2016).

1.2 Personal motivation

Music is always playing in my head. Eventually, I wanted to externalize those imaginary sounds for sharing, creative, or interaction purposes. The guitar initially fulfilled this objective, as I could imagine sounds and play them on my guitar. In the meantime, I developed other skills I believed were unrelated to music: electronics and computing. However, as my musical universe expanded into new rhythms, timbres, and noises, my capacity to recreate my imagined sounds with the acoustic guitar alone diminished. The possibility of using computers and electronics to recreate these imagined (and imaginary) sounds was casually presented to me in a composition course taught by Ignacio de Campos. As De Campos explained, "There is this piece of software in which you can connect blocks [objects] and use them to create any sounds you want; it is called Pure Data." This unpretentious piece of information permanently changed how I relate to sound and music. My composition, improvisation, and teaching practices became entangled with my electroacoustic discoveries, leading to my master's in "sonology" (music technology) and my current Ph.D. research.

As a guitarist, an idea that naturally emerged was to use electronics to create guitar sounds that I could play in my head but not on the instrument. Those sounds were simply not achievable due to the instrument's physical constraints.[3] Between the desire to externalize my imagined sounds and exploration of electroacoustic music, questions such as "how to create meaningful interactions with the computer while still performing the guitar" and "how to organize these interactions to repurpose them in different performances or compositions" arose. These questions led me to my interest in music technology research, how performers learn to play with DMIs/AMIs, and how to organize this knowledge into digital instrumental techniques.

1.3 Statement of the problems

As stated before, there are no fixed connections between gesture and sound in DMIs, i.e., the mappings between the gestural controller and synthesis unit are arbitrary. To develop expertise in DMI performance, performers need to learn to play the instrument. That goal can be achieved by instrument exploration and performing new or already established repertoires (both approaches are discussed in Chapters 4 and 5). Conversely, composers need to access an established instrumental technique to make their compositions accessible for future performances. Without a fixed relationship between gesture and sound, traditional methods to develop and organize instrumental techniques based on sound output are unsuitable for digital instruments.

Latency and jitter can be troublesome for real-time performances, where performers need to adapt to the delay between action and (sonic) effect. Latency and jitter can also be

[3] An in-depth discussion on the GuitarAMI motivations can be found in Section 3.4 and Meneses (2016).

disruptive for the audience, creating a disconnection between the performer's action and the sonic outcome. Even though instrument designers are aware of potential latency/jitter problems, there is often not enough information or tools to mitigate the problem.

In addition, AMIs require the performer to use new gestures beyond the traditional acoustic instrumental techniques while maintaining cognitive control over the new sonic possibilities. Cook (2001) refers to the set of possible gestures that do not impair the playability of the acoustic instrument, as *spare bandwidth*. Cook uses trumpet players to illustrate the concept of spare bandwidth: the trumpet requires performers to use three fingers to engage its valves, leaving the remaining fingers available to act on added sensors. In this dissertation, we use the *instrument's spare bandwidth* when referring to Cook's definition of spare bandwidth while adding the concept of *performer's spare bandwidth* to define cognitive or physical control limitations imposed by the individual (Sullivan 2021, p. 128).

Spare bandwidth is crucial in DMI/AMI design. AMI designers who do not take spare bandwidth into account can create unintentional constraints that adversely affect the acoustic counterpart performance.

Instrument designers cannot fully identify the performer's spare bandwidth when designing instruments for multiple performers or composers. However, they can consult performers regarding the instrument's spare bandwidth. Instrument designers face the challenge of creating and evaluating an instrument's spare bandwidth while properly categorizing the performer's bandwidth. An in-depth discussion on spare bandwidth can be found in Section 2.5.

Instrumental technique is one of the factors contributing to instrument longevity and the development of expertise in performing with DMIs/AMIs. As previously stated, DMI/AMI users—designers, composers, and performers—cannot rely on fixed mappings for instrumental technique organization. Therefore, how can we facilitate digital instrument learning when there is no fixed relation between gestural control and sound generation? One possible solution is to use gestures to explore and build a DMI/AMI instrumental technique.

1.4 Research objectives

The research presented in this dissertation aimed to investigate two interrelated research questions of performance and development of expertise with DMIs/AMIs:

1) How do instrument designers, composers, and performers develop and expand gestural vocabularies for AMIs and DMIs?
2) How do the developed gestural vocabularies impact performance and pedagogy with DMIs/AMIs?

Two digital instruments were used to address these questions: one DMI and one AMI. The research was then divided into three stages: 1) iterative design and the basis of instrument longevity; 2) instrument exploration and the development of instrumental technique for DMIs/AMIs; 3) instrument performance and interaction between performers and DMIs/AMIs.

We investigated how to explore DMIs/AMIs through gesture during the first stage. We also researched aesthetic and practical considerations in digital instrument design, as these aspects directly impact how performers interact with the devices during composition and performance. This stage's objective was to understand the impact of design choices for the selected DMIs/AMIs for instrument longevity and gestural exploration in the following stages.

During the second stage, we realized a research-creation project to explore how composers and performers learn to play a particular DMI, used the sensor data provided by the instrument in their works, and transferred this knowledge to other performers when exploring the DMI. We invited five collaborating composers to create new musical pieces and perform them in a concert. This project also aimed to expand the DMI/AMI repertoire and user community. The objective was to understand the role of an instrument community to keep the instrument relevant and how gestural vocabulary is developed over time and shared among composers and performers.

During the third stage, we realized a second research-creation project to investigate the impact of established gesture-based instrumental techniques for new AMI performers. We invited two guitarists experienced in electroacoustic mixed music performance to incorporate a guitar-based AMI in their musical practice. The project's artistic goals included a recorded performance of mixed music composed for guitar duo and electronics. The project's research goal was to understand the role of gesture-based instrument vocabularies when learning to perform augmented instruments. We also evaluated the impact of the chosen instrument's gestural vocabulary on the communication between musicians during repertoire preparation.

The objectives and stages are summarized in Table 1.1.

Table 1.1 Summary of the research questions presented, with correspondent research stages and the dissertation chapters.

Research Question	**Stage**	**Chapter**
How do instrument designers, composers, and performers develop and expand gestural vocabularies for AMIs and DMIs?	Iterative design and the basis of instrument longevity	Chapters 2 and 3
	Instrument exploration and the development of an instrumental technique for DMIs/AMIs	Chapter 4
How do the developed gestural vocabularies impact performance and pedagogy with DMIs/AMIs?	Instrument performance and interaction between performers and DMIs/AMIs	Chapter 5

1.5 Dissertation contributions

The research presented in this dissertation is essentially interdisciplinary, involving a research component heavily based on technology and an artistic component based on art in the form of music. The artistic component is crucial as it serves as the laboratory where the research component will be applied and validated.

From an artistic perspective, the use of DMIs/AMIs in music, although not new, is still considered a novelty by many. The technological aspects of digital instrument design and performance are deemed challenging for most musicians and require proficiency in both arts and computing.

From a theoretical perspective, even though we can borrow methodologies from Human-Computer Interaction and other literature on the topic,[4] there is no consensus on the proper methodology to research the interaction between DMIs/AMIs and performers. This dissertation aims to understand the interactions between musicians and technology and how to design tools to facilitate this interaction. More specifically, this dissertation presents the following contributions to the music technology field:

1) High-level gestural descriptors: one of the main contributions of the research presented in this dissertation is how a DMI/AMI instrument vocabulary created from high-level gestural descriptors influences the relationship between performer and instrument. An established gestural vocabulary provides tools to communicate musical information, which improves the performer's engagement with the instrument and aids in the development of expertise for DMI/AMI.
2) GuitarAMI: the augmented instrument using a classical guitar was redesigned during this research at IDMIL. Beyond the AMI, the redesign process used in the GuitarAMI

[4] We discuss methodologies for DMI/AMI design and research in Section 3.1

served as a basis to create and maintain other gestural controllers, e.g., T-Stick and Probatio.

3) Communities of practice for DMIs/AMIs: the research-creation projects presented in this dissertation acted as laboratories to create and expand the communities around the GuitarAMI and the T-Stick. The research-creation projects serve as action models to expand DMI/AMI repertoire and the number of active performers.

4) Artistic works: as a vital component of this dissertation, the research-creation projects presented a substantial number of new compositions and performances. The T-Stick Music Creation Project commissioned and premiered five new compositions for the T-Stick, one T-Stick performance, one improvisation session, and one art installation (Chapter 4). The GuitarAMI Research-Creation Project resulted in four recorded adaptations of existing works: three performances for GuitarAMI duo and one performance for GuitarAMI and T-Stick (Chapter 5).

1.6 Dissertation structure

Chapter 2 presents technical and historical information relevant to fully understanding the context and previous work on each topic explored in the dissertation. We introduce historical aspects tied to live electronics and mixed music exploration, aesthetics, and gestural exploration; and historically contextualize instrumental technique and the first artistic experimentations using music controllers. We also discuss design decisions that impact DMI/AMI longevity and how performers interact with the instrument.

Chapter 3 discusses the methodologies applied during this research and, more specifically, on the research-creation projects. We also discuss the instrument choice for the research-creation projects and the redesign process used to prepare the digital instruments for the artistic activities. Finally, we present the concept of high-level gestural descriptors used to explore and build a DMI/AMI gestural vocabulary.

Chapter 4 presents the T-Stick Music Creation Project (TMCP). This project explored how composers and performers who had not interacted with the T-Stick before used the DMI in their artistic creations. We observed how these composers employed the T-Stick's sensor data to create new instrumental techniques or expand the already established DMI's gestural vocabulary. We also discuss the expansion of the T-Stick community and the instrument's repertoire.

Chapter 5 presents the GuitarAMI Research-Creation Project (GRCP). This project explored the use of the existing gestural vocabulary for the GuitarAMI to facilitate the performance of mixed music repertoire. We discuss the use of high-level gestural descriptors to facilitate creative processes and communication between performers and the instrument designer and whether the use of AMIs impacted interpretative freedom in mixed music repertoire.

Chapter 6 presents a summary of the dissertation's contributions and discusses the impact of the findings and research limitations. Concluding remarks and planned future work are also presented.

Chapter 2

Background and initial considerations

Electroacoustic music, defined as the use of electronic technology in sound production/manipulation, composition, and performance practice, has profoundly transformed music—and our relationship with it—since its inception in the twentieth century. The very paradigms of music and the understanding of musical material were remodeled during this period.

Edgard Varèse proposed one of the definitions of music that synthesizes this conceptual change. Varèse defined music as the art of "organized sound" (Goldman 1961), working with its nuances of spatiality, tempo, and directionality beyond the concept of musical and non-musical sounds. Varèse's definition deconstructed the idea of musical and non-musical sounds, and timbre[1] gained more importance in musical composition and performance. Historically, timbre was a musical aspect entrusted mainly to luthiers (instrument builders), relegating the task of acting within the more restricted timbre palette of most traditional instruments to performers. New technologies, however, enabled an expansion of real-time dynamic timbre manipulation and even the creation of new timbres derived from other sonic elements.

[1] Timbre is defined as a description of the tonal quality of a sound, created from the auditory perception of several factors (Campbell n.d.).

2.1 Input devices

Electronic instruments were developed and used as early as the discovery of electricity (Bongers 2000). However, there was a prolific period of electronic instrument design during the last decade of the nineteenth century and the first half of the twentieth century. Some examples of electronic instruments developed during this period include the *Croix Sonore*, *Electronic Sackbut*, *Hammond Organ*, *Mixtur-Trautonium*, *Ondes Martenot*, *Ondioline*, *Telharmonium*, and the *Theremin* (Chadabe 1997). Bongers (2000) and Roads et al. (1996, p. 622) differentiate the above electronic instruments from DMIs, as the latter use computing systems and DSP. Except for the Theremin, currently found in modern digital versions, which often include MIDI output, the mentioned electronic instruments have fixed or somewhat constrained mappings between input triggers/sensors and the embedded synthesis unit.

This difference highlights one possible motivation to use DMIs. As mentioned in Chapter 1, DMIs have both a gestural controller component (Miranda and Wanderley 2006, p. 3) and a sound synthesis component. Examples of gestural controllers include The Hands (Waisvisz 1985), Lady's Glove (Sonami n.d.), the *Lemur*,[2] the *Radio Baton* (Mathews 1991), and the T-Stick (Malloch and Wanderley 2007).

Gestural controllers, including DMIs that provide data output, offer more control possibilities with continuous gesture acquisition and a performance-oriented design than more traditional Human-Computer Interaction (HCI) devices used in computers, i.e., keyboards and mice.

[2] `https://liine.net/en/products/lemur/` accessed on May 20, 2020.

2.1.1 DMIs, expressivity, and aesthetic choices

Gestural controllers allow performers to be "more expressive" (Jordà 2005, p. 230) when performing a predefined computer music piece and improvising with electronics. It is essential to deviate for a moment and conceptualize expressivity regarding DMIs and musical instruments in general.

Expressivity, the quality of being expressive, semantically signifies "effectively conveying meaning or feeling" (*Merriam-Webster.com* n.d.). In the context of computer music performance, Rovan et al. (1997) associate expressiveness with the level of control provided by gestural controllers. Similarly, Rowe (1996) indirectly relates the lack of expressivity in computer-based systems to its quantized (discrete) nature and, therefore, limited options for humans to perform expressively. We then infer that gestural controllers may facilitate expressivity when they provide more "nuance" in their control possibilities. This effect can be achieved on different levels: 1) on the hardware level, e.g., using sensors with high sensitivity and resolution, and 2) on the software level, e.g., using multiple or complex mapping layers.

While performers can be more expressive using gestural controllers as an alternative to non-musical HCI devices, composers can explore gestural possibilities or variability (Meneses, Fukuda, and Wanderley 2020) by using gestural controllers to interact with DSPs in real-time performances. On the other hand, one can argue that controllers may limit the composer's possibilities during compositional practice in deferred time, presumably by adding undesirable constraints and reducing the number of control possibilities (Menezes 2002). In that sense, real and deferred time can be seen as artificial classifications when discussing the use of gestural controllers. A composer can use DMIs to control processes in real-time, record the sonic outcome, and afterwards manipulate this recording in deferred time; therefore, focusing on the nuance and control possibilities available when using gestural controllers.

2.2 AMI and mixed music: aesthetic and historical aspects

Some composers perceive electroacoustic music as a tool that deviates from the traditional music aesthetic standards and the traditional notion of music. Other composers recall Schaeffer's (1966) concept of sound objects and recognize in mixed music the possibility of expanding the existing sound universe using synthesized and acoustic sounds altogether. The natural dichotomy (Lalitte 2006) between electroacoustic and acoustic extends beyond fusion/contrast (discussed in Section 2.3), manifesting in differences between seen and unseen, tangible or intangible, real time or fixed. Composers can use these perceptual differences to create ambiguity or explore this dichotomy, as fusion and contrast can be established or dissolved during the performance. Even loudspeaker placement can influence fusion and contrast. Acoustic sounds reproduced by loudspeakers and real-time electroacoustic sounds controlled on-stage are disruptive possibilities. Electroacoustic sounds can be embodied by using, for example, active control techniques (Benacchio et al. 2016), and acoustic sounds can be disembodied by using a loudspeaker placed away from the acoustic instrument (McNutt 2003).

However, composing for AMIs presents another layer of complexity than composing for DMIs. Composers need to consider both the electronic and acoustic musical material. From a compositional perspective, working with AMIs can be compared to music composed or performed using both acoustic instruments and electronics, commonly referred to as mixed electroacoustic music or *musique mixte* (Landy 2007).

One of the aspects explored in mixed music related to AMIs is the desire to control pre-recorded material—produced in deferred time—during a real-time performance (Menezes 2009). This scenario lies at the origins of live electronics.[3] Acoustic instruments and computers running DSP can thus both be seen as sound generators. The latter, however, needs an electronic/digital interface to control the sonic outcome: a mouse, keyboard, gestural controller, or even another sound generator. On the other hand, acoustic instruments rely on the physical actuation of the sound source, the vibrating body of the instrument.

The emergence of mixed music as one of the aesthetic possibilities in electroacoustic music and, consequently, its influence on AMI design and usage is directly related to early performance experimentations using electronic devices and traditional instruments. John Cage is considered one of the first composers to use electronic devices and traditional instruments in live performances (Griffiths 1994). *Imaginary Landscape No. 1* (1939) is an example of Cage's electronics exploration. The piece was composed for string piano,[4] china-type cymbals, and two variable-speed turntables reproducing long play (LP) recordings containing sinusoidal signals produced by Victor Talking Machine Co. (Victor Frequency Record 84522A and Victor Frequency Record 84522B). Cage first conceived the piece as acousmatic music[5] to be played as a recording (Menezes 2009), but this work presented the possibility of interaction between acoustic instruments and electronic devices. The electronic equipment used in *Imaginary Landscape No. 1* was treated as traditional instruments in this composition, using a conventional music score for the turntables even though the performers were only responsible for selecting the device's playing speeds—78 revolutions per minute (RPM) or 33 1/3 RPM. Cage also composed four more pieces between 1942

[3]In this dissertation we use the term *live electronics* as any real-time performance using electronics or DSP (Landy 2007).

[4]*String piano* is the term originally used by Henry Cowell to describe the hand manipulation of the piano strings by the performer (Williams 1990).

[5]A term first used by Pierre Schaefer (*acousmatique*) to designate electroacoustic works composed for diffusion in loudspeakers (Peignot 1960).

and 1952—*Imaginary Landscapes No. 2-5*—using electronic devices in performance.

In 1952 Bruno Maderna composed *Musica su Due Dimensioni* for flute and stereo magnetic tape. The composer explored the interaction between the acoustic instrument and electronic devices (Menezes 2009). A technician performs the magnetic tape operation and, together with the flutist, assumes the soloist's role during the performance. Even though some authors consider *Musica su Due Dimensioni* to be the first mixed music composition, Emmerson (2007, p. 89) suggests that *Still Point*, composed by Daphne Oram for orchestra, recorded sounds, and live electronics around 1949, is the first known mixed music composition.

Transición II (1958–59), composed by Mauricio Kagel for piano, percussion (using the internal part of the piano), and two magnetic tapes, was one of the first musical works to manipulate sounds generated by acoustic instruments in real-time during the performance. Kagel used one of the magnetic tapes to reproduce previously recorded sounds. Meanwhile, the second tape recorded acoustic sounds during the performance, performing manipulation processes (mostly cut and paste) and reproduction, generating an echo effect which referred to memories of past events (Manning 2004).

Imaginary Landscape No. 1, *Musica su Due Dimensioni*, and *Transición II* synthesize the development of natural interaction between electroacoustic music and the sonic universe of traditional musical instruments. Even though electroacoustic music offers new timbre and sound organization possibilities, traditional compositional processes and acoustic sounds still aroused the interest of twentieth-century composers. Mixed music brings to the performance elements of variability and interaction, which are impossible with acoustic music.

Other composers experimenting with live electronics and mixed music in the same period include Karlheinz Stockhausen, David Tudor, David Behrman, Gordon Mumma, and Robert Ashley. Ashley and Alvin Lucier founded the Sonic Arts Union in 1966 to promote and perform music and electronic performances (Nyman 1999).

The possible aesthetic choices in mixed music and all technological advances mentioned can make composers, performers, and instrument designers inadvertently confuse aesthetic choices with technical solutions or allow hardware or software technical specifications to guide what should be an aesthetic decision. Even considering that software and hardware exert undeniable influence in the artistic outcome (Lansky 1990), DMIs/AMIs can be seen in the same fashion as traditional acoustic instruments: tools to achieve a sonic outcome, whether this outcome is a new timbre or a musical composition.

Mixed music performance requires the performer/composer to choose the hardware tool most suited to the artistic goal. The task has a seemingly simple premise: the electronics used in mixed music performance need to be controlled using data mapped to the process of synthesis or sound manipulation. The process of controlling electronics can be straightforward (e.g., pressing a single button to start playback) or rather complex (e.g., controlling several synthesis parameters in real-time).

The data required for the process can be generated by pressing computer keyboard keys or using any other devices capable of converting the performer's movements into data. Although the laptop is often considered inappropriate for performance, it has been widely used. Aceituno (2015) and Fiebrink, Wang, and Cook (2007) provided some clever examples of the laptop as a musical instrument in performance.

However, considering the discussion of mixed music practice and performance control in Section 2.3, using laptops in mixed music would make more sense in a chamber music scenario when one (or many) performers would be responsible for the acoustic part. At the same time, another performer(s) would control the electroacoustic counterpart. In electroacoustic processes using real-time acoustic sounds, AMIs can give performers complete control over the sonic crafting process—from acoustic generation to process manipulation.

One can infer that using DMIs can be interesting when working with synthesized sounds that do not depend on the acoustic instruments used during the performance. In contrast, AMIs can be appealing when there is sound manipulation using acoustic instruments as sonic sources and performed in real time. However, there are no technical restrictions to using AMIs and DMIs to control any sound synthesis processes or any other device or process, e.g., light, video, or mechanical/haptic devices. We cannot directly associate DMIs/AMIs with their relationship to the generated sound but consider using and mapping gestural controllers to DSP parameters as arbitrary and guided by aesthetic choices.

2.3 Technique and performance control in AMIs

The use of computers and electronic resources in musical practice became more common and accessible during the last decade of the twentieth century, offering new possibilities for generating and manipulating sounds (Griffiths 1998, p. 146). The new possibilities offered by mixed music practice raise relevant performance and composition issues.

1) Fusion and contrast, often discussed in music composition as compositional tools, assume new roles between the electronic and the acoustic portions in mixed music (Menezes 2002). Electronic and acoustic parts can be perceived either as two entities—i.e., contrast—or a single AMI—i.e., fusion, according to the processing and relationship between the acoustic sound and electronic processes and the sonic outcome.
2) Performers historically possess some interpretive freedom: the score acts as a guide, and the performer has complete control over the sound produced during the performance. For some musicians, mixed music, especially the compositions that use tape as the support media, can provide the performer with less interpretative freedom. McNutt

(2003) refers to this as the *temporal prison*. The performer needs to follow the unresponsive tape (or any playback device), especially when the temporal relationship between live and electronic parts is strictly set.

3) Finally, DMIs/AMIs address the relationship between electronics and acoustic counterparts in mixed music by an *extension bias* (Manzolli 2013). According to Manzolli, the use of electronic resources presents itself as an expansion of composition and improvisation tools, similar to the technique expansion provided by extended techniques in acoustic instruments. The interaction between the acoustic instrument's technical expansion and the electronic music's expansion can become one of the main motivational axes of AMI research. This interaction between acoustic and electronic parts can also be addressed by providing efficient communication channels between the musician and the device responsible for sound manipulation. Historically in live-electronic mixed music, communication was achieved with a second performer responsible for controlling a computer or device performing electroacoustic processes. AMIs present themselves as an alternative to this practice, using sensors to acquire control data from the performer's inputs, e.g., movements and sounds. Once users map sensor data to synthesis parameters, these inputs can be used to control audio synthesis and processing or DSP algorithms (Machover and Chung 1989; Miranda and Wanderley 2006).

Some AMI designers consider the ease of setup and invasiveness. Although those topics are not directly related to performance control, we can argue that setup time is considered a recurrent problem in DMI/AMI design (Hattwick 2017).

It is not uncommon for the performers and technicians to spend the last hours before concerts setting up the electronics and solving technical issues instead of checking sound and familiarizing themselves with the stage. Franco (2019, p. 36) went further to analyze not only the ease of setup but the incompatibility between the computer operation paradigm—visual navigation with point and click actions—and artistic expectations. Franco also points to an expectation of immediacy, i.e., the expectation of an immediate return for every (inter)action performed when comparing acoustic/electric instruments and digital ones.

We consider AMI designs non-invasive where sensors or actuators are not permanently installed into the acoustic counterpart. Designing non-invasive AMIs can present new challenges; however, invasiveness can directly impact setup time in large instruments, e.g., grand pianos (McPherson 2015). Moreover, non-invasive AMI designs enable performers to augment distinctive instruments providing diverse timbre possibilities, e.g., steel or nylon-string guitars, allowing performers to choose a particular instrument with which they have familiarity.

Finally, it is worth mentioning that different DMIs/AMIs present distinctive learning curves, and this subject has been extensively discussed in the literature. Wanderley and Orio (2002) use *learnability* to designate the time required to control DMIs accurately. Levitin, McAdams, and Adams (2002) define the *rewarding point* as the specific point in a learning curve when using a particular instrument becomes rewarding. Wessel and Wright (2002) discuss the *low entry fee/high ceiling* concept, advocating that a good instrument design should allow musical results easily and simultaneously provide "continuous development of musical expressivity" (p. 12). Jordà (2004) explores the efficiency of music instruments, including performer freedom, control complexity, and musical outcome complexity. Learnability, the development of musical expressivity, and the efficiency of music instruments are valuable concepts that can also be applied in DMI/AMI evaluation or validation.

2.4 DMI longevity

Before discussing the elements that influence the exploration and development of an instrumental technique for a particular DMI/AMI, it is useful to discuss instrument longevity. We believe instruments that keep being performed have a higher probability of being deeply explored, allowing performers to develop more advanced instrumental techniques. This probability can be increased if the instrument is performed by multiple musicians, as it may create opportunities for transferring or teaching techniques, leading to the standardization of instrumental techniques.

To the best of our knowledge, there is no definitive answer to why some instruments keep being performed over time and others do not. However, several authors address factors that influence DMI/AMI longevity.

Mamedes et al. (2014) discuss the creation of instrumental techniques for the DMIs, the importance of musical notation, and a dedicated repertoire to foster the preservation of DMIs. The authors suggested that exploration and documentation during composition are essential to establish an instrumental technique for DMIs. Mamedes et al. also suggest that gestures can be used to develop a digital instrument notation. It is important to note that the concept of music notation presented by Mamedes et al. extends beyond score writing and include other methods to register music, such as video recordings.

Marquez-Borbon and Stapleton (2015) discuss the concept of communities of practice (Wenger 1999, p. 45) with examples from within the NIME community. The authors presented the use of laptops as musical instruments as an example of a practice that increases its longevity by finding many practitioners in the community. As a follow-up to this work, Marquez-Borbon and Martinez-Avila (2018) propose creating a pedagogical system and instrument community to sustain and promote a particular DMI artistic practice.

Sullivan and Wanderley (2018) reviewed over 40 years of DMI publications to discuss the design of DMIs for long-term use in performance. The authors concluded that factors limiting the long-term usage of DMIs include instrument stability, reliability, and compatibility. In a subsequent publication, the authors conducted a survey to compare previous research on academic and research-based DMIs with active performer's practices (Sullivan and Wanderley 2019).

Calegario, Tragtenberg, Frisson, et al. (2021) discuss the replicability issue in the NIME community. Instruments that cannot be replicated by researchers and users other than the original designer will invariably face difficulties with respect to the creation or expansion of a community of practice. Availability is fundamental to engaging potential users.

Analyzing these factors, one can appreciate the complexity of instrument longevity. While the arguments mentioned above individually may impact DMI longevity, several of them are interconnected:

- musical notation impacts repertoire creation by composers;
- repertoire impacts the development of instrumental technique;
- replicability and repertoire impact the development of DMI communities;
- stability and reliability impact how these communities engage with the instrument; and
- compatibility impacts the ability to maintain repertoires over time.

In practice, it is not possible to address each of those factors separately.These factors and their influence on DMI/AMI longevity with their associated references are displayed in Table 2.1.

Table 2.1 Some factors that contribute to DMI/AMI longevity. Each factor influences aspects that may promote the use of digital instruments.

Factor	**Influence in DMI/AMI longevity**	**Reference**
Musical notation	Fosters the development of instrumental techniques and repertoire	Mamedes et al. (2014)
Repertoire	Fosters the preservation of DMIs/AMIs	Mamedes et al. (2014)
Instrumental technique	Establishes a DMI/AMI's vocabulary based on gesture	Mamedes et al. (2014)
Communities of practice	Increases instrument relevance and generates pedagogical material	Marquez-Borbon and Stapleton (2015)
Pedagogical system	Promotes artistic practice and the developement of expertise	Marquez-Borbon and Martinez-Avila (2018)
Stability reliability compatibility	Provides minimal requirements for artistic use	Sullivan and Wanderley (2018)

2.5 Affordances and constraints

There is another layer of complexity in AMIs. Designing AMIs implies inheriting all aspects associated with the acoustic instrument, including instrumental technique. Cook's (2001) introduction to the principles for designing computer music controllers is still considered relevant and worthy of discussion and revisiting 20 years after its initial publication (Wanderley 2017). Cook's fourth rule—"some players have spare bandwidth, some do not"—has been understandably cited in several AMI-related works. The reasoning behind this rule lies in the "A" of AMI: one presumes augmented instruments will enhance, or augment, their capabilities rather than impair their performance. While the enhanced expectation may not be fulfilled due to design problems or aesthetic choices, there is undeniably a ceiling where a particular instrument shape or playing technique does not allow extra gestures/movements.

While Cook's examples for his rule focused on acoustic instruments' physical characteristics, the same principle applies to performers themselves. One could then rewrite the rule as *some instruments have spare bandwidth, some do not; some performers have spare bandwidth, some do not.* This differentiation is crucial for AMI design. It guides the instrument designer not only to look for objective constraints (e.g., sufficient fingers to press buttons, the possibility to move the torso, or available feet to press footswitches), but also to provide tools to adjust the instrument to the performer's bandwidth, e.g., setting sensor sensitivity, range, or response time.

Magnusson (2010) proposes a model of constraints divided into three categories:

- objective constraints: physical or environmental limitations of the physical instrument;
- subjective constraints: thinking, creative, or performance constraints, related to human limitations; and
- cultural constraints: created by available technology and the cultural values that support the instrument design.

From Magnusson's constraint categories, one can infer that objective constraints are directly related to the instrument's spare bandwidth, while subjective constraints are directly related to the performer's spare bandwidth. Cultural constraints are tied to instrument design, either by choice and availability of technology or design choices that add constraints to digital instruments.

Conceptualizing affordances, however, can be more troublesome, given the number of often contrasting definitions. In this dissertation, we consider the definition of affordances provided by Gibson (1979, p. 127): the properties or characteristics of the object which afford different interactions relative to the user. For Gibson, affordances are not an inherent part of the object but emerge from the relationship between the object, the environment, and the (human) agent.

Another important aspect of affordances is the agent's perception. Gaver (1991) proposes a classification based on the relationship between perceptual information and the existence of a given affordance. Existing affordances can be either perceptible or hidden; non-existent affordances can be mistakenly perceived as existing ones, and the agent can correctly perceive the absence of particular affordances.

As stated in Section 2.1.1, while the sonic outcomes guide exploration in acoustic instruments, DMIs/AMIs do not have a fixed mapping between gesture and sound. Gestural exploration can guide how performers interact with DMIs/AMIs, as affordances and constraints emerge from an instrument's shape and sensors and their potential interaction with the performer. This emergence occurs at the level of the gestural controller. It is usually detached from specific mappings or sonic outcomes in the DMI/AMI created between the gestural interface and the sound synthesis system.

Perceptible affordances guide how performers interact and explore an instrument by presenting or suggesting interaction modes according to its characteristics. For example, smaller and lighter instruments facilitate fast gestures, while instruments that appear robust can be explored by hitting or tapping the device. Objective constraints have the opposite effect: they prevent performers from doing specific actions, inviting the user to change strategies or devise a workaround. For example, a large instrument may require the user to support the device on the floor or table and rotate around a fixed axis, preventing the effective capture of orientation.

Subjective constraints in DMIs/AMIs are usually related to mapping decisions. Mapping discrete sensors (e.g., buttons) to a potential continuous parameter (e.g., the oscillator's frequency of a synthesizer) limits the user's control over the parameter. However, performers and composers have the flexibility to modify these relationships between pieces, performances, or even within the same piece.

2.6 Time responsiveness

The level of time responsiveness, i.e., how fast a system responds to input, is characterized mainly by latency and jitter (McPherson, Jack, and Moro 2016; Wessel and Wright 2002). For DMIs/AMIs, we define overall (end-to-end) latency as the elapsed time from gestural input to the expected system output, while jitter is the variability of the latency value over time (Brandt and Dannenberg 2002). We can also consider latency and jitter as subjective constraints, substantially impacting gesture exploration. Even though composers and performers can creatively explore instrument limitations (Goudard 2019), users also quickly abandon interactions they deem as unsuitable for controlling—or mapping between—gesture and synthesis parameters (West 2020).

Musicians are used to playing with latency. If we consider the speed of sound at ∼343 m/s, electric guitarists playing 3 meters away from their amplifiers experience a latency of approximately 8.7 ms, while musicians playing in an orchestra 15 m away experience 43 ms latency between the furthest players. French horn players have to wait up to ten cycles of the instrument's fundamental frequency (F_1, or 43.7 Hz) before sound stabilization (standing wave), for a latency between 50 to 100 ms (Fletcher and Rossing 1998, p. 452).

Professional musicians adapt to these situations and can play with latency values considered limiting for DMIs. In an experiment to qualify perceived latency in DMIs/AMIs, Jack et al. (2018) noticed a difference regarding the ability to perceive, discuss, and adjust to action-sound latency between professional percussionists with and without orchestral experience. This observation shows how orchestral and acoustic instrument experience plays a role in fast latency adaptation.

While performers can manage larger latency values, jitter becomes more problematic even in small amounts. The same study above points out that variability of ± 3 ms is enough to impact the performance negatively.

Even though jitter presents itself as a more problematic metric for musical instruments and is not easily compensated for (McPherson, Jack, and Moro 2016), the search for low latency in DMIs/AMIs design tools has been further explored in the literature. In most publications from the NIME international conferences,[6] researchers aim for less than 10 ms latency, an arbitrary value stated by Wessel and Wright (2002), yet very few mention jitter measurements and jitter's impact on playability. Moreover, most experiments use artificial scenarios that provide lower latency and jitter values than in performance situations, with researchers often neglecting computer workload and digital-analog conversion parameters in latency measurements.

2.7 Gestural exploration

A crucial aspect of DMI/AMI instrument design is how users interact with the device. After the design and prototyping stages, composers and performers must learn to play the instrument by repeatedly interacting with and exploring the device. The iteration of this learning process and continued exploration leads not only to the development of performance expertise (Brown 2017) but potentially to the creation of an established instrumental technique.

In traditional acoustic/electric instruments, the instrumental technique is heavily based on the resulting sound. Taking the classical/acoustic guitar as an example, composers primarily convey the desired *musical gesture* in the score, i.e., the characteristics that can be observed through the sonic output independently from the physical gesture/movement responsible for generating the sound (Cadoz and Wanderley 2000). This trend is also

[6]NIME proceedings can be found at `https://www.nime.org/archives/`.

reflected in music notation for acoustic and electric instruments, where most musical information is written as descriptive notation, i.e., specifying the expected sonic outcome rather than gesture/movement (Seeger 1958). Arpeggios and tremolos are examples of classical guitar techniques defined by the expected sound effect and written using descriptive notation rather than the physical movement required to generate these sonic outcomes (see examples 1 and 2 in Figure 2.1). Even when one uses prescriptive notation, which specifies the gesture/movement rather than the expected sonic outcome (Seeger 1958), in acoustic/electric instrument scores, the written information is interpreted according to the expected sound. Performers will execute the *rasgueado* noted in Figure 2.1 (example 3) according to the sound they envision for the piece.

The traditional technique's focus on sonic outcome does not imply that gesture is neglected on an instrumental level. Music teachers and students spend considerable time learning to control their instruments, and there are many studies and articles focusing on proper physical gestures. A notorious classical guitar example of a learning method for instrumental technique with little explanation in terms of movement is the *Série Didactica* (Carlevaro 1967a,b,c,d). In Carlevaro's book series, apart from some introductory notes on hand and arm position, all exercises are written in the score focusing on traditional music elements: pitch and duration. The instrument teacher using these—or similar—methods is usually responsible for all gesture work and evaluation, focusing on the best movements to achieve a particular sound outcome with minimal effort and prevent injuries.

Since the connection between gesture (through sensor data) and sound is arbitrary in DMIs, instrument explorations that rely on sonic outcome are only valid within a particular mapping in a specific musical context. Users who rely on the sonic outcome to guide their gestural exploration cannot abstract a general instrumental technique for DMIs as a different mapping will potentially break the previously set relationship between gesture

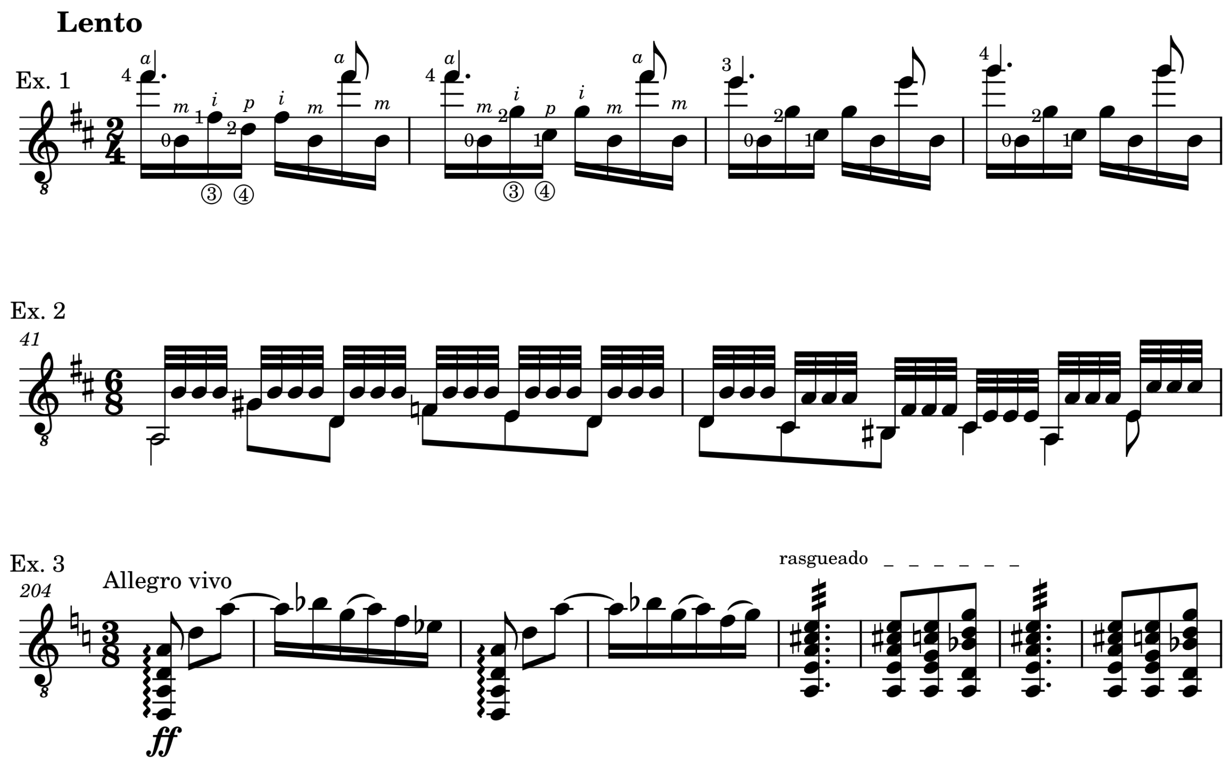


Figure 2.1 Examples of technique notation extracted from public domain classical guitar repertoire, where the strict relationship between instrumental technique and the expected sound output in descriptive and prescriptive notations can be seen. Example 1 is an arpeggio extracted from *La Catedral, I - Preludio saudade* (Agustin Barrios Mangore). Example 2 is a tremolo extracted from *Nocturne "Reverie" Op. 19* (Giulio Regondi). Example 3 is a rasgueado extracted from *Sonata Op. 61* (Joaquin Turina Pérez).

and sound. This instrument exploration demands a new organizational method for creating instrumental techniques in digital instruments.

Chapter 3

Design strategies and methodologies

3.1 DMI/AMI research and design methodologies

There is no standardized methodology or research method in music technology. Since music technology is an interdisciplinary field, researchers are often required to fulfill the methodology requirements—or follow methodological procedures—of different fields related to the proposed research. As the research presented in this dissertation focuses on using high-level gestural vocabulary applied to designing and performing with DMIs, we tackled questions related to engineering, computer science, and the arts—specifically music. It is not within the scope of this dissertation to exhaustively discuss methodology in music technology. However, it is imperative to understand the possibilities and limitations of using multiple research paradigms and clearly define the methodology used in this research.

For DMI/AMI design and evaluation, researchers often borrow concepts from HCI, as presented by Wanderley and Orio (2002) and Zimmerman, Forlizzi, and Evenson (2007). While these tools are handy when applied to DMI/AMI research with a strong technological focus, they do not fulfill all needs when the research delves deeper into aesthetic judgements, social aspects, or the creative exploration of musical devices. The work presented in this dissertation is an example of the latter. Therefore, the observations and findings from the collaborations and DMI/AMI explorations influence the hypothesis, device, and methodology.

In the NIME community, some publications discuss the methodologies used when researching DMIs/AMIs from designer and user perspectives. Gurevich (2016) presents a survey on what the author refers to as NIME research styles or genres. From the presented styles, practice-based research (PBR) most closely fits the scope of this dissertation. According to Gurevich's definition, PBR contributions are related to instrument design or "a theoretical position that the design articulates" (p. 81). Likewise, Cantrell (2017) also suggests categorizing NIME practices. Among the author's categories, practical research (PR) is closest to the scope of this dissertation. PR is also closely related to instrument design; however, Cantrell states that the primary focus of PR is to investigate specific hypotheses related to sound, interaction, and performance. Although both publications provide essential insight on how researchers investigate DMIs/AMIs, there is little information on the specific methods. Nonetheless, NIME-related publications are diverse in scope and, even within PBR and PR contributions, one can expect different methodologies.

Aguinis (1993) compares the scientific method with action research (AR), an alternative research method more suitable for social sciences and arts when the research tackles aesthetic features or moral values. In a slightly different approach, Dresch, Lacerda, and Antunes (2015) present design science research (DSR) as an alternative to AR, where the researcher focuses on developing "artifacts that enable satisfactory solutions to practical problems" as opposed to explaining and potentially solving problems while generating theoretical and practical knowledge (p. 95).

The development of DMIs/AMIs is an intrinsic part of this research's outcome. However, in contrast to the DSR proposed by Dresch, Lacerda, and Antunes, we also aim to understand performers' relationships with digital instruments. Moreover, the researcher's role is also an aspect to consider because they fulfill the function of instrument designer and conduct the investigation.

Finally, to have all the necessary tools to investigate the research objectives proposed in Section 1.4, we need to define the research's practical (active) aspects. One approach that can be used for this task is research-creation. Stévance and Lacasse (2018) define research-creation as an approach that combines creative practices and research methods involving multiple agents (p. 123). The outcome can be 1) academic, e.g., findings and publications; and 2) artefactual, e.g., devices and artistic outcomes.

Therefore, we formulated a method to answer the problems stated in Section 1.3 using characteristics from DSR and AR. At the same time, the development and action stages follow a research-creation approach. All collaborators involved in each of the projects described in Chapters 4 and 5 are either project proponents or invited artists, and they have a profound impact on the project cycle. Their actions and feedback modify the project outcome and the hypothesis reevaluation.

Both research-creation projects described in Chapters 4 and 5 follow the activity organization below:

- Problem definition
- State hypothesis
- Conceptual structure definition
- Plan actions
- Project proposal
- DMI/AMI design/modification

- Iterative cycle
  - Implement actions
  - Evaluate
  - Modify the design and reevaluate the hypothesis
- Final evaluation
- Project Outcomes
  - Artistic: concerts, recordings
  - Academic: reports, publications

3.2 Controller and instrument used during the research-creation projects

As discussed in Section 2.4, it is not typical for DMIs/AMIs or gestural controllers to be used by performers other than the original designer or continuously played after the initial design or the research has concluded. Most digital instruments do not have established instrumental techniques or communities of practice. We cannot use them to investigate how different performers and composers explore and interact with gestural data during artistic creation. We extracted the criteria to choose the controller and AMI used in the research-creation projects developed in this research from the information shown in Table 2.1.

The T-Stick already has an established musical notation, a small community, is a robust instrument (reliability), keeps backward compatibility, and has well-maintained documentation for replicability (Meneses, Fukuda, and Wanderley 2020). Another critical factor for choosing the T-Stick is its extensive repertoire. The T-Stick is a relatively well-known gestural controller, with more than ten pieces composed for the instrument. A list of T-Stick works composed up to the time of the submission of this dissertation in early 2022, including the works developed during the research-creation projects, can be found in Appendix B.

The GuitarAMI is considerably newer than the T-Stick but similar to the latter; it is considered a stable instrument with well-maintained documentation for replicability (Meneses, Fukuda, and Wanderley 2020). The GuitarAMI was also used as a pedagogical tool (Meneses and Fornari 2015), applied in an educational environment, and played in improvisatory performances. These explorations allowed the programming of algorithms to process sensor data, crucial for the explorative processes proposed in the research-creation projects. Sections 3.3 and 3.4 describe the T-Stick and GuitarAMI in more detail.

3.3 The T-Stick

The T-Stick is a gestural controller initially conceived by Joseph Malloch and extensively used by D. Andrew Stewart, created at IDMIL and the Centre for Interdisciplinary Research in Music Media and Technology (CIRMMT) in 2006. The controller has capacitive sensors, an inertial measurement unit (IMU), a piezoelectric transducer, a force-sensing resistor (FSR), and, in some prototypes, an infrared, air pressure, and light sensor, enclosed in a tube (cylindrical shape) (Nieva et al. 2018). The T-Stick has been widely used in compositions and performances in Canada, the USA, Brazil, Argentina, Italy, Norway, France, South Korea, Mexico, and Portugal. During the research presented in this dissertation, six additional performers: Diego Bermudez Chamberland, Erich Barganier, Michał Seta, Kasey Pocius, Takuto Fukuda, and Ana Dall'Ara-Majek were added to the list of T-stick performers/composers, joining Xenia Pestova, Fernando Rocha, Aaron Lindh, and D. Andrew Stewart. Several T-Sticks of different sizes can be seen in Figure 3.1.

Figure 3.1 T-Stick instrument "family" including wireless 2nd generation (2GW) Sopraninos (30 cm in length) on top, wired 2nd generation (2G) Tenor (120 cm in length) in the middle, and wired 2nd generation (2G) Soprano (60 cm in length) on the bottom (Meneses, Fukuda, and Wanderley 2020).

From 2006 to 2007, the first T-Stick instrumental techniques were explored in a collaborative project created by McGill Digital Orchestra performers and a CIRMMT Student Award conducted by Malloch and Stewart.

3.3.1 Established T-Stick techniques

After the first exploration period, the T-Stick was often described as a DMI that can "sense":

- where and how much of the instrument is touched;
- orientation/movement (tilt/roll);
- discrete/time-based interactions such as:
  - tapping,
  - twisting,
  - tilting,

– squeezing,
– shaking,
– brushing,
– jabbing, and
– swinging.

Originally the T-Stick sensor data was sent over Universal Serial Bus (USB) to Max.[1] All data was processed to create gestural information—or gesture semantics—mapping layer (Malloch and Wanderley 2007). Users can create mappings between the gesture information layer and sound synthesis parameters. Both raw sensor data and gesture information were organized according to the namespace hierarchy shown in Table 3.1.

Even though the T-Stick output data only contained raw, normalized, and basic orientation sensor data, extra gesture information was created using algorithms in Max. Exploration based on the controller physicality and affordances was used to build a gestural vocabulary, observed by the *jab*, *grip*, and *lasso* descriptions for the piece *Everybody to the power of one*, composed for soprano T-Stick by Stewart (Stewart and Malloch 2010). Stewart describes mapping as the association between sound synthesis parameters and sensors or gestures. One can observe a clear categorization between lower-level sensor data and the created gestural vocabulary. Stewart referred to the lower-level sensor data as *gestural data* and the gestural vocabulary as *instrumental techniques*. However, Stewart does not detail how gestures are acquired by sensors and processed in the system. We infer that raw sensor data was exclusively accessed to create high-level gestural information, defining the T-Stick playing technique used for other mapping processes.

This explorative approach seems to have been part of the T-Stick framework from the beginning. Malloch and Wanderley (2007) present sensor and gesture information as two

[1] `https://cycling74.com/`, accessed on February 4, 2020. Formerly known as Max/MSP/Jitter.

Table 3.1 T-Stick sensor data sent to Max prior to the T-Stick Music Creation Project (Malloch 2008, p. 29). Malloch used an accelerometer at each end of the instrument to extract angular accelerations from the difference between the axes. As the z-axes of both accelerometers were aligned, one of them was suppressed.

Namespace	**Data**
/tstick/n/raw/piezo	i [0 - 255]
/tstick/n/raw/pressure/1	i [0 - 255]
/tstick/n/raw/pressure/2	i [0 - 255]
/tstick/n/raw/accelerometer/1/x	i [0 - 255]
/tstick/n/raw/accelerometer/1/y	i [0 - 255]
/tstick/n/raw/accelerometer/1/z	i [0 - 255]
/tstick/n/raw/accelerometer/2/x	i [0 - 255]
/tstick/n/raw/accelerometer/2/y	i [0 - 255]
/tstick/n/raw/capacitive	i... [0 - 255]
/tstick/n/cooked/piezo	f [0 - 1]
/tstick/n/cooked/pressure/1	f [0 - 1]
/tstick/n/cooked/pressure/2	f [0 - 1]
/tstick/n/cooked/accelerometer/1/amplitude	f [0 - 1]
/tstick/n/cooked/accelerometer/1/angle	f [0 - 360]
/tstick/n/cooked/accelerometer/1/elevation	f [-90 - 90]
/tstick/n/cooked/accelerometer/2/amplitude	f [0 - 1]
/tstick/n/cooked/accelerometer/2/angle	f [0 - 360]
/tstick/n/cooked/accelerometer/2/elevation	f [-90 - 90]
/tstick/n/cooked/capacitive	f... [0 - 1]

distinct but dependent layers and imply the user will access sensor information only to create gesture information, relying on gesture information to map controller and synthesis portions of the DMI. Malloch and Wanderley also suggest the same procedure for the synthesis portion, creating *musical information* from *synthesis information*.

Malloch and Wanderley provided enough information to hint that not only was high-level gestural information naturally explored, but the designer planned its use on the T-Stick from the beginning. The first composers and performers to use the DMI worked with Malloch (2008) to iteratively design and create the first instrumental techniques from sensor data

(p. 34).

Brush, *fret*, and *touch* are examples of T-Stick techniques created from capacitive sensor data and defined during the early T-Stick explorations. After defining such techniques, one might think users could take advantage of a standard gestural vocabulary right away. However, these techniques, and the extraction algorithms associated with them, were not always available to users. We hypothesize that since the gesture information layer was programmed in Max and the patch was not readily available,[2] T-Stick users exploring the instrument without the guidance of Malloch or Stewart could only use the instrument output available at that time—raw sensor data.

Stewart (2010) states that the algorithms to create the gesture information layer are integral to each composition and the DMI itself (p. 18). This information is contained within each piece's Max patch, even if the same techniques are used in multiple compositions.

3.3.2 T-Stick revisited

The first two iterations, or generations, of the T-Stick, were designed during the McGill Digital Orchestra Project. This interdisciplinary project was developed from 2005 to 2008 and involved researchers from music technology, performance, and composition (Pestova et al. 2009). During the project, several DMIs were developed, including the T-Stick, the T-Box, FM Gloves, and the Rulers (Ferguson and Wanderley 2010). After the McGill Digital Orchestra Project, several modifications were made to the T-Stick building process, mainly to accommodate sensor or electronics replacement due to unavailable, discontinued products (Nieva 2018, p. 15).

An actual upgrade happened only years later through Nieva's work during his master's degree at IDMIL at McGill University. A complete review of the T-Stick sensors and

[2] The first T-Stick repository was created only in October 2018.

implementation was needed for a replication round. These replication rounds are conducted as assignments in an undergraduate music technology course at McGill University. Below, we highlight two important upgrades to the DMI:

1) replacing the accelerometers and gyroscopes used in some T-Sticks with an IMU, and
2) replacing the Arduino Mini and Pro-Mini used in former designs for ESP-based microcontrollers.

The IMU allows the usage of sensor fusion algorithms to produce a T-Stick orientation estimation. The ESP8266 and later the ESP32 allow wireless communication for the T-Stick. The ESP8266 provides 802.11 b/g/n Wi-Fi, while the ESP32 provides the same Wi-Fi capabilities and Bluetooth 4.2 and Bluetooth Low Energy (BLE).

Nieva implemented Open Sound Control (OSC) over Wi-Fi on the T-Sticks 2GW. This implementation allowed backward compatibility with Max patches designed for older versions of the DMI since the OSC namespace remains the same in all T-Stick versions. Users need only modify the Max patch to receive T-Stick data over User Datagram Protocol (UDP).

These modifications, along with revised building instructions, allowed another replication round during Summer and Fall 2019, resulting in 17 new Sopranino 2GW T-Sticks. The availability of T-Sticks enabled the execution of multiple simultaneous projects, including the T-Stick Music Creation Project (TMCP), *Le Vivier Mobile* (2019), and TorqueTuner (Kirkegaard et al. 2020).

3.3.3 Documentation, redesign and replicability/reproducibility on the T-Stick

Documentation is crucial for the redesign process and DMI/AMI longevity. The discussion of replicability and reproducibility in science is not new (Baker 2016; JP. 2005). However, the literature presents inconsistent concept definitions and diverging considerations in social sciences or music technology.

The first issue is the conceptualization of replicability and reproducibility. The definition of the terms is inconsistent in the literature and often conflicting (National Academies of Sciences and Medicine 2019). For this dissertation, we considered the up-to-date definitions proposed by Computing Machinery (2016) and interpreted them in the light of the research presented:

- *replicability*, or *results reproducibility* is described by Goodman, Fanelli, and Ioannidis (2016) as pursuing and finding similar results—or producing similar hardware or software artifacts—using *different* methods, building processes, software code, or hardware specifications that may or may not be inspired by the original experiment or artifact design, and
- *reproducibility*, or *methods reproducibility* is described by Goodman, Fanelli, and Ioannidis (2016) as pursuing and finding similar results—or producing similar hardware or software artifacts—using the *same* methods, building processes, software code, or hardware specifications of the original experiment or artifact design.

In contrast to scientific experimental scenarios, we are more flexible in the DMI/AMI context when describing similar setups and processes. Replicate or reproduce DMIs/AMIs presupposes the construction of a new device with the same characteristics as the original. This construction is not always trivial due to hardware obsolescence, shortage of components, or other unforeseen problems. Therefore, we consider hardware specifications—sensor model, manufacturer, and specifications—criteria for replicability and reproducibility. Moreover, documentation is crucial in defining the building processes, thus ascertaining whether a particular instance of a DMI/AMI can be considered a replication or a reproduction.

An example of this fine line between both processes can be seen in a replicated T-Stick built at Made Makerspace Barcelona.[3] The building process employed at Made, partially documented on GitHub,[4] is noticeably different from the current building process available on the official T-Stick repositories. The models of the sensors used for the Made version of the T-stick are also different from current and older T-Stick designs described by Nieva (2018), keeping similar functionality while adding an ultrasonic sensor that is not part of the original design by Malloch. It is important to note that the Made T-Stick repository presents links for T-Stick documentation previously developed at IDMIL, including the document created by Malloch for a reproducibility round in 2014,[5] and a similar build manual created by Nieva for another reproducibility round in 2017.[6]

The T-Stick documentation organized by Nieva was crucial for the preparation of the T-Stick Music Creation Project, as this documentation was the basis of the reproducibility process that took place before the project. The reproduced T-Sticks used by the project collaborators (see Section 3.3) were built according to the instructions available on IDMIL's public repositories. However, as the instrument building instructions and subsequent builds were updated based on collaborator feedback, one can argue that the redesign process is closely connected with the replicability process.

The reproducibility document created by Malloch, the Made T-Stick, and the work by Nieva are examples of the importance of documentation for DMI/AMI longevity, as building instructions and technical reports contribute to the availability of DMIs/AMIs.

Most instruments lack proper documentation despite the importance of reproducibility and replicability in DMI/AMI longevity. The impossibility of providing information due

[3]`http://made-bcn.org/`, accessed on June, 23, 2020.
[4]`https://github.com/mademakerspace/tstick`, accessed on February, 18, 2020.
[5]`https://josephmalloch.wordpress.com/mumt619/`, accessed on June, 23, 2020.
[6]`http://www-new.idmil.org/education/mumt620-t-stick/`, accessed on August, 10, 2020.

to patents and closed-source software or lack of interest in the documentation for an artifact meant to be ephemeral are possible reasons for documentation deficiency. Calegario, Tragtenberg, Frisson, et al. (2021) surveyed three years of documentation and replicability in the NIME community. The authors found that approximately 67% of the papers did not provide basic information for replicability or reproducibility (build instructions, manuals, videos) other than the publication itself. The authors suggest guidelines for replication-driven documentation in a checklist form and propose further discussion in the NIME community on how to store the replicability information and make it accessible for future reference.

3.4 The GuitarAMI

The GuitarAMI was first designed and built in 2014 in Brazil as a non-invasive AMI intended for artistic exploration using the classical guitar, an ultrasonic sensor, an IMU, and a laptop responsible for audio manipulation and feature extraction using Pure Data.[7] The sensors were initially connected to an Arduino through wires, and the Arduino was connected to the computer through USB. The initial motivation for building the GuitarAMI was to overcome some of the classical guitar's intrinsic sonic limitations, such as the short sustain and the lack of sound intensity control after the attack (Meneses, Fornari, and Wanderley 2015), characteristics commonly associated with the instrument.

Shortly after the first artistic explorations, the GuitarAMI was used during a master's research project at Campinas State University (UNICAMP). Three GuitarAMI prototypes were built between 2014 and 2015 (Figure 3.2). The AMI was used in comprovisation performances and educational projects during this period. Examples of GuitarAMI performances

[7] `https://puredata.info/` accessed on February 4, 2020.

include works from the Brazilian Electronic Aleatorium Trio (B.E.A.T.)—Alê Damasceno (drums and live electronics), Edu Meneses (GuitarAMI), and Walmir Gil (trumpet and live electronics).[8] The GuitarAMI was also used in pedagogical projects at Programa Guri,[9] including a workshop and a course entitled *Música, Tecnologia, e Criatividade*,[10] held in 2015 at one of Programa Guri's music schools in São Paulo, Brazil (Meneses 2016).

(a) First GuitarAMI prototype used in live performances, built using a wired sensor module, DSP unit, audio interface, and laptop.

(b) Wireless RF GuitarAMI sensor module using an Arduino Nano. This module replaced the wired module between versions 2 and 3.

(c) Third GuitarAMI prototype, redesigned to have a modular system and simultaneously improve portability and setup time.

Figure 3.2 Early GuitarAMI prototypes built between 2014 and 2015 in Brazil. The AMIs were used in performances and educational projects during that time (Meneses 2016).

The first two prototypes used a Pure Data patch (*Experimentos_ 0.3.pd*), which contained algorithms addressing the mentioned classical guitar limitations and adding customized

[8]A B.E.A.T. lecture-performance using the GuitarAMI can be found at `https://youtu.be/DlNT8i8i6s0`.

[9]Programa Guri is a music educational program in São Paulo State, managed by the Santa Marcelina Organização Social de Cultura (`http://www.gurisantamarcelina.org.br/`).

[10]Music, Technology, and Creativity.

DSPs: 1) a spectral freeze algorithm to act on the classical guitar's short sustain limitation; 2) a frequency and amplitude modulation algorithm to act as an extension of standard guitar techniques, e.g., vibrato and tremolo; and 3) a looper algorithm to add flexibility in solo performances. The GuitarAMI sensor module and footswitch were mapped to control the algorithm parameters. A multi-effects processor was added to the third prototype to address the classical guitar's amplitude control limitation and provide standard audio effects, e.g., delay, reverb, distortion, and wah-wah.

The third GuitarAMI prototype (GuitarAMI v3) used a Pure Data patch entitled *Time-Machine*. This patch uses the direct FFT/IFFT approach (Arfib et al. 2011) to perform time-stretching operations on a buffer fed with the classical guitar audio captured in real time (Meneses, Freire, and Wanderley 2018). The Pure Data implementation used in the Time-Machine patch is based on the *phase vocoder time bender* patch by Puckette (2007). In addition, a gesture-controlled infinite reverb based on Eli Fieldsteel's original reverb[11] was added as an alternative to the spectral freeze, and the performer can choose between both approaches to sustain sounds for extended periods, depending on the desired aesthetic result.

The Time-Machine mappings use three footswitches, the ultrasonic sensor, and the IMU. The first footswitch activates the spectral freeze (lock and release), the second footswitch activates the time-stretching operations and enables controlling the buffer's reading speed, and the third footswitch returns the buffer's reading to regular speed. The GuitarAMI v3 predefined mappings use the IMU data to control buffer reading speed and the ultrasonic sensor data to modify time-stretching rates.

The motivations to redesign DMIs/AMIs arose during the first two years of performance and educational activities using the GuitarAMI. The redesign process led to improvements

[11] `https://www.youtube.com/watch?v=_2N7lG5uzJI`, accessed on January 5, 2018.

in reliability and accessibility in the fourth and fifth versions of the GuitarAMI. While the sensors and envisioned interactions remained similar to the former versions, the building process, firmware, and software were modified with each version, including updates motivated by feedback from the experiments and projects reported in Chapters 4 and 5.

The fourth GuitarAMI version uses the Prynth framework,[12] while the fifth version uses a modified Linux-based distribution with multiple open-source software to ensure compatibility with several music programming languages. The current GuitarAMI version is designed as a non-invasive modular system, containing a Sound Processing Unit (SPU) and a module attached to the guitar body. The SPU improved portability and setup time, being responsible for receiving the module(s) data, managing mappings, and executing algorithms.

Also, the GuitarAMI SPU (version 5a) contains an audio interface with two inputs and two outputs, allowing either stereo input/output or multiple performers to send audio simultaneously to the same AMI. The SPU provides four USB ports to connect other devices, including external audio interfaces, to expand the number of inputs/outputs. The prototypes currently in use can be seen in Figure 3.3.

3.4.1 Platform and latency investigations on the GuitarAMI

The designer was also the main performer of the first four GuitarAMI prototypes, following a trend already described by Morreale and McPherson (2017). In-depth knowledge of the system allows the designer/performer to predict and fix problems during rehearsal or performance, providing a false sense of reliability that cannot be transferred to other performers. Moreover, when the designer is the performer, there is little incentive to create an intuitive and polished user interface. The designer can usually access system functionalities directly.

[12] `https://prynth.github.io/`, accessed on July 2, 2020.

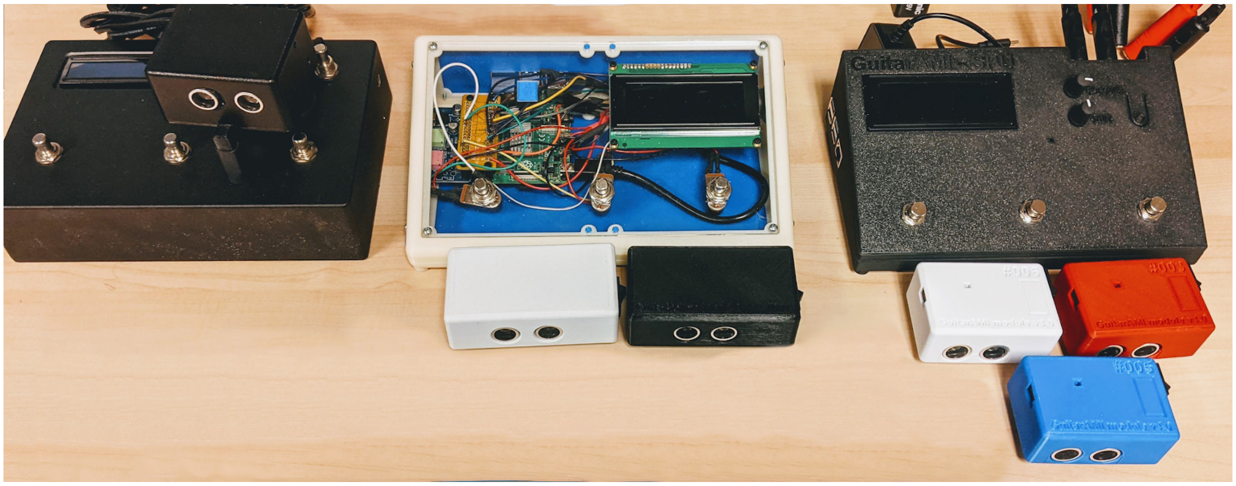


Figure 3.3 Current GuitarAMI versions and their modules: version 4 (left), version 5 (center), and version 5a (right). Version 4 is based on Prynth and runs SuperCollider code, while versions 5 and 5a use a modified Linux distribution capable of running SuperCollider code, Pure Data patches, audio plugins in LV2 format, and custom software installed by the user.

To allow other performers to use the AMI, we needed to improve instrument robustness and user interface. We also wanted to allow flexibility in programming DSP and mappings, as accessibility for composers was an important requirement for the GuitarAMI Research-Creation Project.

The comparison between open-source Linux-based frameworks for AMIs (Meneses, Wang, et al. 2019) showed that choosing a single-board computer or operating system (OS) is not crucial as long as the system is powerful enough to execute the desired code. We tested two Raspberry Pi-based systems and Bela,[13] ultimately creating a custom system to run different music programming languages in parallel.

We also conducted end-to-end latency tests comparing data transmission using wired and wireless connections (Meneses, Wang, et al. 2019). A follow-up study was conducted on Wi-Fi scalability for embedded systems (Wang, Meneses, and Wanderley 2020). These

[13] `https://bela.io/`, accessed on July 2, 2020.

studies allowed us to evaluate appropriately and ultimately choose OSC through Wi-Fi as the primary data communication method for the GuitarAMI, the T-Stick, and other wireless DMIs/AMIs developed at IDMIL.

The first significant result of this latency investigation is that even though wired connections are still more reliable than wireless, the difference is not as critical as designers and performers used to experience ten or more years ago. This result is aligned with other researchers' and instrument designers' perception that the performance gap between wired and wireless technologies is diminishing over time (Cook 2009).

When transmitting data using serial universal asynchronous receiver-transmitter (UART), 98% of the data is received with a latency lower than 7 ms. In comparison, when transmitting data through Wi-Fi using the OSC protocol, 87% of the data is received with a latency of at most 9.7 ms and 96% of the data is received with less than 12.4 ms latency. A graphical comparison using the empirical cumulative distribution function can be seen in Figure 3.4.

For jitter, we found little difference in the systems. Based on the buffer and UDP specificities, the empirical cumulative distribution function allows us to see how most data is received in specific time windows.

Our empirical tests showed that wireless performance is suitable for the DMIs/AMIs developed at IDMIL, including the GuitarAMI. The GuitarAMI could theoretically be more affected by latency and jitter values, as the performer is constantly exposed to acoustic and DSP-based sounds with different response times. Nevertheless, there was no mention of latency and jitter problems during both research-creation projects presented in Chapters 4 and 5 while using wireless transmission.

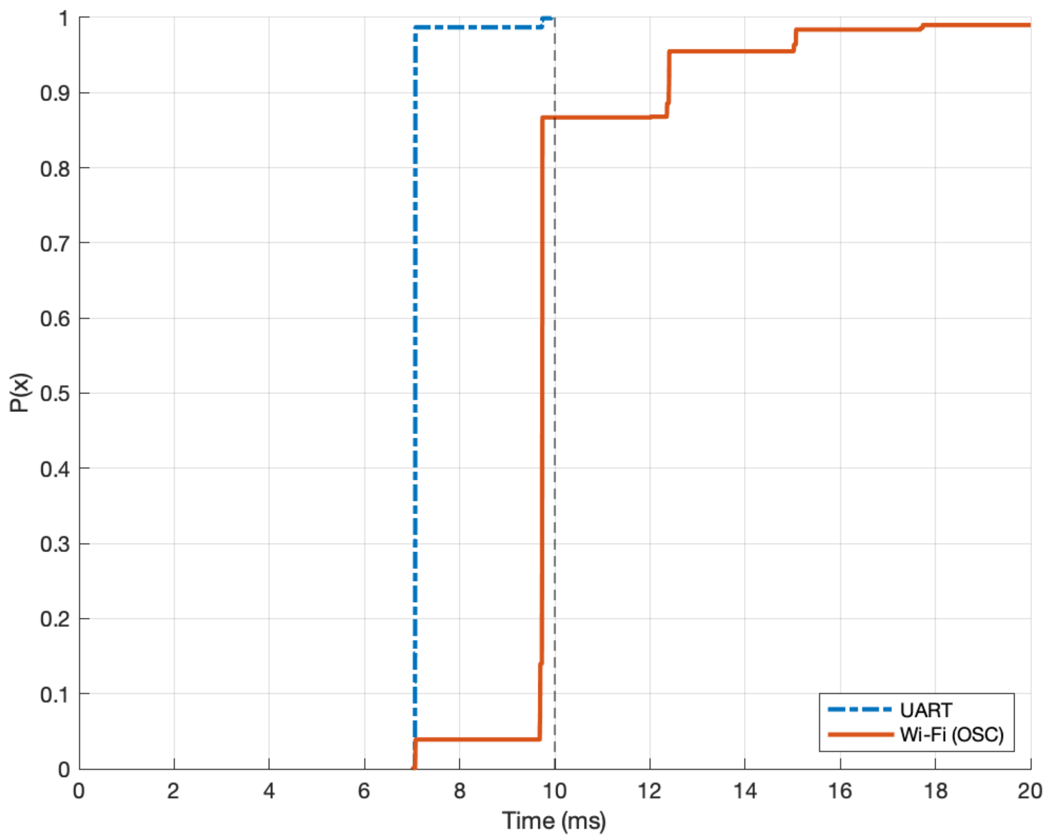


Figure 3.4 The empirical cumulative distribution function of wired (UART) and wireless (OSC over Wi-Fi) sensor data acquisition. Vertical line projections on the x-axis represent the value around which we found a higher concentration of measurements.

3.4.2 Classical guitar and AMIs' spare bandwidth

As discussed in Section 2.5, affordances in AMIs are complex as they are subjected to the interaction between acoustic instrument, augmentation system, and the performer. Magnusson's (2010) definition of subjective, objective, and cultural constraints adds to AMI exploration complexity. Instrument exploration and the consolidation of instrumental technique—crucial aspects of this Ph.D. research—arise from the discovery of designed and "unintentional" affordances/constraints by users.

Therefore, the GuitarAMI required special attention during the iterative redesign, as it is built to augment classical guitars. The sitting position and the use of both hands to perform with classical guitars are constraints the AMI designer should initially consider.

During multiple GuitarAMI redesigns, there was no significant modification of the physical interactions chosen based on performance experience and observations. The physical movements were chosen based on availability. Pedal action (at least one foot is available when the classical guitarist is using a footstool), arm/hand movement captured by the ultrasonic sensor, and guitar/torso movements are still relevant after more than five years of AMI usage. However, performance experience, practical experiments, and user observations allowed us to relate the gestures initially proposed to established classical guitar techniques.

Some objective constraints on the GuitarAMI emerged from the relationship between the acoustic and digital counterparts, but they were case-oriented and usually related to performing concurrent gestures. Ideas for particular compositions or improvisation required incompatible gestures, e.g., strumming and modulating the ultrasonic sensor distance.

Interestingly, we observed that performers found different solutions for these incompatibilities according to the situation, the aesthetic priorities, and the feasibility assessment (Section 5.5).

3.4.3 Performer's spare bandwidth

While some feasibility issues were related to instrument physicality, most constraints discovered during the project exploration discussed in Chapter 5 refer to subjective constraints. More specifically, most interactions deemed impractical were classified as such based on cognitive load, i.e., passages that required intense concentration or physical demands could not be coupled with added gestures.

For all gestures incorporated into the GuitarAMI vocabulary, the range of motion in which sensor data is functional is another layer of complexity related to the performer's spare bandwidth and subjective constraints. We hypothesized that the range of motion for a particular gesture would be determined by two factors related to the performer's spare bandwidth: 1) the amount of movement a particular performer can execute before physically impairing the acoustic instrument's performance, and 2) the cognitive load required to control the movement before affecting the performer's focus.

We designed an experimental setup using an electromagnetic motion capture system to assess the impact of certain gestures in classical guitar performance. While optical motion capture systems, such as the Qualisys,[14] can be used to acquire a range of motion based on the marker's displacement, electromagnetic systems such as the Polhemus Liberty[15] provide relative position and full orientation.

In addition to the orientation information using a single sensor, we chose to use the Polhemus Liberty since the gestural capture does not require a field of vision. This feature allows us to position the sensor on the exact location the performer would use the GuitarAMI Module during performances.

We performed a motion capture session to measure the limits within which a particular performer can move their torso and the guitar while performing a piece in their repertoire. Only one magnetic sensor (receiver) was required for that experiment, as once the sensor is attached to a rigid body—the classical guitar—it provides the body's orientation.

For this particular gesture in the GuitarAMI's vocabulary (tilt), we were interested in three values: the minimum and maximum angles (angular displacement), and the

[14] `https://www.qualisys.com/`, accessed on July 5, 2020.

[15] `https://polhemus.com/motion-tracking/all-trackers/liberty`, accessed on July 5, 2020.

angle the performer considered as *neutral* or *regular position*, i.e., the guitar position in which the performer consider playing without any displacement. The measurement of the maximum angular displacement the performer can move the acoustic guitar without impairing playability can filter and normalize gestural data, thus making the GuitarAMI more responsive and giving the performer greater control over synthesis and manipulation processes during the performance.

The test procedure was initially to measure the theoretical neutral position for a particular performer. This measurement was performed while the performer was seated still before performing the selected excerpt. The performer then played the excerpt while the sensor data and video were recorded asynchronously. Video and data were later synced based on a cue performed at the beginning of the session—a quick shake on the guitar allowed synchronization when the movement changed direction. The performer tried to establish three regions during the performance: the already mentioned neutral position, a forward displaced position, and a backward displaced position. Also, the performer displaced the instrument and body as further as possible before perceiving performance impairment. Finally, the performer also executed the forward displacement using two different methods: following the guitar movement or displacing only the instrument.

The setup can be seen in Figure 3.5.[16] The x-axis ($\Delta_{distance}$) refers to the displaced distance in centimetres between the minimum angle (maximum backward displacement) and the sensor (receiver) position relative to the Polhemus transmitter during the performance. The elevation (Δ_{Θ}) refers to the angular displacement between the minimum angle (maximum backward displacement) and the sensor orientation during the performance.

The data were offset so that the reference ($0°$ and $0\,\text{cm}$) refers to the maximum backward

[16]A video recording of the capture session with motion data can be seen at `https://youtu.be/Ng2ZKBwWN1g`.

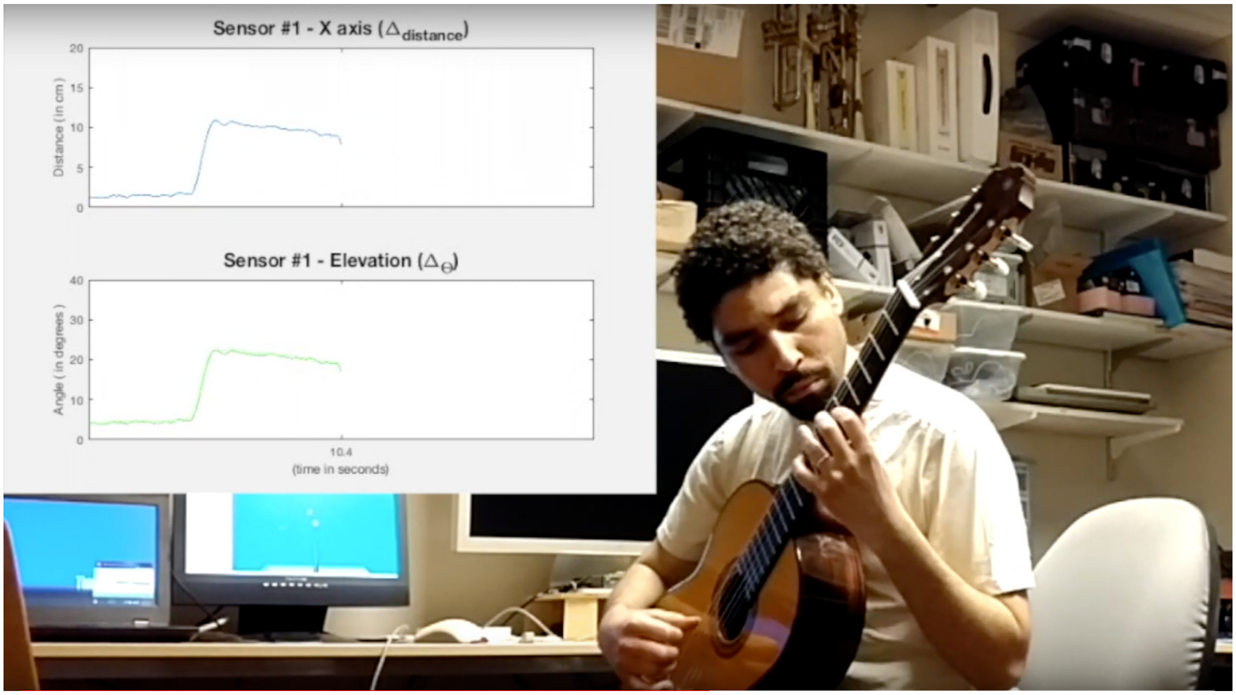


Figure 3.5 Motion capture session using the Polhemus Liberty electromagnetic motion capture system to acquire minimum/maximum angles for nonimpaired performance and neutral position.

displacement. Analyzing Figure 3.6, which comprises approximately 20 seconds of performance, we were able to verify two well-defined regions in time. In the first region—between 0 and 5 seconds—the performer plays in a neutral position, i.e., with the guitarist's torso straight and the guitar back parallel to the chest. In the second region—between 5 and 10 seconds—the performer plays in the *displaced* position, that is, by tilting the instrument at an angle (forward, away from the guitarist's torso) from the neutral position. The average angular displacement reached during the performance in the second region was 16.47° (presented in Figure 3.7).

We also measured two other values: the maximum forward angular displacement was 29.81°, and the backward displacement was −4.377° for this performer. During maximum forward displacement, the performer could not play the guitar naturally while moving

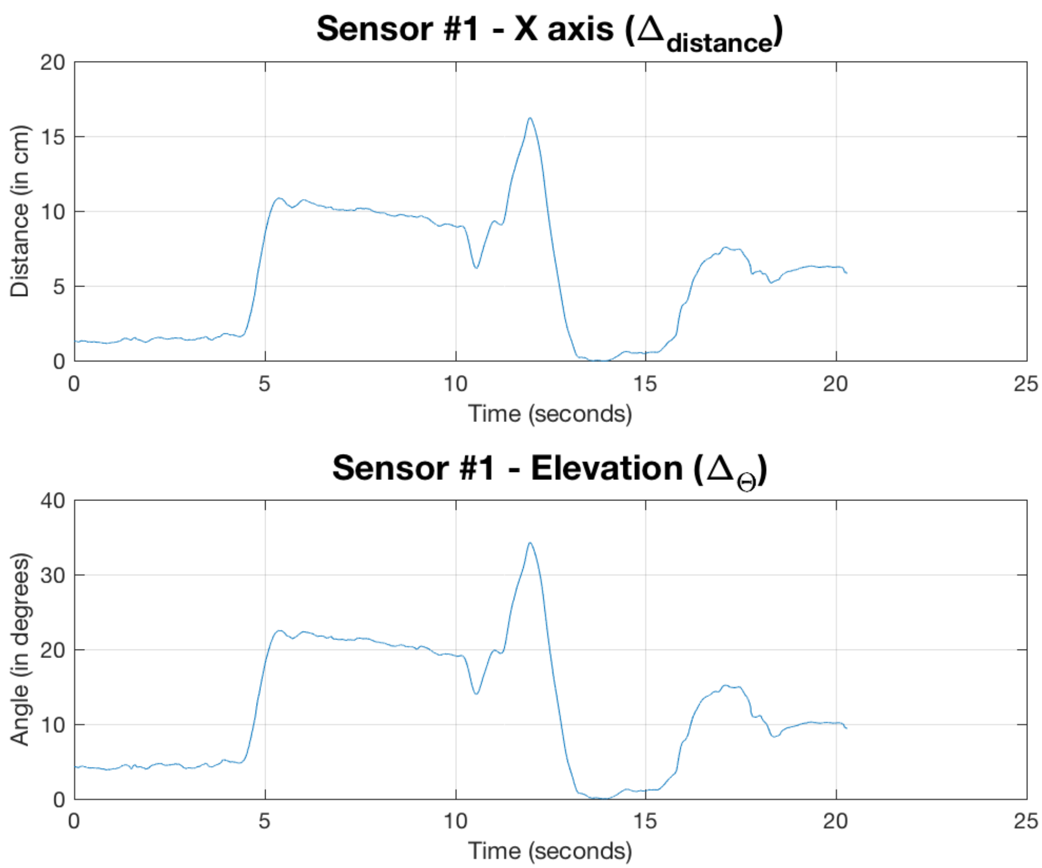


Figure 3.6 Acoustic guitar sensor data (x and elevation). These data were offset so that the greatest distance between the electromagnetic source and the sensor is presented as a reference (0° and 0 cm).

only the instrument, leaving the torso in a neutral position. However, it was possible to accurately control the instrument's position regardless of the displacement angle. The negative angular displacement implies control possibilities when the performer tilts the body and instrument backwards. However, the displacement in that direction is noticeably smaller for this particular performer.

This motion capture session provided a fascinating insight into the performer's expectations and which parameters need to be available for configuration. There is some fluctuation

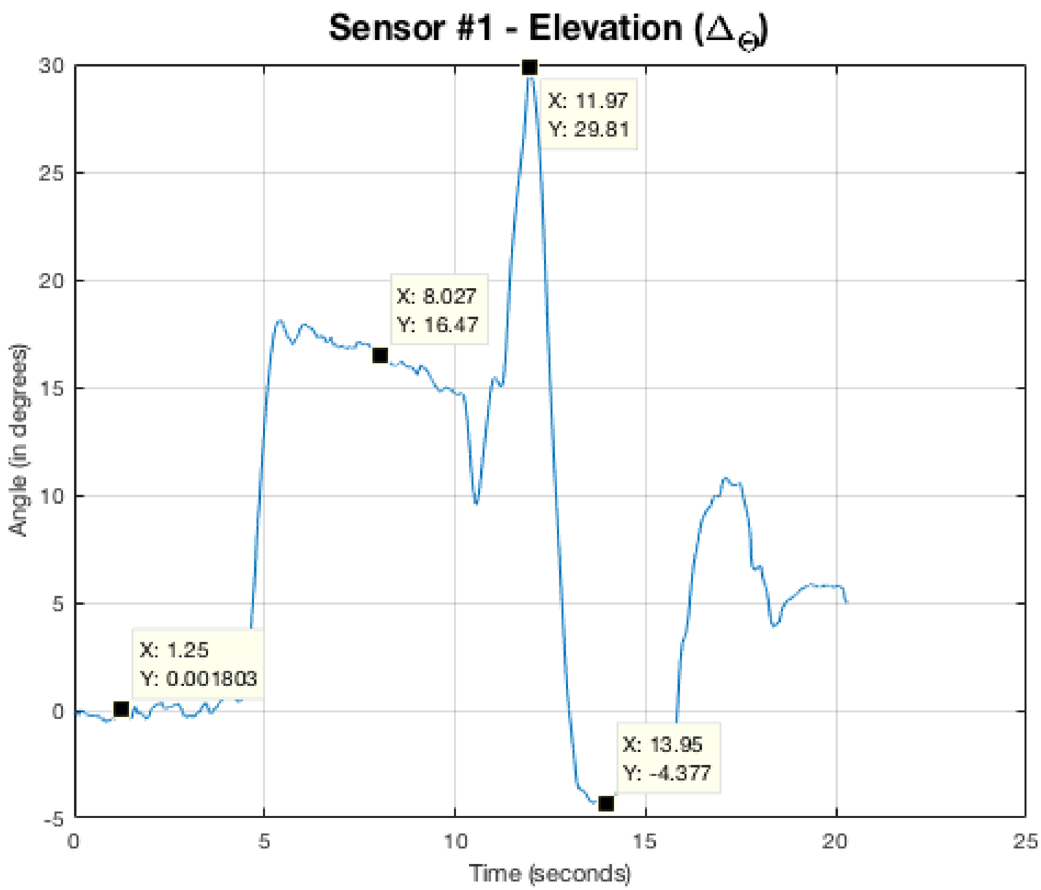


Figure 3.7 Acoustic guitar angular displacement, using the mean angle measured with the neutral position evaluated during a performance.

within a particular position during the performance. The difference between the maximum and minimum measured values during the performance in the forwardly displaced position was 3.35°. This fluctuation implies that programming a sensitivity setting might be helpful to improve instrument controllability.

The redesign following this experiment added parameters to set the *tilt* behaviour and minimize the impact on the performer's spare bandwidth. The parameters added were the minimum angle, maximum angle, and sensitivity, and the same parameters were also used on the distance and roll gestures. We hypothesize that having some parameters to accommodate different performers allows particular gestures to be performed by other musicians and facilitates knowledge transfer.

3.5 Iterative design

The iterative design methodology is central in this research. Expanding on Nielsen's (1993) definition of iterative design, we define *iterative design* as the development and modification of user interfaces by (re)refining their desired features through user testing, feedback, and other evaluation methods.

Applying iterative design methodologies to DMI/AMI (re)design processes is not new. Examples include some gestural controllers previously mentioned in Section 2.1: The Hands, Lady's Glove, and the T-Stick. Other applications of iterative design in the NIME community include the redesign processes of *Codetta* (Ford and Nash 2020) and *SqueezeVox Maggie* (Cook 2009). *Codetta* is a block-based music notation system developed as an educational tool, while *SqueezeVox Maggie* is an augmented concertina used by Cook in music performances.

Some researchers also tackle concepts of interactive design in slightly different scenarios. Vallis, Hochenbaum, and Kapur (2010) propose software and hardware redesign cycles for musical interfaces—iterative controller development—as a development alternative. However, it is notable that Vallis, Hochenbaum, and Kapur's definition of iterative redesign differs from the definition proposed in this dissertation, as the iterations aim to create new musical interfaces rather than refine the existing ones.

As discussed in Section 3.1, the projects presented in this dissertation used feedback and knowledge acquired during the research-creation projects to redesign the DMIs/AMIs and reevaluate the research questions. We used the iterative cycle to improve devices and human interactions inside each project. However, before the research-creation projects described in Chapters 4 and 5, we investigated three aspects discussed in Chapter 2: the instrument's spare bandwidth, performer's spare bandwidth, and latency/jitter.

The objective behind the investigations was to establish metrics or methods to access these aspects during the research-creation projects' experimental stages. Additionally, the latency and jitter investigation provided information on optimal communication protocols and hardware according to each task. A detailed discussion of latency and jitter findings can be found in Section 3.4.1.

In DMI/AMI design, each iterative cycle is often composed of the definition, exploration, prototype, and evaluation phases (Calegario, Tragtenberg, Wang, et al. 2020). The definition phase organizes ideas around interaction, usage, and restrictions for new instruments. The exploration phase comprises sessions for designing the instrument, both physical design (if any) and user interface, based on the directives determined in the design phase. The instrument is built and programmed in the prototype phase, and designers/users can test the device during the evaluation phase. The cycle can be repeated multiple times and even revisited after the instrument is considered mature.

3.6 Gestural exploration paradigms

The Hands is an example of a DMI updated using iterative design processes. Over the years, Michel Waisvisz redesigned The Hands, implementing slight changes improving reliability and playability while keeping functionality and the instrument's main characteristics (Dykstra-Erickson and Arnowitz 2005). However, Waisvisz often halted development for periods (sometimes years) to perform and master the instrument (Waisvisz 2000). It seems to be more difficult for users to develop expertise in DMIs/AMIs if the designer modifies the instrument's characteristics in each new version.

Waisvisz talks about "learning how to play the instrument." Still, designing and performing are two very different activities, even if the designer is also the performer. Once Waisvisz halted development, he could engage in performance practice. Learning how to play can be divided according to how performers engage with the instrument.

Stapleton, Walstijn, and Mehes (2018) suggest two non-exclusive modes of engagement when interacting with DMIs/AMIs: exploratory and performatory. Users in exploratory mode test and search for instrument affordances, while users in performatory mode use the affordances previously discovered in predefined setups, i.e., gestures/instrumental techniques or compositional processes.

There are similarities in exploration methods for DMIs and acoustic instruments. Composers and performers can use established instrumental techniques but sometimes envision new creative ways to interact with the instrument, e.g., tapping in melodic instruments to achieve percussive sounds or bowing percussion instruments to create sustained tones. During the 20th century, these new explorations in acoustic/electric instruments were classified as extended techniques.

New gestural exploration using raw data is the norm for DMIs/AMIs with little to no established vocabulary, and direct access to the sensor output. However, how gestural information is presented influences how performers and composers explore the instrument, in a process similar to how mapping possibilities are visually presented influences how users visualize and create mappings (Wang, Malloch, et al. 2019).

Regarding gestural controllers, raw sensor data usually represents the lowest level of information available to the user. Raw sensor data is acquired at the sensor output, often as a voltage level that can be converted to a unit related to the physical phenomena measured by the sensor, e.g., an IMU outputs acceleration on the x-axis in gravitational force equivalent (g-force). Conversions of the unit, e.g., from g-force to m/s^2, are usually treated as raw sensor data.

Hunt, Wanderley, and Kirk (2000) suggest creating an intermediate mapping layer to provide to the user abstract parameters using sensor fusion or data processing. This processed (cooked) data is usually at a higher level and represents gestural data (data associated with physical gestures) using multiple sensors (sensor fusion), filters, and custom algorithms. Malloch and Wanderley (2007) applied the same principle, where the authors use the term *gesture information* to denominate this layer.

The hypothesis is that sensor data transformation and presentation influences composers and performers to use more complex or more straightforward direct mappings. Unless composers and performers specifically aim to create convergent mappings, i.e., the performer is knowledgeable about DMIs/AMIs and versed in mappings for digital instruments; most interactions will likely be direct, either one-to-one or divergent (West 2020, p. 56).

Moreover, the mere presence of any processed data may shift the user's exploration paradigm from the DMI's affordances to the application of the provided data. DMI users can explore and interact with already processed sensor data or use raw sensor data to create their gestural information layers. Each distinct gestural exploration paradigm has particular advantages and challenges.

3.6.1 Exploring raw sensor data

Probably the most common method to explore DMIs is accessing raw sensor data and mapping directly to synthesis parameters. Many DMIs have well-defined controller and synthesis parts, providing mapping possibilities and digital instrument flexibility, but very few outputs processed data out-of-the-box. As previously stated, how sensor data is presented influences mapping decisions. This influence is especially strong when users other than the instrument designer are responsible for the mapping. When raw sensor data is presented as a list of options for mappings, users experiment with creating mappings from an available item on the list to another on the synthesis parameters. As observed in West's (2020) experiment, users try mappings empirically. Even if it is possible to experiment with convergent mappings, the process is not intuitive, as we observed during the projects presented in this dissertation.

3.6.2 Exploring physicality and affordances

For designers and users, an alternative to exploring and using raw sensor data is to explore the physicality of the DMI and the interaction with that object. One noticeable advantage of creating an abstraction layer for the high-level gestural vocabulary is that the abstractions can already contain complex mappings, mitigating the lack of convergent mappings created by composers and performers while connecting gestural and musical information.

In acoustic/electric instruments, the sonic result is usually dependent on the combined effects of multiple inputs. One example of this phenomenon in acoustic instruments is single-reed instruments (e.g., clarinets and saxophones), where both breath and lip pressure (mouthpiece pressure) account for loudness variation (Cadoz and Wanderley 2000). Clarinetists learn to control these input parameters by practicing and, once they internalize the technique, professional performers can focus on the sonic result and unconsciously realize the necessary movements. Designers can use convergent mappings to create a gestural information layer that behaves similarly to an acoustic/electric instrumental technique. In that scenario, the composer can request a particular technique and the performer empirically deals with the data convergence during execution.

Also, the connection between the sensor and gestural information often needs signal conditioning, filtering, or sensor fusion algorithms. Even though one can search in communities such as NIME, measuring how many instrument designers and performers rely on creating a gestural information layer as part of their workflow is difficult. Unless authors describe their framework in detail or provide source code, it is impossible to confirm if sensor fusion techniques or higher gestural representation were employed (Medeiros 2015).

3.7 High-level gestural descriptors

The gestural information mapping layer allows performers to create high-level gestural descriptors, i.e., interactions with the instruments using complex human gestures/movements such as *push*, *jab*, *wave*, *point*, or *shake*, rather than using raw sensor data such as *ultrasonic sensor at 5 cm*, or *accelerometer x-axis at 30 degrees*. Subjective constraints are more difficult to grasp and not often explicitly explored in DMI/AMI design; however, mappings can be set to increase the difficulty to access specific outcomes, emulating physical constraints in acoustic instruments. One example can be found on the gestural vocabulary for the T-Stick: the *shake* gesture is extracted using a leaky integrator and, the more the performer shakes the instrument, the more difficult it is to reach a higher value. Mappings using this gesture are often adjusted in a range that makes it impossible for the performer to reach the maximum value accepted by the DSP algorithm, creating a ceiling that mimics the sound intensity expected behaviour from acoustic instruments.

The instrumental technique, or gestural vocabulary for DMIs/AMIs, can be organized by creating high-level gestural references when exploring new instruments. These high-level gestural descriptors (e.g., jab, rub, tap, mute, or bend) are created from gestures used during the assessment of the DMI/AMI physical affordances and acquired using raw sensor data. These high-level gestural descriptors are not always explicit. It is reasonable to assume that composers and performers use these mechanisms to organize their practice similarly to how dancers use *chunking* to memorize choreography (Starkes et al. 1990).

For this dissertation, we differentiate the high-level gestural descriptors from mappings. High-level gestural descriptors refer to the connections between raw sensor data and a representation of a physical gesture. Usually, they assume the role of intermediate mappings, created so users can grasp the physical movements associated with that parameter. Mappings will, by consequence, be connections to the parameters exposed at the destination: synthesis, manipulation, and control parameters.

For instrumental techniques based on gesture description, different mappings between pieces can take advantage of prior practice, as the instrumental techniques are based on movements previously learned. This organization based on gesture seems to be the case for the T-Stick examples discussed in Section 3.7.1. Stewart and Malloch organized the T-Stick techniques based on the gestures deemed affordable by the instrument's shape and sensors.

Instrumental techniques based on gesture description can be more complex for AMIs, as the new gesture-based and traditional instrumental techniques are organized differently. Regardless, the performers organized the new set of traditional and extended instrumental techniques without any issues, as presented in Chapters 5 and 6.

3.7.1 Instrument affordances: From gesture...

The T-Stick is an interesting example of an instrument with an established playing technique created by performers and composers using high-level gestural references from different musical works. T-Stick composer D. Andrew Stewart notably explored the instrument affordances, envisioning meaningful gestures according to the instrument shape and data acquired by the sensors (Stewart and Malloch 2010). Stewart classified gestures according to the nature of the movement, as *malleable* or *intractable*. Repeatable gestures (malleable) are easily repeatable and reproduce gestures used to trigger sounds, while fluid, subtle movements (intractable) are used for sound modification. The gestural vocabulary created with this approach is developed from the interaction with the physical object as the primary exploration method to access the DMI affordances. The vocabulary is mostly focused on the instrument's physicality and the movements required to perform the gestures. This focus can be perceived when comparing the notation in Figure 2.1 with the notation used by Stewart in *Catching Air and the Superman*[17] (D. Andrew Stewart) for chamber orchestra, 2 T-Sticks, and live electronics in Figure 3.8. Stewart's notation for the two T-Sticks in *Catching Air and the Superman* is entirely prescriptive, with gestures presented without any sonic implication.

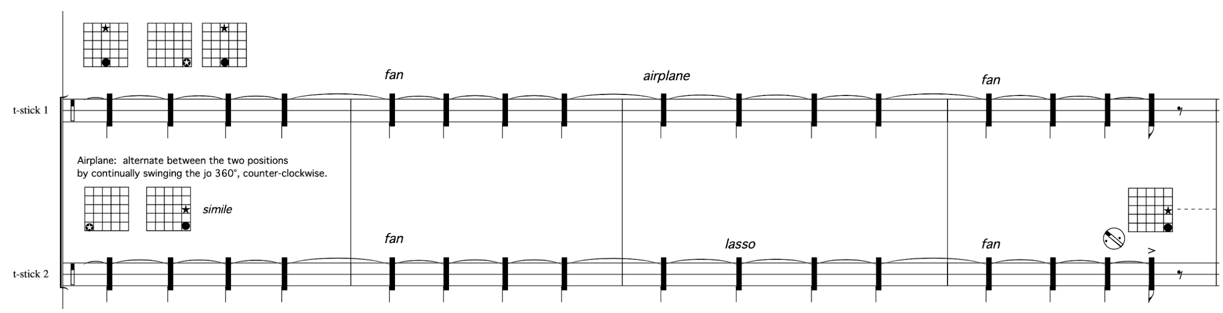


Figure 3.8 Example of technique notation for the T-Stick, extracted from *Catching Air and the Superman* (D. Andrew Stewart). The three-line staff shows fingering and *grip* using a T-Stick representation of the touch area divided into three sections. The custom tablature refers to a dynamically changing visual interface for controlling the synthesis processes. The text names—fan, lasso, airplane—refer to specific movements. The round symbols refer to the T-stick orientation (Stewart 2010).

There is no sound-related information on the T-Stick notation used by Stewart:

- the custom tablature controls a dynamically-changing interface for controlling the synthesis processes;
- *fan*, *lasso*, and *airplane* refer to specific gestures;

[17] An excerpt of this piece can be seen at `https://vimeo.com/77112292`.

- the T-stick orientation symbols invoke postures; and
- even if the score presents some rhythmic notation, there is no certainty that the sonic outcome will follow the gesture rhythm.

3.7.2 ... to sound

Both sound (for acoustic/electric) and gesture-centric (for some DMIs/AMIs) technique organizations offer tools for performers/composers to develop instrument expertise. However, users often need to map between the established gestural vocabulary and synthesis parameter(s) in digital instruments. This extra step differentiates digital and acoustic/electric instruments, even if the former already has established instrumental techniques. It is only possible to use sonic feedback to practice performance with DMIs/AMIs after the mapping process. Consequently, exploring new gestural possibilities and expanding the gesture vocabulary in digital instruments creates a feedback process where new gestural descriptors are created using sensor data, incorporated into the gestural vocabulary, and eventually made available for new mappings.

3.8 GuitarAMI gestural vocabulary

The GuitarAMI is an example of an AMI that is in the process of establishing a standard gestural vocabulary transferable between different performers and pieces. GuitarAMI performances were created exploring the desired sonic outcome, based on digital sound modifications designed to modify the classical guitar's sustain and audio feedback. Gestures used to control the sound modification parameters were chosen according to the expected sonic outcome (Meneses 2016) and became part of the gestural vocabulary later explored in other contexts, as described in Chapter 5. Due to the nature of augmented instruments and the GuitarAMI design choices, the AMI's gesture vocabulary is primarily used on controlling sound modification algorithms. These gestures correlate with the resulting sounds, similar to the relationship between the tremolo technique for classical guitars and the inseparable sonic outcome expected by the physical act.

Using algorithms from former GuitarAMI performances and the established high-level gestural descriptors for the T-Stick, we embedded custom high-level gestural descriptor algorithms into the GuitarAMI version 5a firmware. While GuitarAMI's high-level gestural descriptors using the IMU share similar algorithms with the T-Stick, the ultrasonic-based gestures were created from the gestural interactions explored in older GuitarAMI versions. Finally, we added the capacitive touch on the GuitarAMI version 5 with the primary objective of enabling access to the AMI configuration mode. However, we later created gestural descriptors using the capacitive sensor, allowing new mapping possibilities for the GuitarAMI. The list of gestural vocabulary embedded in the GuitarAMI firmware versions 5 and 5a can be seen in Table 3.2.

Table 3.2 List of the gestural vocabulary embedded in the GuitarAMI firmware versions 5 and 5a, separated per sensor.

Gesture	Unit	Description
Ultrasonic Sensor		
Distance	mm	Distance between module and hand/body
Wave Trigger	[0 or 1]	Waving the hand in front of the module
Capacitive Touch		
Tap	[0 or 1]	Capacitive sensor single touch
Double Tap	[0 or 1]	Capacitive sensor double touch
Triple Tap	[0 or 1]	Capacitive sensor triple touch
Count	n/a	count number of touches
IMU		
Tilt (Pitch)	[0 – 1]	Calculated Euler angle (normalized)
Roll	[0 – 1]	Calculated Euler angle (normalized)
Shake (X, Y, Z)	n/a	Amount of “energy” when shaking the GuitarAMI
Jab (X, Y, Z)	n/a	Amount of “energy” when jabbing the GuitarAMI
Footswitch		
Button 1	[0 or 1]	Footswitch 1
Button 2	[0 or 1]	Footswitch 2
Button 3	[0 or 1]	Footswitch 3

Chapter 4

T-Stick Music Creation Project

4.1 Project Overview

The T-Stick Music Creation Project (TMCP) was a research-creation project organized by CIRMMT, IDMIL, and the Digital Composition Studios (DCS), with the support of the Nomura Foundation. I led the technical and research aspects of the project, and Takuto Fukuda, a Doctoral Student in Composition (D.Mus) at the DCS (McGill University) and the second project leader, was responsible for the artistic (composition/performance) aspects. We also had the mentorship of Joseph Malloch (Dalhousie University), the T-Stick designer, and D. Andrew Stewart (University of Lethbridge), considered the most prominent T-Stick performer and composer. We describe the project's main objectives regarding the research-creation aspect in Section 4.3, including the possibility of embedding algorithms that provide high-level gestural descriptors on the T-Stick.

Embedding high-level gestural descriptors on the T-Stick is significant. More direct access to the instrument's gestural vocabulary may reduce the extra time spent in programming related to composing and rehearsing, as discussed in Sections 4.6 to 4.8. In the context of this dissertation, the TMCP serves as a case study to explore the impact of high-level gestural descriptors in DMIs design. We were also able to qualitatively assess the impact of different levels of accessibility for sensor data and established instrumental techniques for organizing creative thought. In other words, we evaluated how access to information and the lack of a formal pedagogical system impact a DMI learning process.

Exploration and expansion of the T-Stick gestural vocabulary were issues of critical importance. The hypothesis was that users rely on standard instrumental techniques to learn and teach instruments. Conversely, we also expect that the exploration process promotes new techniques for the instrument, leading to an expansion of the already existing high-level gestural descriptors for the T-Stick.

The TMCP used the new generation of wireless Sopranino T-Sticks (Section 3.3.2). It was organized in two workshops, a work period for composers to develop their musical works, and two concerts, where the T-Stick repertoire was premiered.

4.2 Justification

The TMCP tackles instrument longevity (see Section 2.4) and how performers employ raw sensor data to create high-level gestural descriptors that will become part of the instrument's gestural vocabulary (see Section 3.7).

As discussed in Chapter 2, DMIs have been increasingly used in new forms of musical expression since the last quarter of the 20th century (Miranda and Wanderley 2006). Along with all musical possibilities enabled by electroacoustics, DMIs have continuously changed the performer's role over the last century.

Similar to the *problem of the second performance*, where compositions are rarely performed after their premiere; most DMIs suffer from the *problem of the second performer*, where instruments are rarely performed by performers other than the DMI designer (McPherson and Kim 2012).

The problem of the second performer is directly related to the longevity discussion (Section 2.4), more specifically the TMCP, communities of practice, and repertoire. The artistic outcomes of the TMCP, namely music compositions and new composers/performers using a particular controller, simultaneously address instrument longevity and have the potential to give us insights on suitable actions to extend it further.

As discussed in Section 3.2, the T-Stick has the characteristics required to fulfill the research and artistic objectives of the TMCP (Section 4.3). As a gestural controller used in new media exploration for more than ten years, the T-Stick already has a community of practice and repertoire. Simultaneously, the controller's gestural vocabulary is still in development, and the expansion of the repertoire for the T-Stick may expand the communities of practice.

For the scope of this dissertation, the key elements suitable for research exploration using the T-Stick in the TMCP are musical notation and instrumental technique. The hypothesis is that these are the elements used to create high-level gestural descriptors and transfer techniques between performers and between pieces.

4.3 Objectives of the T-Stick Music Creation Project

The objectives of the TMCP fall into two categories:

1) research, involving an investigation of how composers and performers interact with new instruments and expand established instrumental techniques during artistic practice; and

2) creation, involving establishing and expanding a community around the T-Stick, and expanding the instrument's repertoire.

We presented and discussed the background of the first objective in Sections 3.6 and 3.7, while the discussion and background for the second objective can be found in Section 2.4.

4.4 Timeline and Activities

As discussed in Chapter 3, the methodology applied to the TMCP was organized using concepts and methods from practice-based research (PBR) and design science research (DSR).

The TMCP activities were structured as shown in Table 4.1.

During the Winter of 2019, the project leaders organized the overall schedule, determined the data collection methods for their analysis, and submitted the project application to CIRMMT. As a collaborative research-creation project, it was essential to establish the artistic outcomes from the beginning. Framing the artistic goals allowed the artistic collaborators to plan their actions—composing and practicing for the performance—and the research team to develop an observation and data collection strategy. Given the chosen PBR methodology (Cantrell 2017; Gurevich 2016), the schedule was defined to give the collaborating composers enough time to explore the T-Stick and practice the performance. Simultaneously, the collaborating composers needed full access to instruments and, ideally, to have their own instruments.

Even though the T-Stick had good documentation that supported the replication process, building DMIs in an academic laboratory setting is a demanding task. As discussed in Chapter 2, some of the challenges in replicating academic DMIs/AMIs include maintaining replicability documentation or redesigning to replace discontinued/obsolete components. To

Table 4.1 T-Stick Music Creation Project (TMCP) activity timeline.

2019		
	Winter	Official project submissions to CIRMMT for Student Awards, improv@CIRMMT, workshops, and live@CIRMMT
	Spring/Summer	Preparing T-Stick firmware and supporting Max and Pure Data patches for the project
		Call for collaborating composers and performers
	Fall	First T-Stick Workshop with mentors D. Andrew Stewart (performance and composition) and Joseph Malloch (design and mapping)
		T-Stick composition period and gestural exploration
2020		
	Winter	Second T-Stick Workshop, for the collaborating composers. Mentor D.Andrew Stewart (performance/composition)
		improv@CIRMMT concert: The 2019 T-Stick Music Creation Project
		live@CIRMMT DMI Concert: CIRMMT Composers
	Spring/Summer	Embedding expanded gestural vocabulary (high-level gestural descriptors) on the T-Sticks
		Publishing results

ensure instrument availability for all collaborators involved—project leaders, mentors, and composers—we limited the number of collaborating composers to five.

In preparation for the first workshop on the T-Stick, we programmed Max and Pure Data support patches. These patches made available the raw sensor data from the T-Stick and D. Andrew Stewart's algorithms (used in his compositions), containing the instrument's established gestural vocabulary. These resources allowed project collaborators to access both gestural exploration paradigms—raw data and existing high-level gestural descriptors—during the project.

CIRMMT also released a public call for five composers interested in collaborating on the TMCP[1] (shown in Appendix A). The call aimed to promote new live electroacoustic solo works for the T-Stick and accepted other musical formations and formats, including installations and interdisciplinary artistic works, as long as they used the T-Stick. The collaborating composers were selected based on feasibility, originality, and potential to explore and expand the T-Stick gestural vocabulary. The selection jury was composed of the TMCP project leaders and mentors.

The first significant milestone of the project was the *Workshop on the T-Stick*, held on November 16, 2019 (Figure 4.1), and conducted by the TMCP's leaders and mentors. This workshop was open to the public. The topics presented include an introduction to the T-Stick history, motivation, design details, sensors, established techniques, compositional approaches, instrument notation, and projects related to the instrument. The workshop participants could also interact with the instrument during a hands-on session for the T-Stick, including setup, access to the raw sensor data, and processed signals related to the high-level gestures in the existing vocabulary. The five selected composers actively participated in the first workshop, where they were introduced to the current modes of performance on the T-Stick. The composers explored the DMI playing techniques and sounds commonly associated with the instrument in previous compositions. At the end of the event, collaborating composers could schedule one-on-one meetings with Malloch and Stewart, the project mentors, to discuss the next steps and compositional ideas.

During the following two months, the T-Stick composers could work on their pieces, supported by IDMIL for solving technical difficulties and algorithm implementation, and D. Andrew Stewart for composition and instrumental techniques. Throughout this period, the

[1] The *Workshop on the T-Stick* announcement and the call for composers can be found at `https://www.cirmmt.org/activities/workshops/research/stick_workshop/workshop_tstick` and in Appendix A.

Figure 4.1 First workshop on the T-Stick, organized by CIRMMT, IDMIL, and the DCS, with the support of Nomura Foundation on November 16th, 2019 (`https://www.cirmmt.org/activities/workshops/research/stick_workshop/workshop_tstick`).

composers reported reliability problems using the T-Stick, which led to a redesign in the building and maintenance processes, improving the robustness of the current instrument version. The redesigns are discussed in Section 4.8.

The compositional/exploration period and the feedback given at the end of the project were the most prolific phases regarding data/information collection for the research portion of the project. The collaborators received their instruments, which remained in their possession during the entire project duration. The composers were also given the following two patches:

1) Stewart's Soprano T-Stick Max patch (Figure 4.2), containing the instrument's established gestural vocabulary developed by Stewart and Malloch over the years; and

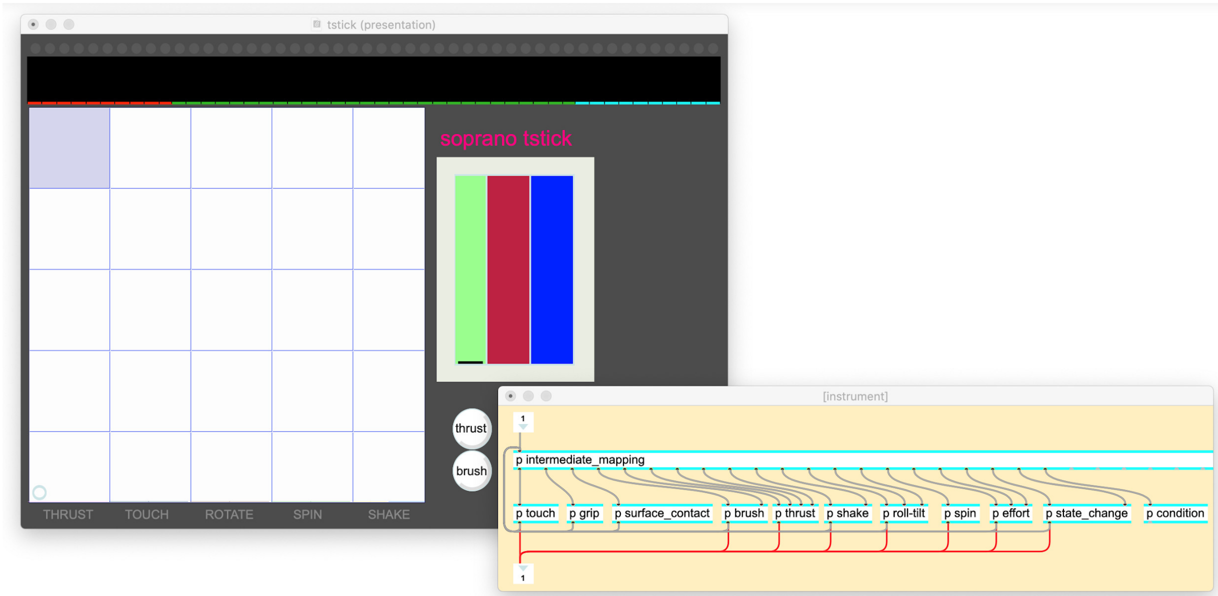


Figure 4.2 D. Andrew Stewart's Soprano T-Stick Max patch. The patch was presented at the first workshop on the T-Stick and made available for the collaborating composers, giving them access to the instrument's established gestural vocabulary.

2) a Pure Data patch programmed at IDMIL as a tool to test T-Stick during the building or repair process (Figure 4.3).

The second T-Stick workshop took place at CIRMMT on February 9, 2020, and it was primarily oriented to the T-Stick composers working on the Music Creation project. During the event, the composers could present their work in progress and receive feedback from Stewart in preparation for the concerts premiering the new T-Stick works. The participants presented each piece's algorithms, notation, and gestures and prepared an improvisation session for the improv@CIRMMT concert (see Figure 4.4).

Finally, the TMCP ended with two concerts. The first concert was part of the improv@CIRMMT series and was entirely dedicated to T-Stick works, featuring collaborating composers—Erich Barganier, Diego Bermudez Chamberland and Yanik Tremblay-Simard,

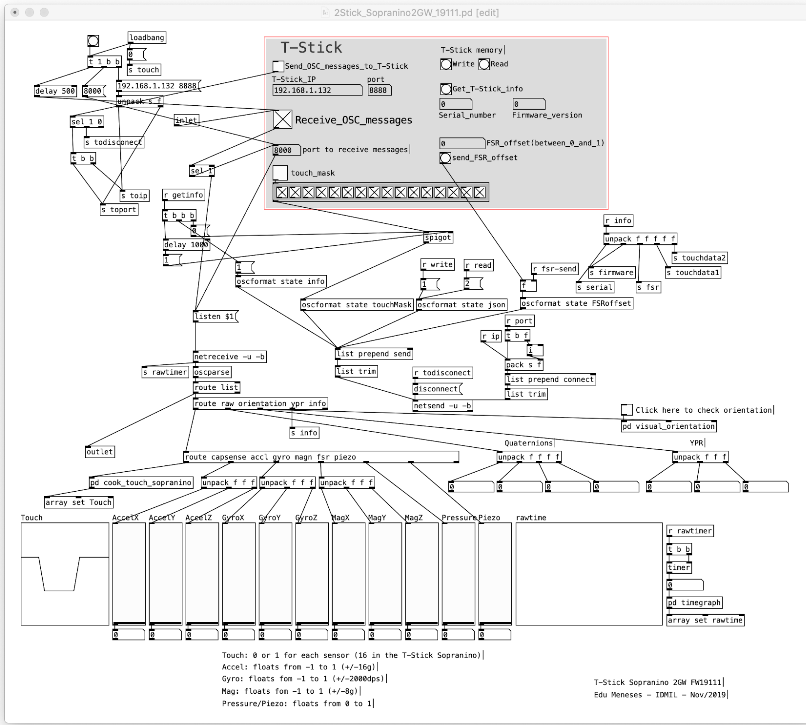


Figure 4.3 T-Stick Sopranino test patch exposing raw sensor data. The patch was programmed in Pure Data at IDMIL to test instruments during the building phase, and was available to the collaborating composers during the TMCP.

Vincent Cusson, Kasey Pocius, and Michał Seta. The concert also included the participation of Vânia Eger Pontes, D. Andrew Stewart, and the Concordia Laptop Orchestra (CLOrk). Four musical works and one installation were premiered during the *improv@CIRMMT: The 2019 T-Stick Music Creation Project* (Figure 4.5) on February 11, 2020, at La Vitrola in Montreal, Canada.[2] An in-depth discussion on the T-Stick usage of the artistic works can be seen in Section 4.6.

The second performance related to the project took place at the *live@CIRMMT:*

[2]More information about the *improv@CIRMMT: The 2019 T-Stick Music Creation Project* event can be found at the official event page: `https://www.CIRMMT.org/activities/special/improv_feb2020_tstick`.

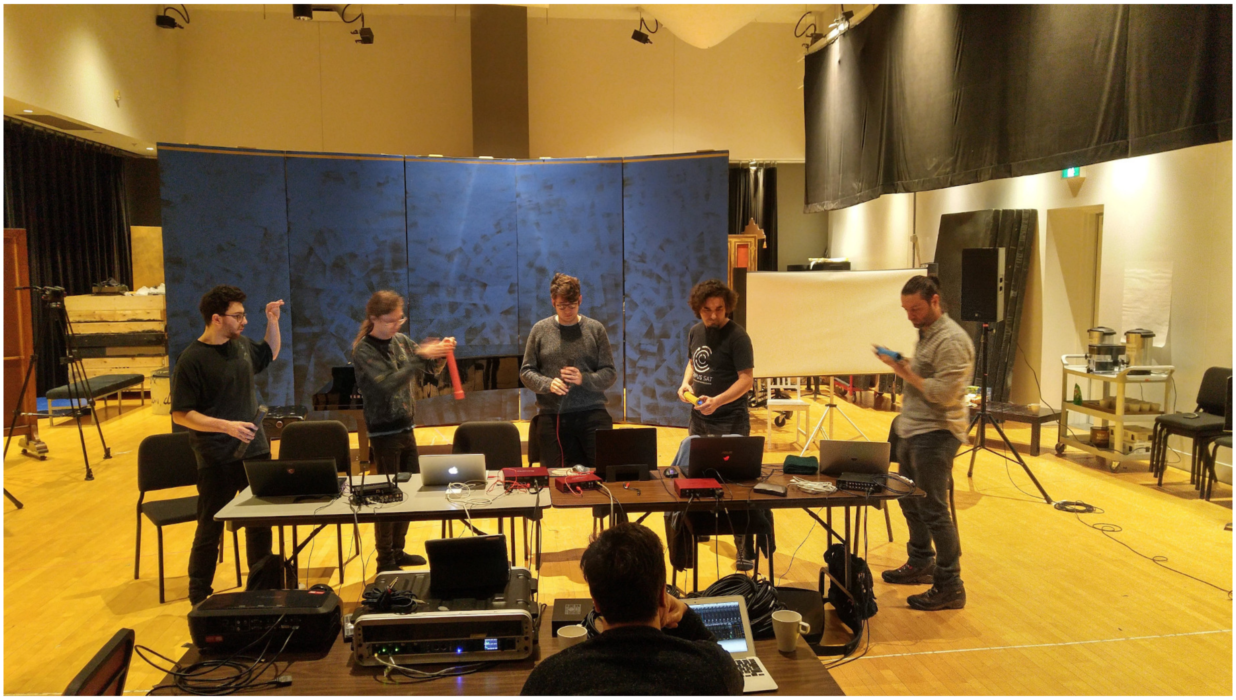


Figure 4.4 Free improvisation during the second T-Stick workshop on February 9, 2020. The collaborating composers and D. Andrew Stewart prepared this short improvisation session for the improv@CIRMMT concert (picture: Yuval Adler).

CIRMMT Composers event (Figure 4.6). This event presented a joint concert with several CIRMMT members and included two T-Stick performances: *Dweller within* for soprano T-Stick—a solo composition performed and composed by D. Andrew Stewart; and *Higher order gestalt fromage* for two T-Sticks and a three-dimensional loudspeaker array—composed and performed by *blablaTrains duo* (Ana Dall'Ara-Majek and Takuto Fukuda). This event took place February 13, 2020, at the Music Multimedia Room (MMR), located within CIRMMT facilities.[3]

A description of the workshop and artistic works was presented at the International

[3]More information about the *live@CIRMMT: CIRMMT Composers* event can be found at https://www.cirmmt.org/activities/live-cirmmt/cirmmt_composers_february/.

Figure 4.5 improv@CIRMMT: The 2019 T-Stick Music Creation Project, on February 11th, 2020. The event was part of a series of concerts devoted to electro/acoustic performances with musicians from the CIRMMT community.

Conference on New Interfaces for Musical Expression (NIME) (Fukuda et al. 2021).

4.5 Impact of high-level gesture descriptors on compositions and performers

As discussed in Section 3.7.1, before the TMCP, the T-Stick already possessed an established gestural vocabulary created by D. Andrew Stewart. This gestural organization allows the use of a DMI similar to a traditional musical instrument, where interacting with the physical

(a) Higher Order Gestalt Fromage, for two Sopranino T-Sticks and 24-ch loudspeaker array, 2020 (blablaTrains: Ana Dall'Ara-Majek and Takuto Fukuda).

(b) Dweller within, originally for Soprano T-Stick, 2012 (D. Andrew Stewart).

Figure 4.6 live@CIRMMT: CIRMMT Composers event, on February 13th, 2020. The live@CIRMMT Performance Series is a concert series organized annually and open to the general public. It includes four to six performances per year that highlight the artistic research output of CIRMMT.

object is the primary exploration method to access the DMI affordances (Meneses, Fukuda, and Wanderley 2020). However, the feedback provided by the composers working with T-Sticks during the TMCP allowed us to identify two distinct methods of instrument exploration:

1) directly mapping raw sensor data into arbitrary sound parameters, and
2) processing and organizing raw gestural data into high-level gestural descriptors along with the already established gestural vocabulary and subsequently mapped into sound parameters.

Composers who used the first method expressed the desire to create an instrument with novel behaviour that could be physically explored. In contrast, composers aiming for the second exploration method expressed the desire for repeatability so that a particular piece could be learned by different performers, similar to how performers can master traditional musical instrument compositions.

In contrast, Stewart presented an example of the second method during the initial workshop, highlighting the use of a gestural taxonomy that is coherent across different performances. One argument for this approach is that it creates instrumental techniques that are relatable across different works and, consequently, recognizable by the audience.

Creating a high-level gestural vocabulary for a DMI can vary from simply rescaling the data to employing complex sensor fusion algorithms. This project was driven by instrument exploration during composition and performance—after instrument design and construction. The main issues in this process are that DMI/AMI users (e.g., composers and performers who are not instrument designers) do not necessarily understand technical sensor specifications. Consequently, they do not consider factors such as drift, precision, or resolution (Medeiros 2015) when creating their gestural descriptors. A possible solution could be for the instrument designer to reprogram and embed high-level gesture descriptors created by composers and performers in DMIs/AMIs as a possible solution. This process can lead to potentially more computationally efficient solutions by considering composers' and performers' previously neglected technical demands.

Interestingly, the collaborating composers that used the established vocabulary were the ones using Max in their compositions. In contrast, the remaining composers using other music programming languages—Pure Data and SuperCollider—mostly explored raw sensor data.

Even though the composers still had access to the T-Stick Max patch containing the gestural vocabulary created by Andrew Stewart, porting these algorithms to other programming languages was beyond the scope of the composition/performance activity. In other words, porting the T-Stick algorithms from Max to the language of choice would have spent precious time otherwise used for composing and creating the mappings. Moreover, the composers considered the type of sensor data they could access—raw or processed—as the instrument's available vocabulary for their work, even though they became familiar with the already established T-Stick techniques during the TMCP workshops.

When considering the composers' feedback, it is apparent that the instrumental techniques presented at the first workshop inspired some of the mappings and abstractions created by composers exploring raw sensor data; nevertheless, the composition structure is fundamentally different depending on the data employed.

Section 4.6 presents the music pieces and exploration methods used by the composers.

4.6 Repertoire produced during the TMCP

The artistic outcome of the TMCP consisted of five new artistic works: four musical compositions from the collaborating T-Stick composers and one new media installation from the fifth collaborator. One extra music composition was created during the period by the *blablaTrains* (Ana Dall'Ara-Majek and Takuto Fukuda).

4.6.1 The Taxidermy of Negative Space

The Taxidermy of Negative Space was composed for T-Stick, video, and dancer by Erich Barganier. Barganier invited Vânia Eger Pontes to perform the improvisatory dance with the T-Stick. The performance consisted of a dancer using the T-Stick to control sound and video synthesis, programmed in Pure Data. The composer made use of granular synthesis applied to pre-recorded children's voices. There was also an exploration of the T-Stick's theatrical aspect in conjunction with the physicality of the dance. A video recording of the presentation can be found at `https://youtu.be/RFVvmboxgcg`.

Figure 4.7 The Taxidermy of Negative Space, composed by Erich Barganier and performed by Barganier (electronics) and Vânia Eger Pontes (dance/T-Stick). Performed on February 11, 2020 (improv@CIRMMT).

4.6.2 Mimoidalaube

Mimoidalaube was composed for T-Stick and audiovisual projection by Michał Seta, programmed in SuperCollider and the Godot game engine. This piece was designed as a gamified composition where the performer controls the T-Stick as a game joystick to navigate an avatar in a game world. The composer/performer's gestures invoke a different audiovisual reaction from the game mechanics. Seta uses the *roll* gesture: a motion of the T-Stick rolling along the floor. The roll gesture results from an exploration of a physical affordance of the T-Stick: as a cylindrical instrument with wireless connectivity, the DMI performer can roll the device. The new gesture is an example of interaction created for a particular

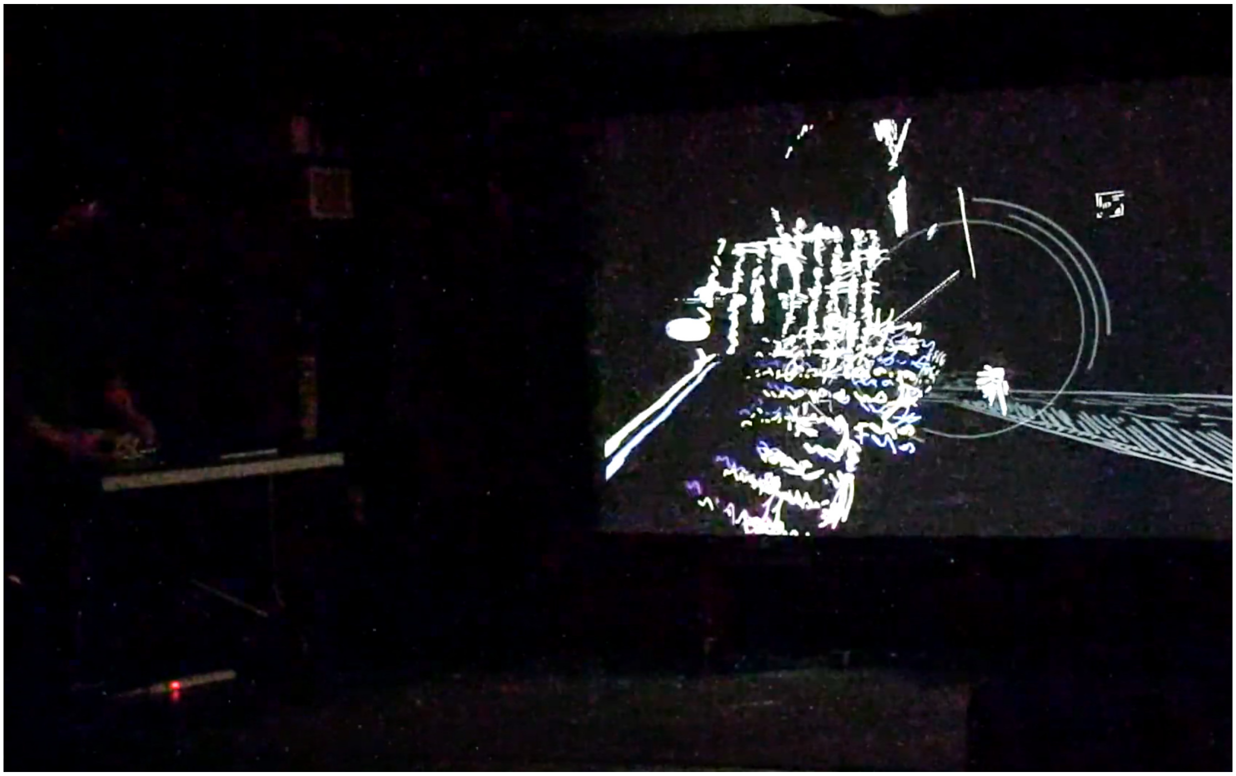


Figure 4.8 Memoidalaube, composed and performed by Michał Seta, for T-Stick. Performed on February 11, 2020 (improv@CIRMMT).

piece that can expand the T-Stick's idiomatic gesture vocabulary. A video recording of the presentation can be found at `https://youtu.be/ndWjdQLATDg`.

4.6.3 Balance

Balance was a sound installation created by Vincent Cusson. The visitor finds the T-Stick attached to a stand that allows only one degree of freedom: a single-dimensional rotation. The system uses a stepper motor to resist movement and force the system to return to the rest position. Cusson stated that the T-Stick shape and lack of wires invite the audience to rotate the instrument when interacting with the interface. The artist mapped the sensor data directly to multiple voices and a sequence of eight musical notes. A video recording of

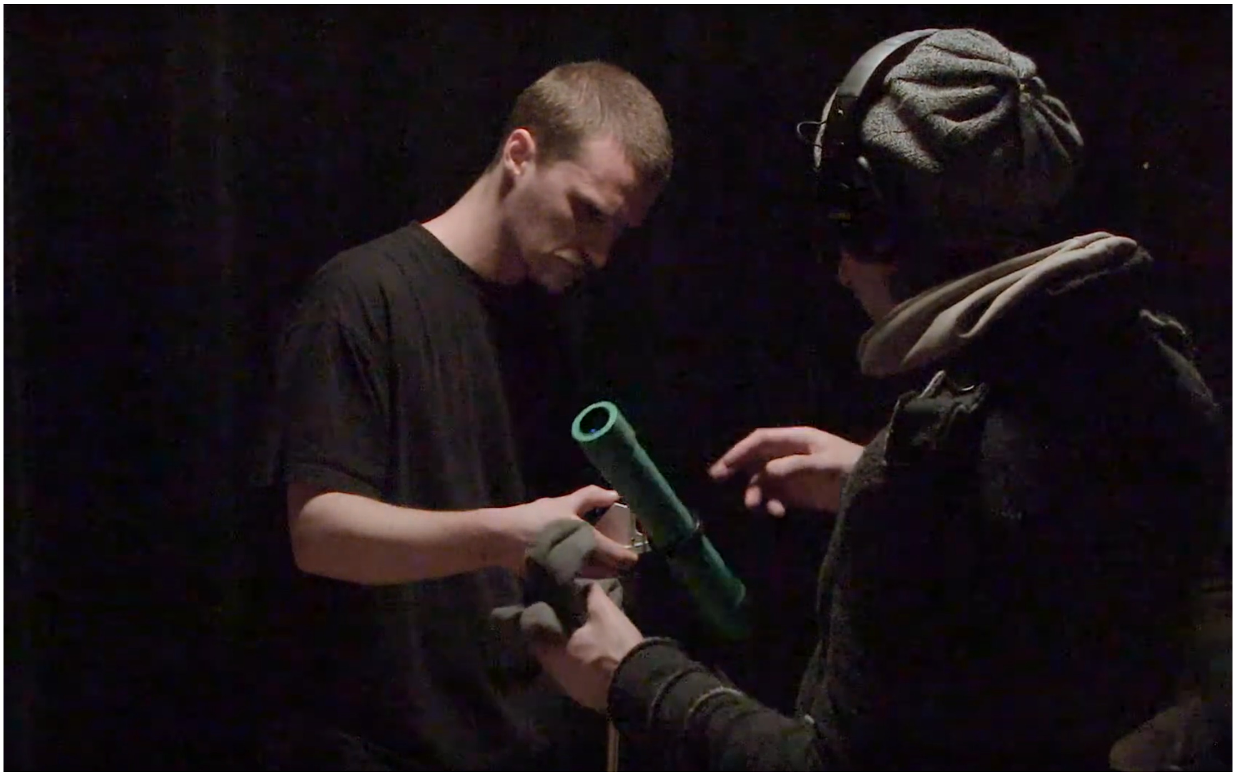


Figure 4.9 Balance, a sound installation created by Vincent Cusson, for T-Stick. Performed on February 11, 2020 (improv@CIRMMT).

visitors interacting with the system can be found at `https://youtu.be/B_4XV3seIXg`.

4.6.4 Reflexion

Reflexion was composed for T-Stick and Theremin by the Macroplasm Duo (Diego Bermudez Chamberland and Yanik Tremblay-Simard). Chamberland and Tremblay-Simard developed new idiomatic gestures, including the *tiny picking* gesture, which explores the T-Stick's capacitive sensor sensitivity with a fine finger motion. The *smash the side* gesture is similar to the *jab* gesture already used by Stewart, but the movement involves a strong tap on the instrument. The *rolling up and down* gesture resembles Seta's rolling gesture, but it is performed from hand to hand. The gestures created by the Macroplasm Duo were used

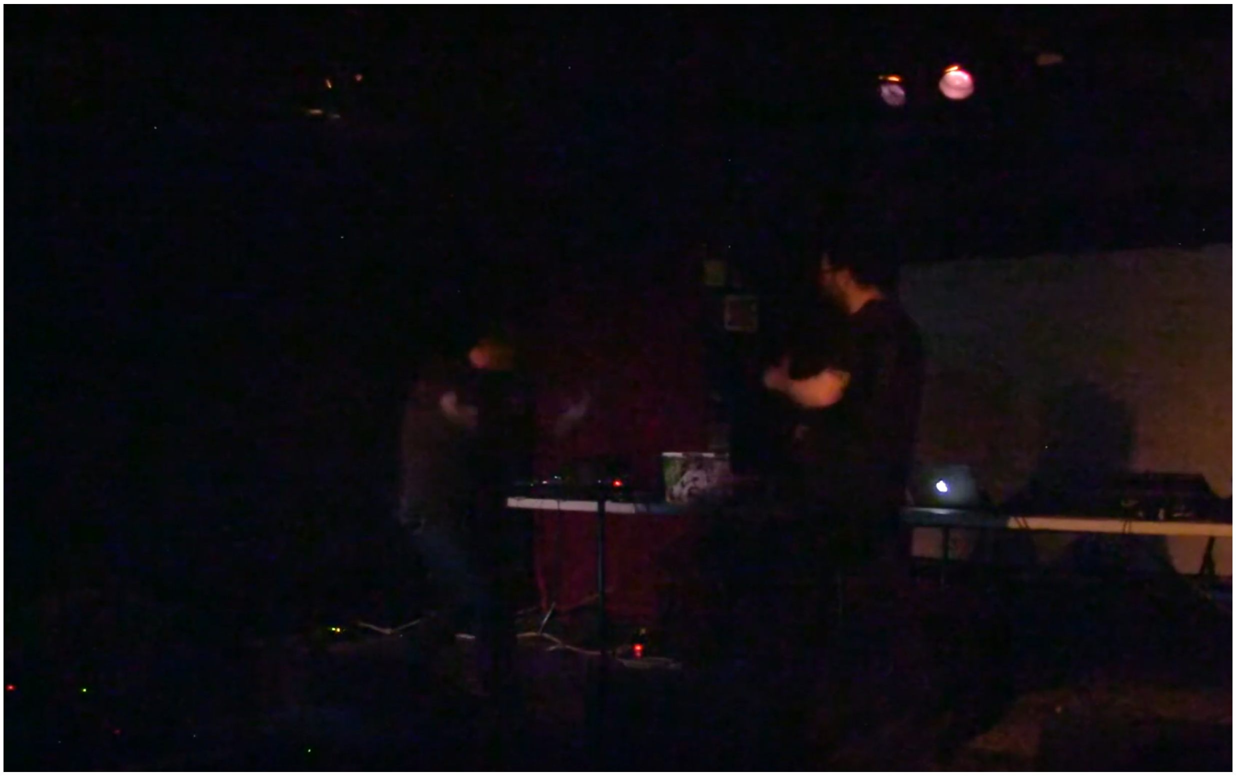


Figure 4.10 Reflexion, composed and performed by the Macroplasm Duo (Diego Bermudez Chamberland and Yanik Tremblay-Simard), for T-Stick and theremin. Performed on February 11, 2020 (improv@CIRMMT).

to modulate the audio stream from the Theremin and control the synthesis unit directly mapped from the T-Stick.

4.6.5 Synthetic Icescapes

Synthetic Icescapes was composed for T-Stick and five laptops by Kasey Pocius. The piece was performed by Pocius, D. Andrew Stewart on a laptop, and the CLOrk: Danielle Savage, Arturo Hidalgo, Gabriel Gustafsson, and Liam Mansfield. Pocius extensively explored the established T-Stick vocabulary inherited from Stewart and took full advantage of existing T-Stick-related Max patches. Some of the gestures include *windmills*, *windows intervene*,

Figure 4.11 Synthetic Icescapes, composed by Kasey Pocius for T-Stick and laptop orchestra. Performed by Pocius, D. Andrew Stewart, and the CLOrk on February 11, 2020 (improv@CIRMMT).

jab, and *framing* gestures. This piece explored complex mutable mappings between a T-Stick and CLOrk. While the T-Stick controlled specific synthesis processes, it also shared gestural data with other orchestra members, who could use webmapper to map the T-Stick data to any other sound synthesis they executed on their laptops in real time. CLOrk members were free to map the data in any way they felt fit using libmapper and webmapper during the performance.[4] A video recording of the presentation can be found at `https://youtu.be/Jn8iVg3SdPk`.

[4]Libmapper is a software/library to share data signals and create mappings through networks. webmapper is a browser-based interface for libmapper. More information at `https://libmapper.github.io/`.

4.6.6 Higher Order Gestalt Fromage

Figure 4.12 Higher Order Gestalt Fromage, composed and performed by blablaTrains (Ana Dall'Ara-Majek and Takuto Fukuda) for two Sopranino T-Sticks. Performed on February 13, 2020 (live@CIRMMT).

Higher Order Gestalt Fromage was composed for two Sopranino T-Sticks and 24 channel loudspeaker array by blablaTrains (Ana Dall'Ara-Majek and Takuto Fukuda). The piece used audio samples (created using Reaper[5]) mapped through Max and independent granular synthesizers. Dall'Ara-Majek and Fukuda exclusively explored established T-Stick gestures, e.g., *jab*, *squeeze*, *airplane*, and *tilt* (orientation). The composers used predefined mapping presets triggered and changed during the performance, exploring the mutable relationship between gesture and sound. Unlike the other TMCP works, this piece premiered at the live@CIRMMT: CIRMMT Composers event, where two pieces using the T-Stick were

[5]`https://www.reaper.fm/` accessed on February 4, 2020.

performed: *Higher Order Gestalt Fromage* and *Dweller Within*, by D. Andrew Stewart (Figure 4.6). A video recording of the presentation can be found at `https://youtu.be/e10h27TBzRk`.

4.7 Feedback from composers and project observations

The collaborating composers, designers, and mentors provided constant feedback on the TMCP, including project organization, performance aspects, technical difficulties using the T-Stick, and updated suggestions. Technical aspects and updates can be seen in Section 4.8.

All TMCP composers stated the importance of spending time with the DMI and the advantage of having dedicated T-Sticks during the project. However, contrary to expectations, the collaborators reported that they spent most of the time programming their composition algorithms rather than practicing and purely exploring the T-Stick. We asked the TMCP composers to estimate the percentage of time they spent on each task.

1) Mostly programming/composing: the primary task was to compose or write code related to the composition, e.g., audio synthesis, mappings. Some time could be spent testing the code and exploring the instrument, but always in service of composition.
2) Mostly practicing/exploring: The primary task was practicing the performance or improvising with the instrument. Some time could be spent on mapping or fixing algorithms, but always in the service of performing.

Most composers answered that around 90% of the time was spent programming or composing. Some composers stated they had virtually no time to practice their performances before the second workshop, two days before the premiere.

Furthermore, the TMCP composers stressed the importance of working with their particular instrument rather than a generic T-Stick. After the first month in possession of the T-Sticks, the collaborators started to know their instrument enough to notice slight behaviour differences when using other T-Sticks. This phenomenon was revealed when some composers had hardware problems, and we provided spare instruments while fixing the defective T-Sticks. Users instantly noticed minor differences between the instruments, e.g., sensitivity response for the touch sensor or FSR, small-angle differences for orientation.

Another interesting observation was the influence of the T-Stick design aesthetics and build quality in motivating instrument usage and exploration. TMCP composers chose their T-Sticks among the available instruments during individual meetings after the first workshop. The primary choice was by the colour—the T-Sticks were available in "boring-black," "romantic-red," "bumblebee-yellow," "baby-blue," "fern-green," and "transparent." Also, some T-Sticks had minor aesthetic imperfections, e.g., slightly wrinkled heat-shrink tubing and misaligned tube endcaps. Users were provided with an opportunity to physically hold and inspect the T-Sticks, unconnected, before making a choice. Once the collaborator chose the instrument, they signed an agreement and could take the instrument out of IDMIL. Composers immediately discarded instruments that did not "feel right" or a colour that displeased them. During feedback, composers often stated that choosing the instrument among the available T-Sticks helped with intimacy, going as far as to consciously declare that the chosen colour helps motivate instrument exploration and asking to keep the same colour if a replacement instrument was needed.

Finally, we could observe a trend among collaborators to choose the exploration method according to the available data. TMCP composers predominantly explored raw sensor data when using the T-Stick's Pure Data patch or other music programming languages. Nevertheless, composers predominantly using the T-Stick's Max patch explored the established instrument vocabulary. Discussion on the exploration methods can be found in Section 4.5; however, the observed trend leads to the conclusion that data presentation strongly influences instrument exploration.

4.8 Design lessons and T-Stick improvements

In addition to the feedback reported in Section 4.7, the collaborators suggested improvements to be incorporated into new firmware versions of the T-Stick.

The T-Stick uses OSC to send data to other devices, allowing the instrument to interact with different software, including Ableton Live, Max, SuperCollider, and Pure Data. All gesture extraction and instrumental techniques/vocabulary were programmed in Max (as discussed in Section 4.5). However, the lack of algorithms in other music programming languages limited access to high-level gestural descriptors. This lack of access forced users to port the original sensor fusion algorithms if other software or systems were needed, e.g., Linux-based systems and devices without Max support. Moreover, some software commonly used in composition and performance use other protocols such as MIDI or requires specific ranges or normalized data.

Even though the T-Stick repository contains technical documentation covering building and basic configuration, there was little information on how to play the instrument or a guide for the established instrumental technique. Knowledge of performing the T-Stick has been transmitted orally among users or by accessing videos of past performances.

One last concern was the lack of feedback on battery status. Several hours can pass between soundcheck and concert, and users require feedback to control battery power better.

In the subsequent months after the TMCP, we made improvements to the T-Stick based on the composers' feedback. An integrated battery monitor was incorporated through OSC messages and LED blinking patterns. Also, the subsequent firmware releases include embedded gesture extraction algorithms for part of the T-Stick gestural vocabulary. The gestural signals extracted by the firmware are summarized in Table 4.2.

The high-level gesture descriptor algorithms were ported from Max to C++ on the ESP32, and all firmware versions after FW200422 output both raw sensor data and the embedded gestural vocabulary. An excerpt of the resulting code showing the high-level gestural description implementation can be seen in Appendix C.

Table 4.2 List of the gestural vocabulary embedded on the T-Stick firmware at the end of the TMCP, separated per sensor. The gesture acquisition algorithms were updated to provide idiomatic gestures and some of the techniques created during the research-creation project.

Gesture	Unit	Description
Capacitive touch		
Touch All	n/a	Amount of touch (area) through all T-Stick
Touch Top	n/a	Amount of touch (area) in the "top" region of the T-Stick
Touch Middle	n/a	Amount of touch (area) in the "middle" region of the T-Stick
Touch Bottom	n/a	Amount of touch (area) in the "bottom" region of the T-Stick
Brush	cm/s	Brush your hand along the surface of the instrument
Multi-Brush	cm/s	Similar to Brush, but outputs up to four velocities simultaneously
Rub	cm/s	Rub your hand on the surface of the instrument
Multi-Rub	cm/s	Similar to Rub, but outputs up to four velocities simultaneously
IMU		
Yaw, Pitch, Roll	degrees	Calculated Euler angles (from -180 to 180)
Shake (X, Y, Z)	n/a	Amount of "energy" when shaking the T-Stick
Jab (X, Y, Z)	n/a	Amount of "energy" when jabbing the T-Stick

During the TMCP, we had to fix some T-Sticks due to malfunction. The original DMI design involves splitting a PVC tube lengthwise, securing sensors and microcontrollers using 3D-printed beds glued to the tube and sealing with a heat shrink tube. Even though this solution was robust enough for performance, performing maintenance on these T-Sticks can be time-consuming.

An improved building process also contributed to the consistency of sensor placement inside the instrument, ensuring repeatable and coherent sensor data among different T-Sticks. The difference between old and new building processes is illustrated in Figure 4.13.

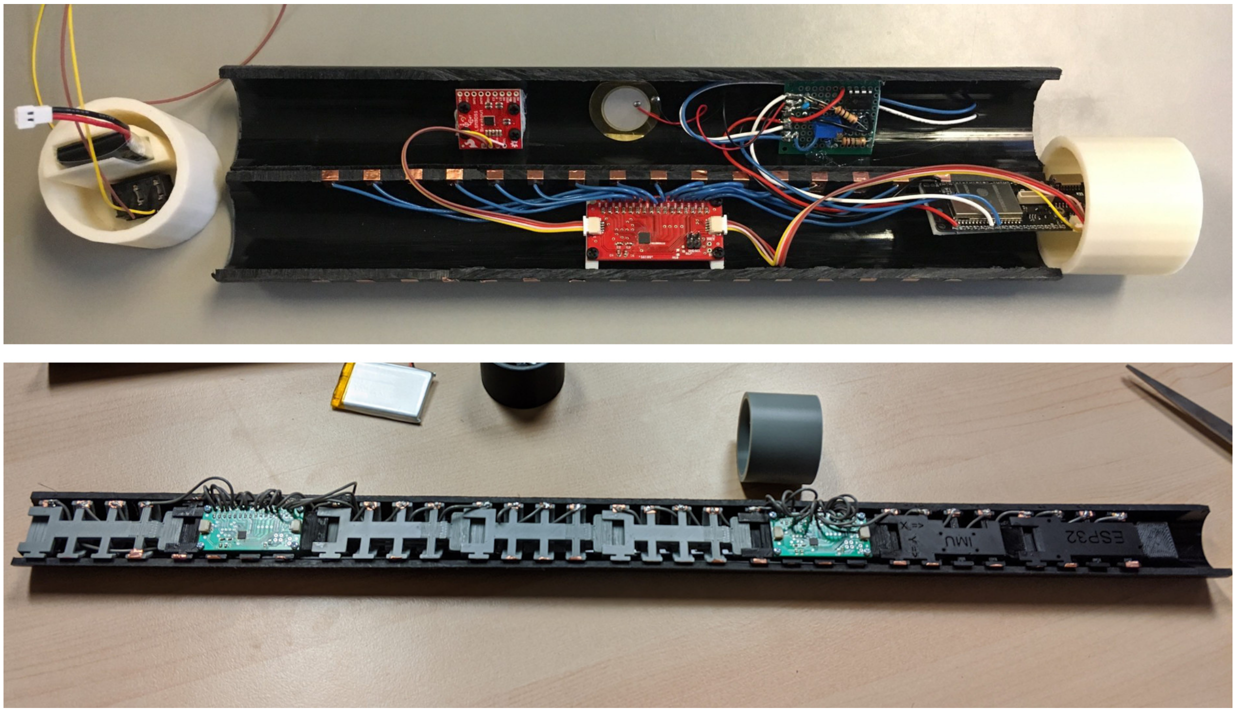


Figure 4.13 T-Stick sopranino and soprano using different building processes: the sopranino (top) uses 3D printed "beds" under sensors.[6] The soprano (bottom) uses a newly designed modular 3D-printed frame. It is unnecessary to cut the PVC tube lengthwise when using the modular 3D-printed frame (the cut tube was used in the image for illustrative purposes).

4.9 Discussion

The artistic outcomes of the TMCP addressed the second objective stated in Section 4.3 by expanding the T-Stick's repertoire and the community of composers and performers. As discussed in Section 4.6, the T-Stick repertoire was substantially expanded with five new musical compositions and one art installation. After the TMCP, four composers kept working with the instrument. Kasey Pocius, Michał Seta, and blablaTrains regularly perform

[6] More information on the former T-Stick building process can be found at `https://github.com/IDMIL/TStick/blob/master/Sopranino/2GW/FW19101/Docs/T-Stick_2GW_building_instructions.md`.

with the Sopranino T-Stick, while Mimoidalaube was presented at Linux Audio Conference 2020.[7] Seta also currently works in a DMI Trio project with Stewart (Karlax) and Dirk Stromberg (Phallophone).[8]

The TMCP allowed us to observe how the accessibility of the established instrumental techniques impacts the exploration of DMIs by composers and performers. Previous knowledge of existing instrumental techniques presented to the composers during the first workshop had less impact on instrument exploration than the type of data available (raw sensor data or high-level descriptors). Composers working directly with the T-Stick Max patch had the instrument vocabulary available without any need for porting or configuration. These composers took advantage of the established vocabulary, which offered more mapping options and the advantage of the inherent complexity present in the algorithms used for sensor fusion.

Composers using the raw sensor data of the T-Stick followed two different approaches: 1) mapping raw data directly to synthesis parameters, creating complexity by increasing the number of direct mappings or simpler divergent mappings; or 2) creating their high-level descriptors in an intermediate mapping layer. Both approaches led to more direct mappings; in most cases, the intermediate mapping layer did not involve sensor fusion or data manipulation. Instead, the intermediate layer served more as a sensor data tagging role, e.g., the *roll* gesture was achieved by directly reading one of the accelerometer axes, and *tiny picking* was implemented directly from the touch (capacitive) sensor. It is important to state that complex mappings are not a requirement for good DMI composition or performance. Nevertheless, data accessibility and, more specifically, high-level gestural descriptors directly

[7] The Mimoidalaube performance at the Linux Audio Conference 2020 can be seen at `https://tube.aquilenet.fr/videos/watch/2e4e4115-bc09-49c6-b7fb-cdd76dd59cec?playlistPosition=13`.

[8] A video performance with Seta, Stewart, and Stromberg entitled *Alt F in Front of the Body - Rehearsal* can be seen at `https://vimeo.com/538235209`.

influence instrument exploration.

Another interesting observation from the interaction between composers and mentors was transferable knowledge. During the second workshop, the dynamics between the mentor and the collaborating composers were fundamentally different between composers using raw data and composers using the established T-Stick techniques. Composers in the latter group could communicate better with the mentors on their intentions. Stewart could "access" the interaction between gesture and synthesis by observing the composer/performer performing an established gesture. According to the synthesis processes employed, the interaction flowed similarly to a traditional music workshop, where the mentor could connect knowledge from gesture and sound expectations. Composers using raw data often needed to provide the mentor with further explanations for mapping sensor data and the expected sonic result. Discussions and support, in that case, were dealt with at a lower level, often focusing on technical aspects of data handling.

Based on our findings from the TMCP, we were able to define a coherent set of high-level gestural descriptors that could be embedded into the T-Stick's firmware. These gestural descriptors were embedded starting with firmware version FW200422, shown in Table 4.2. Current T-Sticks updated with the latest firmware and output data relative to the established instrumental technique, providing the T-Stick instrument data to users in any system capable of receiving OSC. This embedding process also standardizes instrument behaviour and gesture extraction between performers and music programming languages.

Chapter 5

GuitarAMI Research-Creation Project

5.1 Project Overview

The GuitarAMI Research-Creation Project (GRCP) was a project organized by IDMIL and CIRMMT, under my leadership in collaboration with the Cicchillitti-Cowan Duo (Adam Cicchillitti and Steve Cowan). Adam Cicchillitti is a Montreal-born classical guitar performer, composer, teacher, and arranger. Cicchillitti has won awards in several national and international competitions and is considered "one of the most promising guitarists of his generation" (*Adam Cicchillitti official website* n.d.). Steve Cowan is a Montréal-based classical guitar performer and teacher. Cowan is an active chamber musician and multiple award winner, referenced as "one of Canada's top contemporary classical guitarists" by Classical Guitar Magazine (*Steve Cowan official website* n.d.).

The project's main research-creation objectives are described in Section 5.3, which focuses on the temporal prison and extension bias discussed in Section 2.3. For the temporal prison, we focused on using the GuitarAMI to mitigate the lack of interpretative freedom in mixed music composed with pre-recorded material. For the extension bias, we focused on exploring the GuitarAMI's gestural vocabulary as an extension of the classical guitar techniques. In addition, the GRCP provides insight into the conceptual tools performers and designers use to communicate. The hypothesis is that high-level gestural descriptors facilitate communication between collaborators when used as the basis for the instrument's gestural vocabulary.

The music pieces were chosen from the Cicchillitti-Cowan Duo repertoire and cover several composition techniques for mixed music, e.g., tape music, live electronics, ensemble (laptop and acoustic instruments), multi-effects sound modifications. The pieces are presented in detail, and the modifications are discussed in Section 5.5. The selected pieces were part of *Focus*, a full album recorded by the Duo, and also performed live during the first 21st Century Guitar Conference (21CGUITAR).[1]

The performers were particularly interested in the GuitarAMI as a non-invasive self-contained system. As a non-invasive system, the GuitarAMI allows performers to choose their preferred guitar without permanent modification or damage and even change instruments according to the desired timbre. As a self-contained system, the GuitarAMI helps the performers with practical issues during setup and sound check (Hattwick 2017), a common problem presented in Section 2.3, by reducing the number and complexity of components. As experienced contemporary music performers, Cowan and Cicchillitti have experienced several difficulties during setup and sound tests in previous mixed music concerts. As mentioned by Cook (2009) and McPherson and Kim (2012), portability and ease of use and setup heavily influence the design and usage of DMIs/AMIs. A survey by Sullivan and Wanderley (2018) showed the difficulty of configuration as one of the main factors influencing DMI users to stop using a particular instrument. We present a more in-depth discussion on that topic in Section 5.8.

[1] `https://www.21cguitar.com/`, accessed on March 26, 2021.

A live performance was initially planned as a primary artistic outcome of the GRCP. In addition to the compositional and performative aspects, the concert would highlight mapping explorations, gestural control of DSP, and the system's portability. However, due to restrictions on public gatherings during the COVID-19 pandemic, the concert was replaced by a recording session. While the recording session can explore the GuitarAMI flexibility in sending audio and data to other computers and Digital Audio Workstations (DAWs), it requires a different setup from the live performance. For the latter, all samples, pre-recorded audio, and DSP were embedded into the GuitarAMI's SPU. In contrast, for the former, some of the audio processing was done in the control room to allow the recording of separate audio tracks. Section 5.5 presents two possible setups: the embedded setup for live performances, and the recording setup, sending data/audio to external computers and DAW.

5.2 Justification

The observations from the TMCP, described in Section 4.9, revealed how the availability of high-level gestural descriptors influences DMIs learning, composition, and performance. These findings suggest that creating a gestural vocabulary during the DMI design process may help designers and users improve knowledge transfer and instrument learnability. Still, there are unanswered questions when applying the same principles for AMIs while considering the topics presented in Section 2.3. The GRCP objectives focus on two specific questions:

1) Does the conscious use of high-level gestural descriptors in AMIs facilitate mapping, controlling, or modifying synthesis/DSP in mixed music performances? In other words, do AMIs with established instrumental techniques aid mixed music exploration beyond the extension bias?

2) Does the use of AMIs in mixed music performance have any impact on interpretative freedom and tackle the temporal prison problem?

We established some guidelines to design a research-creation project from these questions. It is necessary to use an AMI with an established gestural vocabulary and usage in artistic performances to explore the first question—as opposed to a purely academic research-oriented prototype. Moreover, it is desirable to observe and participate in the experimentation sessions rather than simply provide the instrument to the users. Lastly, it is helpful to apply the creative process with distinct scenarios, as performers may behave differently according to the music requirements, e.g., the performer's spare bandwidth can vary according to what the musical piece technically demands.

Additionally, it is essential to have a baseline to explore the second question. The reference will most likely be provided by previous performance experience in this scenario. This demand is particularly challenging to fulfill, as it requires motivation from collaborators, including composers and performers, to adapt or modify previous works that were not initially composed to use gestural controllers. At the same time, the pieces simultaneously have the potential to be expanded when adding AMIs.

There are very few AMIs used in artistic performances with an accessible established gestural vocabulary, i.e., a set of high-level gestural descriptors implemented either as an external algorithm or embedded into the device. The GuitarAMI is one of these devices, as discussed in Section 3.4, and it is suitable to fulfill the research and artistic objectives of the GRCP.

5.3 Objectives of the GuitarAMI Research-Creation Project

The objectives of the GRCP can be placed into two categories:

1) the research aspect on the questions presented in Section 5.2:
    (a) an investigation on how the use of AMIs in live-electronics mixed music affects the temporal prison problem (Section 2.3);
    (b) an investigation on how the use of high-level gestural descriptors in AMIs affects the extension bias (Section 2.3); and
2) the creation aspect, involving promoting the use of the GuitarAMI beyond the instrument's initial motivations, more specifically as a tool for composers and performers to create interactions between performer and electronics in mixed music.

To properly tackle the research aspect of the GRCP, we chose mixed music compositions which cover different common compositional strategies in the genre. The collaborating performers had direct participation in all project steps. In contrast with the TMCP, the instrument exploration took place in virtual and in-person mapping sessions with the project collaborators. The discussion on piece selection and mapping sessions can be seen in Section 5.4.

5.4 Timeline and activities

The GRCP activities were structured as shown in Table 5.1.

During Spring and Summer 2020, the project leaders started defining the pieces, activities, and project scope. The original plan included the following:

1) instrument building/replication to allow all collaborators to have individual GuitarAMIs,
2) multiple sessions with all collaborators to explore possible mappings and the performers' spare bandwidth,

Table 5.1 GuitarAMI Research-Creation Project (GRCP) activity timeline.

2020		
	Spring/Summer	Project preparation, including meetings with performers, scope definition, and repertoire selection
	Fall	Official project submissions to CIRMMT for a live@CIRMMT concert at MMR
		Building process for GuitarAMI version 5a
2021		
	Winter	Mappings sessions and rehearsals
		live@CIRMMT MMR recordings
	Spring/Summer	Feedback sessions and future work
		Mixing/mastering, audio editing
		Video releases
	Fall	Publishing results

3) embedding patches/algorithms into the SPU, and

4) a recording session for video, audio, and gestural data shortly before the concert.

Modifications due to pandemic restrictions included postponing sessions in item 2 from Fall 2020 to Winter 2021 and reducing the number of in-person meetings. Also, item 4 was expanded to accommodate the extra recording sessions.

Based on the results from latency studies discussed in Chapter 3 and the flexibility available when using custom Linux distributions, we built the GuitarAMI versions 5 and 5a using off-the-shelf audio interfaces and a modified OS, capable of running algorithms in multiple music programming languages. GuitarAMI versions 5 and 5a are similar in software, OS, and functionality, differing only in the enclosure design and embedded audio interface. Three SPUs and several GuitarAMI modules were built in Fall 2020. During the same period, we received the musical material from the composers—scores, audio samples, recordings, Max patches, DAW sessions, and auxiliary files—to port the algorithms when needed. For example, Max patches cannot run on Linux-based OSs, and some DAWs require modifications to work properly.

With the live concert canceled, the project was divided into two stages. During the first stage, the rehearsals and algorithms were explored in a setting suitable to fulfill the recording requirements. All audio tracks and effects were sent to the mixing desk and studio DAW as individual channels. The GuitarAMI sent gestural data to a computer running the algorithms, which connected directly to the studio's DAW to ensure maximum audio quality. During the second stage, algorithms were ported and embedded into the SPU, allowing the performers to tour using the GuitarAMI easily.

Meetings with the performers to explore the research questions associated with instrument practice took place at CIRMMT facilities from March 1 to March 7, 2021. These exploration sessions included instrument experimentation, mapping sessions, rehearsals, and a data/audiovisual recording session. We applied the findings discussed in Section 3.4.3 to adjust according to the performer's spare bandwidth aspects, e.g., set *tilt* range to allow the performer to control mapped parameters without impairing playability.

The main recording session at the MMR took place on March 8th and 9th, 2021. Using the recording setup during the MMR and the self-contained, embedded setup during the exploration sessions allowed performers to experience both scenarios and provide more informed feedback, discussed in Section 5.7.

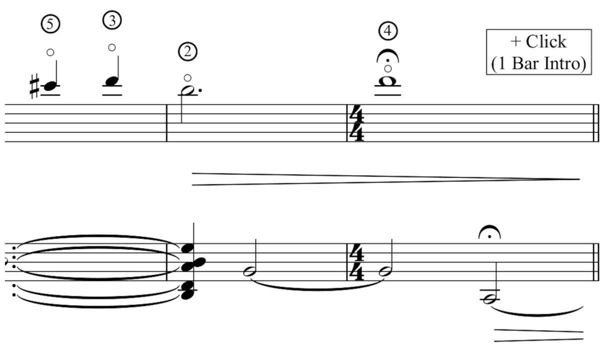
5.5 Live-electronics mixed music repertoire

The artistic outcome of the GRCP consisted of three audiovisual recordings of Cicchillitti-Cowan Duo performing the pieces described in Sections 5.5.1 to 5.5.3. The three pieces initially chosen for the GRCP were originally commissioned by the first 21CGUITAR conference and premiered by the Cicchillitti-Cowan Duo on August 23, 2019, at Dominion Chalmers Hall in Ottawa, Canada. During the project, one extra piece was recorded at CIRMMT for the second 21CGUITAR conference, described in Section 5.5.4. As stated in Section 5.1, the collaborators prepared the artistic works for recording sessions and live performances.

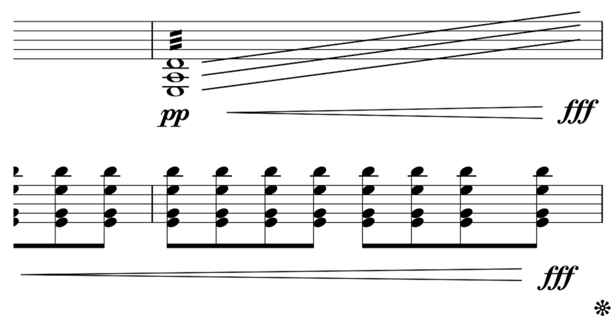
5.5.1 The Turing Test

The Turing test is a composition by Alex Burtzos, composed initially for amplified Guitar Duo, electronics (Loop pedal), and Fixed Media (Tape), with a duration of approximately 6 minutes and 30 seconds. Burtzos adapted the piece to be performed without the loop pedal by mixing a pre-recorded guitar loop track into the fixed media and adding a click track, all programmed in Max. The piece instructions state that "the fixed media electronic track should be activated during the fermata in measure 9. The click track should be sent in-ear to the performers and not included as part of the master mix. The track begins with a four-count introduction and concludes after measure 182."

The composition explores technical aspects intimately related to the original GuitarAMI exploration of classical guitar's limitations. Some examples of this exploration can be seen in Figure 5.1. Figure 5.1a reveals that Burtzos wrote a relatively high-pitched note that must be sustained for an extended period. Even though the composers use note dynamics to make the musical gesture feasible, i.e., marked the note *fortissimo* to make the sound last longer, the gesture is particularly challenging for performers, and the sonic result lacks the sustain effect (holding the note with the same intensity for a period of time). In Figure 5.1b, Burtzos uses the tremolo technique to facilitate the execution of the glissando/crescendo, creating a texture familiar in classical guitar compositions. The tremolo is fundamental in the execution of this passage, and it is impossible to achieve the crescendo effect or *slide* for the entire required duration without the tremolo, i.e., a plucked string instrument cannot increase or hold the volume of a note after the note has been played. In this situation, the composer was forced to use the tremolo, even if the intention was to achieve a glissando/crescendo effect without it.



(a) Long, sustained high-pitched note. The full duration of the musical gesture is difficult to achieve due to the natural amplitude envelope in classical guitars.



(b) Glissando/crescendo using tremolo as facilitator. The full duration of the musical gesture would be impossible to achieve without the tremolo.

Figure 5.1 Examples of musical gestures related to classical guitar's limitations used on *The Turing Test*.

Before the exploration sessions, the performers expressed their desire to execute the impossibly long sustained notes and the glissando/crescendo sections. We could use the GuitarAMI features to achieve musical gestures that were physically impossible. As the composer provided a fixed media containing the acousmatic and the pedal loop tracks, the collaborators focused on using the GuitarAMI to achieve the mentioned musical gestures and to embed all algorithms and tracks into the SPU.

We observed a lower spare bandwidth for the second guitar during the exploration sessions. This part is responsible for the chord accompaniment and provides little rest time.

Even though the first guitar was the most prominent for most of the piece, the melodic aspect and long notes allowed more exploration of other performative aspects and added gestures. Cicchillitti performed the first guitar while Cowan performed the second guitar. Interestingly, the performer's spare bandwidth heavily influenced how each performer started exploring the GuitarAMI. Cowan had less spare bandwidth to move his torso or guitar due to the musical density of his part in the piece. However, as he was more familiar with electric guitar's effect pedals, Cowan could comfortably use the GuitarAMI footswitches to trigger events. On the other hand, Cicchillitti's part was less dense and often focused on melodic events and long notes. He could comfortably move his torso while performing this piece, allowing him to use the GuitarAMI's *tilt* and *roll* gestures to control algorithms or effects in real-time.

Based on observations during the sessions, we set the live performance mappings employing SuperCollider[2] and a Linux Audio Developer's Simple Plugin API Version 2 (LV2) plugin host to replicate the composer's original Max patch, thus embedding the composition elements within the GuitarAMI. In SuperCollider, we played and controlled the fixed media control and click track (audio player), the GuitarAMI's Infinite Reverb, and an OSC to MIDI translator used to send MIDI messages to the plugin host and control the LV2 plugins. The plugins activated on the LV2 host were bypass/gain, pitch-shifter, and distortion.

Cicchillitti used the *tilt* gesture to control the infinite reverb algorithm and perform the musical gesture in Figure 5.1a, sustaining the required sound for as long as necessary. The *tilt* gesture was repeated throughout the piece when needed. Cowan was responsible for the fixed media and click track control, triggered once after the piece's introduction. We observed some of the mentioned interactions in the recording session around 00:53

[2] `https://supercollider.github.io/` accessed on February 4, 2020.

seconds, 00:57 seconds, and 8:01 minutes. These are discussed in Section 5.6.

Cicchillitti also used the footswitches to activate or deactivate (bypass) the pitch-shifter and distortion plugins. *Roll* and *tilt* GuitarAMI gestures were used to control the dry/wet parameter on the plugins, performing the musical gesture in Figure 5.1b without moving the left hand. While the pitch-shifter and distortion effects were mapped and available, the performers decided not to use them during the recording session. There was insufficient practice time to incorporate all available gestures into the performance with the desired proficiency.

The live performance mapping connections for *The Turing Test* are presented in Figure 5.2. The video recording for the performance is available at `https://www.youtube.com/c/CIRMMT/`.

5.5.2 Insomnia Rain

Insomnia Rain is a composition by Derek Cooper, composed initially for Guitar Duo with Electronics, with a duration of 6 minutes and 45 seconds. The electronics part contains a fixed track and a set of effects for each guitar: reverb, distortion, and a combination of pitch-shifter and echo effects to mimic a raindrop effect from the acoustic instrument sounds. All effects were programmed initially in Max and set using timed events triggered by either a footswitch or a third performer controlling the Max patch (Figure 5.3).

The triggered events have a predefined duration from 1 second to around 10 seconds in a fixed order. For each event, the amplitudes of the chosen effects are automated using a simple triangular-shaped amplitude envelope. The cues are marked on the second guitar's score and the second performer was originally responsible for triggering the events throughout the piece.

Before the exploration sessions, the collaborators expressed concern about the lack of

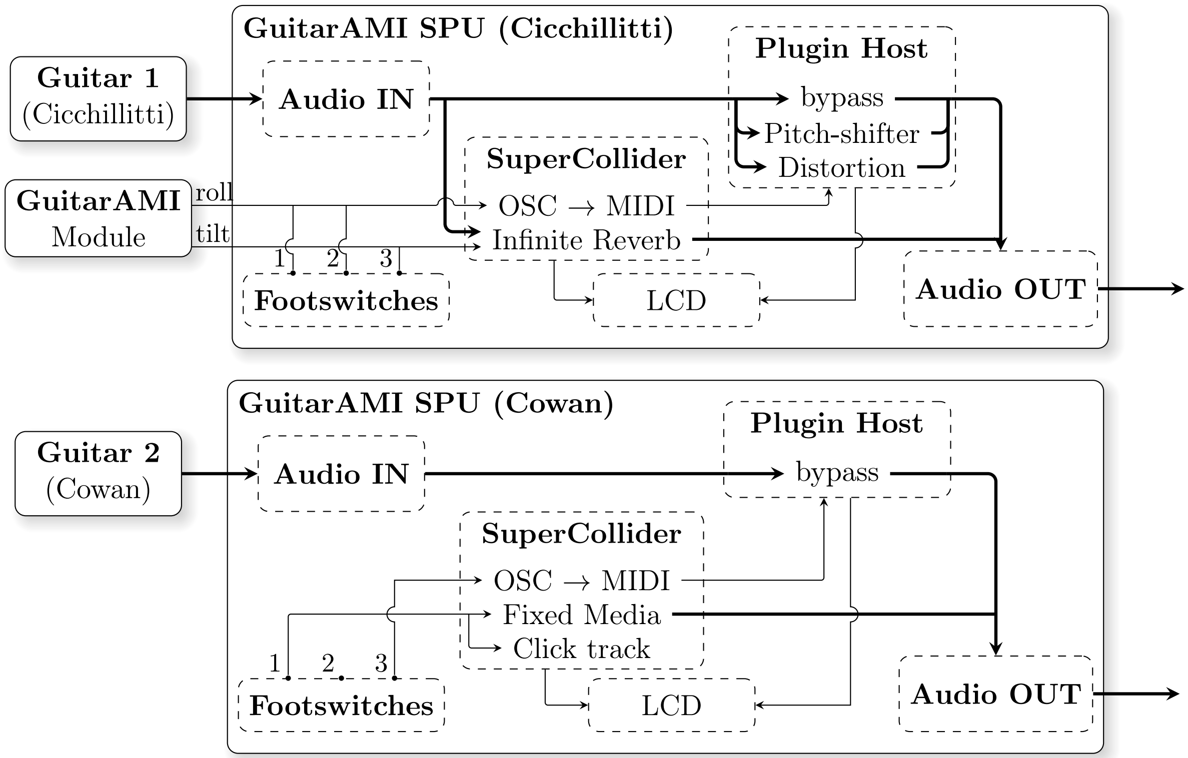


Figure 5.2 Algorithms and mappings for the live performance of *The Turing Test*. Thick lines represent audio connections while thin lines represent wired and wireless data connections.

performer's spare bandwidth for the piece, as keeping synchronization between guitars is particularly challenging in this composition. Conversely, the effects and fixed media are lenient regarding synchronization, acting as musical drones or ethereal textures. Cowan and Cicchillitti also stated there is implicit freedom on the usage of the effects. The natural "flow of the music" suggested they could exert some "influence" or control on how reverb, distortion, and raindrop effects emerge on the electronic portion of the piece, i.e., shift the placement of the electronics in the foreground or the background of the musical texture.

As we experimented with different mappings during the exploration sessions, the collaborators' initial assessments regarding the lack of performer's spare bandwidth were confirmed. The collaborators decided to maintain the original performance structure where the second

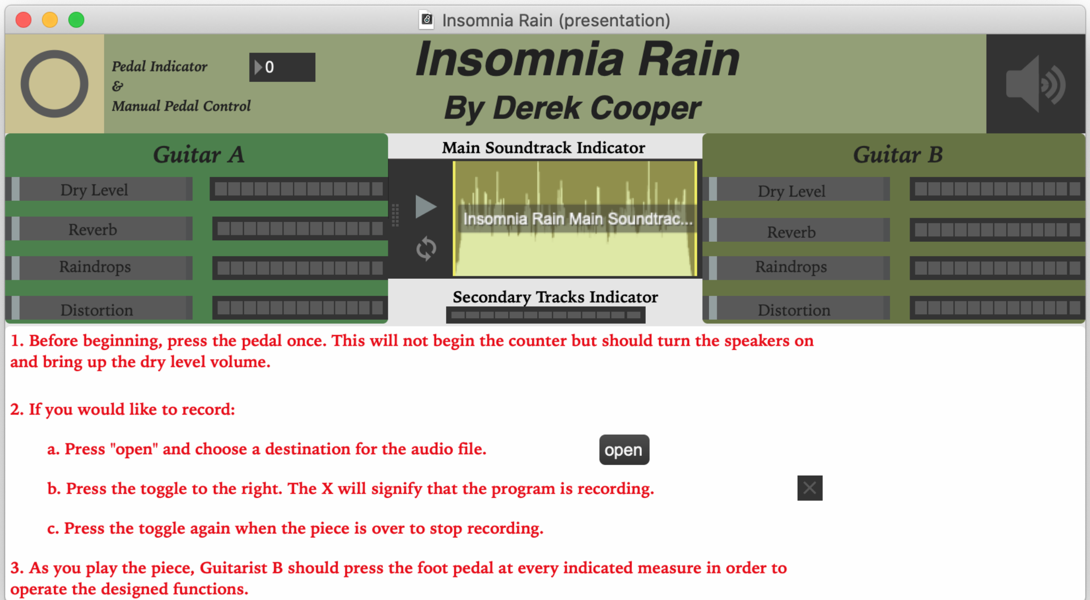


Figure 5.3 *Insomnia Rain*'s original Max patch, programmed by Derek Cooper. The patch is either controlled using a computer-compatible footswitch or manually by a third performer.

guitar triggers the events. Additionally, new mappings were created to control the effects' amplitudes using the first guitar's *roll* and *tilt* GuitarAMI gestures. Cicchillitti played the first guitar and performed the GuitarAMI gestures, while Cowan performed the second guitar and triggered the events.

For the live performance setting, the triggering and the fixed media player were implemented on Cowan's SPU. Both SPUs contained the same plugin set: bypass, reverb, distortion, pitch-shifter, and echo. As Cicchillitti's gestures and Cowan's triggers control both performers' plugin sets, we needed data transferred between both SPUs. All module(s) and SPUs are connected to the same network, and SuperCollider was responsible for exchanging OSC messages between the devices. Alternatively, all SuperCollider code and plugins could be embedded within a single SPU, although the impact on the individual practice of the piece is yet to be determined. The live performance mapping connections for *Insomnia Rain* are presented in Figure 5.4. The video recording of the performance is available at `https://www.youtube.com/c/CIRMMT/`.

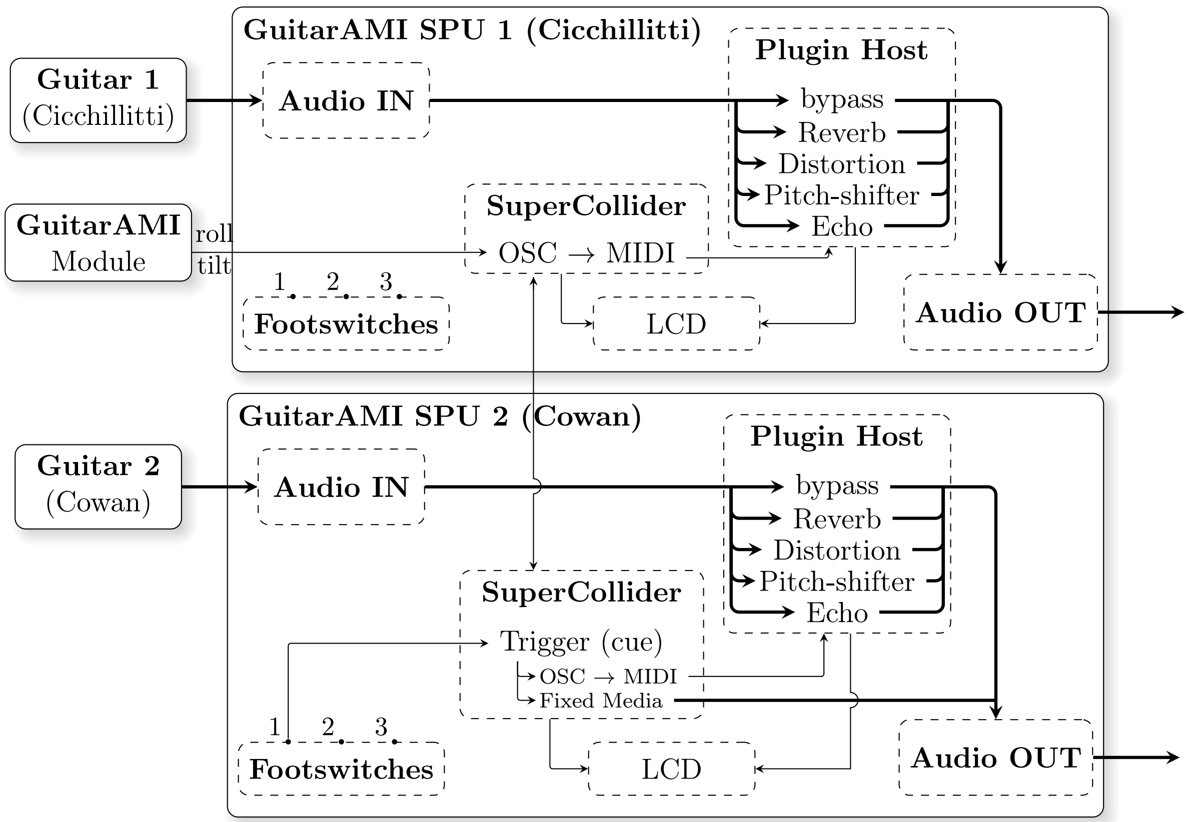


Figure 5.4 Algorithms and mappings for the live performance of *Insomnia Rain*. Thick lines represent audio connections, while thin lines represent wired and wireless data connections.

5.5.3 Focus (Van Tilburg Remix)

Focus is a two-movement composition by Harry Stafylakis and remixed by Adam Pietrykowski. The piece was composed initially for Guitar Duo with Electronics, and the two movements—*Radial Glare* and *Inward Gaze*—have a duration of 9 minutes and 45 seconds.

The electronic part was entirely built using Reaper and performed by Pietrykowski during the premiere. Pietrykowski's performance was similar to traditional mixed-music performances where the musician acts directly on the computer or mixing desk sliders, controlling the sound diffusion. The electronics were composed using a series of fixed media (musical drones), including a low-frequency backtrack and common electric guitar effects: bypass, reverb, octave delay, dual tap delay, and stutter.

All elements used in the Reaper session can be ported or reproduced on the SPU in a setup similar to *Insomnia Rain*. However, the mapping exploration for *Focus* presented challenges related to spare bandwidth due to the density of the music. The rhythmic character of the piece leaves little room for most of the GuitarAMI gestures, and Pietrykowski's deep exploration of the electronics parameters during the performance was challenging for guitarists to emulate without impairing guitar performance.

Confronted with the lack of spare bandwidth that precluded additional gestures, the performers desired to explore the synchronous relationship between guitars in the piece. The resulting solution was to create triggers to virtual events tied to the musical gestures already available in *Focus*. These events include amplitude and plugin parameter automation, following the directions given by Pietrykowski in the edited score for the remix version. The role of triggering events was divided between the performers. The relationship between these events created a hidden rhythm performed between the guitarists as another rhythmic layer of the composition. The sharing approach also reduced the individual performer's bandwidth requirements since some events must be triggered simultaneously or in rapid succession.

Focus required complex internal mappings since some triggers started multiple automated events (divergent mappings), acting on different algorithms/plugins. However, performers only had to trigger once to start the automated routine. This single input was mapped from three different GuitarAMI gestures/triggers. The performers could choose between these gestural possibilities during the performance to facilitate the interaction according to the available bandwidth.

The performers could use footswitch 1, footswitch 3, or the ultrasonic sensor trigger to call each event. While the ultrasonic trigger provided an easy visual cue to the audience, it could not be used during high-density musical passages. The simplicity of the employed direct mappings raised interesting discussions and observations, presented in more detail in Section 5.7.

In contrast to *Insomnia Rain*, in *Focus*, we programmed all DSP and playback processes in a single SPU, using centralized code management without additional communication and keeping all composition material fully embedded. With this approach, the electronic part is fully available to each guitarist. The *Focus* setup was used as a model for recording all three pieces at the MMR, as described in detail in Section 5.6. The live performance mapping connections for the piece are presented in Figure 5.5. The video recording for *Focus - Van Tilburg Remix* is available at `https://www.youtube.com/c/CIRMMT/`.

5.5.4 Trouveur

Trouveur is a composition by Víctor Báez, composed initially for Acoustic Guitar and Electronics, with a duration of 10 minutes and 40 seconds. The piece is distinct from the remaining compositions performed in the GRCP as it is a piece of live-electronics mixed music with all sounds generated in real-time from the acoustic guitar. The electronics were initially programmed in Max and employed a dual tap delay, a harmonizer, and a ring modulation effect.

Báez composed the piece as a duo, with guitarist and *electronics operator* as performers. This performance configuration presents a scenario where one of the performers—the

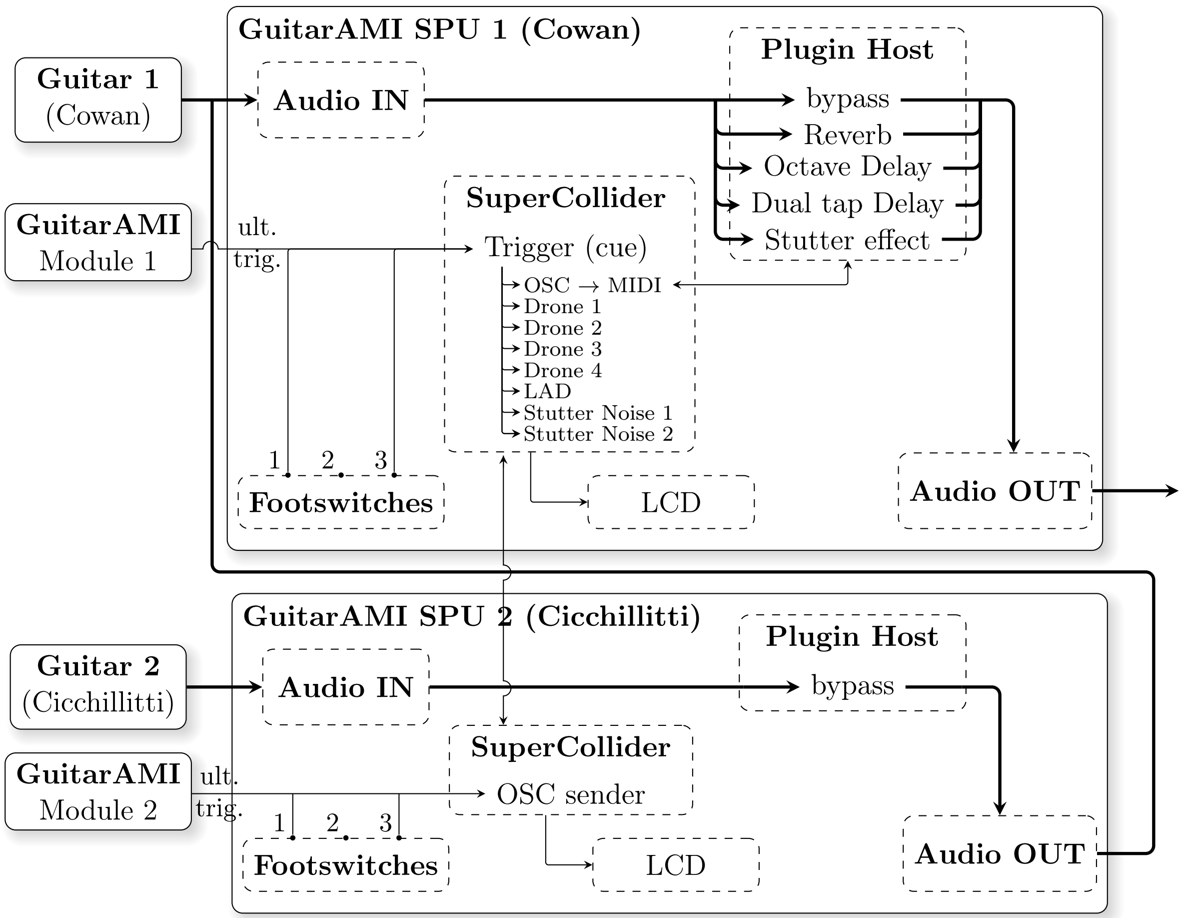


Figure 5.5 Algorithms and mappings for the live performance of *Focus*. Thick lines represent audio connections while thin lines represent wired and wireless data connections.

electronics operator—is not generating any sound but controlling DSP to modify sounds created by the guitarist. This interaction can be seen in an excerpt of the music score in Figure 5.6.

The composer conceptually organized the piece around a guitarist starting the piece solo and the electronics operator gradually reacting and interacting with the guitarist. Báez also suggested in the music score that *Trouveur* may also be performed as a solo piece with or without electronics, even stating the possibility of the guitarist operating the electronics

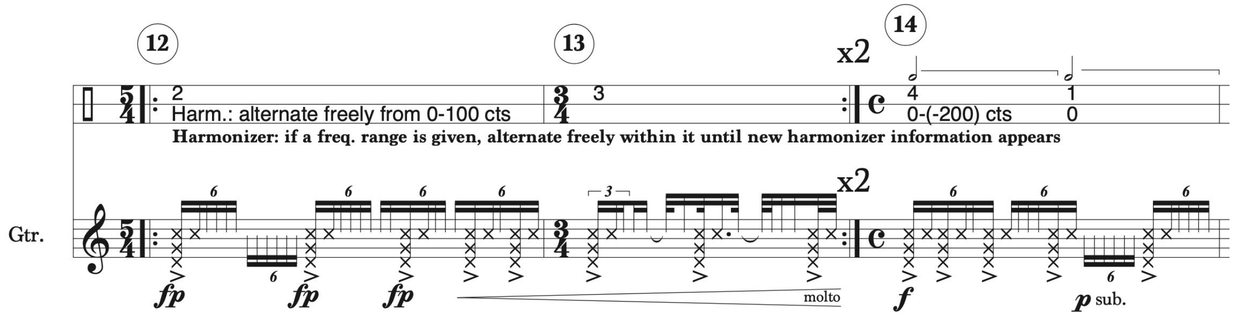


Figure 5.6 Excerpt from the music score for *Trouveur*, composed by Víctor Báez. The score highlights the interaction between the guitar and electronic parts in a chamber-music or ensemble setting.

using external musical controllers.

The dual tap delay, harmonizer, and ring modulator were built in Max. The delay control used nine different presets: presets one through eight are set to gradually increase the values of delay parameters—delay time, rate, and feedback—while the last preset resets all parameters. There are no exact values for the algorithm parameters. Still, the composer provides a graph with parameter curves, and the electronics operator can adjust the parameter proportionally to the graph. Báez also stated a maximum delay value between 5000 and 8000 ms. Performers are encouraged to adjust preset parameters to add variety between preset delay values. There are also guidelines for setting the "transposition" (pitch shifting) parameter for the harmonizer: minimum and maximum values must be set to −2000 and +2000 cents, respectively.

Trouveur was not originally part of the GRCP; however, Cowan suggested recording an audiovisual performance for the second 21CGUITAR conference, at the Universidade Nova de Lisboa, in Portugal, March 22-26, 2021. Cowan also mentioned that the guitar part of the piece is highly demanding. During most of the performance, it would not be possible to execute additional gestures. Consequently, it would not be possible for Cowan to play the guitar part and control the electronics using the GuitarAMI gestural vocabulary.

We suggested using *Trouveur* to explore the approach proposed by Báez: have a second performer using a gestural controller specifically for the electronics. The relationship between the acoustic and electronic parts in *Trouveur* provides new mapping possibilities when the electronics operator uses DMIs.

We chose the T-Stick as a music controller for *Trouveur*, as it shares hardware similarities and is easily interfaced with the GuitarAMI SPU. The parameters exposed to the electronics operator in the Max patch—delay presets and harmonizer transposition value—were mapped from the T-Stick *touch* and *tilt* parameters. The T-Stick performer could use hand position on the instrument to control delay presets and the DMI's tilt angle to finely control the harmonizer transposition. This mapping allowed the T-Stick performer to read the electronics score while keeping the same interpretative freedom as the guitarist.

Despite the demanding aspect of *Trouveur* for the guitarist, Cowan was particularly interested in exploring GuitarAMI gestures for a specific passage during the piece. During the densest passage, the guitarist stops playing, and the sounds are entirely generated from the delay and harmonizer. Cowan suggested mapping the shake gesture on the GuitarAMI as a modifier of the signals from the T-Stick gestures, forming a convergent mapping. While shaking the guitar, Cowan would modulate the T-Stick touch and *tilt* gestures, effectively amplifying the T-Stick performer's gestures.

Finally, the collaborators noticed the ring modulation was not used in the piece: the algorithm was programmed in Max, but there was no exposed parameter to the electronics operator. We then mapped the *tilt* data, associated with the guitar/torso movements, to the ring modulation effect. This extra mapping allowed ring modulation to be used during the piece. The result was a subtle timbre variation caused by the guitarist's natural body movement. The live performance mapping connections for *Trouveur* are presented in

Figure 5.7, while the lab recording can be seen at `https://youtu.be/lEWevEhnPPg`.

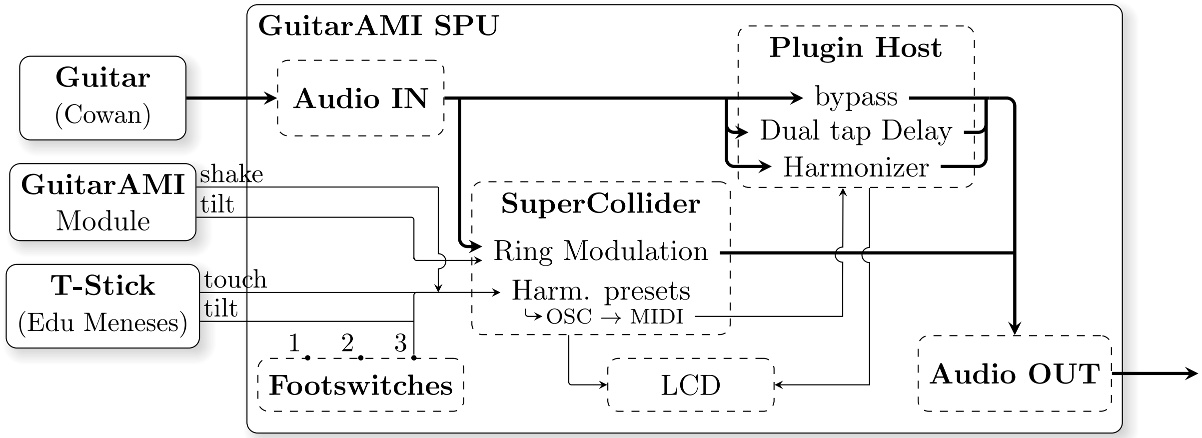


Figure 5.7 Algorithms and mappings for the live performance of *Trouveur*. Thick lines represent audio connections while thin lines represent wired and wireless data connections. The connection transporting the shake gesture from the GuitarAMI actuates as a modifier to the gestures from the T-Stick.

5.6 MMR recording session

As the live@CIRMMT event was changed from a live performance to a recorded session, we took advantage of the benefits of a pre-recorded session. Having multiple takes and more video recording/editing possibilities allowed us to explore AMI flexibility further. The collaborators and CIRMMT staff decided to prioritize audio and video quality by sending instrument data and audio directly to the recording control room, allowing the CIRMMT staff and collaborators to record gestural data, classical guitar audio, and effects in separate tracks. The live mappings for *Focus* were used as a model to create the studio mappings and setup for the MMR recording sessions, although the DSP was transferred to an external computer.

The GuitarAMI SPU has two audio inputs and two audio outputs, making it impossible to send more than two audio signals to the control room. A new recording session mapping was created to send the mono guitar signal directly (bypass) from the GuitarAMI to the control room. The gestural data was sent using ethernet connections. A dedicated computer received audio and data for both guitars, executed all algorithms, and sent the audio tracks to a second DAW for recording. The DAW was also responsible for controlling the performer's in-ear returns. This setup ensured minimal data latency between SPUs and the recording computer when using the ethernet connection while sending audio synchronously to the DAW. The complete MMR recording session can be seen at `https://www.youtube.com/c/CIRMMT/`.

The recording setup also provides the future possibility of recreating the session in real-time using the piece's algorithms, the recorded gestural data, and the unmodified classical guitar sounds.

5.7 Performers' feedback and project observations

The creation of different setups for recording and live performances is common. While we modified the configuration to be indistinguishable to the performers, i.e., there was no difference in the interaction between performers and GuitarAMI using live or studio configurations, the DSP processes employed were different in live and studio configurations. The live configuration employed audio plugins as much as possible, as they provide sounds closer to those expected for standard effects, e.g., distortions, delays, reverbs. The studio configuration used the composer's algorithms, often a recreation of guitar effects in Max. The collaborators could frequently hear the difference, describing the Max recreations as more "digital" or "artificial."

Even though the sounds from the different GuitarAMI configurations used in the project differ, the collaborating performers were particularly interested in using the live configuration.

This configuration would allow them to tour without technical support or complicated concert setups. As discussed in Chapter 2, complex setups with several electronic devices and connections constitute a strong barrier for performers to adopt live electronics without dedicated technical support. The premiere of *The Turing test*, *Focus*, and *Insomnia Rain* in 2019 faced technical challenges to set and perform, requiring technical support and the presence of all the composers for troubleshooting. Composers and technicians are not always available and, most likely, will not tour with the performers.

During the exploration sessions, performers could use *The Turing test* and *Trouveur* entirely embedded in the GuitarAMI. While the *Insomnia Rain* and *Focus* algorithms were ported to SuperCollider and are available for embedding, time constraints prevented the performers from testing the configuration thoroughly.

The collaborating performers were interested in using as many GuitarAMI gestures as possible. Yet, the collaborators soon realized additional gestures required extra hours of practice. While the GuitarAMI shake gesture was incorporated almost instantaneously in *Trouveur*, the *tilt* gesture used in *The Turing test* required some hours of practice to yield the expected sonic result and more time after that for the performer to feel complete in control of the sound.

Another example was the trigger mappings for *Focus*. Cowan was used to triggering events using the pedal; however, Cicchillitti needed extra time to get used to the pedal without shifting focus from the guitar part. Moreover, Cicchilliti's part also presented some fast passages that required the trigger to be shifted or passed to Cowan.

Cowan faced a similar situation in *Insomnia Rain*, where one of the tested ideas was to use the performer's ancillary gestures[3] to modify Cichillitti's gestural data. The tested

[3] Ancillary gestures are performer's gestures that are not required to sound production but modify the resulting outcome (Wanderley and Depalle 2001).

mapping consisted in using Cowan's torso movements to modulate Cichillitti's *tilt* gesture. For these mappings to have the expected sonic effect, Cowan needed to actively control how he shifted the center of gravity during the performance. Even though it was possible to practice and get the expected results, the performers estimated that modifying their muscle memory would be time-consuming and challenging; therefore, this mapping was discarded.

Another observation was the gestural choice to trigger events in *Focus*. Both performers had the option of using footswitches one, three, or the ultrasonic trigger. As discussed in Section 5.5.3, most cues would be triggered using one of the footswitches. However, each performer chose a different trigger: Cowan used Footswitch 1 while Cicchillitti footswitch 3, even though both performed with the guitar in the right-handed position, resting on the left leg. When asked the reasoning, Cowan stated that it made the most sense positioning the pedal further to the right to reach footswitch 1, reducing the risk of accidentally hitting one of the other pedals. Cowan also stated that since he has extensively used a looper pedal while playing electric guitar, it felt natural to maintain the same pedal configuration, with the triggering pedal on the performer's left. Cicchillitti preferred a pedal position slightly shifted to the performer's right, positioning the third footswitch closer to his right foot.

Footswitch position was also reflected on *The Turing test* mappings, where Cicchillitti used all three footswitches to control different effects. In this case, we prioritized footswitch 3 by mapping it to the Infinite Reverb as it was used more often during the piece, followed by pitch-shifter and, lastly, distortion.

All communication regarding mapping decisions between collaborators occurred via high-level gestural descriptors, i.e., the GuitarAMI's gestural vocabulary. The gestural vocabulary was helpful in the mapping design process, especially given that there was limited exploration and rehearsal time. After a short period of adaptation, Cowan and Cicchillitti could actively choose gestures and assess their feasibility in a given piece passage. The AMI gestural vocabulary facilitated music score annotation and, in some passages such as the example in Figure 5.1a, the standard music notation was enough to convey the GuitarAMI gestural idea. Cicchillitti's choice of the *tilt* gesture to perform the Infinite Reverb and some rehearsal time made the passage execution intuitive to the performer. At the end of the process, the performer had the effect constantly active and was able to control the amount of reverb during the performance naturally.

This approach is an example of employing ancillary gestures to generate data and control digital effects continuously (Wanderley and Depalle 2001). However, we noticed that the GuitarAMI gestural vocabulary, rather than ancillary gestures, became part of the performer's instrumental techniques, i.e., effective gestures (Cadoz and Wanderley 2000).

High-level gestural descriptors also allowed the communication between performers to extend beyond discussions of mere movement execution to more interesting aspects of performance aesthetics. We observed the guitarists discussing the GuitarAMI *tilt* speed and displacement to convey a musical idea, similar to what is commonly observed between musicians discussing standard guitar techniques to achieve a particular sonic result.

5.8 Design lessons and future GuitarAMI improvements

As the performers became familiar with GuitarAMI techniques, they suggested mappings for the four pieces performed during the GRCP. Even though the mapping process was straightforward to instrument designers—receiving OSC messages from the GuitarAMI module and mapping using SuperCollider, creating mappings was challenging to the performers as they were not proficient in the music programming language.

The GuitarAMI software is built upon PatchboxOS,[4] and uses open-source tools con-

[4]`https://blokas.io/patchbox-os/`, accessed on August 2, 2021.

nected through OSC and MIDI. The installed tools include SuperCollider, Pure Data, and a plugin host, although users can install and connect any software compatible with Linux-based systems. Since the GuitarAMI SPU was designed to operate headlessly (without a monitor connected to the device), the configuration or modification of mappings requires a web browser or Secure Shell Protocol (SSH) access. Except for the plugin host, accessible using any web browser, Max patches or SuperCollider code must be uploaded and programmed accordingly.

The system embedded in the GuitarAMI allows the performers to tour with a plug-and-play setup, and they could perform minor modifications when needed. Instrument designers can facilitate this process, but performers should have some familiarity with the music programming language used to program the algorithms.

The performers quickly mastered the plugin host; however, they could not create mappings using SuperCollider during the exploration sessions without technical support. Moreover, the audio connections were made using the JACK Audio Connection Kit (JACK), accessible only through SSH. The difficulty in modifying JACK audio connections prevented performers from mapping the Pure Data or SuperCollider audio output to the plugin host, restricting the ability to add post-effects to synthesized audio.

Therefore, we suggest a mapping library such as libmapper and webmapper be used in future work to create and manage connections using a client's web browser. Libmapper and webmapper would allow performers to access the SPU using another computer and create or destroy mappings with little to no programming knowledge.

Additionally, the audio hardware connections presented a challenge for the MMR recording. The GuitarAMI was designed to facilitate connections and, as a non-invasive system, to allow performers to choose any acoustic instrument. The performers explored this flexibility by choosing the classical guitar according to playability and timbre. To ensure maximum audio recording quality, both performers and the sound recording staff decided on condenser microphones. As the GuitarAMI audio interface is configured for unbalanced instrument inputs, condenser microphones require additional hardware. While additional hardware does not pose a challenge for most studio setups, it might impose extra requirements that increase the complexity of live performance setups. The AMI is explicitly designed for classical guitars; therefore, it is desirable to provide the usual requirements for professional classical guitar performances, including built-in microphone input capabilities with proper level and phantom power.

5.9 Discussion

The high-level gestural descriptions played a pivotal role in the GRCP. Since the communication between the collaborators relies on the GuitarAMI gestural vocabulary, the practice sessions developed similarly to acoustic ensemble rehearsals. The collaborators could focus on mappings and aesthetic aspects without translating instrument interactions to sensor data.

This interaction was familiar to the musicians and, according to the performers, "felt more like a rehearsal and less like a troubleshooting session for the devices." However, the workflow was still occasionally interrupted as the mappings needed to be coded in SuperCollider. The code grew in complexity, so the troubleshooting time in case of extensive code modifications. The use of libmapper/webmapper suggested in Section 5.8 may mitigate the issue; however, the effects of improving mapping tools for the system are yet to be evaluated.

Moreover, composers still need to create algorithms for each piece and ensure the patch or code works on the intended system. When analyzing the original Max patches for

Insomnia Rain and *Focus* to port them to SuperCollider, we noticed that the composers made an effort to create user-friendly patches for the performers. Simultaneously, parts of the Max code were not visually cleaned up, suggesting misplaced connections and objects, most likely from last-minute modifications to adapt the patch for a particular performance setting. Replacing the laptop with the GuitarAMI (or any other DMI/AMI) will likely shift a composer's focus from programming a user interface for the patch to creating meaningful sound-gesture relationships/mappings.

Adaptations were also often necessary for the visual feedback. The current SPU version uses a 20x4 Liquid-crystal display (LCD) to provide basic visual feedback. During the GRCP the LCD was used to show the algorithms in use, the module battery level, and some piece information, e.g., current cue. The collaborators used visual feedback very sparingly, as we stipulated the feedback should not disrupt performance. There are rare examples of publications mentioning the disruptive nature of visual feedback, e.g., Martin (2013). To this author's knowledge, there are currently no case studies or experiments deeply exploring or observing the effects of visual feedback in shifting the performer's attention.

The SPU position and bare-minimum visual feedback led to an interaction closer to a regular guitar pedal, with little distraction and guiding the performers to rely on the sonic feedback. We observed a noticeable difference in concentration during the exploration sessions when performers played *Insomnia Rain* using the laptop with the screen as visual feedback compared to using the GuitarAMI alone. The information available on the Max patch included all amplitude levels and modifications in all effects. Conversely, the GuitarAMI LCD only displayed the current cue.

The performer's spare bandwidth also affects the engagement with visual feedback and possible focus shift. The available performer's spare bandwidth changed during the project based on their time with the GuitarAMI. Some gestures deemed too challenging to perform initially became more feasible later during the exploration process. The performers also started to evaluate some AMI gestures according to rehearsal time, perceiving some gestures as feasible "after some study" or stating that "it can be done, but I need some time to practice it." As the performers became more familiar with the AMI and the associated gestural vocabulary, these gestures became part of their learned techniques. This process allowed the performers to assess the technique's feasibility similar to traditional classical guitar techniques and estimate the practice time needed to perform a particular passage properly.

The assessment of the feasibility of performing a particular gesture in a given situation is directly related to the concept of transparency (Fels, Gadd, and Mulder 2002). Fels, Gadd, and Mulder presents transparency as a measure of mapping understanding, i.e., the relationship between the performer's actions and the sonic outcome. The authors suggest a two-axis scale relating audience and performer transparency. Fels, Gadd, and Mulder also propose using known music interactions as metaphors to create allegedly more transparent mappings. However, the guitarists' assessment of how feasible a particular gesture is after practicing suggests the performer's transparency may not be a static characteristic of the mapping, but it also depends on the individual.

The interaction paradigm also changed during the project, and the exploration of mappings between multiple instruments/devices became familiar to the collaborators. The collaborators explored convergent mappings in *Trouveur*, applying the idea of a gestural data stream modifying another one. A similar procedure was employed on *Focus*, where Cicchillitti's gestures could drastically modify the sonic outcome of both guitars. A remarkable interaction between the performers emerged from that relationship, as we could observe from the discussions during rehearsals. The performers discussed the simultaneity of Cowan's right hand and Cicchillitti's speed/displacement for the GuitarAMI *tilt* gesture, rehearsing that passage to achieve the desired result.

The shake gesture used in *Trouveur* is generated using specific algorithms (Section 3.8), in a convergent mapping between the GuitarAMI and T-Stick gestures, and subsequently used in a divergent mapping to each DSP parameter. In another example, the mapping layer exposed in *Focus* shows multiple direct mappings between the gestures and the trigger, hiding the complex trigger process connected in multiple divergent mappings.

In both examples, most connections are fixed mappings on the GuitarAMI or internal mappings for a particular piece. The performers were solely aware of a single mapping layer between the GuitarAMI high-level gestural descriptors and the exposed piece parameters. The acquired proficiency in the AMI gestural vocabulary and understanding of each piece's parameters allowed performers to autonomously and independently perform the musical pieces at an expert level.

These interactions were possible since the performers learned the GuitarAMI gestures and clearly understood the gesture-sound relationship. The transparency provided by the high-level gestural descriptors after practice and the DSP parameters hid the complexity of the algorithms used to create the embedded gestural vocabulary or the synthesis algorithms.

Chapter 6

Conclusion

In this dissertation, we presented an investigation of the development and expansion of gestural vocabularies in DMIs/AMIs. We also investigated the impact of these gestural vocabularies in musical performance, learning processes, and the development of expertise in DMIs/AMIs.

The motivation for this work is that as gesture and sound have arbitrary relationships in DMIs/AMIs, users need different approaches to organize instrumental techniques. Once established, the gestural vocabularies of a DMI/AMI can help performers and composers learn and interact with digital instruments.

We carried out a literature review to understand the aesthetical and historical aspects of DMI usage, highlighting the issue of instrument longevity and the interaction between performer and DMI/AMI. We then applied iterative design processes to embed the gestural vocabularies into the instruments used in the research presented in this dissertation: one DMI—the T-Stick—and one AMI—the GuitarAMI. Finally, we conducted two research-creation projects to investigate the impact of gestural vocabularies in performance practice.

6.1 Contributions

We presented several contributions, specifically in DMI/AMI design, research methods, composition, and performance. For the first research question presented in Section 1.4, we studied how instrument designers, composers, and performers develop or expand gestural vocabularies for AMIs and DMIs. For the second research question presented in Section 1.4, we explored how the developed gestural vocabularies impact performance and pedagogy with DMIs/AMIs.

In Chapter 2, we reviewed electroacoustic music through the lens of AMIs. This review allowed us to understand the relationship between live-electronic mixed music and the paradigms for instrument exploration in the 21st century. We also tackled design strategies and methodologies for NIME-related research. The reflections on methodologies are particularly important as there is no consensus on DMI/AMI research methodology. During the investigation of the first research question, we organized an action plan using established research methods—DSR and AR—employed in research-creation projects. While investigating digital instrument research methodologies was not originally intended for this dissertation research, the organization proposed in Section 6.2.1 can serve as the basis for future music technology research, primarily when the research question comprises a strong artistic component.

In Chapter 3, we proposed the creation of a methodology for (re)design and investigated research questions regarding DMIs/AMIs, based on DSR and AR, and organized around research-creation projects. This methodology comprises an action plan based on research-creation, iterative cycles (redesign) from DSR, and evaluation methods from AR. The GuitarAMI and T-Stick redesign processes represent a contribution to the NIME community while also contributing to instrument interconnectivity and a model for the design and maintenance of DMIs, AMI, and gestural controllers.

In Chapter 4, we investigated how composers and performers interact with the T-Stick, and how new gestural descriptors emerge from the use of raw sensor data. Composers using the T-Stick gestural vocabulary took advantage of available gestural descriptors, focusing on composing and developing expertise. Composers using raw sensor data spend extra time creating their gestural descriptors with different levels of complexity. Moreover, composers created interactions with the T-Stick that diverge from the established gestural vocabulary when accessing exclusively raw sensor data. The artistic outcomes of the TMCP contribute to the factors influencing DMI/AMI longevity: musical notation, repertoire, instrumental technique, communities of practice, pedagogical system, stability, reliability, compatibility, and replicability.

In Chapter 5, we investigated the influence of an instrument vocabulary created from high-level gestural descriptors on the exploration of DMIs/AMIs and the development of expertise. Performers organize their practice according to the embedded gestural descriptors, when available. Also, performers using the GuitarAMI organized their instrumental techniques in three categories: traditional classical guitar techniques, extended guitar techniques, and gestural techniques. This organization method allowed the performers to transfer knowledge, interact with other performers using a common vocabulary, and transfer playing expertise between different music pieces.

6.2 Discussions

6.2.1 Methodology for DMI and AMI-based research

The iterative design process was fundamental in preparing the T-Stick and the GuitarAMI to complete the research-creation projects. As the evaluation process was heavily based on user feedback for the methodology applied, the ability to return to the design process and implement the required or suggested modifications expanded the volume and quality of the received feedback. Fixing or upgrading the instruments as part of the methodology process improved instrument reliability and the performers' perception of ownership.

One can argue that researchers in the NIME community are still trying to find an identity of their own. Apart from borrowing methodologies from HCI, there is still a pursuit for a definition that properly represents research related to digital instruments and musical interaction with music technology. Examples of this search for self-identification can be seen in the emergence of terms such as Internet of Musical Things (IoMusT) (Hazzard et al. 2014) and Human-Sound Interaction (HSI) (Di Donato, Dewey, and Michailidis 2020). We expect to see more discussion in the academic and NIME communities addressing methodological issues when researching DMI/AMI design, use, and evaluation.

6.2.2 DMI/AMI instrumental technique

Based on our observations during the TMCP, we can conclude that high-level gestural descriptors, in the form of a gestural vocabulary for the T-Stick, heavily influenced how the collaborating composers explored the controller (see Section 4.9). Also, the composers using the established gestural vocabulary communicate better with the mentors, especially D. Andrew Stewart, already a proficient T-Stick player. Conversely, composers accessing only raw sensor data were more prolific in creating their gestural descriptors by mimicking an already established gestural descriptor or creating new gestures based on the exploration of T-Stick affordances.

Our observations during the GuitarAMI Research-Creation Project and the T-Stick Music Creation Project suggest that a vocabulary of instrumental techniques created from high-level gestural descriptors facilitates gestural controller exploration, mastery, and transfer of knowledge between performers. Composers using the T-Stick established gestural vocabulary built upon existing techniques. Once exposed to the GuitarAMI gestural vocabulary (Section 3.8), the guitarists incorporated the AMI gestures to the set of classical guitar techniques employed in the compositions performed during the research-creation project. Moreover, the guitarists replaced the units associated with raw sensor data, such as the acceleration on the IMU's x-axis—in g-force—for subjective gestural indicators associated with the gestural vocabulary, such as tilting forward or backwards.

A notable example of this phenomenon was the *tilt* gesture. During the GRCP, we used subjective gestural indicators for movement speed, direction, and posture (position). For example, the guitarists could slowly move from neutral position to far-forward, or rapidly move from near-forward to mid-backward position. These directions could be directly translated into a displacement measurement in degrees from the IMU accelerometer and an angular velocity measure in radians per second from the IMU gyroscope. Even knowing the gestural descriptor and raw sensor data equivalence, asking the guitarists for the tilt movement using angle instead of the *tilt* subjective indication led to confusion.

Moreover, some hours into the exploration session, the performers mastered the tilt's physical movement and fully associated it with the gestural descriptor and the musical notation (Section 5.5.1). The guitarist could sight-read the same notation in different parts of the music score, simultaneously playing the requested note, tilting the guitar to control note duration, and maintaining a given angle to control amplitude. There is, in this case, a connection between gesture, sound, and notation.

The GRCP collaborators aimed to use the GuitarAMI to perform musical gestures deemed impossible with the classical guitar alone. Most explored musical gestures were already notated using traditional music notation, e.g., Figure 5.1, allowing an association between high-level gestural descriptors, the Time-Machine algorithm (discussed in Section 3.4) and music notation.

In this scenario, the performer learned the *tilt* gesture as an extended classical guitar technique using the GuitarAMI. We also observed that the association is mapping-dependent, as the gesture could be arbitrarily mapped to other synthesis algorithms. It is, however, unknown if this mapping will become part of the guitarist's instrumental practice outside the scope of the GRCP.

We can conclude that DMIs/AMIs created from gestural controllers, fixed mappings, and sound synthesis may have, in addition to a gestural vocabulary, an extended technique and an instrumental technique vocabulary based on sound output. For AMIs specifically, the sound-based techniques are often incorporated within the acoustic instrument techniques, with the acoustic instrument sounds being used as audio input for the sound processes. Based on feedback from the GuitarAMI performers, we suggest the following organization for the GuitarAMI instrumental techniques, divided into three categories:

1) Traditional classical guitar techniques, e.g., arpeggio or rasgueado;
2) Extended guitar techniques, e.g., bi-tones or tambora (Lunn 2010), including GuitarAMI gestures that control the guitar sound (e.g., tilt the guitar to sustain the sound); and
3) Gestural techniques that can be mapped according to the piece (e.g., the SPU footswitches or capacitive touch sensor module, that can control different parameters depending on the mappings).

Mastering techniques in all categories requires practicing physical movements and postures, even though the reference—or feedback—used to practice the techniques is essentially different between Item 3 and Items 1 and 2. Items 1 and 2 rely on sound output as reference/feedback, while Item 3 relies on data observation or any feedback modality provided by the instrument, e.g., visual, tactile, or auditory (arbitrarily mapped).

It is important to notice that organizing techniques based on gesture description, i.e., Item 3, does not imply that performers will not consider or rely on auditory feedback when practicing a particular music piece. Once performers have understood and mastered the DMI/AMI gestural vocabulary, they still need to practice the gesture when mapped to a specific sonic outcome in a given composition. This process is similar to the learning process for compositions using standard acoustic instrument techniques. Performers master a particular technique but still extensively practice passages in the context of that piece. During the GRCP, we observed the difference between the learning/practicing gestures and performance practice/rehearsal.

Through the gestural learning stage, the performers focused only on the gesture: how to perform a particular movement at different speeds, how sensitive the sensors are, and the range of movement. When practicing gestures, the auditory feedback aided performers in creating a generalized mental image of the gestures and the parameters they could modulate.

Through the performance practice stage, the performers used the learned gestural vocabulary to create and consolidate the mappings for each piece. When rehearsing, the auditory feedback was crucial for performers to practice modulating the gestural parameters, i.e., modify how they executed each gesture, to achieve the desired sonic results.

6.2.3 Electronic-acoustic relationship in the GRCP

The pieces performed during the GuitarAMI Research-Creation Project were previously composed without considering gestural controllers. Fusion and contrast between acoustic and electronic counterparts, a compositional aspect discussed in Section 2.3, were explored by the pieces' composers prior to the project.

However, we noticed the influence of the mapping process and, consequently, gestural vocabulary in amplifying, diminishing, or shifting this fusion/contrast relationship. The acoustic and electronic fusion created in *The Turing Test* is an example of this phenomenon (Section 5.5.1). In this piece, Cicchillitti controls the Infinite Reverb (artificial reverberation) using the GuitarAMI *tilt* gesture, effectively sustaining the played note.[1] Even though the resulting sound has inherently electronic characteristics, there is a clear attempt of fusing the acoustic guitar sound and the sustained electronic sound synthesized with the Infinite Reverb (DSP) algorithm.

The relationship between the *tilt* gesture and sustained note also has a direct connection with the extension bias proposed by Manzolli. In the example above, mapping *tilt* and Infinite Reverb created a tilt-sustain relationship perceived as an instrument's extended technique, as discussed in Section 6.2.2.

As the guitarist performs the gesture, the tilt-sustain relationship becomes transparent to the performer and the audience. Additionally, the mappings using footswitches to trigger events became more transparent to the performers than having an electronics operator controlling a laptop offstage or events triggered without human input.

6.2.4 Instrument longevity

We presented some factors that contribute to DMI/AMI longevity in Table 2.1. Longevity for a DMI or AMI is not ensured simply by organizing a workshop or a research-creation project. Nevertheless, the research-creation projects discussed in this dissertation achieved tangible results.

The preparation for both the TMCP and GRCP required upgrading, replicating, and ensuring the compatibility of the instruments. The research-creation projects also served as a laboratory for T-Stick and GuitarAMI stability and reliability, which improved during the

[1] `https://www.youtube.com/c/CIRMMT/`, accessed on December 2, 2021.

projects through an iterative design.

The T-Stick repertoire grew substantially, as we can observe in Appendix B. The number of T-Stick compositions increased from 13 to 19 in six months. The new compositions also expanded the DMI/AMI community, especially for the T-Stick. The number of T-Stick composers and performers documented by IDMIL archives grew from 4 to 10 (Section 3.3).

Most of the established gestural vocabulary for the T-Stick was embedded into the device at the end of the TMCP (Table 4.2), along with expanded multi-touch functionality, e.g., multi-brush and multi-rub (Section 4.8). Even though a pedagogical system for the T-Stick was not addressed during the TMCP, availability and ease of access to the established gestural vocabulary had a substantial impact on the GRCP. Musical notation was not directly addressed during the TMCP; however, as discussed in Section 6.2.2, the GRCP explored the relationship between gesture descriptors mapped to specific synthesis algorithms and traditional music notation.

The GuitarAMI was essentially an AMI used in comprovisation and improvisatory performances. Prior to the research-creation project, the only composed piece for the instrument was *Improviso em 3 Dimensões*, a comprovisation by B.E.A.T. (Section 3.4). The GRCP allowed the arrangement of 4 mixed music pieces for the GuitarAMI duo and GuitarAMI/T-Stick (Section 5.5). The GuitarAMI gestural vocabulary, already developed before the research-creation project (Table 3.2), was embedded into the instrument firmware and crucial to investigating the proposed research questions.

Finally, we observed an increase in performances with the T-Stick and GuitarAMI after the research-creation projects, and a commitment of some collaborators to becoming T-Stick and GuitarAMI performers, e.g., Michał Seta and Kasey Pocius for the T-Stick, Adam Cicchillitti and Steve Cowan for the GuitarAMI. Even though projects, activities, and events are not the only factors influencing instrument longevity, they provide substantial support for a particular DMI/AMI community, repertoire, and demand.

6.2.5 Towards digital instrument interconnectivity

IDMIL is a prolific research laboratory in creating DMIs, AMIs, and gestural controllers. Beyond communities for instruments built at the laboratory, there is a community of current and former IDMIL instrument designers that share technology, conduct research, and interact with broader NIME communities. The benefits of a common architecture and communication protocol became evident during the research presented in this dissertation. The T-Stick, the GuitarAMI, and recently Probatio (Calegario, Tragtenberg, Wang, et al. 2020) share firmware code, hardware specifications, and communication protocols.

Sharing specifications allows easier replicability and connectivity when multiple projects share the same architecture and microcontroller, e.g., Espressif ESP32.[2] By sharing the same communication protocols, it was possible to arrange *Trouveur* (Section 5.5.4) for T-Stick and GuitarAMI using a single SPU. The SPU can receive gestural data from any gestural controller using OSC or MIDI. However, it contains embedded code to receive and map T-Stick, GuitarAMI, and Probatio messages.

In their current versions, the GuitarAMI, T-Stick, and Probatio support libmapper, allowing the discovery, creation, and destruction of mappings in real time. An example of connectivity between devices can be seen in a video demonstration using the GuitarAMI and T-Stick (performed by Alex Nieva) at `https://youtu.be/iA749TLpZ4A`.

[2] `https://www.espressif.com/en/products/socs/esp32`, accessed on August 4, 2021.

6.3 Limitations and future work

In many DMI/AMI research projects, the instrument designers are also performers and, eventually, composers of the compositions used during the process. We actively engaged with different performers and composers to investigate the impact of high-level gestural descriptors in music practice using digital instruments and gestural controllers.

Due to COVID-19 pandemic restrictions, the GuitarAMI Research-Creation Project was reduced from eight performers/composers to two and from two one-hour live performances to three pre-recorded pieces.

However, five composers performing a DMI—the T-Stick—during the T-Stick Music Creation Project is already a large number of musicians compared to other instruments/-controllers. There are few DMI projects with as many composers. Although we observed guitarists interacting and exploring the GuitarAMI, the composers did not participate directly in the project. The composers provided the material needed to perform the pieces: scores, patches, audio files, and instructions while the project leader made the arrangements. Therefore, there is no assessment on how composers would tackle embedded gestural descriptors compared to raw sensor data. One area of future work on gesture/data usage could be a new research-creation project commissioning new compositions for the GuitarAMI. This project would allow us to verify how composers use and notate the embedded gestural vocabulary while possibly creating new gestures using raw sensor data, in a similar manner to the TMCP.

During the GuitarAMI Research-Creation Project, the guitarist had visual feedback and mentoring from the project leader to learn the GuitarAMI gestural vocabulary. There is little information on the effects of different feedback modalities on developing expertise in AMIs. Further studies should investigate the effect of visual, tactile, or auditory feedback when learning a DMI/AMI gestural vocabulary.

There is still a considerable amount of work to be done in DMI/AMI interconnectivity. While libmapper is already supported, it was not often used, and the mappings were hard-coded directly. More work is needed to improve libmapper capabilities in the SPU, adding the ability to store, load, and ensure mappings are persistent for a given piece. Libmapper would allow quick experimentation with different mappings during composition/rehearsal stages. Once satisfied, users can store the configuration for performance.

Finally, the GuitarAMI module was adapted for the piano[3] and could be applied to other instruments beyond the acoustic guitar. One proposal could use the iterative design process to design modules for other acoustic instruments.

[3] https://www.cirmmt.org/activities/workshops/research/Vania_motion_matrices, accessed on August 3, 2021.

Appendices

Appendix A

T-Stick Music Creation Project - call for composers

2019 T-Stick Music Creation Project Call for composers

CIRMMT Research Axes 1 and 4 are delighted to announce a call for composers and sound artists who are interested in creating a new work for the T-Stick digital musical instrument. This project will bring together five composers (any nationality, all ages) to develop new live electroacoustic works for the wireless 33.5 cm. long **Sopranino T-Stick** with the support of two tutors: D. Andrew Stewart, composer/T-Stick instrumentalist (University of Lethbridge) and Joseph Malloch, T-Stick designer and developer (Dalhousie University). Selected participants will be introduced to the current modes of performance on the T-Stick during two workshops that will focus on the technical foundations of the instrument, while developing participants' unique musical vocabularies for integrating physical playing gestures and electroacoustic sounds. The program will culminate with the performances of participants' new works for the Sopranino T-Stick, performed by the artists, with support from D. Andrew Stewart, in February 2020 at improv@CIRMMT in Montreal.

We are looking for composers or sound artists who envision a unique project that both illustrates a wide-ranging use of the Sopranino T-Stick and seeks to expand the performance practice of the instrument. For this project, we also welcome group submissions for collaborative projects, including projects with other types of performance (e.g., art, dance, drama).

Applicants are invited to submit (1) their biographies along with (2) a one-page project proposal, identifying anticipated software requirements and audio equipment specifications, if possible. Additionally, applicants are invited to send (3) one optional sample of their music or sonic art that illustrates a concept or musical idea they find applicable to the T-Stick. Applications must be written in English or in French and should be emailed to **takuto.fukuda[at]mail.mcgill.ca** no later than October 16, 2019.

More information about the T-Stick and the tutors can be found in the links below;

T-Stick: http://www-new.idmil.org/project/the-t-stick/
D. Andrew Stewart: http://dandrewstewart.ca/
Joseph Malloch: https://josephmalloch.wordpress.com/

The program is supported by the CIRMMT, Nomura Foundation, McGill University, the University of Lethbridge and IDMIL–Input Devices and Music Interaction Laboratory.

Timeline

October 16, 2019: the application deadline
Applications should be emailed to **takuto.fukuda[at]mail.mcgill.ca** no later than October 16, 2019.

November 16, 2019: the first workshop at CIRMMT
This workshop will include presentations on the instrument functionalities and compositional approaches for the T-Sticks, and a hands-on workshop on the practice of the T-Stick, including how to set up, perform and notate. A sopranino T-Stick will be provided. Participants should bring their laptop.

From November 17, 2019 until February 8, 2020: the individual working period
Following the first workshop, the selected participants will work on their proposed compositions with the provided T-Stick under the support of the tutors and the IDMIL T-Stick development team.

February 9, 2020: the second workshop at CIRMMT
This workshop will be dedicated to finalizing the participants' compositions with the aid of the tutors.

February 11, 2020 : improv@CIRMMT at Café Resonanca
The improv@CIRMMT will showcase the participants' compositions in Montreal.

Biographies

D. Andrew Stewart is a composer, pianist and digital musical instrumentalist. A convergence of acoustic and electroacoustic instrumental praxis is at the centre of Stewart's oeuvre. His music is dedicated to exploring composition and performance for new interfaces for musical expression by adapting and evolving traditional praxis. Stewart's work asks whether musical idea – concept, theory, material, technique and means – has kept pace with developments in digital lutherie; furthermore, what are the essential constituents for creating a viable digital instrument for the twenty-first century performer. Stewart has contributed to the field of music technology through his demonstrations at: the International Conference on New Interfaces for Musical Expression, International Computer Music Conference / International Computer Music Association, Electroacoustic Music Studies Network, Electronic Music Foundation, ACM SIGCHI Conference on Human Factors in Computing Systems, Society for Music Theory, and the Guthman Musical Instrument Competition. Andrew Stewart's music has been featured in countries such as: The UK, Netherlands, Switzerland, Czech Republic, Poland, USA, Germany, France, Mexico, Norway, Denmark, Austria, Italy, Korea Republic and his home country of Canada.

Dr Joseph Malloch is an Assistant Professor with the Graphics and Experiential Media (GEM) lab and the HCI, Visualisation & Graphics research cluster in the Faculty of Computer Science at Dalhousie University. Previously, he was a postdoctoral fellow with Ex(Situ (Extreme Situated Interaction Lab), part of the Laboratoire de Recherche en Informatique (LRI) at Université Paris-Sud XI and INRIA. He holds a Ph.D. in music technology from the Input Devices and Music Interaction Laboratory at McGill University. Malloch's research focuses on Human Computer Interaction, especially as applied to creative and expressive interaction with digital tools. His new "digital musical instruments" – including the T-Stick and the prosthetic "Spine" – have been performed and demonstrated across Europe, North and South America in dozens of concerts, including at international conferences, new music festivals, and performances with dancers.

Appendix B

List of T-Stick works composed to date

Table B.1: T-Stick works composed until the submission of this dissertation. Compositions marked with an asterisk (*) were composed during the T-Stick Music Creation Project.

Name (year)	**Composer**	**Instrumentation**
Les multiples usages du mot "geste" (2021)	Antoine Goudreau	sopranino t-stick
Alt F in Front of the Body (2021)	D. Andrew Stewart, Dirk Stromberg, Michał Seta	Karlax, sopranino T-Stick, and Phallophone
Higher Order Gestalt Fromage* (2020)	blablaTrains: Ana Dall'Ara-Majek, Takuto Fukuda	two Sopranino T-Sticks and 24-ch loudspeaker array
The Taxidermy of Negative Space* (2020)	Erich Barganier	sopranino T-Stick, video, and dancer
Memoidalaube* (2020)	Michał Seta	sopranino T-Stick and audiovisual projection
Balance* (2020)	Vincent Cusson	sound installation
Reflexion* (2020)	Macroplasm Duo: Diego Bermudez Chamberland, Yanik Tremblay-Simard	sopranino T-Stick and Theremin
Synthetic Icescapes* (2020)	Kasey Pocius	sopranino T-Stick and laptop orchestra
Soundwalk Comprovisation no.1 (2014)	Darren Copeland	soprano t-stick
Still Life: Eviction (2013)	D. Andrew Stewart	soprano T-Stick
Dweller within (2012)	D. Andrew Stewart	soprano T-Stick
Concerto for T-Stick and Two Laptop Orchestras (2011)	Eldad Tsabary, D. Andrew Stewart, David Ogborn	T-Stick and two laptop orchestras
Packing a lunch ! (2011)	D. Andrew Stewart	soprano T-Stick
With Winds (2011)	D. Andrew Stewart	soprano T-Stick
E pluribus unum | Out of many, one (2009)	D. Andrew Stewart	soprano T-Stick

Name (year)	Composer	Instrumentation
One Ton (2009)	D. Andrew Stewart	soprano T-Stick
Everybody to the power of one (2008-09)	D. Andrew Stewart	soprano T-Stick
Catching Air and the Superman (2008)	D. Andrew Stewart	keyboard, T-Sticks, and orchestra
The One, for t-stick (2006)	D. Andrew Stewart	T-Stick
Dancing with a Tiger (2006)	D. Andrew Stewart	T-Stick

Appendix C

Code for the embedded T-Stick high-level gestural descriptors

```

//*******************************************************//
// Sopranino T-Stick 2GW - LOLIN D32 PRO - USB -WiFi      //
// Input Devices and Music Interaction Laboratory (IDMIL) //
// Created:  February 2018 by Alex Nieva                  //
//           March 2020 by Edu Meneses - firmware version //
//                                200330 (2020/Mar/30) //
// Notes: Based on test program for reading CY8C201xx     //
//        using I2C                                       //
//        by Joseph Malloch 2011                          //
//                                                        //
// Adapted to work with Arduino IDE 1.8.10 and T-Stick    //
// Sopranino 2GW                                          //
//*******************************************************//


// Code for sensor reading, Wi-Fi management, and
    microcontroller options suppressed for code brevity
// Full code accessible at https://github.com/edumeneses/TStick/
    tree/master/Sopranino/2GW/FW200422


struct RawDataStruct {
    byte touch[8];          // /raw/capsense, i..., 0--255, ...
                            // (1 int per 8 capacitive stripes
                            // -- 8 bits)
    byte touchStrips[64];
    int fsr;                // /raw/fsr, i, 0--4095
    int piezo;              // /raw/piezo, i, 0--1023
    float accl[3];          // /raw/accl, iii, +/-32767 (integers)
    float gyro[3];          // /raw/gyro, fff, +/-34.90659
                            // (floats)
    float magn[3];          // /raw/magn, fff, +/-32767 (integers)
    float raw[10];          // /raw (IMU data to be send to
                            // callibration app)
    float quat[4];          // /raw/quat, ffff, ?, ? ,? ,?
    float magAccl;
    float magGyro;
    float magMagn;
    byte buttonShort;       // /raw/button/short, i, 0 or 1
```

```
    byte buttonLong;        // /raw/button/long, i, 0 or 1
    byte buttonDouble;      // /raw/button/double, i, 0 or 1
} RawData;

struct blob {
  byte blobArray[8];        // shows the "center" of each array
  int blobPos[4];           // position (index) of each blob
  float blobSize[4];        // "size" of each blob
} BlobDetection;

struct InstrumentDataStruct {
    float touchAll;         // /instrument/touch/all, f, 0--1
    float touchTop;         // /instrument/touch/top, f, 0--1
    float touchMiddle;      // /instrument/touch/middle, f, 0--1
    float touchBottom;      // /instrument/touch/bottom, f, 0--1
    float brush;            // /instrument/touch/brush, f, 0--?
                            // (~cm/s)
    float multiBrush[4];    // /instrument/touch/brush/multibrush,
                            // ffff, 0--? (~cm/s)
    float rub;              // /instrument/touch/rub, f, 0--?
                            // (~cm/s)
    float multiRub[4];      // /instrument/touch/rub/multirub,
                            // ffff, 0--? (~cm/s)
    float ypr[3];           // /instrument/ypr, fff, +/-180,
                            // +/-90, +/-180 (degrees)
    float shakeXYZ[3];      // /instrument/shakexyz, fff, 0--?
    float jabXYZ[3];        // /instrument/jabxyz, fff, 0--?
} InstrumentData;

byte touchSizeEdge = 4;     // amount of T-Stick stripes for top
                            // and bottom portions of the T-Stick
                            // (arbitrary)
byte nCapsenses;            // autodetected
byte touchStripsSize = nCapsenses*16;

void updateInstrument() {

    // InstrumentData.touchAll: get the "amount of touch" for the entire capsense
    // normalized between 0 and 1
```

```
    InstrumentData.touchAll = touchAverage(RawData.touchStrips, 0, touchStripsSize);

    // InstrumentData.touchTop: get the "amount of touch" for the top part of the capsense
    // normalized between 0 and 1
    InstrumentData.touchTop = touchAverage(RawData.touchStrips, 0, touchSizeEdge);

    // InstrumentData.touchMiddle: get the "amount of touch" for  the central part of the capsense
    // normalized between 0 and 1
    InstrumentData.touchMiddle = touchAverage(RawData.touchStrips, (0+touchSizeEdge), (touchStripsSize - touchSizeEdge));

    // InstrumentData.touchBottom: get the "amount of touch" for  the botton part of the capsense
    // normalized between 0 and 1
    InstrumentData.touchBottom = touchAverage(RawData.touchStrips, (touchStripsSize-touchSizeEdge), touchStripsSize);

    // Save last blob detection state before reading new data
    for (byte i=0; i < (sizeof(BlobDetection.blobPos)/sizeof(BlobDetection.blobPos[0])); ++i) {
      LastState.blobPos[i] = BlobDetection.blobPos[i];
    }

    // 1D blob detection: used for brush
    BlobDetection = blobDetection1D(RawData.touch,(nCapsenses*2));

    // InstrumentData.brush: direction and intensity of capsense  brush motion
    // InstrumentData.rub: intensity of rub motion
    // in ~cm/s (distance between stripes = ~1.5cm)
    for (byte i=0; i < (sizeof(BlobDetection.blobPos)/sizeof(BlobDetection.blobPos[0])); ++i) {
      float movement = BlobDetection.blobPos[i] - LastState.
```

```
blobPos[i];
        if ( BlobDetection.blobPos[i] == -1 ) {
            InstrumentData.multiBrush[i] = 0;
            InstrumentData.multiRub[i] = 0;
            brushCounter[i] = 0;
        }
        else if (movement == 0) {
            if (brushCounter[i] < 10) {
                brushCounter[i]++;
                // wait some time before dropping the rub/brush values
            }
            else if (InstrumentData.multiBrush[i] < 0.001) {
                InstrumentData.multiBrush[i] = 0;
                InstrumentData.multiRub[i] = 0;
            }
            else {
                InstrumentData.multiBrush[i] = leakyIntegrator(movement*0.15, InstrumentData.multiBrush[i], 0.7, leakyBrushFreq, leakyBrushTimer);
                InstrumentData.multiRub[i] = leakyIntegrator(abs(movement*0.15), InstrumentData.multiRub[i], 0.7, leakyRubFreq, leakyRubTimer);
            }
        }
        else if ( abs(movement) > 1 ) {
            InstrumentData.multiBrush[i] = leakyIntegrator(0, InstrumentData.multiBrush[i], 0.6, leakyBrushFreq, leakyBrushTimer);
        }
        else {
            InstrumentData.multiBrush[i] = leakyIntegrator(movement*0.15, InstrumentData.multiBrush[i], 0.8, leakyBrushFreq, leakyBrushTimer);
            InstrumentData.multiRub[i] = leakyIntegrator(abs(movement*0.15), InstrumentData.multiRub[i], 0.99, leakyRubFreq, leakyRubTimer);
            brushCounter[i] = 0;
        }
    }
```

```
    InstrumentData.brush =  arrayAverageZero(InstrumentData.
       multiBrush,4);
    InstrumentData.rub = arrayAverageZero(InstrumentData.
       multiRub,4);

    // InstrumentData.shakeXYZ
    for (byte i=0; i<(sizeof(RawData.gyro)/sizeof(RawData.gyro
       [0])); ++i) {
        if (abs(RawData.gyro[i]) > 0.1) {
            InstrumentData.shakeXYZ[i] = leakyIntegrator(abs(
               RawData.gyro[i]*0.1), InstrumentData.shakeXYZ[i],
                0.6, leakyShakeFreq, leakyShakeTimer[i]);
        }
        else {
            InstrumentData.shakeXYZ[i] = leakyIntegrator(0,
               InstrumentData.shakeXYZ[i], 0.3, leakyShakeFreq,
               leakyShakeTimer[i]);
            if (InstrumentData.shakeXYZ[i] < 0.01) {
                InstrumentData.shakeXYZ[i] = 0;
            }
        }
    }

    // InstrumentData.jabXYZ
    if (arrayMax(LastState.gyroXArray,5)-arrayMin(LastState.
       gyroXArray,5) > 15) {
        InstrumentData.jabXYZ[0] = arrayMax(LastState.gyroXArray
           ,5) - arrayMin(LastState.gyroXArray,5) - 10;
    }
    else {
        InstrumentData.jabXYZ[0] = 0;
    }
    if (arrayMax(LastState.gyroYArray,5)-arrayMin(LastState.
       gyroYArray,5) > 3) {
        InstrumentData.jabXYZ[1] = arrayMax(LastState.gyroYArray
           ,5) - arrayMin(LastState.gyroYArray,5) - 3;
    }
    else {
        InstrumentData.jabXYZ[1] = 0;
    }
```

```
    if (arrayMax(LastState.gyroZArray,5)-arrayMin(LastState.
       gyroZArray,5) > 10) {
        InstrumentData.jabXYZ[2] = arrayMax(LastState.gyroZArray
           ,5) - arrayMin(LastState.gyroZArray,5) - 10;
    }
    else {
        InstrumentData.jabXYZ[2] = 0;
    }
}

float touchAverage (byte * touchArrayStrips, byte firstStrip,
   byte lastStrip) {
    int sum = 0;
    for (int i = firstStrip; i < lastStrip; ++i)
    sum += touchArrayStrips[i];
    return  ((float) sum) / (lastStrip - firstStrip);
}

float arrayAverage (float * Array, byte ArraySize) {
    float sum = 0;
    for (int i = 0; i < ArraySize; ++i)
    sum += Array[i];
    return  (sum / ArraySize);
}

float arrayAverageZero (float * Array, byte ArraySize) {
    float sum = 0;
    byte count = 0;
    float output = 0;
    for (int i = 0; i < ArraySize; ++i) {
        if (Array[i] != 0) {
            sum += Array[i];
            count++;
        }
    }
    if (count > 0) {
        output = sum / count;
    }
    return output;
}
```

```
blob blobDetection1D (byte * touchArray, byte arraySize) {
    // creating local variables
    blob blobDecect;
    byte tempArray[8];
    int beginBlob = -1; // -1 means it will not count stripes
    byte blobCount = 0;
    for (byte i=0; i < sizeof(blobDecect.blobPos)/sizeof(
       blobDecect.blobPos[0]); ++i) {
        blobDecect.blobPos[i] = -1;
        blobDecect.blobSize[i] = 0;
    }
    for (byte i=0; i < sizeof(blobDecect.blobArray)/sizeof(
       blobDecect.blobArray[0]); ++i) {
        blobDecect.blobArray[i] = 0;
    }
    // fixing capsense byte order
    byte order[8] = {1,0,3,2,5,4,7,6};
    for (byte i=0; i < arraySize; ++i) {
        tempArray[i] = touchArray[order[i]];
    }
    // shifting and reading...
    for (byte i=0; i < arraySize*8; ++i) {
        bitShiftArrayL(tempArray, blobDecect.blobArray,
           arraySize, i);
        if ((blobDecect.blobArray[0] & 128) == 128 && beginBlob
           == -1) {
            beginBlob = i;
        }
        if ( ((blobDecect.blobArray[0] & 128) == 0 || i == (
           arraySize*8)-1) && beginBlob != -1) {
            blobDecect.blobPos[blobCount] = (i + beginBlob) / 2;
            blobDecect.blobSize[blobCount] = float(i - beginBlob
               ) / (arraySize * 8);
            beginBlob = -1;
            blobCount++;
        }
    }
    for (byte i=0; i < sizeof(blobDecect.blobArray)/sizeof(
       blobDecect.blobArray[0]); ++i) {
```

```
        blobDecect.blobArray[i] = 0;
    }
    for (byte i=0; i < sizeof(blobDecect.blobPos)/sizeof(
        blobDecect.blobPos[0]); ++i) {
        if (blobDecect.blobPos[i] != -1) {
            bitWrite(blobDecect.blobArray[blobDecect.blobPos[i
                ]/8], (7-(blobDecect.blobPos[i]%8)), 1);
        }
        else {
            break;
        }
    }
    return blobDecect;
}

void printBinary (byte number) {
    byte reading;
    for (int i=7; i >= 0; --i) {
        reading = number >> i;
        Serial.print(reading & 1);
    }
}

void bitShiftArrayL (byte * origArray, byte * shiftedArray, byte
    arraySize, byte shift) {
    for (byte i=0; i < arraySize; ++i) {
        shiftedArray[i] = origArray[i];
    }
    for (byte k=0; k < shift; ++k) {
        for (byte i=0; i < arraySize; ++i) {
            if ( i == (arraySize-1)) {
                shiftedArray[i] = (shiftedArray[i] << 1);
            }
            else {
                shiftedArray[i] = (shiftedArray[i] << 1) | (
                    shiftedArray[i+1] >> 7);
            }
        }
    }
}
```

```
void bitShiftArrayR (byte * origArray, byte * shiftedArray, byte
    arraySize, byte shift) {

for (byte i=0; i < arraySize; ++i) {
    shiftedArray[i] = origArray[i];
}

for (byte k=0; k < shift; ++k) {
    for (int i=arraySize; i >= 0; --i) {
        if ( i == 0) {
            shiftedArray[i] = (shiftedArray[i] >> 1);
        }
        else {
            shiftedArray[i] = (shiftedArray[i] >> 1) | (
               shiftedArray[i-1] << 7);
        }
    }
}
}

float arrayMin (float *inputArray, byte arraySize) {
    float output = inputArray[0];
    for (byte i=1; i<arraySize; ++i) {
        output = min(output,inputArray[i]);
    }
    return output;
}

float arrayMax (float *inputArray, byte arraySize) {
    float output = inputArray[0];
    for (byte i=1; i<arraySize; ++i) {
        output = max(output,inputArray[i]);
    }
    return output;
}


// Simple leaky integrator implementation
```

```
// Create a unsigned long global variable for time counter for
   each leak implementation (timer)

float leakyIntegrator (float reading, float old_value, float
   leak, int frequency, unsigned long& timer) {
    float new_value;
    if (frequency == 0) {
        new_value = reading + (old_value * leak);
    }
    else if (millis() - (1000 / frequency) < timer) {
        new_value = reading + old_value;
    }
    else {
        new_value = reading + (old_value * leak);
        timer = millis();
    }
    return new_value;
}
```

Bibliography

Aceituno, Jonathan (2015). "Direct and expressive interaction on the desktop: increasing granularity, extent, and dimensionality". Ph.D. dissertation. Lille, France: Université de Lille.

Adam Cicchillitti official website (n.d.). URL: https://adamcicchillitti.com/ (visited on 05/09/2021).

Aguinis, Herman (1993). "Action Research and Scientific Method: Presumed Discrepancies and Actual Similarities". In: *The Journal of Applied Behavioral Science* 29.4, pp. 416–431.

Arfib, Daniel et al. (2011). "Time-frequency processing". In: *DAFX: Digital Audio Effects*. Hoboken, USA: John Wiley and Sons. Chap. 7, pp. 237–295. ISBN: 978-0-470-97967-9.

Baker, Monya (2016). "1,500 Scientists Lift the Lid on Reproducibility". In: *Nature* 533, pp. 452–454.

Bellona, Jon (2017). "Physical Intentions: Exploring Michel Waisvisz's The Hands (Movement 1)". In: *Organised Sound* 22.3, pp. 406–417.

Benacchio, Simon et al. (2016). "Active control and sound synthesis – two different ways to investigate the influence of the modal parameters of a guitar on its sound". In: *Acoustical Society of America* 139.3, pp. 1411–1419.

Bongers, Bert (2000). "Physical Interfaces in the Electronic Arts Interaction Theory and Interfacing Techniques for Real-time Performance". In: *Trends in Gestural Control of Music*. Paris, France: Editions IRCAM – Centre Pompidou, pp. 41–70. ISBN: 978-2-844-26039-0.

Brandt, Eli and Roger B. Dannenberg (2002). "Low-Latency Music Software Using Off-The-Shelf Operating Systems". In: *Proc. International Computer Music Conference (ICMC)*. San Francisco, USA.

Brown, Andrew R. (2017). "Expert Commentary: Developing a Digital Virtuosity". In: *A NIME Reader*. Cham, Switzerland: Springer International Publishing AG, pp. 295–296. ISBN: 978-3-319-47214-0.

Cadoz, Claude and Marcelo M. Wanderley (2000). "Gesture-music". In: *Trends in Gestural Control of Music*. Paris, France: Editions IRCAM – Centre Pompidou, pp. 71–94. ISBN: 978-2-844-26039-0.

Calegario, Filipe, João Tragtenberg, Christian Frisson, et al. (2021). "Documentation and Replicability in the NIME Community". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Shanghai, China.

Calegario, Filipe, João Tragtenberg, Johnty Wang, et al. (2020). "Open Source DMIs: towards a replication certification for online shared projects of digital musical instruments". In: *Proc. 22nd International Conference on Human-Computer Interaction (HCII)*. Copenhagen, Denmark.

Campbell, Murray (n.d.). *Timbre (i). In Grove Music Online*. URL: http://www.oxfordmusiconline.com/grovemusic (visited on 05/01/2021).

Cantrell, Joe (2017). "Designing Intent: Defining Critical Meaning for NIME Practitioners". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Copenhagen, Denmark.

Carlevaro, Abel (1967a). *Serie Didactica: Cuaderno Nº 1 - Escalas diatónicas*. Buenos Aires, Argentina: Barry.

— (1967b). *Serie Didactica: Cuaderno Nº 2 - Técnica de la mano derecha*. Buenos Aires, Argentina: Barry.

— (1967c). *Serie Didactica: Cuaderno Nº 3 - Técnica de la mano izquierda*. Buenos Aires, Argentina: Barry.

— (1967d). *Serie Didactica: Cuaderno Nº 4 - Técnica de la mano izquierda conclusión*. Buenos Aires, Argentina: Barry.

Chadabe, Joel. (1997). *Electric sound : the past and promise of electronic music*. Upper Saddle River, N.J.: Prentice Hall.

Computing Machinery, Association for (2016). *Artifact Review and Badging*. URL: https://www.acm.org/publications/policies/artifact-review-and-badging-current (visited on 06/21/2021).

Cook, Perry (2001). "Principles for designing computer music controllers". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Havana, Cuba.

— (2009). "Re-designing principles for computer music controllers: A case study of SqueezeVox Maggie". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Pittsburgh, USA.

Di Donato, Balandino, Christopher Dewey, and Tychonas Michailidis (2020). "Human-Sound Interaction: Towards a Human-Centred Sonic Interaction Design Approach". In: *Proc. International Conference on Movement and Computing (MOCO)*. New Jersey, USA.

Dresch, Aline, Daniel Lacerda, and José Antunes Jr. (2015). *Design Science Research: A Method for Science and Technology Advancement*. Cham, Switzerland: Springer International Publishing AG. ISBN: 978-3-319-07373-6.

Dykstra-Erickson, Elizabeth and Jonathan Arnowitz (2005). "Michel Waisvisz: The Man and the Hands". In: *Interactions* 12.5, pp. 63–67.

Emmerson, Simon (2007). *Living Electronic Music.* Aldershot, UK: Ashgate Publishing Limited. ISBN: 978-0-754-65546-6.

Fels, Sidney, Ashley Gadd, and Axel Mulder (2002). "Mapping Transparency Through Metaphor: Towards More Expressive Musical Instruments". In: *Organised Sound* 7.2, pp. 109–126.

Ferguson, Sean and Marcelo M. Wanderley (2010). "The McGill Digital Orchestra: An Interdisciplinary Project on Digital Musical Instruments". In: *Journal of Interdisciplinary Music Studies* 4.2, pp. 17–35.

Fiebrink, Rebecca, Ge Wang, and Perry Cook (2007). "Don't forget the laptop: using native input capabilities for expressive musical control". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME).* Copenhagen, Denmark.

Fletcher, Neville H. and Thomas D. Rossing (1998). *The Physics of Musical Instruments.* 2nd. Cham, Switzerland: Springer International Publishing AG. ISBN: 978-0-387-21603-4.

Ford, Corey J and Chris Nash (2020). "An Iterative Design 'by proxy' Method for Developing Educational Music Interfaces". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME).* Birmingham, UK.

Franco, Ivan (2019). "A Framework for Embedded Digital Musical Instruments". Ph.D. dissertation. Montreal, Canada: McGill University.

Fukuda, Takuto et al. (2021). "The T-Stick Music Creation Project: An approach to building a creative community around a DMI". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME).* Shanghai, China. URL: `https://nime.pubpub.org/pub/7c4qdj4u`.

Gaver, William W. (1991). "Technology Affordances". In: *Proc. International Conference on Human Factors in Computing Systems (CHI).* New Orleans, USA.

Gibson, James J. (1979). *The ecological approach to visual perception.* Boston, USA: Houghton Mifflin. ISBN: 978-0-898-59959-6.

Goldman, Richard Franko (1961). "Reviews of Records". In: *The Musical Quarterly* XLVII.1, pp. 133–134.

Goodman, Steven N., Daniele Fanelli, and John P. A. Ioannidis (2016). "What does research reproducibility mean?" In: *Science Translational Medicine* 8.341, 341ps12–341ps12.

Goudard, Vincent (2019). "Ephemeral instruments". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME).* Porto Alegre, Brazil.

Griffiths, Paul (1994). *Modern Music: A Concise History.* 2nd. London, UK: Thames and Hudson. ISBN: 978-0-500-20278-4.

— (1998). *A Música Moderna - Uma história concisa e ilustrada de Debussy a Boulez.* 1st. Rio de Janeiro, Brazil: Jorge Zahar Ed. ISBN: 978-8-571-10004-6.

Gurevich, Michael (2016). "Diversity in NIME Research Practices". In: *Leonardo* 49.1, pp. 80–81.

Hattwick, Ian (2017). "Creating Novel Digital Interfaces for Professional Artistic Productions in Collaborative Art-Science Research Projects". Ph.D. dissertation. Montreal, Canada: McGill University.

Hazzard, Adrian et al. (2014). "Musical intersections across the digital and physical". In: *Proc. DMRN+9: Digital Music Research Network One-day Workshop*. London, UK.

Hunt, Andy, Marcelo M. Wanderley, and Ross Kirk (2000). "Towards a Model for Instrumental Mapping in Expert Musical Interaction". In: *Proc. International Computer Music Conference (ICMC)*. Berlin, Germany.

Jack, Robert H. et al. (2018). "Action-sound Latency and the Perceived Quality of Digital Musical Instruments". In: *Music Perception: An Interdisciplinary Journal* 36.1, pp. 109–128.

Jordà, Sergi (2004). "Digital Instruments and Players: Part I — Efficiency and Apprenticeship". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Hamamatsu, Japan.

— (2005). "Digital Lutherie: Crafting musical computers for new musics' performance and improvisation". Ph.D. dissertation. Barcelona, Spain: Universitat Pompeu Fabra.

JP., Ioannidis (2005). "Why Most Published Research Findings Are False". In: *PLOS Medicine* 2.8, e124.

Kirkegaard, Mathias et al. (2020). "TorqueTuner: A self contained module for designing rotary haptic force feedback for digital musical instruments". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Birmingham, U.K.

Lalitte, Philippe (2006). "Towards a semiotic model of mixed music analysis". In: *Organised Sound* 11.2, pp. 93–100.

Landy, Leigh (2007). *Understanding the art of sound organization*. Cambridge, Mass.: MIT Press. ISBN: 978-0-262-52925-9.

Lansky, Paul (1990). "A View From the Bus: When Machines Make Music". In: *Perspectives of New Music* 28.2, pp. 102–110.

Le Vivier Mobile (2019). URL: `https://levivier.ca/en/cultural-outreach` (visited on 04/18/2021).

Levitin, Daniel J., Stephen McAdams, and Robert L. Adams (2002). "Control parameters for musical instruments: a foundation for new mappings of gesture to sound". In: *Organised Sound* 7.2, pp. 171–189.

Lunn, Robert Allan (2010). "Extended Techniques for the Classical Guitar: A Guide for Composers". Ph.D. dissertation. Columbus, USA: Ohio State University.

Luthier. In Grove Music Online (n.d.). URL: `http://www.oxfordmusiconline.com/grovemusic` (visited on 02/26/2021).

Machover, Tod and Joe Chung (1989). "Hyperinstruments: Musically Intelligent and Interactive Performance ande Creativity Systems". In: *Proc. International Computer Music Conference (ICMC)*. Columbus, USA.

Magnusson, Thor (2010). “Designing Constraints: Composing and Performing with Digital Musical Systems”. In: *Computer Music Journal* 34.4, pp. 62–73.

Malloch, Joseph (2008). “A consort of gestural musical controllers: Design, construction, and performance”. MA thesis. Montreal, Canada: McGill University.

Malloch, Joseph and Marcelo M. Wanderley (2007). “The T-Stick: From Musical Interface to Musical Instrument”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. New York, USA.

Mamedes, Clayton et al. (2014). “Composing for DMIs - Entoa, Music for Intonaspacio”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. London, UK.

Manning, Peter (2004). *Electronic and Computer Music*. New York, USA: Oxford University Press. ISBN: 978-0-195-14484-0.

Manzolli, Jônatas (2013). “Interpretação Mediada: pontos de referência, modelos e processos criativos”. In: *Revista Música Hodie* 13.1, pp. 48–63.

Marquez-Borbon, Adnan and Juan Pablo Martinez-Avila (2018). “The Problem of DMI Adoption and Longevity: Envisioning a NIME Performance Pedagogy”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Blacksburg, USA.

Marquez-Borbon, Adnan and Paul Stapleton (2015). “Fourteen Years of NIME: The Value and Meaning of ‘Community’ in Interactive Music Research”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Baton Rouge, USA.

Martin, Charles (2013). “Performing with a Mobile Computer System for Vibraphone”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Daejeon, Republic of Korea.

Mathews, Max V. (1991). “The Radio Baton and Conductor Program, or: Pitch, the Most Important and Least Expressive Part of Music”. In: *Computer Music Journal* 15.4, pp. 37–46.

McNutt, Elizabeth (2003). “Performing Electroacoustic Music: A Wider View of Interactivity”. In: *Organised Sound* 8.3, pp. 297–304.

McPherson, Andrew P. (2015). “Buttons, Handles, and Keys: Advances in Continuous-Control Keyboard Instruments”. In: *Computer Music Journal* 39.2, pp. 28–46.

McPherson, Andrew P., Robert H. Jack, and Giulio Moro (2016). “Action-Sound Latency: Are Our Tools Fast Enough?” In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Brisbane, Australia.

McPherson, Andrew P. and Youngmoo E. Kim (2012). “The Problem of the Second Performer: Building a Community Around an Augmented Piano”. In: *Computer Music Journal* 36.4, pp. 10–27.

Medeiros, Carolina B. (2015). “Advanced Instrumentation and Sensor Fusion Methods in Input Devices for Musical Expression”. Ph.D. dissertation. Montreal, Canada: McGill University.

Meneses, Eduardo A. L. (2016). “GuitarAMI: desenvolvimento, implementação e performance de um instrumento musical aumentado que explora possibilidades de modificação de características intrínsecas do violão.” MA thesis. Campinas, Brazil: Universidade Estadual de Campinas (UNICAMP).

Meneses, Eduardo A. L. and José Fornari (2015). “Educação musical através da improvisação livre com recursos computacionais: contribuições e desafios”. In: *Proc. XXV Congresso da Associação Nacional de Pesquisa e Pós-Graduação em Música (ANPPOM)*. Vitória, Brazil.

Meneses, Eduardo A. L., José Fornari, and Marcelo M. Wanderley (2015). “Projeto e Construção de uma Interface Gestual para Improvisação Livre”. In: *Proc. 13rd AES-Brasil Audio Engineering Congress*. São Paulo, Brazil.

Meneses, Eduardo A. L., Sérgio Freire, and Marcelo M. Wanderley (2018). “GuitarAMI and GuiaRT: two independent yet complementary Augmented Nylon Guitar projects”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Blacksburg, USA.

Meneses, Eduardo A. L., Takuto Fukuda, and Marcelo M. Wanderley (2020). “Expanding and Embedding a High-level Gesture Vocabulary for the Digital and Augmented Musical Instruments”. In: *Proc. 22nd International Conference on Human-Computer Interaction (HCII)*. Copenhagen, Denmark.

Meneses, Eduardo A. L., Johnty Wang, et al. (2019). “A Comparison of Open-Source Linux Frameworks for an Augmented Musical Instrument Implementation”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Porto Alegre, Brazil.

Menezes, Flo (2002). “For a morphology of interaction”. In: *Organised Sound* 7.3, pp. 305–311.

— (2009). *Música Eletroacústica: Histórias e Estéticas*. 2nd. São Paulo, Brazil: EDUSP. ISBN: 978-8-531-40301-9.

Merriam-Webster.com (n.d.). URL: https://www.merriam-webster.com/ (visited on 07/13/2021).

Miranda, Eduardo R. and Marcelo M. Wanderley (2006). *New Digital Musical Instruments: Control and Interaction Beyond the Keyboard*. Middleton, USA: A-R Editions, Inc. ISBN: 978-0-895-79585-4.

Morreale, Fabio and Andrew P. McPherson (2017). “Design for Longevity: Ongoing Use of Instruments from NIME 2010-14”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Copenhagen, Denmark.

National Academies of Sciences, Engineering and Medicine (2019). *Reproducibility and Replicability in Science*. Washington, DC, USA: The National Academies Press. ISBN: 978-0-309-48619-4.

Nielsen, Jakob (1993). “Iterative User-Interface Design”. In: *Computer* 26.11, pp. 32–41.

Nieva, Alex (2018). "Maintenance Strategies and Design Recommendations on Input Devices for Musical Expression". MA thesis. Montreal, Canada: McGill University.

Nieva, Alex et al. (2018). "The T-Stick: Maintaining a 12 year-old Digital Musical Instrument". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Blacksburg, USA.

Nyman, Michael (1999). *Experimental Music: Cage and Beyond*. 2nd. New York, USA: Cambridge University Press. ISBN: 978-0-521-65383-1.

Palacio-Quintin, Cléo (2011). "Compositions d'œuvres pour hyper-flûtes et traitement audionumérique interactif". Ph.D. dissertation. Montreal, Canada: Université de Montréal.

Peignot, Jerome (1960). "De la musique concrète à l'acousmatique". In: *Esprit* 280.1, pp. 111–120.

Pestova, Xenia et al. (2009). "The Digital Orchestra Project". In: *Proc. International Computer Music Conference (ICMC)*. Montreal, Canada.

Puckette, Miller (2007). *The Theory and Technique of Electronic Music*. 1st ed. Singapore: World Scientific Pub Co Inc. ISBN: 978-9-812-70077-3.

Roads, Curtis et al. (1996). *The Computer Music Tutorial*. Cambridge, USA: MIT Press. ISBN: 978-0-262-18158-7.

Rovan, Joseph et al. (1997). "Instrumental Gestural Mapping Strategies as Expressivity Determinants in Computer Music Performance". In: *Proc. Kansei Workshop*. Genoa, Italy.

Rowe, Robert (1996). "Incrementally improving interactive music systems". In: *Contemporary Music Review* 13.2, pp. 47–62.

Schaeffer, Pierre (1966). *Treatise on Musical Objects: An Essay across Disciplines*. Trans. by Christine North and John Dack (2017). Berkeley, USA: University of California Press. ISBN: 978-0-520-96746-5.

Seeger, Charles (1958). "Prescriptive and Descriptive Music-Writing". In: *The Musical Quarterly* 44.2, pp. 184–195.

Sonami, Laetitia (n.d.). *Lady's Glove*. URL: `https://sonami.net/?page%5C_id=118` (visited on 07/13/2021).

Stapleton, Paul, Maarten van Walstijn, and Sandor Mehes (2018). "Co-Tuning Virtual-Acoustic Performance Ecosystems: observations on the development of skill and style in the study of musician-instrument relationships". In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Blacksburg, USA.

Starkes, J. L. et al. (1990). "Motor recall of experts for structured and unstructured sequences in creative modern dance". In: *Journal of Sport & Exercise Psychology* 12, pp. 317–321.

Stévance, Sophie and Serge Lacasse (2018). *Research-Creation in Music and the Arts: Towards a Collaborative Interdiscipline*. 1st. London, UK: Routledge. ISBN: 978-1-315-60557-9.

Steve Cowan official website (n.d.). URL: `https://www.stevecowanmusic.com/` (visited on 05/09/2021).

Stewart, D. Andrew (2010). “Catching Air and the Superman”. Ph.D. dissertation. Montreal, Canada: McGill University.

Stewart, D. Andrew and Joseph Malloch (2010). “Everybody to the power of one, for soprano T-Stick”. In: *Proc. International Conference on Human Factors in Computing Systems (CHI)*. New York, USA.

Sullivan, John (2021). “Built for Performance: Designing Digital Musical Instruments for Professional Use”. Ph.D. dissertation. Montreal, Canada: McGill University.

Sullivan, John and Marcelo M. Wanderley (2018). “Stability, Reliability, Compatibility: Reviewing 40 Years of DMI Design”. In: *Proc. Sound and Music Computing Conference (SMC)*. Limassol, Cyprus.

— (2019). “Surveying Digital Musical Instrument Use Across Diverse Communities of Practice”. In: *Proc. 14th International Symposium on Computer Music Multidisciplinary Research (CMMR)*. Marseille, France.

Torre, Giuseppe, Kristina Andersen, and Frank Baldé (2016). “The Hands: The Making of a Digital Musical Instrument”. In: *Computer Music Journal* 40.2, pp. 22–34.

Vallis, Owen, Jordan Hochenbaum, and Ajay Kapur (2010). “A Shift Towards Iterative and Open-Source Design for Musical Interfaces”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Sydney, Australia.

Waisvisz, Michel (1985). “The Hands, a set of remote MIDI-controllers”. In: *Proc. International Computer Music Conference (ICMC)*. Burnaby, Canada.

— (2000). “Round table: Electronic Controllers in Music Performance and Composition”. In: *Trends in Gestural Control of Music*. Paris, France: Editions IRCAM – Centre Pompidou, pp. 415–438. ISBN: 978-2-844-26039-0.

Wanderley, Marcelo M. (2017). “Expert Commentary: Perry Cook’s Principles Still Going Strong”. In: *A NIME Reader*. Cham, Switzerland: Springer International Publishing AG, pp. 11–12. ISBN: 978-3-319-47214-0.

Wanderley, Marcelo M. and Marc Battier, eds. (2000). *Trends in Gestural Control of Music*. Paris, France: IRCAM - Centre Georges Pompidou. ISBN: 978-2-844-26039-0.

Wanderley, Marcelo M. and Philippe Depalle (2001). “Gesturally-Controlled Digital Audio Effects”. In: *Proc. of the COST-6 Conference on Digital Audio Effects (DAFx-01)*. Limerick, Ireland.

Wanderley, Marcelo M. and Nicola Orio (2002). “Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI”. In: *Computer Music Journal* 26.3, pp. 62–76.

Wang, Johnty, Joseph Malloch, et al. (2019). “Webmapper: A Tool for Visualizing and Manipulating Mappings in Digital Musical Instruments”. In: *Proc. 14th International Symposium on Computer Music Multidisciplinary Research (CMMR)*. Paris, France.

Wang, Johnty, Eduardo A. L. Meneses, and Marcelo M. Wanderley (2020). “The Scalability of WiFi for Mobile Embedded Sensor Interfaces”. In: *Proc. International Conference on New Interfaces for Musical Expression (NIME)*. Birmingham, U.K.

Wenger, Etienne (1999). *Communities of Practice: Learning, Meaning, and Identity*. 1st ed. Cambridge, UK: Cambridge University Press. ISBN: 978-0-521-66363-2.

Wessel, David and Matthew Wright (2002). "Problems and Prospects for Intimate Musical Control of Computers". In: *Computer Music Journal* 26.3, pp. 11–22.

West, Travis (2020). "Making Mappings: Examining the Design Process with Libmapper and the T-Stick". MA thesis. Montreal, Canada: McGill University.

Williams, Barry M. (1990). "The Early Percussion Music of John Cage: 1935 - 1943". Ph.D. dissertation. East Lansing, USA: Michigan State University.

Zimmerman, John, Jodi Forlizzi, and Shelley Evenson (2007). "Research through Design as a Method for Interaction Design Research in HCI". In: *Proc. International Conference on Human Factors in Computing Systems (CHI)*. San Jose, USA.