Making Mappings: Examining the Design Process with Libmapper and the T-Stick

Travis J. West

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

August 2020

An thesis submitted to McGill University in partial fulfillment of the requirements for the degree of MA.

2020/08/06

Abstract

Electroacoustic musical instruments (EAMIs), those new instruments for musical expression that use an electronic transducer such as a loudspeaker to produce sound, are characterized by the seperation of control and sound. This modular separation produces the need for EAMI makers to decide how the modules that produce control data should be connected to the modules that expose control paramters. These connections form the mappings in an EAMI, which have been shown to have important influences over the success of any EAMI design.

Although many tools have been developed to help EAMI designers and users to develop effective mappings, relatively little research to date has considered the how these tools are actually used. This thesis presents an exploratory study examining how skilled music technology users make mappings. Our investigation paints a clear picture of mapping design as an iterative process of trial and error, with stages that reflect typical models of creative activities [1] such as learning, exploring, and evaluating. These stages progress in rapid cycles, gradually building up the mapping from individual connections.

We also consider the mappings participants made, and the reasons given for what makes them effective. Recognizing that such evaluations are highly subjective and dependent on the context and approach of each mapping designer, we nonetheless found a common thread that was unanimously recognized as important for effective mappings: in one way or another, effective mappings were always considered highly comprehensible. Whether this understanding came from a clear correspondence between gesture and sound, from an analogy with existing instruments and systems, or from ineffable intuition, all participants justified mappings by the ease with which they could be understood.

These results have clear implications for developers of mapping design tools, as well as EAMI designers making mappings. However, signicant future work remains to be done. In order to fully realize the potential of any mapping design tool, further inquiry must be devoted to the ways in which different design paradigms influence the design process, the diversity of creative goals with which EAMI designers approach mapping design, and the properties of different mappings which allow them to effectively meet those diverse goals.

Resumé

Les instruments de musique électroacoustiques (IMEAs), ces nouveaux instruments de musique qui utilise un transducteur électronique pour la production sonore, sont caractérisés par la séparation du contrôle et du son. Ce système modulaire entraîne la nécessité pour les conceptuers de l'IMEA de décider comment les modules de production de contrôle doivent être connectées aux modules qui exposent des paramètres de contrôle. Ces connexions forment les mappings d'un IMEA, qui ont été dont dont il a été démontré qu'ils ont une influence importante sur le succès de toute conception EAMI.

Bien que de nombreux outils aient été mis au point pour aider les concepteurs et les utilisateurs de l'IMEA à élaborer des mappings efficaces, relativement peu de recherches ont été menées à ce jour sur la manière dont ces outils sont utilisés. Cette thèse présente une étude examinant comment les utilisateurs compétents de la technologie musicale réalisent des mappings. Notre enquête brosse un tableau clair de la conception de le mapping comme un processus itératif d'essais et d'erreurs, avec des étapes qui reflètent des modèles typiques d'activités créatives [1] telles que l'apprentissage, l'exploration, et l'évaluation. Ces étapes se déroulent en cycles rapides, construisant progressivement le mapping à partir de connexions individuelles.

Nous prenons également en compte les mappings réalisées par les participants et les raisons invoquées pour les rendre efficaces. Reconnaissant que ces évaluations sont très subjectives et dépendent du contexte et de l'approche de chaque concepteur de mapping, nous avons trouvé néanmoins un fil conducteur qui a été unanimement reconnu comme important pour des mappings efficaces: les mappings efficaces ont toujours été considérées comme très compréhensibles.

Ces résultats ont des implications claires pour les développeurs d'outils de conception de mapping, ainsi que pour les concepteurs de l'IMEA qui réalisent des mappings. Toutefois, il reste encore beaucoup à faire. Afin de réaliser pleinement le potentiel de tout outil de conception de mapping, il convient d'étudier plus avant la manière dont les diffèrents paradigmes de conception influencent le processus de conception, la diversité des objectifs créatifs avec lesquels les concepteurs EAMI abordent la conception de mapping, et les propriétés des différentes mappings qui leur permettent d'atteindre efficacement ces divers objectifs.

Acknowledgements

Many thanks to my supervisors, my peers at IDMIL, and especially to wife, my parents, and my brother.

Contents

Chapter 1

Introduction

In the last hundred years, the rapid development of technology has fundamentally altered the way music is created, disseminated, and enjoyed in every tradition and practice. Compositions are written with the collaboration of artificially intelligent computer agents. Performances are amplified, mixed, modified, and recorded electronically. These recordings are stored and transmitted with unprecedented fidelity, allowing listeners to enjoy the performances of the world's most celebrated musicians wherever they are, whenever they want.

Musical instruments themselves are changing in particularly dramatic ways. Whereas acoustic sound production is inseparable from acoustic sound control, a new generation of musical instruments are characterized by the disentanglement of control and sound production. Whereas acoustic musical instruments are constrained by the laws of physics to adopt certain acoustically optimal forms and produce certain acoustically efficient sounds, the new kind of instruments are characterized by their lack of design constraints, allowing them to adopt any imaginable form and to produce any possible sound. Whereas acoustic musical instruments each have a fixed form and function, new instruments for musical expression are malleable, customizable, and inherently reprogrammable.

The development of electronic telecommunication technologies in the 1800s and the creation of the loudspeaker around the year 1900 coincided with the development of the very first of this new generation of musical instruments. A huge quantity of these instruments have emerged since then. There are analog electronic instruments, digital computer-based instruments, augmented acoustic instruments, and many other forms taken by these new instruments, which can make it difficult to describe them. All of these instruments use a transducer, typically but not always a loudspeaker, in order to convert an electrical signal into an acoustic excitation, producing sound. Recognizing this unambiguous and uncontroversial commonality between all of these new instruments for musical expression, we will call them “electroacoustic musical instruments”, or EAMIs[1]. Whether analog, digital, or electronic/acoustic hybrid, all EAMIs are faced with the challenges and opportunities given by the programmability and design freedom that accompany the separation of sound control and sound production.

Programmability means that an instrument can be optimized for a particular performer, a particular performance, or even a particular moment in a performance. But if an instrument is constantly changing, is it possible to master it? The development of virtuosic performance prowess (often considered a worthwhile pursuit) may be jeopardized by endless customization or careless design [2]. On the other hand, endless customization may allow an instrument to be configured one way for a beginner, and another for an expert [3]. The possibility of instruments with low barriers to entry and high ceilings for mastery is attractive to many designers of EAMIs [4].

Design freedom means that any imaginable musical instrument could be realized. Every aspect of the design of an EAMI falls under the influence of the designer. Every detail of look, feel, functionality, sound, and movement can be deliberately crafted to serve a particular aim. With this immense freedom and power comes an incredible burden. When every detail can be designed, the designer becomes responsible for every detail. And since everything imaginable is realizable, every decision made by the designer necessarily excludes an infinite range of other possible choices. It ultimately falls to the designer to impose constraints–to reduce the range of possible decisions–simply to overcome the paralysis of possibilities evoked by endless choice [5].

The separation of control and sound production is implied by the structure of EAMIs. The sensors, computers, sound synthesizers, and transducers that make up an EAMI can be independently swapped out and interchanged with freedom that is unheard of in acoustic musical instruments. Because EAMIs are inherently modular, their parts are usually produced as standalone products: synthesizers, controllers, and sound effects are developed by independent organizations and designers using interoperable standards, allowing these manufacturers to specialize in one kind of musical module. Because each part is distinct, the combination of these parts in a musical instrument requires the connections between parts to be defined. To some extent this separability or modularity is what leads to many of the opportunities and challenges just described. Even just a few modules can be reconfigured and reconnected in unimaginably many ways, offering the user

---

[1] Although the term “electroacoustic musical instrument” is sometimes used to refer specifically to instruments which employ a hybridisation of electronic and acoustic sound production, we will use this term to refer to all musical instruments that require an electronic transducer to make their characteristic sound. “EAMI” is chosen over other commonly used terms, such as “electronic musical instrument”, “digital musical instrument”, or sometimes simply “NIME” (after the International Conference on New Interfaces of Musical Expresion). The former two of these terms imply needless restrictions on the kind of technologies involved–analog and digital electronics respectively. The term NIME on the other hand is overly broad, potentially referring to art installations, computer programs, and other interfaces for musical expression that are not necessarily musical instruments. The common requirement for a loudspeaker seems to more adequately encompass the class of musical instruments relevant to this discussion.

of these modules a whole realm of musical instrument design possibilities.

Many of the design decisions which ultimately distinguish one EAMI from another are essentially choices about the way that commonly employed parts should be combined into a unified whole. For example, relatively few sensors are employed in the majority of EAMIs presented at the NIME conference each year [6]; sound synthesis techniques such as subtractive, FM, additive, and granular synthesis are all workhorse standards with well understood capabilities and boundaries; and the toolbox of sound effects such as reverb, distortion, and delay are equally well codified and understood. So in the scenario where a designer intends to develop an EAMI in which a gestural interface controls a sound synthesizer with some effects, they need “merely” choose from the standard sensors, synthesizers, and effects, and “simply” connect them together. Of course, deciding how signals from one module should control another is not trivial. This particular kind of connection, between the modules in an EAMI, is called a mapping. The precise meaning of this term will be elaborated in section 2.2.

The mappings in an EAMI have been demonstrated to have important effects on the overall effectiveness of the instrument. Changing nothing other than the mapping, it is possible to improve the expressivity of the instrument design [3]. There is evidence that more complex mapping designs may allow greater long term improvement and better performance on complex tasks than simpler designs [7], [8]. Mapping designs that incorporate multiple layers may allow control and sound production to be dynamically swapped out by users [2], [9]. Furthermore, the mapping is an important determinant of the perceived causality of an instrument, which has been demonstrated to play an important role in audience enjoyment of performances with EAMIs [10].

Despite its recognized importance, the mapping is very often left for end users to determine, rather than e.g. expert EAMI designers. Most EAMIs are conceived, designed, and implemented by individuals or small groups who become the sole users of their own unique instrument [11]. The mapping is necessarily designed as part of this process. This is a non-issue when the users are EAMI designer or researchers, but this is most often not the case. The average EAMI player cannot build their own EAMI from scratch, meaning they must rely on music technology manufacturers to make the parts for them. Individual EAMI players combine various ready-made modules to form an amalgam-instrument unique to their artistic practice. Designing the mapping and building the instrument become collapsed into the same activity; since all parts of the instrument are already developed except for the mapping, making this connection between the parts is what transforms them into a musical-instrument-whole.

The effect of this situation is that almost all EAMI players must participate in mapping design. This reality is reflected in the wide range of tools which are readily available to assist in mapping design. Programs like OSCulator and junXion are dedicated to this task, and programming environments such as Max/MSP and Pure Data readily support it. Projects such as libmapper and libossia are also geared towards assisting in the design of mappings, and alongside protocols such as MIDI and OSC, these tools make it easier for developers to build software which can be easily interconnected.

1.1 Thesis Overview

This thesis explores the way tools such as these are used to make mappings. By examining the mapping design process in action, we seek to better understand how this task is accomplished, the strategies users employ to achieve their goals, and how we might improve these tools to help mapping designers work more quickly and achieve their aims more easily. We present a study where participants were asked to make "an effective mapping for a live musical performance" to connect a gestural controller, the T-Stick (section 3.1.1), to a typical subtractive synthesizer (section 3.1.1). The libmapper+Webmapper environment (section 3.1.1) was used to facilitate the mapping task. Afterwards, participants were interviewed about their experience of making the mapping, and their approach to the task.

Chapter 2 provides an overview of the relevant literature on the importance of mappings and prior studies looking at how mappings are made, as well as offering some preliminary recommendations regarding the murky terminology surrounding mappings in EAMI design. Chapter 3 details the design of the experiment and its methodology. Chapter 4 discusses in depth the process of making a mapping suggested by the results of the study, while chapter 5 discusses the mappings participants made and what they considered effective. Finally, chapter 6 points to open questions and concludes the overall discussion. The experiment protocol is fully elaborated in the online appendices [2], alongside the full dataset produced by the study including the interview transcriptions and mappings produced by users. In addition, an up to date overview of libmapper is provided in appendix A.

---

[2] `http://idmil.org/making_mappings`, `http://traviswest.ca/making_mappings`

Chapter 2

Background

In 1985, Michel Waisvisz presented The Hands at the International Computer Music Conference (ICMC) [12]. Waisvisz wrote that "an important experience gained from playing with the hands was of possibilities for control signal manipulation. Rescaling, patching, and other treatments, i.e. the application of a control-signal algorithm, are the main factors through which a controlling device derives its effectiveness." In other words, effective processing of control signals, what might now be termed effective mapping, makes for an effective instrument.

In mathematics, the term "mapping" simply refers to a relationship between two sets of objects, most often in the form of a function [See e.g. 13, Ch. 2]. So to say that "A is mapped to B" suggests that every element of the set A is associated with some element in the set B. The term is used quite casually, as it implies next to nothing about the nature of the association, so it can be used to refer to almost any association. It is useful in the same way that a pronoun is useful, as a shorthand placeholder which refers to some object of discussion which is clear from context. The term "mapping" pops up frequently even in the earliest publications of the ICMC[1] and Computer Music Journal[2], but the usage is in the casual sense meaning "some association".

Some time starting around 1990, "mapping" started to acquire the particular connotation which it now bears in the context of EAMI and computer music research. In 1990, Bowler et al [16] described a technique for mapping from an N-dimensional control space to an M-dimensional parameter space, employing the term mapping in the general mathematical sense but in a musical context. Jeff Pressing [17] referred to "gestural mapping between musical effect and body motion", and Rubine and Mcavinney [18] claim to "introduce the idea that an instrument controller is both a measurer of gestural parameters and a mapper from gestural to sound control parameters", both

---

[1] E.g. "These programs can be useful in the processes of specifying and compiling composing grammars, as well as the process of mapping the syntactic structure of a composition into a lexicon of sound objects." [14]

[2] E.g. "The mapping of the tablet coordinates to screen coordinates leaves a margin area at the top of the tablet coordinate space." [15]

using the term in roughly the way it is used in contemporary discourse, emphasizing the association between gestural control signals and sound parameters. In 1995, Todd Winkler wrote “Making Motion Musical: Gesture Mapping Strategies for Interactive Computer Music” [19], writing that “physical parameters can be appropriately mapped to musical parameters”. These are just a few examples of the trend in the early 90’s towards the current usage of the term “mapping”.

By 1997, Rovan et al. wrote that “the mapping layer is a key to solving such control problems, and is an undeveloped link between gestural control and synthesis by signal models” [3]. As well as introducing the now familiar interface-mapping-synthesis diagram of an EAMI, reproduced in fig. 2.1, this paper may be recognized as a landmark in the path toward formal study of mapping design. Whereas previous work had frequently emphasized the importance of mapping, Rovan et al. were among the first to begin to formally compare mapping strategies. By this time, the special meaning of “mapping” to EAMI designers had also become clear. Rather than referring to any association between two sets, a mapping is specifically the association of control signals to sound parameters.

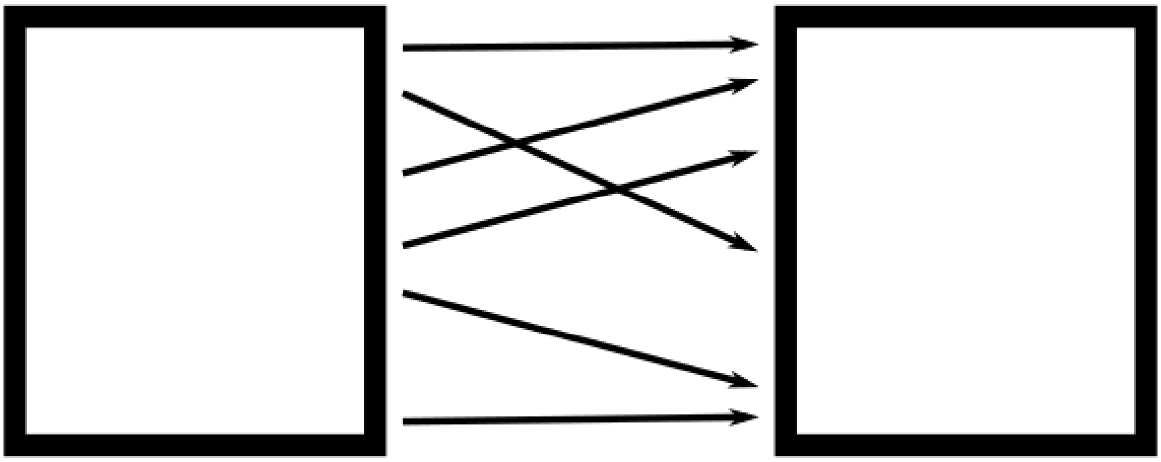


Fig. 2.1 A typical representation of a mapping as a layer of associations between two functional blocks.

Since 1997, there have been two special issues in important journals, one in the Organised Sound [20], the other in Computer Music Journal [21], which have focused on mapping in this sense, further solidifying the specific connotation of the term in EAMI research and design. In [9], Hunt et al. describe mapping as “the designed link between an instrument’s playing interface and its sound source”. In the same journal issue, Doornbusch states that “mapping concerns the connection between gestures, or structures and audible results in a musical performance or composition” [22]. These definitions give a good impression of roughly what “mapping” means to

EAMI designers and researchers, and with this in mind we will consider in section 2.1 just how important mapping is in EAMI design.

2.1 The Importance of Mapping

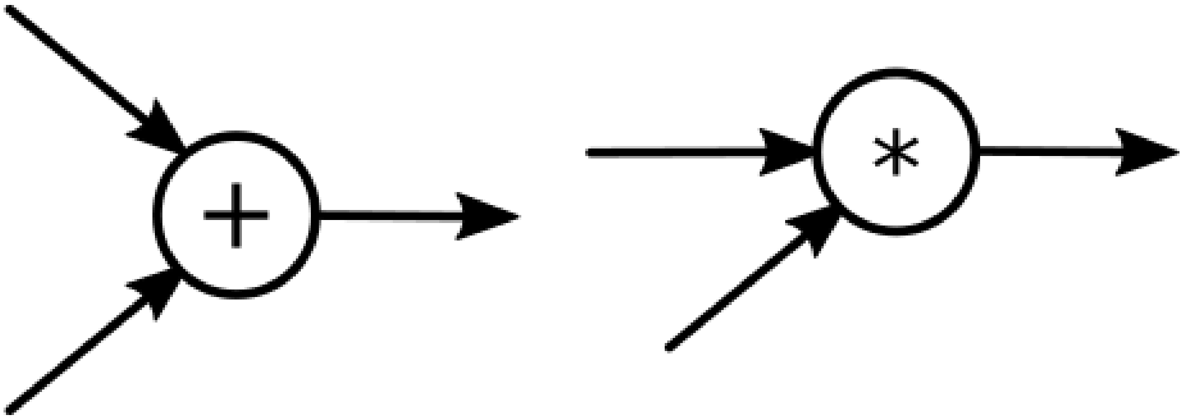
Although mapping was recognized as an important aspect of EAMI design even before the term was used in its current specialized sense and before the mapping became the focus of concerted formal study, since the 1990s numerous results have empirically demonstrated the important role mapping plays. Many of these results are summarised in [8].

In [3], Rovan et al. compared several different mapping strategies for a MIDI wind controller mapped to an additive resynthesis model of a clarinet. The mappings compared varied from a simple mapping in which breath, embouchure pressure, and fingering control amplitude, vibrato, and pitch respectively, through to a complex mapping based on the physical behavior of a vibrating reed that coupled breath and embouchure to vary timbre and loudness together. The authors introduce the terms "divergent mapping" and "convergent mapping" to describe the more complex mappings in their study. A divergent mapping is one in which a single control signal is connected to multiple control parameters. A convergent mapping is a mapping in which the effect of one control parameter is modulated by the state of another control parameter, introducing a coupling between the two[3]. Skilled performers were found to prefer the more complex mapping, whereas beginners often preferred the simpler mappings. This study demonstrated that without the addition of more sophisticated sensors or sound synthesis algorithms, merely changing the mapping can produce a positive change in the behavior and feel of an EAMI, with the potential to "provide higher levels of expressivity" [3] to the instrument.

In [7], Hunt and Kirk presented two studies comparing simple EAMIs with simple mappings to more complex EAMIs with cross-coupled mappings. They demonstrated that although participants were initially able to imitate a demonstrated sound more effectively with the simpler instrument, over time they were able to improve more and perform more accurately with the more complex instrument, especially when imitating more complex sounds. This study was limited by the fact that both the interface as well as the mapping strategies were varied across the compared instruments, making it impossible to determine the extent to which the mapping specifically influenced the results. However, this study is still often cited (e.g. by [5], [23], [24]) as providing evidence for the importance of mapping, and particularly for the usefulness of complex mappings.

---

[3] Convergent mappings are sometimes conflated with so called many-to-one mappings, in which many control signals are connected to a single control parameter in an unspecified manner. An important feature of convergent mappings, as originally described in [3] is the coupling introduced between control signals. So a mapping of the form $y = x_1 * x_2$ might be considered a convergent mapping in this sense, but $y = x_1 + x_2$ should not, since the two control signals x_1 and x_2 have independent influence over the control parameter y



(a) many-to-one association (b) convergent association

Fig. 2.2 Comparison of a many-to-one association and a convergent association as per the meaning in [3].

These results are summarised in [8] along with a third study by Matthew Paradis which aimed to replicate some of the results of [7] while varying only the mapping. In this study, participants were asked to explore three different instruments using the same interface and synthesis algorithm, but in which the mapping was varied from very simple to very complex. This study was limited in a few ways: the study was conducted in a single session with each participant, making it impossible to determine long term effects such as the long term improvement seen in [7]; quantitative results were also not reported, although it is stated that quantitative data was collected. The results were nevertheless interesting. Participants were found to reach the point where they felt they could drive the simple mapping after a few minutes, whereas the more complex mappings allowed the participants to engage longer and improve more over the course of the experiment. "The final test presented much more of a challenge to the users, with some of them spending several minutes just trying to make it produce sound. As time progressed the users became more and more immersed in what they were doing musically and forgot about the technical aspects of the experiment." This suggests that more complex mappings may provide greater room for improvement and deep engagement.

Taken together, these studies clearly demonstrate that changes to the mapping can have important impacts on the qualities of an EAMI. The results seem to suggest that more complex mappings, e.g. including convergent associations, may be especially useful by creating a higher ceiling for mastery and allowing performers to develop expertise playing the instrument, however this relationship has still not been unambiguously demonstrated. Further inquiry, and especially longitudinal studies, will be required for us to fully understand the role of complexity in mapping design, and to learn how other aspects of the mapping can be tailored to achieve a certain end result.

Mapping has also been explored in the context of sound design. In [25], Tubb and Dixon introduce a mapping technique for extreme dimensionality reduction using hilbert curves to reduce an N-dimensional space of synthesis parameters to a 2-dimensional zoomable map of all possible configurations of the parameters. They compare the use of this technique to more traditional sound design where multiple sliders each control an individual parameter. The dimensionality reduction technique was found to be especially useful for exploring new sounds which may not have otherwise been discovered. Tubb and Dixon's study suggests that mappings are important in EAMIs which may not necessarily be used for real time performance. Studies examining mappings in EAMIs which are played in real time have provided some suggestions of the kinds of mappings which are most effective; it is unknown to what extent these results generalize to EAMIs which are not played in real time, such as a synthesizer being used for sound design[4].

Regardless of the context, it seems clear that the mapping is of crucial importance to any EAMI design. In my personal experience developing EAMIs, I have consistently found that the mapping is the most difficult and impactful part of the design process. Whereas the choice of sensors and sound production are clearly governed by the goals of the design and can often be selected from off the shelf solutions that "just work", the mapping is more ambiguous and requires significantly greater refinement and experimentation. This is nothing new; it has been over 40 years since Waisvisz recognized mapping processes as "the main factors through which a controlling device derives its effectiveness." Ongoing research into EAMI mappings aims to provide clearer guidelines to help designers achieve their goals through effective mapping design, and to provide tools which will make effective mapping design more accessible to all designers.

2.2 Terminology

Joel Chadabe's well known criticism of mapping as a conceptual model [27] argues that the highly complex networked systems found in interactive instruments are not appropriately modelled in terms of mappings. Chadabe describes interactive instruments in which analysers and autonomous algorithms generate control signals that collaboratively influence the sound produced by the instrument in tandem with control directly by the performer. "Controls for the music and sound variables

---

[4] EAMIs are able to offload much of the cognitive workload traditionally associated with performance [26], blurring the line between instrumental performance and musical activities such as sound design, composition, and other "offline" activities that are not subject to the hard real time constraints of performance. The tools used for these activities have increasingly more in common with EAMIs than they do not, including the separation of control and sound production and requirement for mapping design. As such these tools may be considered as EAMIs even if they are not used for real time performance.

originate simultaneously and asynchronously in different contexts and from different sources; and because the relative importance of each control may vary according to function and in time, the cause-effect relationships within such an instrument, too complex to be usefully described in terms of mapping, are most usefully described as a network.” The implication is that, because some of the control signals in the network are not produced by the performer’s (or performers’) gestures, and because the influence of these control signals changes dynamically, the concept of mappings is of limited use.

Consider however that regardless of the source of a control signal, whether human performer or complex network of autonomous digital agents, it is necessary to define associations from control signals to control parameters in order for the signals to have influence. Thus, even if a control signal is not generated by a performer’s gesture, the instrument designer must still determine a mapping. Perhaps the control signals (regardless of how they are generated) are not connected to sound parameters, but to the parameters of some other process, such as another control signal generator, visual or vibrotactile feedback, or other kinds of actuators. Even the most byzantine network in the most complex imaginable interactive instrument is necessarily full of many associations from sources of information to control parameters. The concept of mappings is therefore crucial even in the complex instruments described by Chadabe; Although mappings may not be sufficient to describe the entire architecture of the system in the way they can be used when describing more deterministic and hierarchical instruments, networked indeterministic interactive instruments still require mappings, and it may be possible to improve such systems by employing more effective mapping strategies.

Chadabe’s criticism is reasonable if the concept of mapping is understood as a high level structural descriptive wherein control is associated to sound parameter, as in fig. 2.1; from this perspective, the mapping appears as a unidirectional flow of information from control source to sound generator. The mapping is treated as a single system which is separable from the source of control and the source of sound which is controlled. In Chadabe’s interactive instruments, this organizational scheme is inadequate, because the numerous sources of control are interconnected in such a complex manner that *the* mapping cannot be readily identified–there isn’t a single monolithic mapping stage which can be recognized. If mapping is instead viewed as a lower-level structural descriptive, then Chadabe’s criticism is called into question, as we have seen. Mappings in this lower-level sense exist even in the most complex network of autonomous agents (fig. 2.3).

I argue that the latter sense of mapping is more flexible and that it better reflects the applicability of mapping research than the high-level version. Indeed, the low-level view of mappings includes the high-level view of mappings as a special case. With the low-level view, we can easily recognize that techniques and guidelines developed for designing mappings can be readily applied in the context of EAMIs of every kind, whether control flows unidirectionally from a performer’s

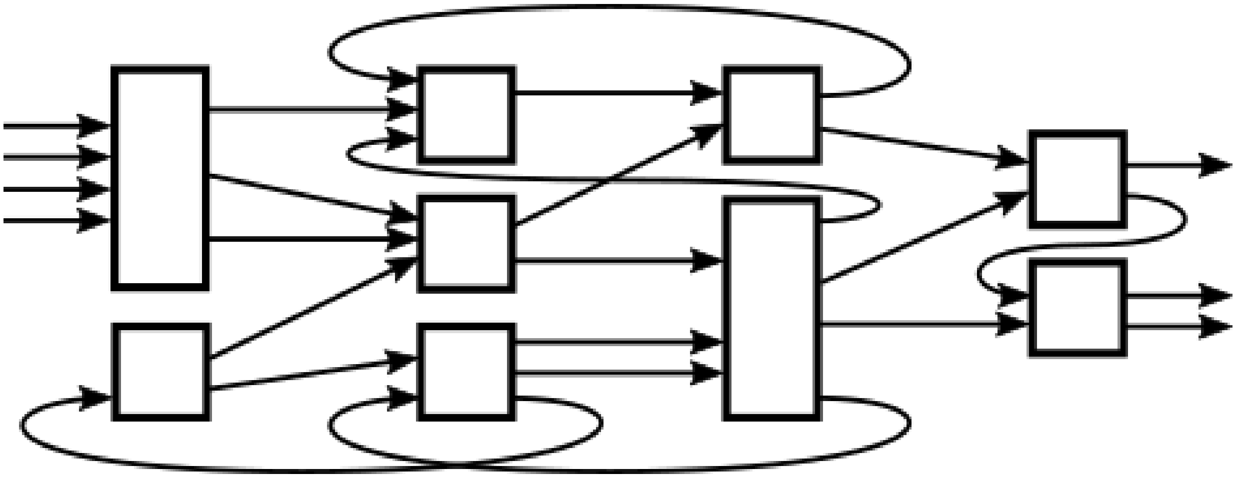


Fig. 2.3 An interactive instrument such as described by Chadabe [27], where numerous autonomous devices generate and are dynamically influenced by various signals in a complex network. If mapping is viewed as a high level structural descriptive as in fig. 2.1 it appears incompatible with this kind of instrument. If instead mapping is viewed as a lower-level structure, several independent mappings can still be identified even in the complex network of an interactive instrument.

gesture to sound parameters, or in a byzantine network of autonomous generators. Every mapping can potentially leverage the whole body of technique and strategy developed by EAMI research.

Given a low-level view of mappings, it is possible to distinguish between numerous types of mappings based on the nature of their origin and destination. If a signal is generated by a performer's gesture and used to control the parameter of a sound synthesizer or processor, the mapping from one to the other could be called a gesture-to-sound mapping; this is the kind of mapping on which much of the research to date has focused. More generally, control-to-sound mappings may refer to mappings in which the source signal is not necessarily gestural, such as an envelope generator, a sequencer, or a random number generator. Mappings with multiple layers have been suggested by numerous authors [2], [28], [29], including mappings from control parameters to perceptual parameters related to control, from sound perception related parameters to sound synthesis parameters, and from control-perceptual to sound-perceptual parameters.

One might reasonably wonder whether it is useful to identify and name every slight variation in mappings. There is a risk that the concept of mappings may become so generic that it ceases to be distinct from the more general mappings of mathematics. Why should we not consider every connection, computation, and transformation as a mapping? The performer has a mental representation of their instrument that is mapped to their motor control system causing their muscles to contract and expand; this results in movements and forces that are mapped through the physical linkages of the body and the instrument to actuate electronic sensors; the sensors determine a mapping from the physical quantities related to force and movement to varying electrical signals, which might then be mapped to digital signals through an analog-to-digital conversion; and so on, until eventually an electrical signal is sent to the voice coil of a loudspeaker where it is finally mapped to an acoustic wave; the acoustic wave is then mapped from the air to the eardrum and through the acoustic channels in the ear; the cilia in the inner ear then map the acoustic signal to a neurological signal that travels into the brain of the listener where it is ultimately mapped into some kind of meaningful percept[5].

All of the stages just described can reasonably be considered as mappings in the general mathematical sense, and indeed they must all be considered in the course of any EAMI design, but the concept of mapping identified in section 2.1 is noticeably inapplicable to the stages just described. Because of the flexibility and ambiguity of the term mapping, I argue that it is necessary to be sensitive to the many different mappings which exist in and around an EAMI so that we remain aware of the boundaries between the specific mappings that are studied by EAMI designers and those mappings which fall outside the scope of this research fig. 2.4.

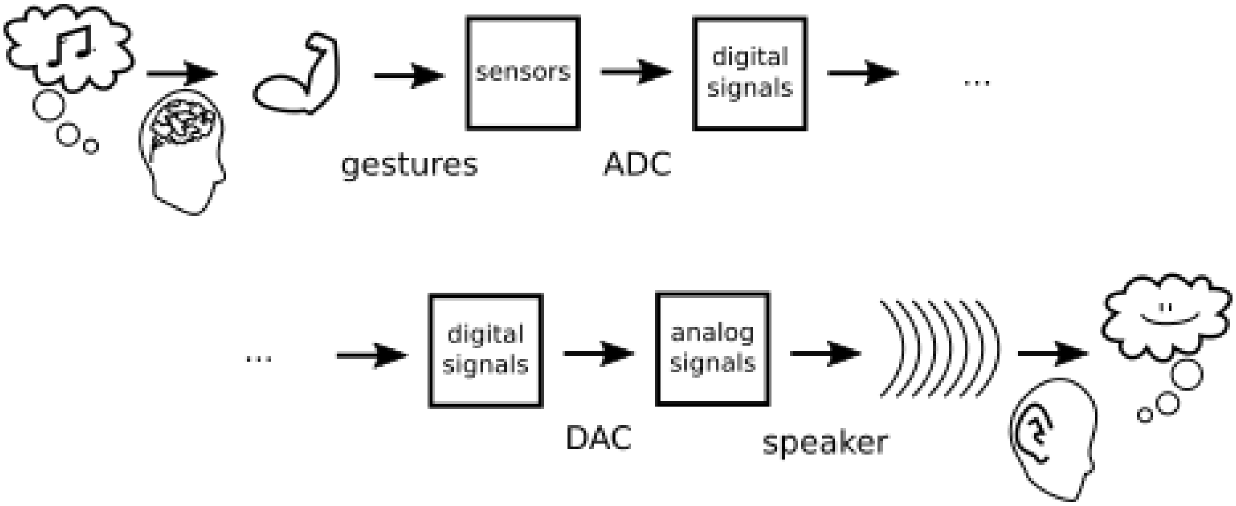


Fig. 2.4 Various "mappings" in the general mathematical sense exist in an around an EAMI; many, such as those depicted here, fall outside of the focus of research into mappings in EAMIs. Stages such as sensor fusion, gesture signal extraction, and raw-to-perceptual parameter extraction may also be considered as distinct from "mappings" in the specialized sense.

As well as the general sort of mappings which take place outside the boundaries of an EAMI (e.g. before any control signal is produced, by gesture or algorithm, and after an acoustic wave is produced by a loudspeaker), there are also certain mappings within EAMIs which should possibly

---

[5] In case the audience has some agency or influence over the performer, we could continue ad nauseum as the audience's reaction to the performance is cyclically mapped onto the performer's next action, and so on.

be excluded from the our focus and rather considered as part of signal conditioning, synthesis algorithm design, or another part of EAMI design distinct from mapping. The programs or circuits which produce audio signals, for example, may rightfully be considered as implementing mappings from sound parameter to sound signal. Similarly a sensor fusion algorithm used to combine multiple sensor measurements into a single more robust or more meaningful measurement [6] may also be considered as a kind of mapping. However, these processes (whether algorithmic or electronic) are selected and evaluated by criteria which are distinct and unrelated to those used to select and evaluate effective mapping strategies. Sound synthesizers and processors are chosen based on the kinds of sounds which the designer wishes to produce, and they are evaluated by their ability to produce these sounds. Sensor conditioning and fusion processes are selected based on the sensors used, and they can be evaluated by how effectively they improve the accuracy, precision, repeatability, and so on of the sensor system. Furthermore, both of these systems can be developed and evaluated apart from the context of EAMI design entirely. Synthesis techniques are routinely developed for use in telecommunications, audio coding, and signal processing, and only later find use in making music. The majority of sensor design and development is done by engineers outside the context of sound, let alone music making. Although, the same phenomenon can also be seen when mapping techniques are developed which appropriate algorithms from domains outside music [30], the point is to recognize that many of the parts of EAMIs are areas of study in their own right; the study of mapping is distinct from the study of sound synthesis or sensor fusion, and so I argue that it is unhelpful to consider sound synthesis or sensor fusion processes as part of mapping.

We may go even further: the layered approach to mapping design suggested by numerous authors often implies so-called mapping layers which are closely associated with a specific sound synthesizer or gestural interface. High-level or related-to-perception parameters can be extracted from raw sensor data or abstracted from underlying synthesis parameters. I would argue that these particular kinds of associations, especially when the perceptual-level parameters are not chosen arbitrarily but based on perceptual measurements or other measures of robustness, may be better recognized as distinct from mapping. Similar to sensor fusion, gesture feature extraction can be studied without regard for music making, and such feature extraction techniques are likely to be specific to a particular set and/or arrangement of sensors. Robust and effective high-level synthesis parameters may also be better understood as part of the synthesis algorithm itself, rather than part of the mapping. Making this distinction sheds light on why certain members of the NIME community have expressed that mapping is not important at all, usually while working with physical modelling synthesis. In this particular case, they are right to be suspicious: there is nearly no mapping to speak of in an EAMI where the synthesizer is controlled by physical parameters and the interface generates exactly such physical parameters. Instead, all of the interesting work happens in the synthesis model and sensor signal conditioning stages, and the mapping itself practically disappears from consideration.

2.2.1 Recommendations

Let us discuss the limitations in existing terminology used in the study of mappings, and the conventions that we will adopt for the remainder of this thesis. The problems generally stem from overuse of the word mapping, and from the vagueness of certain other commonly used terms. "A mapping", "mappings", and "*the* mapping" as nouns, and also potentially "map", "mapped", and "mapping" as verbs, "Mapping" the proper-noun concept or field of research, and "mapping" the generic term used in mathematics–it's all a bit muddled. Ambiguous terms such as "input" and "output" further muddy the situation. I will offer concrete recommendations to try to clear the terminological waters; it should be noted that none of these recommendations are original per se, but rather serve to highlight existing terminological practices that are especially effective with the hope of encouraging more widespread adoption of these conventions.

I recommend to understand "a mapping" as an identifiable signal processing stage in an EAMI, such as the form seen in fig. 2.1, but also the identifiable links in a complex network such as described by Chadabe; mappings are distinct from other modules, such as the interface and the synthesizer, autonomous digital agents and vibrotactile transducers, and other control sources and destinations. What sets a mapping apart from these other modules is two characteristics that mappings alone possess: a mapping is interstitial, an amorphous web situated between other distinct modules, and a mapping is arbitrary in the sense that there are no robust and generalizable metrics which can be used to evaluate a mapping other than its raw efficacy for a given purpose [31]. This is in contrast to (for example) signal conditioning, gesture feature extraction, synthesis parameters, and synthesis algorithms, which may be evaluated according to clear criteria (accuracy, perceptual uniformity, ability to produce life-like and novel sounds, and so on).

The mapping is a stage in a particular instrument, or model of instruments. *A* mapping, or mappings, are instances of this stage, sometimes in a particular instrument, sometimes in general placeholder. This usage is the same as "the interface" compared to "an interface" or "interfaces", or "the synthesizer" compared to "a synthesizer" or "synthesizers". Therefore, in Chadabe's interactive instruments, the mapping cannot be identified because there are many mappings, whereas in a gestural EAMI the mapping might be easy to distinguish from the other parts of the instrument, as there may only be one mapping stage. I would recommend to reserve "mapping", without a determiner, to refer to the broader field of research and practice around mappings, analogous to "sound synthesis" or "signal conditioning".

"Input" and "output" are relative terms that necessarily change depending on the perspective of the speaker. For example, from the perspective of the interface or synthesizer, control signals are outputs from the interface and inputs to control the synthesis parameters, but from the perspective of the mapping stage, the control signals from the interface are inputs to the mapping and the synthesis parameters are outputs. Any given signal or parameter can be considered as both an input and an output, depending on your point of view. This is not to say "input" and "output" should not be used at all; they are sometimes useful when the perspective to be adopted is clear from context. When the perspective is not clear from context, the terms "source" and "destination" along with "signal" and "parameter" can resolve any ambiguity. In the above examples, the interface has signals (interface outputs), the synthesizer has parameters (synthesizer inputs), and the mapping has sources (mapping inputs) and destinations (mapping outputs).

Many of the signals and parameters in an EAMI will be control signals and control parameters; this designation implies that the parameters control some underlying process, and that the signals in question are used to influence these control parameters[6]. When a module is a signal processor, such as a low pass filter, some of its destination parameters will be for the signals to be processed. These destinations are in contrast with control parameters; the signals connected to them will not have influence, but rather be influenced. These destinations can most readily be identified as "input" parameters. Here the use of "input" is explicitly relative to the signal processor, which is hopefully made clear by the use of "input" as an adjective which modifies "parameter", rather than as a noun, which would be ambiguous as just discussed. By avoiding use of the terms "input" and "output" from the mapping-centric perspective, potential ambiguity can be further avoided. In general, "input" and "output" should be used only when necessary, and preferably as adjectives modifying unambiguous terms such as "signal", "parameter", "source", and "destination".

A link from source to parameter has often been called "a map", or even "a mapping", and making these links has been described as "mapping" using the now thoroughly overburdened word as both a noun that refers to several independent concepts and as verb. This situation can be resolved by distinguishing between an individual "association" or "connection" and "a mapping" or "mappings", which include many associations. Use of "association" or "connection" to refer to individual links within a mapping naturally leads to the use of the verbs "to associate" or "to connect" instead of "to map". "A mapping" always refers to an identifiable mapping stage (of which there may be many, as in Chadabe's networked interactive instruments), while "a connection" refers to an individual link, signal to parameter, source to destination, within a larger mapping. More generally, a connection may have multiple sources and/or destinations, such as when using an N-M functional mapping strategy [e.g. 30], [32]; although the terms "connection" and "association" may be used interchangeably, "connection" may be preferred when adopting a systems view of

---

[6] The term "influence" is recommended over "control" to avoid overloading the use of "control", and also because "control" implies significant influence, which is not always the case.

mappings (section 2.3.1, [33]), whereas “association” may be used to imply a functional view of mappings.

By adopting “association” or “connection” into our vocabulary, “a/the mapping(s)” can be used exclusively for identifiable mapping stages and singular “mapping” without a determiner for the field at large. When “mapping” might be used in the mathematical sense, one could instead consider using a more precise term such as “bijection”, “surjection”, or “injection” [13, section 2.1.2], or alternative terms which are equally as flexible as “mapping” such as “function”, “transformation”, or “morphism”.

This distinction between connections and mappings requires a minor change to the group of keywords in the literature related to the cardinality of mapping associations. “Convergent”, “divergent”, “one-to-one”, “one-to-many”, “many-to-one”, and “many-to-many” are terms that are usually applied to modify “a mapping” as in “a convergent mapping.” However, if “a mapping” is taken to refer to an identifiable group of associations or connections, these terms should be modified to apply to associations or connections, as in “a convergent association” or “a one-to-one connection.” This is more useful than retaining the terms as modifying “a mapping” given the more specific recommended usage of this term, since almost any reasonable mapping in an EAMI (e.g. every mapping produced by participants in the study) is likely to be “a many-to-many mapping”; these terms, applied to mappings as collections of connections, are simply not that useful. In contrast, they retain their intended function and utility when describing individual associations between signals and parameters.

At this point we have a reasonably full picture of mapping and where its boundaries lie. Mappings contain the associations of signals from sources to destinations. Sources may be driven by human gestures, or they may be autonomous algorithmic or electronic processes. The destination parameters may be those of sound synthesizers, sound processors, or even other control sources. Individual associations, and thereby the mappings they make up, may introduce arbitrary transformations of the signals which flow through them, i.e. they may be passive connections or active signal processors. Signal processors themselves may also be parameterized, meaning control signals may be connected to mapping processes (which has been termed convergent mappings). An EAMI may be described with a single mapping (“the mapping”), or it may better be modelled as a network of parameterized control sources and processors with numerous mappings forming complex interrelationships between networked nodes.

Mappings appear in most EAMIs, which generally exhibit some separation between control source and sound production. This separation must be closed by a mapping. Consequently, techniques and strategies for designing effective mappings are applicable in most EAMI designs, whether they are better modelled by e.g. fig. 2.1, Chadabe’s notion of the network in an interactive instrument, or any other structural model. The very few EAMIs without mappings are those in which acoustic and electronic signals are fully and inherently coupled in the design of the instrument, such as an electronic guitar played directly through an amp and loudspeaker without any parameterized processing, or in a digital instrument which employs physical modelling in both its interface and synthesis algorithm. For most other cases, it's important to consider the mapping. The terminological recommendations in this section, summarised in fig. 2.5 and table 2.1, may help designers and researchers to more clearly communicate these considerations.

2.2.2 A Note About Efficacy

I have argued in section 2.2 that mappings cannot be evaluated based on robust generalizable criteria, and are instead compared by their overall efficacy in achieving a particular goal. This position also informs the design of the experiment presented in the rest of this thesis, wherein participants were asked to make a mapping that they would consider effective for live performance. This wording was chosen with the intention that participants would bring their own unique creative goals to the experiment; in other words, "effective" is meant to imply *nothing* about how the mapping should be designed. This is important because, as mentioned in section 2.1, it is still not fully known what properties make a given mapping work for a given goal. Indeed, it is not even well documented what creative goals are common or likely amongst music technology users and designers. So rather than ask participants to make a complex mapping, or an expressive mapping, they were asked to make an effective mapping with the intention of observing to some extent what the participants' unbiased creative goals might be, and what properties of the mapping they would prioritize based on those goals.

2.3 Making Mappings

For EAMI designers and players to achieve their creative goals, they must design effective mappings. To do so, it might be helpful for the designer to use effective tools. To this end, much of the study of mappings has focused on the development of novel techniques and tools to facilitate the design of mappings. Relatively little examination has considered the design process itself. In this section we will consider some of the tools developed for making mappings, and what the existing literature suggests about this process.

2.3.1 Mapping Design Tools

Many tools for mapping design have been presented in the literature. These tools have been broadly categorized as *generative* or *implicit*, and *explicit* [2], [9], [34]. Generative mapping tools are characterized by an optimization, training, learning, or other generative stage which determines the final configuration of the mapping based on user specifications. Most commonly, these specifications take the form of demonstrated signal-parameter associations, i.e. when I do this particular thing with the control signals, I want the control parameters to respond in this particular way. These demonstrations are often thought of as a kind of preset, analogous to the preset patches in a keyboard synthesizer or virtual instrument plugin. In generative systems, it is difficult or impossible to make sense of the low-level associations between control signals and destination parameters, because these associations are determined by a complex "black-box" algorithm based on the user's inputs. The demonstrations are used to design the association, but the association itself cannot be directly edited, because it is not known explicitly and may even be unknowable; the actual associations are a result of the adaptation of the system to the users inputs. Neural networks are commonly cited as an example of a generative tool used for mapping design.

In contrast, in explicit mapping tools the signal associations are determined directly. The system is configured explicitly by the user, and the way each signal influences each parameter can be directly calculated based on the configuration of the system alone, without use of an optimization or adaptation algorithm. One of the most common mapping tools, the patch bay (whether physical or virtual), is a tangible example of an explicit mapping tool.

Mappings themselves can be viewed from different perspectives [33], which can be helpful as well when discussing mapping tools. A systems point of view considers the connections between control signals and destination parameters, emphasizing individual associations with a flow-chart or graph-like mental model; this is the perspective which best characterizes most of the present discussion thus far. A functional perspective instead emphasizes the associations between sets of variables, reducing the emphasis on individual signals. A functional view of mappings focuses on the dimensionality and other mathematical properties of the input and output sets, the functions which map from one to the other, and the topology of the resulting transformation. "These two viewpoints act as duals of one another. Shifting between perspectives helps one to have a complete picture of a performance system. In some instances, the views are vastly different and give unique insights into design possibilities" [33]. A third perspective is the perceptual view on mappings [35]. This perspective emphasizes the association between the performers action and the resulting audio-gestural percept experienced by someone participating in the performance, whether performer or audience member. A perceptual view considers the look and feel and the sonic result of a performance, and the role which mappings play in mediating them.

The delineations which have been previously made in the literature consider mapping tools in terms of the structure of their implementation, which perhaps reflects the perspective of the mapping tool maker, rather than the mapping designer. Another important view of mappings and mapping tools, especially when considering them from the designer's perspective, is the way the tools are used to make the mappings.

In digital audio workstation and synthesis plugins, mappings are typically determined by creating direct associations and tweaking parameters such as scaling and offset to adjust the transfer function of the association; this is a direct result of the lineage of the virtual studio, which emerged from analog electronic contexts in which scaling and offset are implemented with relatively simple circuits. In a programming environment such as Supercollider, Max/MSP, or Pure Data, mappings are often designed by hand by specifying the associations and transfer functions so that control signals have the desired influence over their destination parameters. The same approach is taken in libmapper+Webmapper (appendix A), which is structured around connections which can transform and combine signals based on hand-written equations.

These tools could be considered as a group which is a subset of explicit mapping tools, characterized by the emphasis on individual connections and hand-written or manually tuned transformations. In these tools, the output associated with a given input is not represented within the system. Instead, it emerges as a result of the configuration of the system. These tools will be called systems-oriented mapping tools, since their emphasis on connections, individual signals and parameters, and manually tuned signal conditioning is strongly associated with a systems-level view of mappings.

On the other hand, many of the mapping tools presented in the literature instead focus on the direct association of certain input states to certain output states. These tools often work based on the principle that the mapping designer will create presets of certain states in the input signal space along with corresponding snapshots of the output parameter space. Each preset defines an association between a point in control signal space and a point in destination parameter space which the control point should be associated with. The mapping tool then provides a means to interpolate between the demonstrated associations. This may be accomplished through a mathematical formula, making the tool an explicit mapping tool, or through the adaptation of the internal representation of the mapping e.g. through learning or optimization, making the tool an implicit mapping tool. Regardless of the underlying method by which the mapping is determined, these tools emphasize mapping design by demonstration. The designers using these tools need not concern themselves with individual associations or the operations which influence individual signals. Instead, the designer directly represents the input-output association they wish to achieve, and the lower level details are sorted out automatically. These tools will be described as function-oriented mapping tools, since their emphasis on input and output sets is strongly associated with a functional view on mappings. They could also be called demonstration-oriented tools, since they require the user to supply demonstrations of the kind of mapping they wish to achieve.

Systems-oriented and demonstration-oriented mapping tools offer significantly different ways of working to mapping designers. Although many tools have been presented in the literature and are regularly used by music makers, relatively little research specifically considers the influence these different working modes have on the design process. What are the strengths and weaknesses of these different mapping design approaches? Is one approach or the other more suitable for certain parts of the design process? Does the way a designer works ultimately change the kinds of mappings the designer will make? We begin to shed light on some of these questions in chapter 4, where we consider the creative process of mapping designers using a systems-oriented mapping tool in an open design task.

2.3.2 Mapping Design Processes

This thesis is, to my knowledge, one of the first works to specifically examine the creative process of music makers designing mappings. The study presented in chapters 3 to 5 builds on a prior study conducted by visiting researcher Benjamin Muzart in 2017. For context, this section will summarize the design and results of Muzart's study based on an unpublished internal report shared with me by the author [36].

Muzart's study aimed to observe the design process of skilled EAMI designers tasked with making "a complex and expressive mapping" between a particular gestural control interface and synthesis module. Six participants with at least one year experience designing EAMIs were recruited for the study. They were given a T-Stick and a modal synthesizer and asked to create a mapping for the two using libmapper+Webmapper. Participants were given 20 minutes to design a mapping, after which they were asked to explain their design and interviewed about the design process.

Since the T-Stick and libmapper were both used again in the present study, their complete description is reserved for chapter 3. However, as in the present study, in Muzart's study the raw signals from the T-Stick's sensors (a pressure sensor, a bank of capacitive touch strips, and a 3-axis accelerometer) were conditioned before being presented to the study participants. Five gestural signals were extracted from the raw data based on mapping strategies previously designed by Joseph Malloch [e.g. 37] and D. Andrew Stewart [e.g. 38]: kick, brush, move, shake, and squeeze. In addition, the three raw signals from the accelerometer were also made available. Kick represented the intensity of any sudden impulsive movement of the T-Stick. Brush represented the movement of the hand along the capacitive touch sensor. Move was related to the global quantity of movement, calculated from the average of the three accelerometer signals. Shake represented a related signal to move, but high pass filtered so that it corresponded with rapid movements only. Finally, squeeze represented the activation of the pressure sensor.

The synthesizer used was a modal synthesizer designed by Joseph Malloch and Stephen Sinclair with eight sound parameters: excitation, damping, attack time, reverb amount, reverb duration, global bandpass center frequency, and two timbre-related parameters. Damping and attack time were used to control the duration of the overall amplitude envelope of the sound, acting as smoothing time constants to the excitation parameter which was used to set the current overall amplitude. The two reverb parameters acted in the way suggested by their labelling. The overall sound was filtered with a bandpass filter using a fixed Q factor, center frequency being set by the associated parameter. Finally, the timbre parameters were used to interpolate between four pre-determined spectral presets which would determine the frequency, decay rate, and amplitude of the individual modes of a simulated resonating body.

Considering the process of designing a mapping, Muzart's study found several interesting results. Generally, participants had a tendency to commence the experiment by exploring the gestural signals produced by the T-Stick, without making any mappings. This was followed by a phase in which participants tended to make one-to-one connections in isolation, one after the other, in order to explore the effect of each sound parameter. Muzart related these preparatory learning stages to an analytic frame of mind, as described in [7].

Following the learning stage, participants tended to switch modes, beginning to combine gesture-sound mappings so that many gestures would influence many sound parameters simultaneously. Muzart termed this as aesthetic exploration, proposing that the combination of many connections was more similar to performance, involving a holistic frame of mind [7] which is more creative than the analytic learning phase. This claim was partially motivated by the fact that, as many synthesis parameters become modulated simultaneously, it would become increasingly difficult to discern the individual effect of each sound parameter due to the nature of the synthesis module employed. This implies that participants must have been acting based on a more aesthetic holistic approach, since analytic evaluation of the mapping would become difficult under these circumstances.

Muzart's study was subject to notable limitations. Many of the results regarding the actual mappings made by participants and their effectiveness were noticeably influenced by the combination of synthesizer and control signal processing. For example, several control signals were strongly preferred by participants over others that were found to be erratic and difficult to utilize. The choice of a physical-modelling based synthesizer may have also influenced the types of mappings favored by participants, which tended to utilize energy-injecting gestures to generate excitation, similar to how an acoustic instrument would operate. The design period of 20 minutes was also noted by many participants as very limiting.

2.3.3 The Creative Process

Although relatively little research has considered the process of designing mappings, there has been considerable investigation into creative process in diverse contexts and from many disciplinary approaches. R. Keith Sawyer [1] provides a comprehensive overview of the field of creativity research. Most relevant to this thesis, Sawyer describes a unified conceptual framework which models the creative process. Sawyer's framework consists of eight stages:

1. Find and formulate the problem.
2. Acquire knowledge relevant to the problem
3. Gather a broad range of potentially related information.
4. Take time off for incubation.
5. Generate a large variety of ideas.
6. Combine ideas in unexpected ways.
7. Select the best ideas, applying relevant criteria.
8. Externalize the idea using materials and representations.

Sawyer's framework is useful for describing various creative processes, and in chapter 4 we will see that many of its stages can be seen in the process of making a mapping.

2.4 Conclusion

This chapter has reviewed the development of mapping as an area within the broader research into EAMI design, the importance of mappings and mapping research, and the existing research that specifically considers the mapping design process. In addition to this review, several terminological recommendations were made, summarised in table 2.1. Finally, two groups of mapping tools were pointed out, highlighting the mapping designer's perspective when using mapping design tools: systems-oriented mapping tools emphasize individual connections, whereas demonstration-oriented tools are more focused on specifying signal-to-parameter relationships directly. These designations complement the more commonly recognized implicit and explicit categories of mapping tools.

Mappings are ubiquitous across EAMI design. By recognizing the far-reaching applicability mapping, we are well equipped to recognize the potential impact of this research area. By also acknowledging the boundaries of this topic of research, we can better understand its relationship to the ecosystem of interconnected technologies which form an EAMI. Mappings are the glue that hold together control sources and the processes in their influence. Interfaces and synthesizers are elevated to become instruments when they are bound together by effective mappings. The parts are made into a whole, and their final character and capability are ultimately determined as much by mappings as any other part of the system. The importance of mappings cannot be understated.

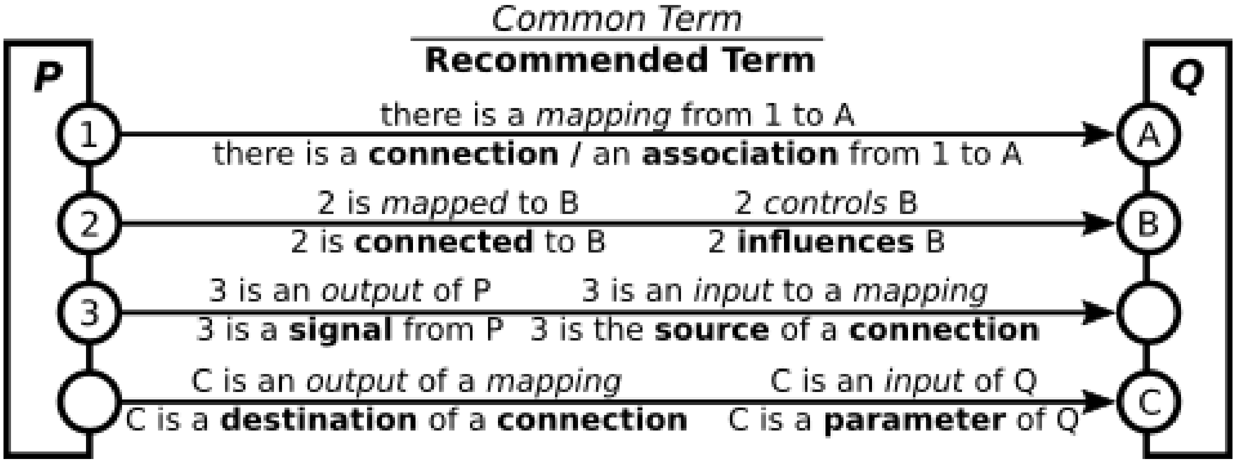


Fig. 2.5 A diagram of a mapping from device P to device Q contrasting common terms with those recommended in section 2.2.1.

Concept	Recommended	Not recommended
The topic or field of research that concerns the design, evaluation, and modelling of mappings	mapping (singular, no determiner)	
An identifiable signal routing and/or processing stage situated between a control source or sources and a control destination or destinations, composed of one or more individual associations, evaluated mainly by its efficacy for a given purpose	the/a mapping(s)	the/a map(s)
An individual link between a control signal and a control parameter within a mapping	association, connection	mapping, map
The action of creating a connection	associate, connect	map
Control information output by a module	control signal	input, output
A free variable which modifies the behavior of a module	control parameter	input, output
A module that outputs control signals	control source	input, output

A module with control parameters	control destination	input, output
The conceptual point at which an association originates	signal, source	input, output
The conceptual point at which an association ends	parameter, destination	input, output
A destination on a control signal processor for one of the signals that it will process	input parameter	input, output
A destination on a control signal processor which influences the behavior of its processing	control parameter	input, output
The effect of a signal on the parameters it is connected to	influence	control
Use of a control signal or signals to influence parameters of a control signal processor such that the influence of one control signal is affected by another	convergent association/connection	convergent/many-to-one mapping
Use of a single signal to influence multiple parameters	divergent/one-to-many association/connection	divergent/one-to-many mapping
Use of two or more signals to influence a single parameter	many-to-one association/connection	many-to-one mapping
A tool in which mappings are generated by an optimization / learning algorithm operating as a black box	generative/implicit mapping tool	
A tool in which mappings are determined by an analytically derived deterministic approach that can usually be directly inspected by the user	explicit mapping tool	
A perspective which emphasizes individual connections with a flow-chart or graph-like mental model	systems point of view (POV)	
A perspective which emphasizes the topology of the transformation from a set of signals to a set of parameters	functional POV	
A perspective which emphasizes the audio-gestural percept resulting from the mapping as experienced by the performer and audience	perceptual POV	

A tool in which the user specifies associations and signal processors directly	systems-oriented
A tool in which the user specifies the mapping mainly or exclusively by providing demonstrations of the desired output parameters for a given set of input signals	functional, demonstration-oriented

Table 2.1: Summary of terminological recommendations made in chapter 2. Although an emphasis is placed here on control signals, the same vocabulary can also be used when discussing audio signal routing and mapping.

Chapter 3

Methodology

The main contributions of this thesis come from the results of an exploratory study investigating the way mappings are devised by skilled music technology users. This chapter presents the design of the study, including a detailed description of libmapper, Webmapper, the T-Stick, the subtractive synthesizer used in the study, as well as the data collection and analysis methods. Further technical details on the tools are presented in the online appendix[1]along with the guide for the interview and full script for the study.

3.1 Setup

The study took place in a quiet room. A computer was set up on a desk with the study software (section 3.1.1. Participants were given a chair in front of the desk with a mouse and keyboard connected by USB for operating the computer; most participants chose to sit, although one participant (P15) chose to stand for most of the experiment. The T-Stick (section 3.1.1) was connected with a long (3 m) USB cable to allow participants some freedom to move about. Overall, the physical set up for the experiment is believed to have had no meaningful impact on the results.

3.1.1 Tools

Libmapper

Libmapper is "a library for connecting things" [39]. It offers a convenient API for developing interoperable multimedia devices, and interconnecting them in a distributed peer-to-peer network. Although originally envisioned as a communications protocol, first implemented in Max/MSP,

---

[1] `http://idmil.org/making_mappings`, `http://traviswest.ca/making_mappings`

today libmapper is primarily manifested by a C library with bindings for C++, Max/MSP, Pure Data, and a variety of other languages and environments.

A libmapper device is configured with a number of input parameters and output signals. When a signal on one device is connected to a parameter on another device, the source device sends the signal data directly to the destination device using Open Sound Control (OSC) protocol. Associations are set up using a shared administrative bus on which all devices listen. A certain message can be sent by any device requesting that a given association be made, at which point the devices involved in the new connection will automatically set up the specified association. This distributed architecture naturally leads to the use of a dedicated program for monitoring and configuring the network. Webmapper was used in the experiment. A more detailed description of libmapper and Webmapper is given in appendix A.

Using libmapper+Webmapper, libmapper devices can be easily recognized and connected together. This mapping environment allows users to easily configure one-to-one, divergent, and convergent associations[2], including arbitrary transformation functions specified using a C-like expression language. Webmapper primarily uses a simple drag and drop interface, making it easy to operate for users with less expertise, but the availability of features such as arbitrary transfer functions ensures that highly skilled users are not significantly limited by the environment.

A more detailed description of libmapper and Webmapper is included in appendix A.

The T-Stick

The T-Stick is a family of gestural DMIs originally designed and created by Joseph Malloch [37]. The conceptual design of the instrument/family is straightforward and ambitious. Each T-Stick is meant to be a robust musical instrument which is simple to set up, easy to make sound with, and both robust and nuanced enough to allow deep long-term engagement and eventually mastery. Ideally, the instrument should sense its spatial orientation, as well as the location, size, and pressure of every touch. The form of the instrument is a simple cylinder; each different T-Stick differs in length and diameter, from the smallest piccolo to the enormous contra-bass.

This conceptual design has been realized using a variety of sensors. Most T-Sticks made to date, including the one used in the study, employ a 3-axis accelerometer to sense motion and orientation, an array of capacitance sensitive copper strips to sense touch, and a long and narrow force sensing resistor to sense pressure. More recently constructed T-Sticks employ magnetic field and angular rate sensors along with accelerometers to more accurately measure the orientation of the instrument. The touch and pressure sensing strategies however, have largely remained the same since the initial implementations of the instrument.

---

[2] support for convergent associations in Webmapper was added part way through the experiment, meaning three participants (P3, P4, and P7) were not able to use this feature

Although the signals from the sensors could be directly mapped to synthesis parameters, it is more common to model the gestures of the performer and present high-level parameters which relate to these gestures. [37] goes into detail on some of the possible gesture modelling strategies, and other approaches have been developed by D. Andrew Stewart [38]and other users of the T-Stick. Eight gesture-related features were exposed as source signals to participants in the study: jab, shake, eggbeater, tilt, average touch position (touch), number of touch sensors activated (hold), brush, rub, and pressure (squeeze). The details of the implementation of the gestural model and corresponding high-level parameters used in the study are presented in the online appendix[3]. A summary of the gesture feature extraction is offered in table 3.1. Note that, aside from the squeeze signal, all of the gestural signals presented to participants are derived from multiple sensors; either a combination of the three accelerometer axes, or a combination of numerous capacitive touch sensors.

Signal	Sensors Utilized
/brush	capacitive touch sensors
/eggbeater	accelerometer x y z
/hold	capacitive touch sensors
/jab	accelerometer x y z
/rub	capacitive touch sensors
/shake	accelerometer x y z
/squeeze	force sensing resistor
/tilt	accelerometer x y z
/touch	capacitive touch sensors

Table 3.1 Summary of gesture signal to T-Stick sensor relationships.

A program in Max/MSP was used to read the raw data from the T-Stick via USB serial, extract the gestural features described above, and connect to the libmapper network. In addition to these roles, the Max program was used to provide a simple visualization of the gesture signals, shown in fig. 3.2.

Sound Synthesizer

For the sound synthesis module in the study, we chose to use a simple subtractive synthesis approach which would be familiar to a wide range of participants. For ease of implementation, a commercial subtractive synthesis plugin (Massive by Native Instruments Gmbh.) was used as the basis for the sound synthesizer. The plugin was configured to follow a simple signal flow as outlined in fig. 3.1. Eight sound parameters were exposed to the user: the fundamental frequency

---

[3] `http://idmil.org/making_mappings`, `http://traviswest.ca/making_mappings`

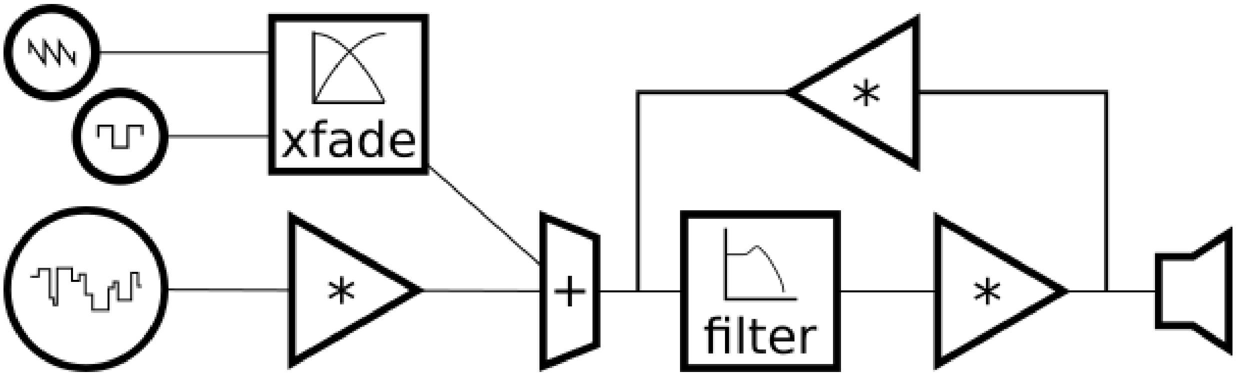


Fig. 3.1 Block diagram of the synthesis algorithm used in the study.

of the oscillators, the cross-fade between the square and sawtooth waveforms, the cutoff frequency of the noise filter, the amplitude of the noise generator, the overall low pass filter cutoff frequency, the overall low pass filter resonance, the amount of feedback from the output of the filter back to its input, the amount of reverb, and the overall amplitude at the output.

The synthesis plugin was hosted in a Max/MSP program, which was also used to expose the parameters described above to the libmapper network and to provide a simple visualization of the current state of the parameters similar to the display used to visualize the T-Stick signals. The visualization is shown in fig. 3.2. The Max program was also used to manipulate the synthesis parameters when they were not being controlled via the mapping, by moving the position of a small slider adjacent to the signal visualization. Finally, in case the user put the synthesizer in a configuration where it would not produce sound, a small reset button was placed in the upper right corner of the display that would restore a preset configuration of the synthesis parameters in which the synthesizer would be audible.

Data Collection

In addition to a demographic questionnaire and the interview conducted after the mapping design activity, three streams of data were recorded during the experiment: the raw sensor data and gestural features from the T-Stick, all changes to the parameters of the subtractive synthesizer, and all communications on the libmapper administrative bus. The latter dataset can be used to reconstruct a timeline of the mapping activity, including the time when any modifications were made to the mapping such as adding an association, removing an association, or changing the transfer function of an association; this is referred to as the activity data throughout the presentation of the results.

These three data streams were recorded using the oscdump utility that is included in the

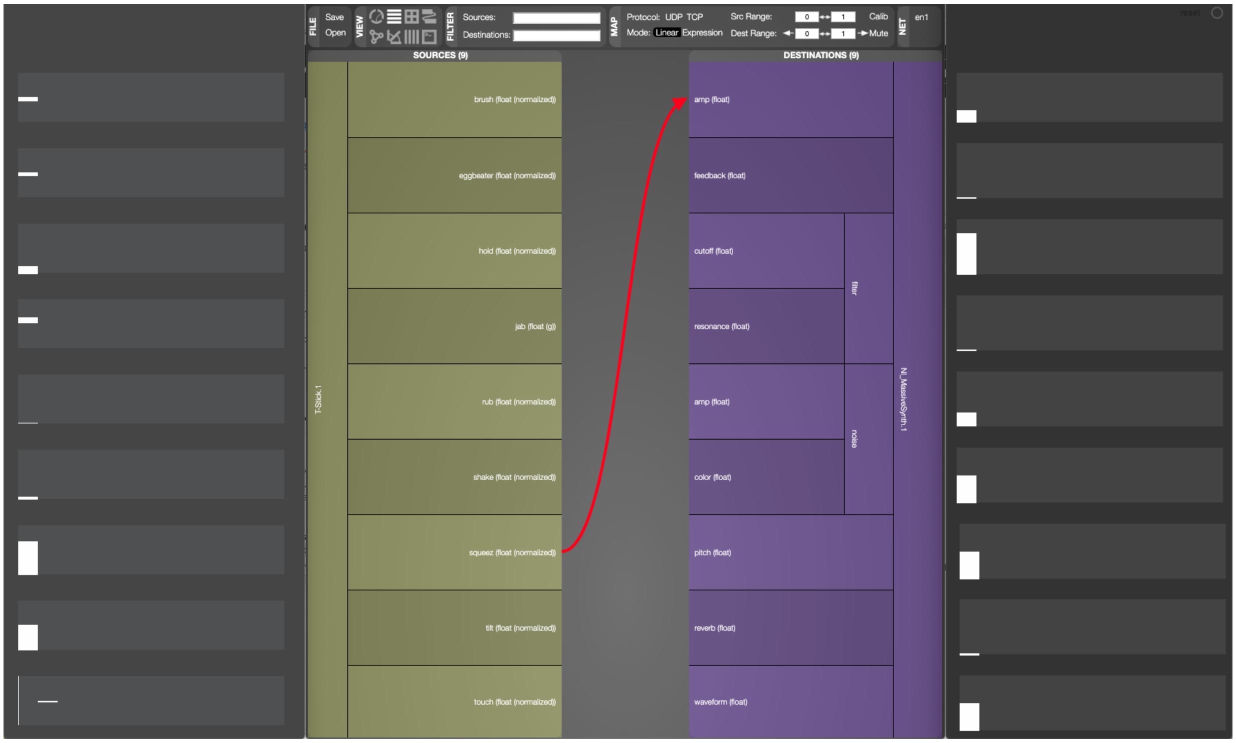


Fig. 3.2 The display seen by participants. From left to right: graphs of recent gesture signals, Webmapper list view, graphs of recent synthesis parameter settings.

liblo OSC library[4]. This command line program listens on a given UDP port, dumping any OSC messages received as text to standard output. This text output was then redirected to a text file using a simple shell script, included in the online appendix[5].

3.2 Participants

The participants were all musicians with a range of backgrounds. All participants had at least four years of experience in composing, performing, or otherwise producing music using EAMIs and related electronic music technology. The most experienced participant (P12) reported 25 years of experience with computer music. Participant age ranged from 24 to 39, with an average age of 30.5 years.

---

[4] `http://liblo.sourceforge.net/`

[5] `http://idmil.org/making_mappings`, `http://traviswest.ca/making_mappings`

3.2.1 A Note About Participant Numbers

Overall, fifteen participants took part in the study, including three in a pilot version of the study (P1, P2, and P5) This leaves twelve participants who were included in the analysis, with code names ranging from P3 to P15.

Due to human error, the data recorder was not started for three participants (P4, P5, and P11), meaning that activity data, synthesis parameters, and gesture signals were not recorded for these participants. To simplify the presentation of the results, these three participants are omitted from the analysis of the mapping design process is chapter 4, where frequent reference is made to the activity data. Since this data was not used in chapter 5, all twelve participants are included in the analysis considering the efficacy of mappings.

3.3 Experiment

Before beginning the mapping activity, the participant was given a brief tutorial on the T-Stick, the synthesizer, and Webmapper. I identified each of the gestural signals from the T-Stick by name, demonstrating each of them using the T-Stick. Each of the synthesis parameters was similarly identified by name and swept through its range of values. I noted that the low pass filter, when set to the lowest possible cutoff frequency, would cut off all sound, and that the pitch parameter, when set to the highest value, would render the oscillator inaudible unless the filter was also near the highest parameter. I also noted that the amplitude of the noise generator was not independent of the overall amplitude, but influenced by both the noise amplitude and overall amplitude controls (see fig. 3.1). In Webmapper, I demonstrated how to create one-to-one, divergent, and (when available) convergent mappings; the signals used for the demonstration were chosen at random. In addition, I demonstrated how to adjust the default linear transfer function by adjusting the input and output ranges, and how to enable the arbitrary transfer function expression editor. Participants were encouraged to request assistance in case they wished to achieve a specific effect using the expression editor (only P15 took advantage of this offer).

After the tutorial, the data collection program was launched, and a timer was started. Participants were reminded that their goal was to “design a mapping that, based on your understanding of the words, would be an effective mapping for a live performance,” and that they would be required to spend at least 20 minutes, but not more than 60 minutes. Participants were then left to work on their design. I remained nearby in case of an issue or other need for assistance.

After time ran out, or the participant notified me that the mapping design was complete, I stopped the data collection program, and exported the participant’s finished mapping using Webmapper’s save functionality. I then helped the participant to fill out the demographic questionnaire, and turned on the camera to record the interview. The interview used the questions in table 3.2 as a starting point for an open-ended discussion.

Table 3.2: Interview guide

Would you please demonstrate (explain) how your instrument works?
According to you, what makes a mapping effective? What did you try to do so that your mapping would be "an effective mapping for a live musical performance"?
What were some of the things that worked really well? (Why, and how did you know?)
What were some of the things you tried that didn't really work? (Why didn't they work, or how did you know?)
Could you describe the process that you used to make your mapping?
Suppose as if you had to give someone step by step instructions, "this is how you make an effective mapping (regardless of the interface or synth)", what would you tell them?
Could you describe a bit how your mapping evolved from start to finish?
If you had all the time in the world to work on this, what would you do next?
If I asked you to do it all again from scratch, would you do anything differently?
At any point did you feel limited by the mapping software, or as if it got in your way?
If you could change anything or add any functionality that you wanted, that you can imagine, how would you improve the mapping software?

3.4 Analysis Methods

The main analysis conducted was a qualitative thematic analysis of the interviews. The interviews were first transcribed to text based on the video recordings. I then formatted the transcriptions based on my interpretation of what ideas the participant was communicating, putting distinct ideas in blocks, with related ideas at a shared level of indentation (similar to indentation style used in computer code). The transcriptions were then read and re-read several times while taking note of possible themes or ideas of interest. The transcriptions were then printed on paper with line numbers, and color coded based on participant ID. Quotations from each transcription were extracted and sorted into piles of related ideas (e.g. remarks about convergent associations, remarks about how to design a mapping, etc); should a quotation fit into multiple piles, it was copied by hand onto a new slip of paper. The piles were then individually analysed; some related piles were merged if the ideas in them were related, some large piles were split into smaller piles with more granular themes, and some piles were discarded if they were too small or the remarks included in them too tentatively related. Based on the remaining piles, a final set of codes was determined. The transcriptions were then re-read several more times, annotating the thematic codes using an xml-like format. A python script was written to collate arbitrary selections of codes based on the xml-formatted transcripts. This was used to facilitate further high-level readings of the transcriptions.

Various ad-hoc methods were employed to analyse the usage data and the final mappings. Since the analysis was exploratory, rather than hypothesis testing, primarily observational metrics were derived from the data. The following list of metrics were considered, though others may have been explored and later dropped:

- period of time between different actions
- period of time during which given associations were present
- number of times a mapping was tested before keeping or rejecting it
- number of times a given mapping was used
- number of times a given signal or synthesis parameter was used

3.5 Summary

The study described in this chapter was designed to facilitate direct observation of the mapping design process, using a systems-oriented explicit mapping tool. The results of these observations, and the analysis of the interview transcriptions, are presented in chapter 4 and chapter 5.

Chapter 4

The Mapping Process

Designing a mapping is a creative process that plays a crucial role in determining the way an EAMI will look and feel to play. Although many tools have been developed to facilitate this process, relatively little research has directly considered how it is carried out, and how the tools used may influence the process. In this chapter we consider the remarks made by our study participants[1] after spending 20-60 minutes developing a mapping for the T-Stick using libmapper+Webmapper. The thematic analysis of these interviews provides a fairly clear picture of the creative process of making a mapping, leading to the development of a preliminary conceptual model of this process. Future work will determine how well this model generalizes to a longitudinal design process, and to a design process using different mapping tools, such as those that are demonstration-oriented.

4.1 Learning

In [36], Muzart et al. found that participants consistently spent time at the beginning of their design process forming simple one to one associations in order to learn the behavior of the input device and synthesizer. This stage of one-to-one associations was not prominent in the usage of the participants in the present study. However, 8 out of 9 described moments during the task where they were focused on learning the capabilities of the input and output systems and how they relate to each other. 5 out of 9 participants described or recommended a period of learning the T-Stick specifically:

> So first I wanted to understand what–even before tying some sort of gesture or sensor acquisition to some sort of synthesis, first you want to see what the sensors can do,

---

[1] As mentioned in section 3.2.1, three participants are omitted from the analysis in this chapter due to an error recording activity data for these participants, meaning there are a total of nine participants considered in this chapter

> and how the computer reads them, and what the scale is of those, and yeah, basically exploring the affordances at that stage. (P7)

3 out of 9 participants specifically mentioned discovering the way the gesture signals of the T-Stick were coupled or interrelated:

> I think the–yeah that sort of organization first into figuring out okay, how sensitive and how easy to use– like all of these sensors are very sensitive, but I found that brush was kind of cumbersome, rub was a little bit cumbersome as well, um... so, so figuring out which ones are going to be more easy to play, but also in this case uh there was a couple that, just because of the nature of the sensors, there was overlap in the information that was being sent out. So also keeping track of that so that you know what elements of the performance or what elements of the sound that are being produced uh can be done in isolation based on the sensor input, and which ones can be... um, which ones can be done sort of independently. (P12)

7 out of 9 participants described or recommended learning both how the T-Stick and the synthesizer work:

> But yeah I think I would start with what is capable from the instrument, what is capable from the side of synthesis, and then you know do some internal organization of those features and actually move this. (P12)

Overall, 8 out of 9 participants described periods of learning about the affordances of the input/output system. Participants' remarks were coded 'learning' 25 times in total. The discrepancy with [36] may be explained by the introduction of sliders which allowed participants to explore the sound synthesis parameters without having to make a connection from the T-Stick. In [36], the only way participants could vary the synthesis parameters was by creating a connection from the T-Stick to the synthesis parameter of interest. Without this requirement, participants still took time to explore the capabilities of the sound synthesizer, as in [36], instead of making simple associations like the participants of the prior study, participants in the present study simply manipulated the synthesis parameters and the T-Stick directly.

Although the participants in the present study did not exhibit the same strategy of exploring using one-to-one associations, it is clear from their remarks that learning was an important part of almost all participants' mapping design process. In addition to recognizing this during the interviews, it can also be observed that most participants spent some time at the beginning of the task simply manipulating the instrument and synthesizer controls. 7 out of 9 participants spent over a minute at the beginning of the task before making any connections. This initial period may be attributable to initial exploration of the input and output systems.

If we consider the time between mapping changes made by a participant, e.g. the time between making two connections, we can see that 8 out of 9 participants had, at some point during the task, periods between mapping changes with duration over two minutes. If we suppose that periods of learning correspond with these long periods of time during which no connections or mapping modifications are made, we may hypothesize that mapping design involves learning throughout all stages of the design. These periods may alternatively correspond to some other process that doesn't require changes to the mapping, such as evaluating the changes just made, or developing ideas for new associations to try.

4.2 Experimentation

Whereas the theme of learning described in the previous section related to learning the properties of the input and output systems, the theme of experimentation (also identified by participants as discovery, exploration, or trial and error) is oriented towards discovering what mapping connections are most effective. 8 out of 9 participants described this kind of discovery process during their mapping design. Participants' remarks were coded as 'trial and error' 47 times, including 'exploration', 'discovery', 'experimentation', and 'trial and error' as keywords.

> It's a lot of experimentation. I tried a bunch of different things and just tried to make sound, it didn't feel right, didn't work, so I changed it, but eventually I came across something that I was like "I like how this works", like the pressure for the amplitude. So once I found that one, I kept that one and moved on to the other parameters. So I would say that mostly it's trial and error. (P4)
>
> Trying these kinds of stuff, and trying to experiment with what you see in front of you would be ideal for me, you know, would be ideal for the person that's trying to map things because when you see this you're not... You're not totally sure of what to do, so experimenting with what seems logical to you would be the first thing, would be the first step that you should take, and then modifying to your liking after understanding a bit how it works. I think that was kind of my process also. (P6)
>
> I think it just evolved from like a less useful system to a useable one, and I think further evolution would come with more experimentation with the instrument, and the system. That sort of thing. But yeah, it doesn't feel like as–too much of an elegant evolution as much as it is just discovery, trial and error, and then... um, sort of trial and confirmation, and satisfaction. (P10)

So if mapping design is a process of trial and error, or sometimes "trial and confirmation, and satisfaction" (P10), it begs the question: how do participants evaluate whether a trial has led to error or satisfaction? This important question is addressed in chapter 5, where we will discuss the results pertaining to what makes one mapping more effective than another. It is interesting to note, however, that participants often described their process as "messing around", "playing around", and then "finding", "realizing", or "discovering" what works and what doesn't. So the process sounds like an experiment: the designer supposes that a certain change to the mapping might improve it; they make the change, and test their hypothesis by playing with the instrument given the new mapping; finally, they evaluate the results and discover whether or not the change works.

Considering the activity timelines, we can observe that most participants alternate between periods of more rapid activity, during which many changes are made to the mapping, and more sparse periods, where the mapping is left unchanged or modified less frequently. Perhaps these alternating changing and static periods correspond with the phases of trial and error or experimentation. During the static periods, participants manipulate the input device and un-connected synthesis parameters and accumulate embodied experiences about the affordances of the instrument given its current mapping; in essence, they are *learning* about the input, the output, and mapping they are making (section 4.1). These experiences lead to the participants' new short term goals. A period of rapid changes follows in which the participant narrows in on something that meets their goal. After making these changes, another period of reflection follows where the newly edited mapping is evaluated once more. By repeating this process, the overall mapping design gradually improves until it is satisfactory, or time runs out. This cyclic model of the mapping design process is also described by some of the participants, 6 out of 9 of whom described the mapping as being built up iteratively.

> So, once you narrow in on a parameter that you think might be good for y'know turning the sound on and off and everything in between, louder and softer, it's then connecting that to the volume, the volume knob, and picking an arbitrary kind of tone to control, and getting a sense, okay, is this shaping the dynamics in a way that I like. [...] And then its... Yeah, and then its, okay, what's the next thing that might be interesting to map? (P7)
>
> The point for me is to layer up all of the controls, so add add add add add. [...] In the case of recommending to someone, I would recommend that they sort of individually go through parameters and ask what sorts of articulation they want to have, and map fitting sources to those parameters, experiment, see if it's behaving how you like, if not, sort of rinse and repeat, and see if you can get results that you like. (P10)

> It's a lot of small tiny little iterations of imagining some kind of gesture and what I want it to do, trying a mapping out, seeing what the range does, seeing what's cool or not cool, maybe re-imagining, (laughs), trying it out, changing the numbers, trying it out. [...] So there was all these tiny little iterations of imagining something, trying it out, tweaking the numbers, maybe changing my mind, like getting rid of the eggbeater or adding the reverb or... All these things that sort of go with that. [...] So it starts with either a sound or a gesture coupled instantly with one or with the other, with the compliment, and then just playing around with the numbers and the things, until it's closer to what I wanted, or what I wanted has changed because I found something cool. But then again, I have to update the ideal and go through the process again, and kind of keep doing that. (P14)

Whereas [36] proposed an analytic exploration phase followed by a holistic mapping design phase, our observations suggest that the task is mostly holistic, especially during the exploratory periods in between changes to the mappings. Analytic processes are likely used only when the designer has a clear and specific vision in mind, such as when tweaking a connection expression to achieve a certain effect. Many participants can be observed to make a large number of rapid changes to connection expressions at one time or another during their design process. These rapid changes likely correspond to an analytic honing in or contracting process. In most other instances, at least in the case of the participants observed in our study, it doesn't appear that there is always a specific goal in mind. In the kind of trial and error design process taken by the participants in this study, most parts of the design process appear to favor a holistic and embodied approach.

4.3 First Impressions

6 out of 9 participants described a moment where their first impression of a new connection immediately informed their decision of what to do with it (coded p1st). Oftentimes an effective change was recognized immediately:

> I could generate some cools sounds with it, the whistle sound. (P3)
>
> That appealed to me when I tried it out first. I think this one was an effect I didn't imagine before doing it, but then it worked out quite well so I retained it. Yeah. (P11)
>
> I liked the um... the hold related to the amplitude of the noise, 'cause I love noise and I... I don't know (hugs T-Stick). Like, touching related to noise... it seems really cool for me, so... that was my favourite move I think. (P15)

Other times a connection didn't work as well as expected:

> I tried putting tilt on amplitude and–no, it was not good at all, because the thing is, with how I placed it, the tilt... like, this is neutral (parallel to ground) like this would be high volume (pointed up) like this would be low volume, but it didn't really work out like I wanted to because if I would put the tilt on amp, then my feedback would have been like hold, and then it would be more complex, more complicated. (P6)
>
> I tried mapping squeeze to amplitude, and I found that I couldn't get exactly what I wanted. It was just a little bit hard to hold down or something or like I wasn't finding the sweet spot well enough. (P10)
>
> Somewhere within that mess of like little iterations like that I added the eggbeater also as something that introduces noise, but I felt like the motion was awkward, so... rather than trying to figure out how those adding and multiplication and all that works, I just decided to take it out because I felt like it didn't add something that I was comfortable with anyway? (P14)

And in these cases of strong first impressions, it seems that the decisions made are often retained until the mapping is complete:

> The first thing I did was definitely the rub. The rub for me–I guess for me reverb, sometimes you put some sometimes you don't, and like, I know if I go like this (rubs the instrument)... For me it's only natural, and I know I was like, "yo, this is what I like." It was the first one that I did, and I've never changed it since. (P6)
>
> I think I had some initial impulses right away, um and those fundamental ones at the beginning such as tilt for volume and um touch for pitch were pretty much unchanged, I think those are the fundamentals that I wanted to get in place. (P12)

Considering the activity timelines, we may also notice that most of the time participants only had to try a mapping connection once to determine whether or not it was worth keeping. Of 182 different connections which were tried by all 9 participants, 133 (73%) were only tried once, and 40 (22%) were only tried twice. The remaining 9 connections were tried three or four times. Furthermore, considering the period of time during which each connection was active, we can observe very different distributions of durations between connections which participants retained and those they rejected. The active duration of retained connections is evenly distributed from nearly the whole duration of the task to just a few seconds. This is consistent with what would be seen if we expected participants to find connections worth keeping from the very beginning of the experiment up to the very final moments. On the other hand, the duration of rejected connections is distributed exponentially with 25% of rejected connections being active for less than 20 seconds, and a full 50% active for less than a minute (fig. 4.1).

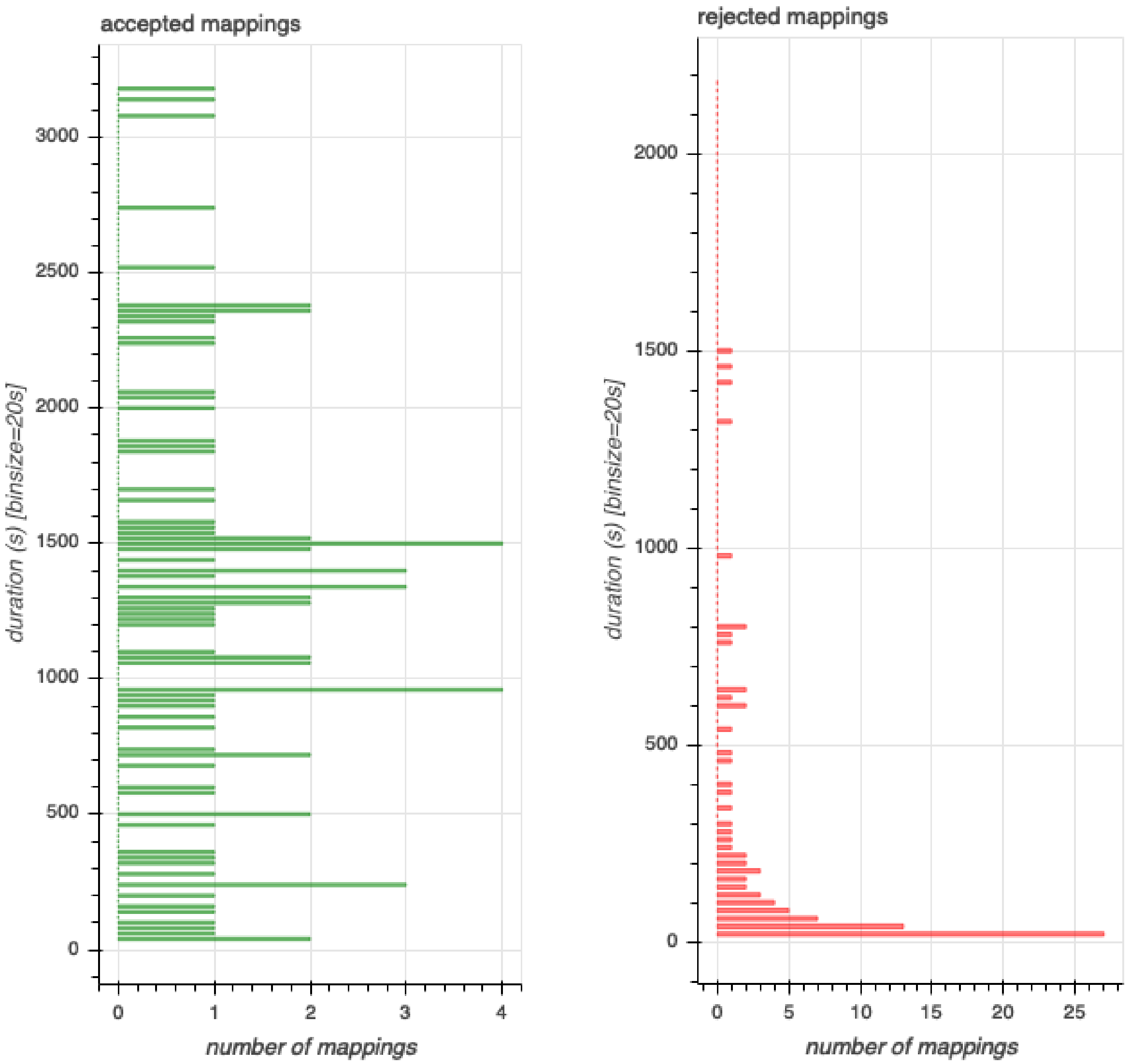


Fig. 4.1 Comparison of the active duration of all the associations kept compared to all those rejected by all 9 participants. The active duration of an association is the number of seconds that it was present during a given participant's design activity.

These observations provide strong evidence that participants are able to determine whether a connection works within a brief period of time based on trying the connection just once. This suggests that the process of evaluating changes to a mapping is not a slow analytic process, but an instinctual one based strongly on immediate first impressions. So we see once again that the process of making a mapping is not a careful analytical process, but a holistic and intuitive one.

4.4 A Model of the Mapping Design Process

Based on these results, we propose a model of the mapping design process that consists of two main modes of activity: diffuse exploration, and directed experimentation. These two modes of activity roughly correspond with the divergent and convergent phases of the balloon model of creative processes [1], or stages 2-5 and 6-8 of Sawyer's eight stages of the creative process [1].

In the exploratory mode, designers mainly manipulate the input device and un-connected synthesis parameters and accumulate embodied experiences about the instrument given its current mapping. They are focused on learning the affordances of the input and output systems, and the way the two relate to each other, presently and in the imagination. They are generally getting a feel for their mapping so far, developing a sense for what the instrument is doing that they like and generating ideas about how they might improve their mapping moving forward. During this mode, designers make relatively few changes to the mapping, with periods of roughly one to two minutes between changes (depending on the designer). This mode comprises a mix of problem finding, learning, incubation, idea formation, and idea externalization.

In the experimental mode, participants mainly make changes to the mapping, briefly moving the instrument in between changes to compare the result of the modification with their expectation. In this mode, participants test the hypotheses they developed while exploring the instrument. They work in a directed manner to try to achieve any short term goals established during their exploration and bring their mapping closer to the ideal. During experimentation, every change is evaluated within seconds. Whether something works or not is recognized immediately. When it works, participants generally keep it. When it doesn't, they move on, perhaps testing other hypotheses or else switching back into the exploratory mode. Designers make many more changes while experimenting, with periods of a few seconds to half a minute between changes (again depending on the designer). This mode is a mix of idea combination, evaluation, and externalization.

These two modes of activity roughly alternate and probably overlap at times. Some designers more heavily favor the exploratory mode, others alternate between the two often, and others spend most of their time making rapid changes in the experimental mode.

Gradually, each designer builds up their mapping one association at a time, until finally they reach a point where either further changes seem unnecessary or their allotted work time has run out. At this point the mapping is considered "finished," but the work is likely not done. Given more time to continue developing their mapping, most designers expect that further refinements could be made, the instrument could be adapted further for a specific context or performance, and generally there remains room for improvement.

Mode	Function	Description
Diffuse Exploration	learning imagining next change forming ideas incubating ideas externalizing ideas	moving the instrument playing around getting a feel for it
Directed Experimentation	testing assumptions evaluating changes combining ideas evaluating changes externalizing ideas	changing the mapping discovering what works honing in on it

Table 4.1 Summary of the mapping design process observed in the study.

4.5 Practical Applications

Based on this model of the mapping design process, we can consider possible improvements to the design of the mapping environment to facilitate designers' work. These practical recommendations may be implemented in a future version of libmapper+Webmapper.

The iterative connection-by-connection approach favoured by the participants in the study suggests a way of organizing the layout of signals. In the current version of Webmapper's list view, signals are displayed in alphabetical order from the top of the screen towards the bottom. This is advantageous insofar as the signals have a fixed order and relative location in the list, but it often results in a jumble of arrows forming in the middle of the display as the mapping is built up. A typical mapping is shown in fig. 4.2a. An alternative would be to display signals in the order with which they are added to the mapping, e.g. shifting signals to the top of the table when a connection is made involving them. Using this approach, connections are less likely to cross on screen, seen in fig. 4.2b, which makes it easier to read each association in the mapping. This also keeps the connection which the user is most likely to want to edit (the one they just made) in a consistent location on screen. Allowing the user to switch between these two approaches may be especially useful, preserving the benefits of both.

To facilitate the experimental mode of working, it may be useful to enable more direct manipulation styles of editing connection expressions. In particular, participants appear to adjust the

(a) signals sorted by name

(b) signals sorted by last edit

Fig. 4.2 Comparison of signal sorting using the same mapping (P10).

range of individual connections especially frequently. Allowing these edits to be made by clicking and dragging rather than typing in a number may allow designers to dial in these adjustments more efficiently by giving continuous feedback.

Its harder to suppose what might help with the explorative mode. Perhaps richer signal visualizations might help users develop their holistic sense of the functionality of the mapping. Similarly, generative visual or haptic signal outputs from the mapping may increase users' informational bandwidth; however, these may be beyond the scope of the mapping environment, delving more into questions of instrument design in general.

Another useful feature might be to allow snapshots of the mapping to be recorded and later restored or combined[2]. This might allow designers to test their ideas using a greater range of strategies. For example, it would be impractical in the current version of Webmapper to remove all connections except one in order to isolate the influence of that one connection, since there is no way to reverse the change where all the connections were removed. Allowing for previous states to be combined could also allow designers to explore ideas and combinations of ideas at a higher level. For instance, a certain behavior involving several connections could be developed, saved, and then cleared away to work on a different behavior, again involving multiple connections. Then

---

[2] some preliminary work is already underway to implement such features, see [40]

the first idea could be restored and combined automatically with the second one. This might allow designers to iterate on larger scale conceptual objects than individual connections, as seen in the study.

4.6 Limitations and Future Work

The participants in this study were asked to take no more than one hour to design a mapping which, according to their understanding of the term, would be an effective mapping for a live performance. They were required to use libmapper+Webmapper to make the mapping, and all used an unfamiliar interface (the T-Stick) and a subtractive synthesizer. It may come as no surprise, given the brief period of time, the participants favored immediate intuitive decision making. Given the subjective goal of making a mapping which they found "effective" may have also encouraged participants to favor intuitive decision making. These factors, along with the unfamiliarity of the interface and synthesizer, may have biased the participants towards their exploratory approach to the task. The specific mapping tool used may have also influenced participants' design approach. For instance, the patch-bay style interface of Webmapper's list view, with its conceptual model of adding connections one by one, may favor the iterative approach taken by most participants. Since most participants had no experience with the T-Stick, this may have encouraged the importance of learning seen in participants' design process.

If participants had had more time to work on their design, if they had been asked to achieve a specific kind of mapping, or if they had been more familiar with the interface before beginning the design, their general approach may have been significantly different. Furthermore, consider the numerous mapping tools that have been discussed in the literature, many of which are radically different in their approach compared to libmapper+Webmapper, especially those which are used in a demonstration-oriented manner. It seems very likely that users might make mappings differently if they were using one of these tools.

How these factors influence the mapping design process remains a question for future work. Because of the limitations of the present study, it is safest to consider its results in context: the mapping process observed here is best thought of as applying to the design of mappings for an unfamiliar interface, using a systems-oriented explicit mapping design tool, without a clear and specific goal in mind. It will be interesting, as future research is reported, to consider which aspects of the model described here can be generalized, and in what new contexts.

4.7 Conclusion

In this chapter, we proposed a model of the mapping design process as a cyclic process which alternates between diffuse exploration and directed experimentation. This model, and other observations of participants design process, naturally lead to practical design improvements applicable to the mapping design tool used in the study. However, future work will be required to better understand how generalizable these results are. The limitations of the present study, especially its relatively short duration (20-60 minutes) and use of a systems-oriented explicit mapping tool (libmapper+Webmapper), should be considered when interpreting these results:

The thematic analysis of the interviews provided a coherent overall picture of what it was like for the participants to design a mapping, and how they approached the task. Participants described making a mapping as a process of exploration, experimentation, and trial and error. Rather than working deliberately to develop a specific ideal mapping, participants would make small guesses at what might work well, quickly test their intuition against the perceived effect of the mapping, and immediately accept or reject each new connection on the basis of intuition, almost always after trying a given connection only once or twice. By repeated cycles of guess and check, the overall mapping would iteratively approach a final form which might be deemed satisfying. An important part of this overall process was for the participant to learn about the system, including the nature of the input device, the synthesizer, and the way these two might interrelate through mapping connections. Overall, the mapping process appears to be directed towards exploration and discovery, rather than towards achieving specific goals. Rather, the goals and preferences of the designer guide and direct their free and intuitive exploration.

Chapter 5

Effective Mappings

One view of a musical instrument is as a tool used to transform a performer's gestures into musical sounds. In an EAMI, gestures are measured by the sensors of the interface, sound is produced by the synthesis algorithm, and the mapping largely defines the transformation from gesture to sound by associating the sensor measurements to the synthesis parameters. From this perspective it is clear that compared to the interface and the synthesis algorithm, the mapping is at least an equal determinant of the quality of an EAMI. In other words, for us to design an EAMI which allows us to achieve our goals, we must at some point devise mappings which allow us to do so. This is what is meant by "an effective mapping". A mapping is effective if it allows the player to achieve their goals.

In this chapter we will consider what qualities were prioritized by the participants in our study when trying to make an effective mapping. Since the goals of EAMI design and performance are not universally shared, it may appear that there should not be universally shared criteria for the efficacy of a mapping. Despite this, we observe that there may be certain factors that the participants in our study generally found useful. This generality is limited at least to the context of the study. Our participants were musicians with a variety of backgrounds and levels of experience. They were asked to devise a mapping effective for a live performance. None of the participants were familiar with the interface, but all were familiar with subtractive synthesis.

We begin by considering the results of the thematic analysis of the interview transcriptions, specifically those which pertain to what makes some mappings more effective than others[1].

---

[1] As mentioned in section 3.2.1, all twelve participants are included in the analysis in this chapter, unlike chapter 4 where three are omitted due to an error recording activity data for those three participants.

5.1 Subjectivity

Given the intentionally open prompt told to the participants, we expected each participant to arrive with their own notions about what sounds good or feels right, what a live performance should look and sound like, and for each participant's ideal mapping to reflect their individual approach. Since no two participants made exactly the same mapping, this expectation may reasonably be considered to have been correct. Furthermore, it is difficult to claim that there was any significant consensus between participants about which gesture signals should control which synthesis parameters; no single association was chosen by a majority of participants, and the association (/touch to /pitch) that was used by the greatest number of participants (6/12) was also tested and ultimately rejected almost as often as it was finally chosen (4/12).

This reflects the first theme identified in the interview transcriptions: mapping is subjective. This theme was touched on by 5 out of 12 participants, with a total of nine quotations being coded with this theme. It suggests that the criteria for the efficacy of a mapping depend on the context of the performance, the instrument, and the player's goals.

> Because most of the time people that make music already, or are using controllers... they try to get the best controller that are adaptive to the style of music that they do and the workflow that they have, so I would definitely advise them to map the T-Stick the way that you would want the controller to be mapped. Yeah. (P6)
>
> With something like that I think its on the person, the user using the interface, it's like they're gunna tune... Like, whatever they map the effects and everything it's going to be based on what they find pleasing, um... and what they might think other people might find pleasing. It's kind of an individual experience. (P13)
>
> Depends on the context, obviously. Yeah, so there isn't a single answer to this question, even for me. Y'know, like, let alone everybody, so it's a hard question to generalize. [...] I get that it's a crucial question. And again, I want to reassert, there's not a single solution to this because y'know, lots of different instruments exist, and it's nice to have the diversity. (P7)

At first glance it seems circular to remark that an effective mapping is one which suits the goals of the player, but in fact it is crucial to recognize the inherent subjectivity of the design of mappings. There cannot be any universal set of truths which clearly delineate good mappings from bad ones, because players do not possess universally shared motivations, and it is the players' motivations which ultimately determine the requirements for their mappings. In fact, given the diversity of music practitioners and their creative practices, it would not be unusual for the goals of two EAMI designers to directly contradict each other. This has important implications for any study of mappings which aims to provide guidelines, tools, or computer models to help designers make better mappings. "Good", "better", and "more effective" are likely not metric spaces which can be submitted to clear delineations or straightforward rules. Any agreement about these notions necessarily stems from a shared understanding of the meaning of these words, and thus from a shared set of goals or intentions. That our participants seem to agree on certain criteria for efficacy implies that they share certain goals or intentions.

5.2 Legibility

Every participant suggested, in one way or another, that an effective mapping is easy to understand, either for the performer, the audience, or both. The legibility of a mapping, in other words, is especially important for the mapping to be effective.

> It's fairly simple, like, the public has to quickly understand what can, like, the link between the gesture and the sound. That's it. (P15)

Three main expressions of this theme were identified: correlation, intuition, and metaphor. 7 out of 12 participants expressed that mappings are effective when there is a correlation, direct relation, or intimacy between the gesture and its resultant sound.

> So that's what I found worked well was the... um, correlating the high intensity motions to high intensity sonic output in my opinion, or in my perception of it. (P10)
>
> I started thinking of things that would make sense to me to be able to control based on that organizational method I alluded to earlier of like large gesture things, large sound control things, fine-tune gesture things fine-tune sound things (P12)

9 out of 12 participants expressed that mappings are effective when they make sense intuitively, when they just work without having to think about it.

> There are certain parameters that I want to be able to control without thinking too much about it. So that's why I went with pressure for the amplitude, I want to be able to start and stop the sound easily without having to think too much about it. (P4)
>
> So for example the hold thing, the fact that I have one hand it gives you half amplitude, and two hands gives you full amplitude, uh, for me it's logical, it makes sense, it's not really something that you have to think about. So if you have a user that's using this instinctively... instinctively you'll be able to do that. (P9)

> The mapping has to be intuitive to the performer. Uh, otherwise he... he might uh intend something but he might perform something else, so that's the main thing. And then it also has to be... uh, intuitive to the listener as well. So the performer uh does some action and the... the person in the audience is observing both his movements and the sound, right? So he has to figure out the mapping by just seeing and hearing the sound. So I think this is the most important thing. (P11)

8 out of 12 participants justified their mappings or described their efficacy by explaining an analogy or metaphor that they employed to make the mapping more intuitive. These metaphors often related to analog synthesis and mixing equipment, or to acoustic instruments.

> I think, in any–because I mix, so like when I put reverb I put it on a return track, so I send a signal, so... for me when I rub it's like I'm sending a signal to the system, and like depending on how high, which would be like the wet or dry, I go, the more reverb it has because I'm still sending reverb to my signal. So that's how I felt when I (rubs) I went like this. I think. (P6)

> It's nice actually to be able to squeeze the instrument and for it to be silent, it's kind of like a resonating body, if you squeeze it it dampens it. So the filter cutoff has a similar relationship there, and I liked that when I stumbled on that. (P7)

Of the three efficacy-related themes found in our study, this seems the most likely to generalize. It seems uncontroversial to claim that most musical practices aim to communicate something. The mapping itself can participate in this musical communication only if it can be understood by the audience. In the context of performance specifically, legibility is important since it allows the performer to initially make sense of the mapping. In a situation such as our study where the interface itself is also unfamiliar, this may make legibility even more crucial. Legibility dominated our participants remarks around efficacy, accounting for just under two thirds (66.1%) of all the quotations which were coded under this broad category. Its importance cannot be missed.

5.3 Control

9 out of 12 participants mentioned the way the mapping allows the performer to create the sounds they intend to create; i.e., the way the mapping allows the performer to control the instrument. Most participants felt that the instrument should give the performer control:

> A mapping is effective when... I mean, the controls work (laughs), y'know? So... it's effective when the input and, well... yeah, when the outputted source is like, reflecting what you kinda want it to do I guess. (P8)

> Um, so... I guess a good mapping is... a system in which the output meets your intentions. (laughs), that seems obvious. But the details of that are kind of... so varied. (P10)

As these two participants' laughter implies, this criterion for efficacy seems circular. It is nevertheless worth acknowledging because of what it implies about the participants' goals. These participants wish to deliberately perform specific sounds as a consequence of performing specific gestures. They wish for the instrument to empower them with musical agency, for it to get out of the way and let them do what they intend to do.

This is not always the case. Although none of the participants in our study completely rejected the desire to control the instrument, some alluded toward another perspective:

> I think the nice things about this kind of instrument is there's always a little bit of random component in the sound. I personally like that. But there are certain things that I want to have a little bit more control. So I would like to make those those super intuitive and then the other ones can add randomization to the sound. (P4)
>
> Yeah, so it has certain things that are controllable. It stops– 'cause I hate things that just drone on infinitely, I need–I like silence in pieces. But it also is a little bit chaotic, and a little bit unpredictable, and that's actually a nice thing for me, for the type of music I would make. (P7)

So as well as deliberate control over some parameters, these two participants also found it effective to relinquish control to the instrument. In this case the instrument is given some sort of agency and allowed to collaborate with the performer.

This theme reveals the desire for control as a common goal amongst the participants in the study, whilst also highlighting that it is not an unyieldingly shared goal. This latter point is well reflected in the literature as well. A paper such as [4] presents a control-oriented perspective, for instance, emphasizing "control intimacy" as the goal of instrument design. The contrasting perspectives of [41] or [27] advocate for giving up control in exchange for other kinds of influence.

The shared goal of control may be explained in part by the design of the study. Live performance may encourage this kind of orientation, the choice of gestural signals may have evoked a more acoustic-like approach, and the T-Stick itself is made is such a way as to "suggest certain gestures that would be consistent with excitation, modification, and damping of sound if the object were actually physically vibrating" [42]. To what extent these factors may have encouraged an approach that emphasizes control, only future studies can determine.

5.4 Analysis of the Mappings

Each participant's mappings were recorded, as well as all of the associations they tested during their design process. From this data, we can determine not only the associations which participants ultimately used in their final design, but also all the associations which were tested and ultimately rejected. In addition, we can consider the usage of libmapper+Webmapper features such as convergent mappings (when they were available) and the expression editor. These data may shed some light on the mapping strategies which participants found effective.

The literature on mapping in EAMIs could reasonably be interpreted to encourage the use of non-linear transfer functions with a mix of divergent and convergent mapping associations [e.g. 3], [7]. Our participants' usage of these strategies, both during their design and in their final mappings, is presented in table 5.1. The majority of participants did use divergent associations. Usage of convergent associations and non-linear transfer functions was less common. Although all participants who were able to make convergent mappings (and for whom the data was recorded) tested convergent mappings in the course of their design process, only 4 participants finally kept a convergent mapping. Non-linear transfer functions were tested even less often, by only 3 out of 12 participants. However, only 2 of these 3 participants retained a non-linear transfer function in their final mappings; in both cases, the absolute value function was used.

Technique	Tried	Used
convergent association	6	4
divergent association	8	8
nonlinear transfer function	3	2
linear transfer function	9	8
only default one-to-one associations	2	2

Table 5.1 Count of how often different mapping techniques were tried vs used in a final mapping. Note that convergent associations were only available to 9 out of 12 participants.

It is impossible to determine exactly why the participants in our study generally chose not to use convergent associations or non-linear transfer functions. We can speculate that usage of these strategies may be related to technical background experience; the participants who used non-linear equations in particular had a high level of experience in computer music and EAMI-related research. Alternatively, the limited usage of these features may provide tentative support for the hypothesis presented in e.g. [2], [28] that mappings made at an intermediary mapping layer from high-level gestural signals to high-level synthesis parameters may be made using simpler strategies while still preserving the benefits seen when using complex cross-coupled mappings. Finally, it's entirely possible that these strategies were avoided by our participants due to the time constraints of the study; future research may shed light on this, either by providing longitudinal inquiries, or simply by asking participants in a shorter experiment about their reasons for using or not using certain features of the mapping tool.

As previously mentioned, certain parameter associations were employed more often than others (fig. 5.1), and some were rejected more often than others (fig. 5.2). Sometimes the same associations were commonly employed *and* commonly rejected. If an association being used in a participant's final mapping is taken as a vote in favor of that association (+1) and an association being rejected as a vote against it (-1), the sum of these votes is illustrated in fig. 5.3. It's difficult to claim that these data provide strong evidence in favor of any particular parameter associations; although some signals are used or rejected more or less often, considering the "final vote", there are not clear winners or losers. Perhaps with a greater number of participants, a more compelling consensus might emerge; this might act as the basis for a participatory mapping design. Unfortunately, in the scope of this research little can be definitively learned from examining the associations participants used.

Another avenue might be to consider the signals alone, rather than the associations; perhaps certain gestural signals are more generally effective than others, for instance. These data are illustrated in table 5.2. Again, there is very little consensus. However, here we do notice that the eggbeater signal was rejected especially often, and that both squeeze and tilt were utilized especially often. The reason for this is not obvious, but perhaps it can be explained by the quality of the signals themselves. Squeeze is derived directly from the output of the force sensor in the T-Stick, and consequently is a very clean and responsive signal. Tilt is derived from the accelerometer according to a simple cartesian-polar conversion, and is also a very clean signal. Eggbeater on the other hand is determined heuristically using an empirically (i.e. arbitrarily) determined algorithm.

5.5 Discussion

The results of the study make at least one thing clear: effective mappings are legible to the performer and the audience. Whether by gesture-sound correlation, analogy, or simply intuition, the best mappings just make sense. It's not clear that participants felt a certain structural property, such as convergent connections or non-linear transfer functions, were important. It's not clear that certain gesture-sound pairings were more useful than others. All that is clear is that legibility is essential.

This implies something interesting about the way the participants in the study might have been thinking about their mappings. If a functional view of mappings had been predominant, we would expect participants to have made more remarks about the dimensionality or topology of the mapping. These kinds of remarks were completely absent, perhaps unsurprisingly given the predominantly non-technical backgrounds of the participants. On the other hand, if a systems-oriented perspective had been predominant, we would expect participants to remark about the structure of the mapping network, the transfer functions, and the connections. Participants did tend to speak about individual connections, and some participants emphasized scaling and transfer function as important for their mapping, but overall this was not a predominant theme; perhaps the few remarks that were made in this direction were prompted by the systems-oriented view presented by libmapper+Webmapper.

Instead of a functional or systems-oriented view of mappings, it appears from our analysis that participants mainly adopted a perceptual view of mappings. What mattered to them most was not the topology or the connectedness of the mapping, but the way that it looks, feels, and sounds. This may be explained by limitations of the study, such as the participants unfamiliarity with the T-Stick, or the 20-60 minute duration of the experiment biasing participants towards intuitive decision making processes, but it is nevertheless an interesting observation. It would be fair to say that most research in mapping to date adopts a functional or systems-oriented view of mappings, but it may be that mapping designers are more concerned with the perceptual view. Perhaps further research into the perceptual aspect of mappings should be prioritized.

5.6 Future Work

One obvious avenue for future research would be to ask participants to evaluate each other's mappings. This might provide better insight into the criteria used to determine efficacy, including participants' individual design goals. It would also shed light on the robustness of a one-off mapping designed in a short period: do participants still prefer the mapping they made during the study? Do they justify the efficacy of their preferred mappings in similar ways? This may provide some of the benefit of a longitudinal design study with considerably less effort.

Using participants' final mapping designs as the basis for an aggregate mapping design might also be interesting; especially with a larger number of participants, perhaps the consensus of the group would narrow in on some kind of optimal mapping design, which could be verified by asking the participants to evaluate the aggregate design in contrast with the individual designs. For example, using the first and second highest valued associations in the consensus-based association matrix, such an aggregate mapping design might be determined fig. 5.4.

It may also be interesting to consider how expert EAMI designers evaluate the mapping designs produced in the study [The Consensual Assessment Technique 1, pages 41-42]. By segmenting different participants' approach to the design task based on how well their final mapping was evaluated, we might gain insight into what makes some design approaches more effective than others, as well as offering an expert perspective on what makes some mappings more effective than others.

These kinds of studies, in which participants or experts evaluate specific mapping designs (whether made by other participants, an aggregation of other participants' designs, or some other means) may provide interesting insights into what makes some mappings more effective than others. However, these results would be severely limited to the devices being mapped and the context in which the mapping is supposed to be used; an effective mapping for a T-Stick may not be expected to work well for a different interface, mappings meant for live performance may not be useful for sound design, and in general it would be difficult to extract generalizable results from any study using only a single specific interface, synthesis device, or usage context. These limitations could be avoided by using a variety of interfaces, synthesis algorithms, and usage contexts, but thoughtful study design would be required to keep the number of test groups from ballooning while also collecting enough data for generalizations to be made across groups. Such a study may be extremely difficult to design and conduct, meaning that direct evaluation of mapping designs may not provide an effective way of studying mapping efficacy in a generalizable way.

Another approach would be to conduct qualitative studies investigating the contexts in which artists use EAMIs, and the creative goals with which they approach EAMI use and mapping design. Since the criteria for mapping efficacy are inherently subjective, this kind of approach may be a more fruitful one. Rather than shedding light on specific criteria for evaluation, a broader qualitative approach could help us to better understand the diversity of contexts and goals upon which such criteria are necessarily dependent. This may serve as a more meaningful basis from which specific criteria for efficacy could be examined.

In general, future research into the efficacy of mappings needs to carefully study this fundamental question: how do criteria for evaluating the efficacy of mappings generalize across different devices and creative goals? Without understanding this, the full implications of any study of mapping efficacy will be difficult to discern.

I hypothesize that the criteria for evaluating mappings are more likely to generalize within groups that share creative goals, rather than within groups that use similar devices. It seems natural that the requirements of live performance would dictate the mapping design criteria more strongly than e.g. the choice of interface or sound synthesizer, and that in the context of sound design the criteria would be radically different even when using the exact same interface and sound synthesizer as the performers. The more precisely the creative goals are shared, the more closely mapping design criteria should resemble each other even as devices are swapped out. Testing this hypothesis is left to future work.

5.7 Summary

By examining the remarks of the participants in our study, as well as the mapping strategies they employed during their design process, we can identify two major factors which participants considered important for a mapping to be effective: mappings should be legible, and they should provide the performer with control over the sound. These criteria suggest related design goals, that the perception of mapping should be clear and meaningful to the performer and the audience, and that the performer should be the main agent responsible for the results of their performance. Future studies will shed light on how the design of the study may have influenced our participants' approaches to the design of effective mappings.

As well as these factors related to efficacy, another crucial theme was found: mapping design is subjective. This theme, although somewhat circular in construction, is worth seriously considering because of the important consequences it implies for future research into the design of effective mappings. Criteria for efficacy are inherently context and goal dependent. If the goal of a study is to determine factors which facilitate efficacy, it is necessary to understand the context and goals from which the otherwise implicit meaning (or meanings) of efficacy is (or are) derived. Future work must consider the goals of artists and other mapping designers so that criteria for efficacy can be properly situated in relation to these goals.

Signal	Used	Rejected	Used Minus Rejected
/brush	7	6	1
/eggbeater	8	14	-6
/hold	10	9	1
/jab	10	9	1
/rub	9	5	4
/shake	17	12	5
/squeez	21	8	13
/tilt	22	11	11
/touch	13	8	5
/amp	18	14	4
/feedback	13	6	7
/filter/cutoff	13	9	4
/filter/resonance	10	9	1
/noise/amp	14	11	3
/noise/color	11	4	7
/pitch	16	15	1
/reverb	11	10	1
/waveform	11	4	7

Table 5.2 Count of how many times each signal or parameter was used or rejected by all participants.

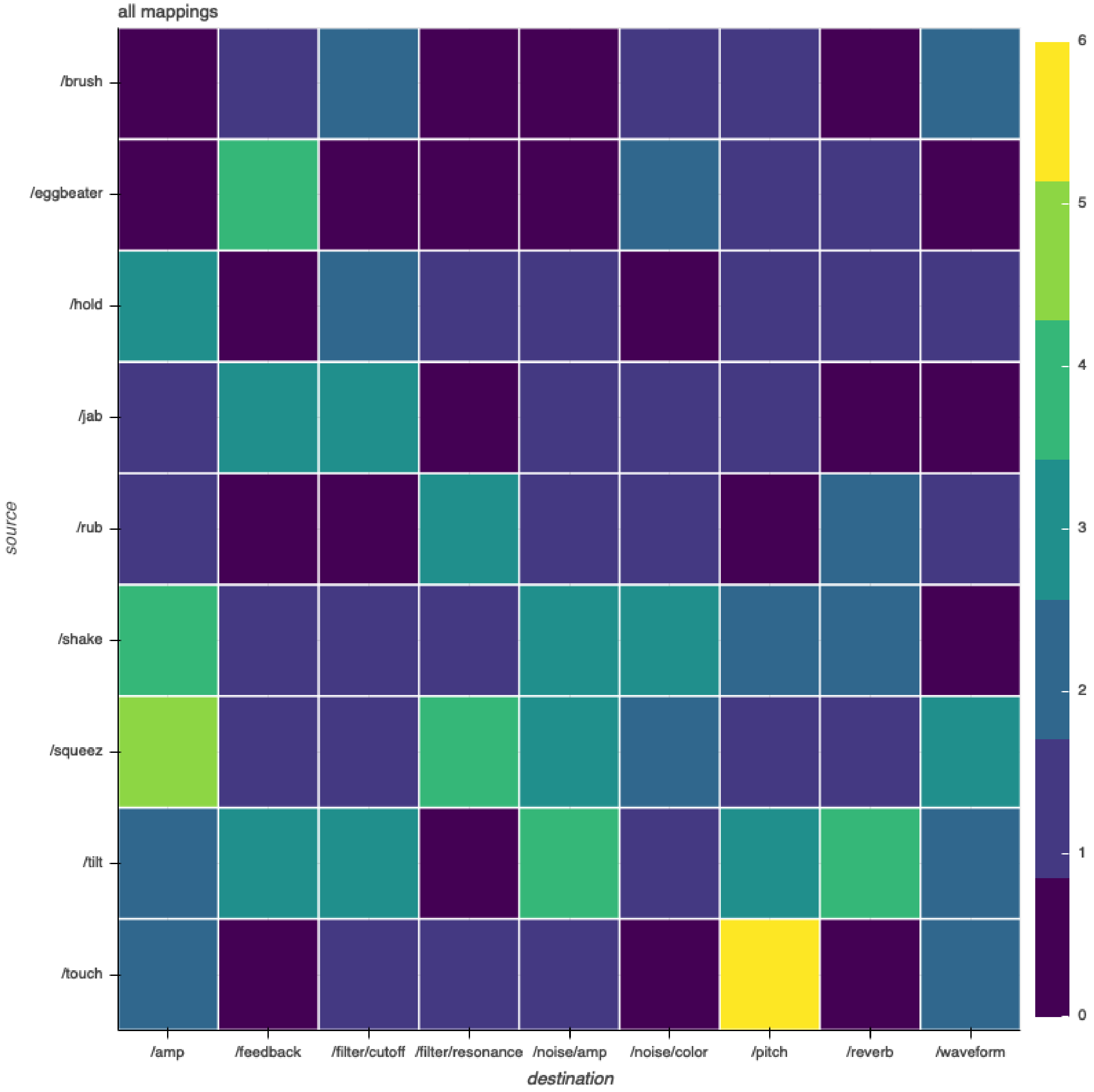


Fig. 5.1 Associations used in final mappings of all participants

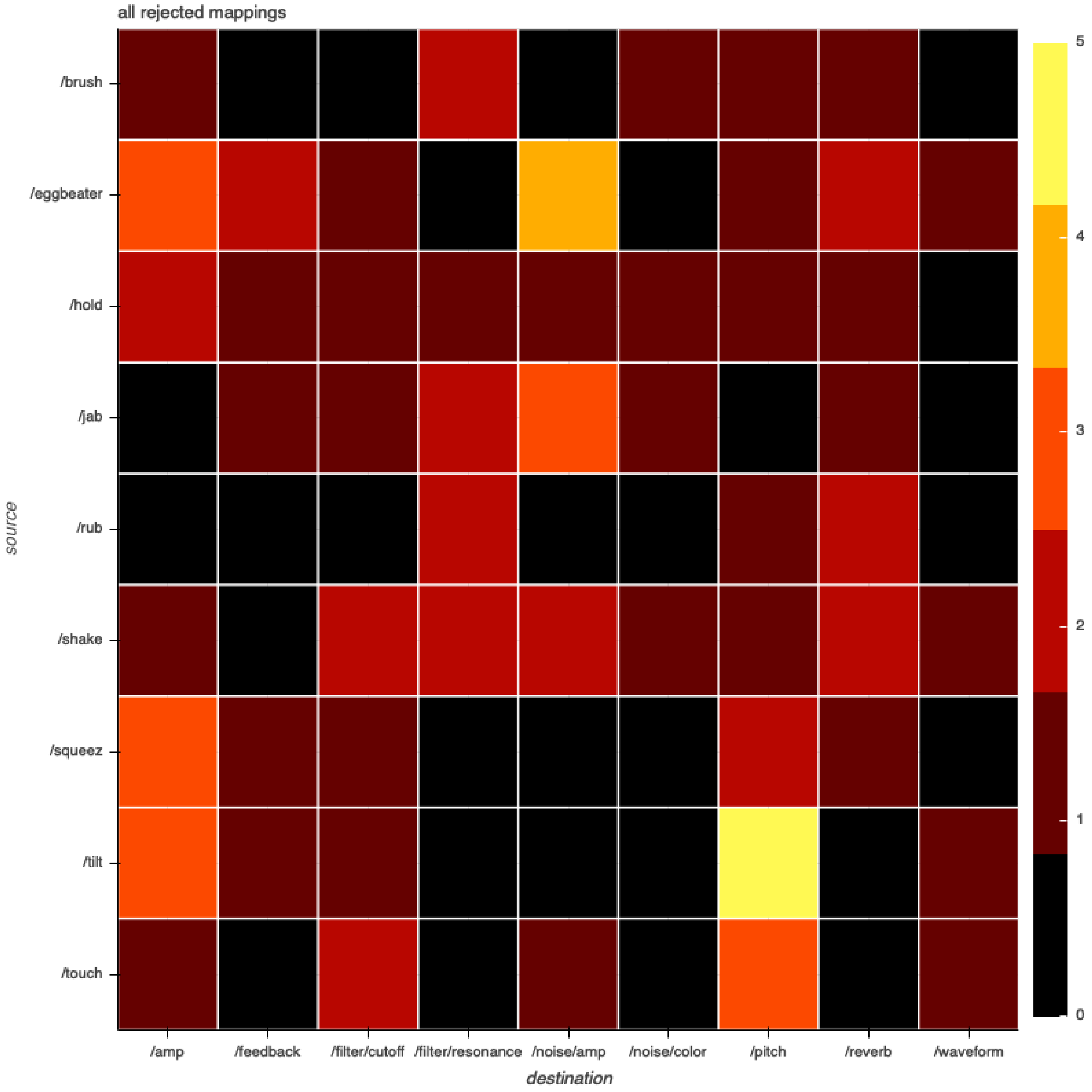


Fig. 5.2 Associations tested but not used in final mappings (i.e. rejected) of all participants

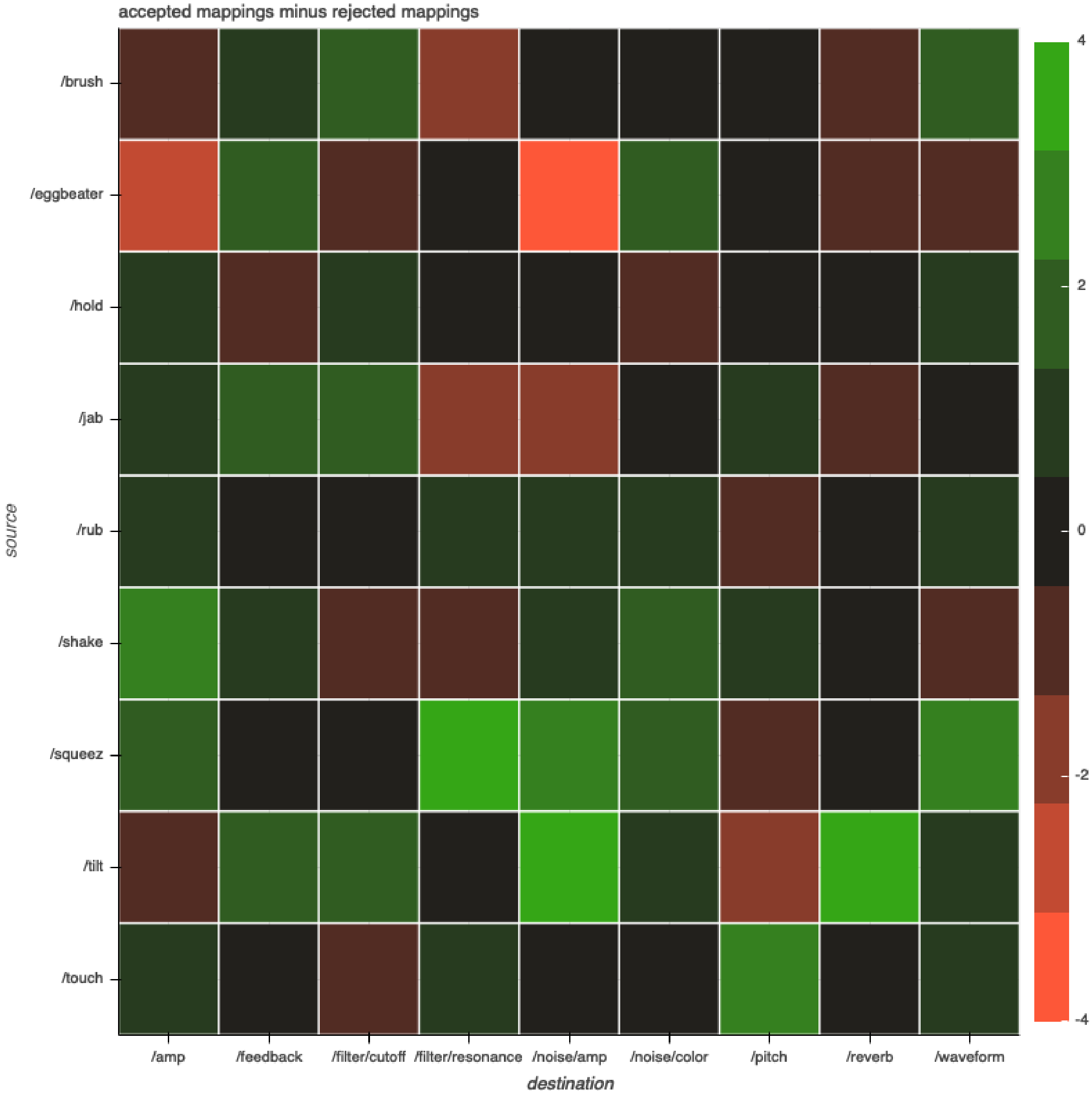


Fig. 5.3 The difference of how often associations were used vs rejected by all participants

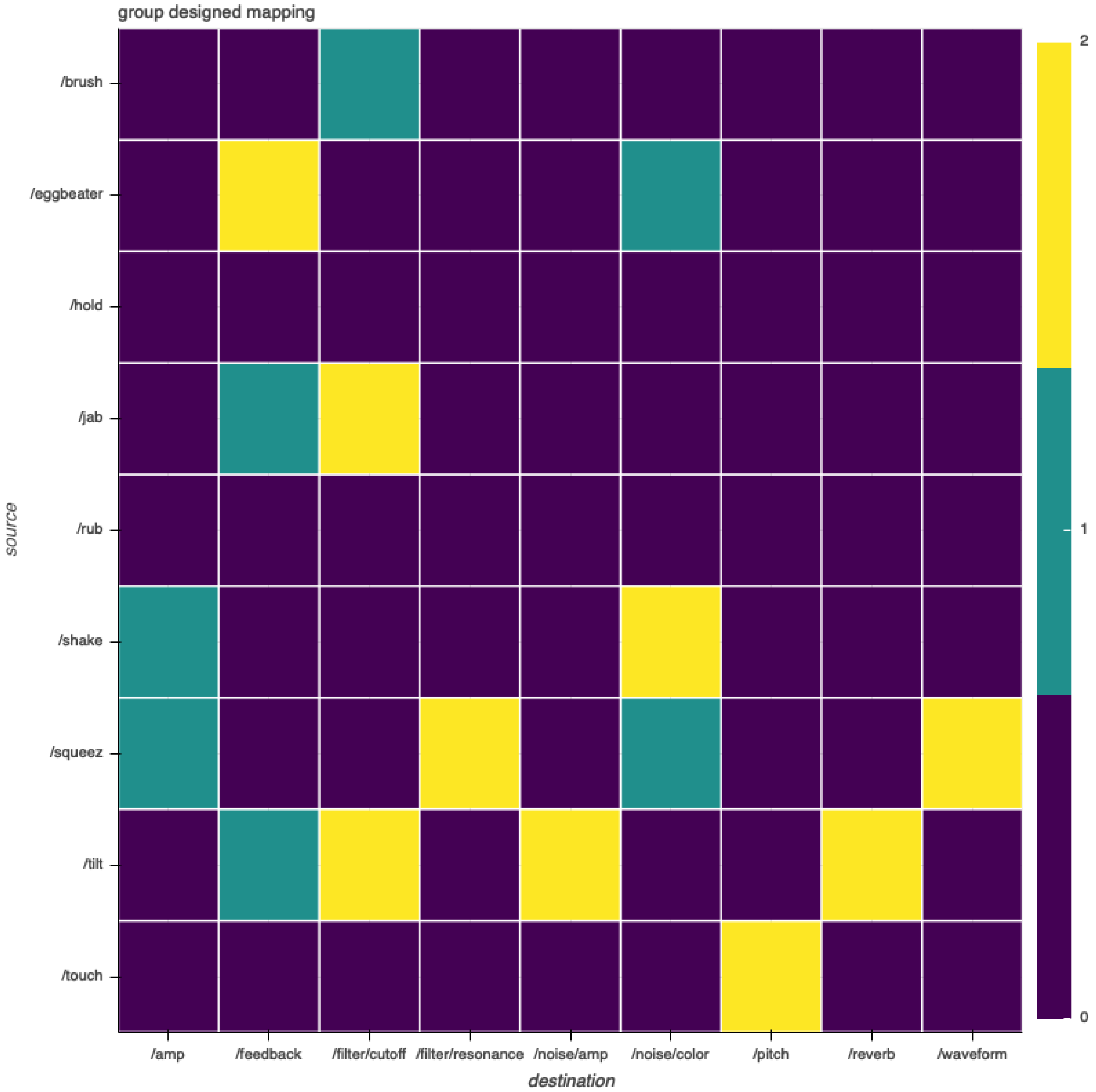


Fig. 5.4 A possible mapping implied by the connections with the most consensus among participants in the study. The brighter associations could be used as the main influence on a given sound parameter, with the other associations having convergent or secondary influence.

Chapter 6

Conclusion

The design of mappings has come to be recognized as a crucial part of EAMI design. To help future researchers examining this topic to communicate unambiguously, it is important to consider the terminology with which our research is described. To this end, numerous terminological recommendations were made in chapter 2.

The mapping design process itself has still received relatively little consideration in the literature, and was the topic of chapter 3, chapter 4, and chapter 5. By observing skilled users making mappings and analyzing their reflections on this process, we can begin to better understand this creative process and how to facilitate it. By considering the way these designers describe the mapping they make, we can understand the factors that they consider important for a mapping to meet their goals, and more importantly, we can begin to document the kinds of goals with which EAMI designers and users approach mapping design.

From this research, we see that the process of designing a mapping is an iterative and cyclic process. Designers alternate between diffuse exploration and directed experimentation, gradually building their mapping up from individual associations. Our observations of this process lead naturally to practical recommendations to improve the mapping environment used in the study.

Considering the effectiveness of mappings, we see that participants in the study emphasized legibility and control as important factors, implying that the design process favours a perceptual view of mappings. As well, we recognize the inherent subjectivity of mappings. This important theme cautions us against seeking universal principles upon which "good" mappings can be based, and encourages further study of the diverse creative goals of mapping designers.

These results represent a first step towards better understanding how skilled designers and users make EAMI mappings, and what considerations might influence their evaluations of the efficacy of certain mappings. Much remains for future research to investigate: How is the design process changed when the design is developed over a longer period of time? How is the design process influenced by the mapping design environment itself, such as the choice of system-oriented vs demonstration-oriented mapping tools? How do skilled designers evaluate each others' mappings, compared to how they evaluate their own mapping? What mapping design goals are most commonly shared, and in what contexts? What criteria for evaluation are these goals related to?

The ultimate goal of this research is to facilitate the design of mappings. This avenue of research complements the significant scholarly effort which has been devoted to the development of novel mapping strategies and techniques. Because of the modular and reconfigurable nature of EAMIs and the parts that make them up, unskilled users are often put in the role of the designer when it comes time to create the mapping. Studying the mapping design process will allow us to fully mobilize novel mapping techniques, allowing music makers to enrich their creative practice with these technologies. Better understanding how effective mappings are made will allow new tools to be developed that will allow these users at all levels of skill to easily create mappings which are complex, engaging, intuitive, and most importantly, which allow them to achieve their unique creative goals.

Bibliography

[1] R. K. Sawyer, *Explaining Creativity: The Science of Human Innovation*, 2nd ed. New York: Oxford University Press, 2012.

[2] D. Arfib, J. M. Couturier, L. Kessous, and V. Verfaille, “Strategies of Mapping Between Gesture Data and Synthesis Model Parameters Using Perceptual Spaces,” *Organised Sound*, vol. 7, no. 2, pp. 127–144, 2002.

[3] J. B. Rovan, M. M. Wanderley, S. Dubnov, and P. Depalle, “Instrumental Gestural Mapping Strategies as Expressivity Determinants in Computer Music Performance,” *Kansei-The Technology of Emotion Workshop. Proceedings of the AIMI International Workshop*, pp. 68–73, 1997.

[4] D. Wessel and M. Wright, “Problems and Prospects for Intimate Musical Control of Computers,” *Computer Music Journal*, vol. 26, no. 3, pp. 11–22, 2002.

[5] T. Magnusson, “Designing constraints: composing and performing with digital musical systems,” *Computer Music Journal*, vol. 34, no. 4, pp. 62–73, 2010.

[6] C. B. Medeiros and M. M. Wanderley, “A comprehensive review of sensors and instrumentation methods in devices for musical expression,” *Sensors (Switzerland)*, vol. 14, no. 8, pp. 13 556–13 591, 2014.

[7] A. Hunt and R. Kirk, “Mapping strategies for musical performance,” in *Trends in Gestural Control of Music*, M. M. Wanderley and M. Battier, Eds., Paris: IRCAM, 2000, pp. 231–258.

[8] A. Hunt, M. M. Wanderley, and M. Paradis, “The importance of parameter mapping in electronic instrument design,” in *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2002, pp. 1–6.

[9] A. Hunt and M. M. Wanderley, “Mapping performer parameters to synthesis engines,” *Organised Sound*, vol. 7, no. 2, pp. 97–108, 2002.

[10] G. Emerson and H. Egermann, “Gesture-sound causality from the audience’s perspective: Investigating the aesthetic experience of performances with digital musical instruments,” *Psychology of Aesthetics, Creativity, and the Arts*, vol. 12, no. 1, pp. 96–109, 2018.

[11] F. Morreale and A. P. Mcpherson, “Design for Longevity: Ongoing Use of Instruments from NIME 2010-14,” in *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2017.

[12] M. Waisvisz, “The Hands: A Set of Remote MIDI-Controllers,” in *Proceedings of the International Computer Music Conference*, 1985, pp. 313–318.

[13] V. Rudraraju, “A Tool for Configuring Mappings for Musical Systems using Wireless Sensor Networks,” Master of Arts, McGill University, 2011.

[14] C. Roads, “Composing Grammars,” in *International Computer Music Conference*, 1978, pp. 54–132.

[15] W. Buxton, W. Reeves, G. Fedorkow, K. C. Smith, and R. Baecker, “A Microcomputer-Based Conducting System,” *Computer Music Journal*, vol. 4, no. 1, pp. 8–21, 1980.

[16] I. Bowler, A. Purvis, P. Manning, and N. Bailey, *On Mapping N Articulation onto M Synthesiser-Control Parameters.pdf*, 1990.

[17] J. Pressing, “Cybernetic issues in interactive performance systems,” *Computer Music Journal*, vol. 14, no. 1, pp. 12–25, 1990.

[18] D. Rubine and P. Mcavinney, “Programmable tracking Instrument Controllers,” *Computer Music Journal*, vol. 14, no. 1, pp. 26–41, 1990.

[19] T. Winkler, “Making Motion Musical: Gesture Mapping Strategies for Interactive Computer Music,” in *Proceedings of the International Computer Music Conference*, 1995, pp. 261–264.

[20] “Special Issue on Mapping,” in *Organised Sound*, M. M. Wanderley, Ed., vol. 7, 2002.

[21] “Special Issue on Mapping,” in *Computer Music Journal*, M. M. Wanderley and J. Malloch, Eds., vol. 38, 2014.

[22] P. Doornbusch, “Composers’ views on mapping in algorithmic composition,” *Organised Sound*, vol. 7, no. 2, pp. 145–156, 2002.

[23] M. Burtner, “The Metasaxophone: Concept, implementation, and mapping strategies for a new computer music instrument,” *Organised Sound*, vol. 7, no. 2, pp. 201–213, 2002.

[24] R. A. Fiebrink, “Real-time Human Interaction with Supervised Learning Algorithms for Music Composition and Performance,” PhD, Princeton University, 2011.

[25] R. Tubb and S. Dixon, “A Zoomable Mapping of a Musical Parameter Space Using Hilbert Curves,” *Computer Music journal*, vol. 38, no. 3, pp. 23–33, 2014.

[26] J. Malloch, D. Birnbaum, E. Sinyor, and M. M. Wanderley, “Towards a new conceptual framework for digital musical instruments,” in *Proceedings of the International Conference on Digital Audio Effects*, 2006.

[27] J. Chadabe, “The limitations of mapping as a structural descriptive in electronic instruments,” in *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2002, pp. 1–5.

[28] A. Hunt, M. M. Wanderley, and M. Paradis, “The Importance of Parameter Mapping in Electronic Instrument Design,” *Journal of New Music Research*, vol. 31, no. 1, pp. 1–12, 2002.

[29] A. Momeni and C. Henry, “Dynamic Independent Mapping Layers for Concurrent Control of Audio and Video Synthesis,” *Computer Music Journal*, vol. 30, no. 1, pp. 49–66, 2006.

[30] D. Van Nort and M. M. Wanderley, “The LoM Mapping Toolbox for Max/MSP/Jitter,” in *Proceedings of the International Computer Music Conference*, 2006, pp. 397–400.

[31] S. D. Thorn and S. X. Wei, “Instruments of articulation: Signal processing in live performance,” in *Proceedings of the International Conference on Movement and Computing*, 2019.

[32] M. Marier, “Designing Mappings for the Sponge: Towards Spongistic Music,” in *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2014, pp. 525–528.

[33] D. Van Nort, M. M. Wanderley, and P. Depalle, “Mapping Control Structures for Sound Synthesis: Functional and Topological Perspectives,” *Computer Music journal*, vol. 38, no. 3, pp. 6–22, 2014.

[34] A. Hunt, M. M. Wanderley, and R. Kirk, “Towards a Model for Instrumental Mapping in Expert Musical Interaction,” in *Proceedings of the International Computer Music Conference*, 2000, pp. 2–5.

[35] D. Van Nort, “Modular and Adaptive Control of Sound Processing,” Ph.D. dissertation, McGill University, 2010.

[36] B. Muzart, M. M. Wanderley, and B. Caramiaux, “Approche centrée utilisateur pour la création de mappings geste-son complexe et expressifs,” 2017.

[37] J. W. Malloch, “A Consort of Gestural Musical Controllers: Design, Construction, and Performance,” Ph.D. dissertation, McGill University, 2008.

[38] D. A. Stewart, “Catching Air and the Superman,” DMus, McGill University, 2010. arXiv: `arXiv:1011.1669v3`.

[39] *Libmapper.*

[40] J. Wang, J. Malloch, S. Huot, F. Chevalier, and M. Wanderley, “Versioning and annotation support for collaborative mapping design,” *Proceedings of the 14th Sound and Music Computing Conference*, pp. 275–278, 2017.

[41] A. D. Campo, “Lose control , gain influence - Concepts for Metacontrol,” in *Proceedings of the International Computer Music Conference*, 2014, pp. 217–222.

[42] J. Malloch and M. M. Wanderley, “The T-Stick: From Musical Interface to Musical Instrument,” *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 66–69, 2007.

[43] J. Malloch, S. Sinclair, and M. M. Wanderley, “From Controller To Sound: Tools For Collaborative Development Of Digital Musical Instruments,” in *Proceedings of the International Computer Music Conference*, 2007, pp. 65–72.

[44] ——, “A Network-Based Framework for Collaborative Development and Performance of Digital Musical Instruments,” in *Computer Music Modeling and Retrieval. Sense of Sounds*, R. Kronland-Martinet, S. Ystad, and K. Jensen, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 401–425.

[45] J. Malloch, “A Framework and Tools for Mapping of Digital Musical Instruments,” Doctor of Philosophy, McGill University, 2013.

[46] J. Wilansky, “A Software Tool for Creating and Visualizing Mappings in Digital Musical Instruments,” Master of Arts, McGill University, 2013.

[47] A. H. Krajeski, “A Flexible Tool for the Visualization and Manipulation of Musical Mapping Networks,” Master of Arts, McGill University, 2013.

[48] X. Pestova, E. Donald, H. Hindman, J. Malloch, M. T. Marshall, F. Rocha, S. Sinclair, D. A. Stewart, M. M. Wanderley, and S. Ferguson, “The CIRMMT / Mcgill Digital Orchestra Project,” in *Proceedings of the International Computer Music Conference*, 2009, pp. 295–298.

Appendix A

Libmapper

Libmapper models the elements of a EAMI (or any other interactive software ecosystem) as *devices* with input and output *signals* which can be *linked* to one another through signal-to-signal *maps*[1]. This modular model is easy to apply to a variety of interactive multimedia systems, but we will focus on the context of EAMI design in which libmapper emerged. Typically, the hardware interface of an EAMI is considered one device and the sound synthesizer another. When both of these elements are implemented using libmapper, the control signals output by the interface can be easily mapped to the parameters of the synthesis algorithm. Furthermore, the mapping is easy to modify, multiple collaborators can simultaneously view and edit the mapping, and multiple different representations of the mapping can be easily generated to facilitate different perspectives and ways of working. These features make libmapper especially useful in the context of collaborative design and development of new mappings–precisely the context in which libmapper was initially developed.

In this thesis, libmapper was used as the environment in which study participants made their mappings. In order to adequately address the ways in which this environment impacted the design of the experiment, we consider here libmapper's structure and functionality. As most prior publications on libmapper [13], [43]–[47] no longer reflects all of its current features, we will also take the opportunity to overview its development from 2007 to the present time.

A.1 The Orchestral Neighborhood

Libmapper originally emerged in the context of the McGill Digital Orchestra Project [48] which took place from 2005 to 2008. This project sought to bring together researchers and research-

---

[1] Input signals correspond to control signals and output signals to control parameters (a mapping-centric input/output perspective), and maps correspond to connections in the terminology outlined in section 2.2. In this section the vocabulary native to the libmapper ecosystem will be adopted.

creators from composition, performance, and music technology in order to develop new musical instruments and compositions for them. With very limited time to meet and experiment with different mapping strategies, existing tools were found to be inadequate [43]. Iterative mapping design in a compiled language is much too slow for fast paced collaborative experimentation. Higher-level environments such as Max/MSP, while much faster, still pose significant difficulties to non-technical composers and performers. Even amongst technically adept collaborators, it can be difficult to build systems which are interoperable. Standards can be adopted, but it is notoriously difficult to design a standard which meets every possible need or requirement. Libmapper was developed to address these kinds of issues. It is meant to provide an environment for quick, collaborative, non-technical design of mappings, while still allowing for the design of complex associations, and all without imposing strong standards for data representation.

In [44], libmapper is presented as a means to facilitate communication between nodes in the "Orchestral Neighborhood", such as novel input devices and sound synthesizers. The fundamental structure and functionality of libmapper was already well established by this time. Additional functionality has been added since then, but the libmapper model of devices, signals, links, and connections has not changed significantly.

The original libmapper implementation was envisaged as establishing a protocol on top of Open Sound Control (OSC). All communication between devices, such as when requesting a connection or publishing metadata, is encapsulated in OSC packets. Device names are used as the root of an OSC namespace, the leaves of which are the signals which a device may send and/or receive.

A device in libmapper terminology is a network node which may produce and/or consume data. A device has a unique name, an IP address, and any number of signals, which represent the streams of data the device can produce, and/or the kinds of data the device can receive. When one device is connected to another, a router is created which is called a link; this intermediary processor is responsible for translating data from the source's representation to the destination's representation. Once a router is made, the two devices' signals can be connected from one to another. In libmapper the connection is not a passive communication channel, but instead represents the translation function, which can include data type conversion (e.g. from integer to floating point), source and destination boundary conditions (such as clipping or wrapping), and arbitrary C-like mathematical transformations, as well as translating from the source namespace to that of the destination. These transformations allow signals to be formatted according to their own individual requirements; by providing flexible transformation between different data formats, libmapper encourages strong device-specific semantics rather than imposing a particular representation.

Devices announce their presence on the libmapper network and share information about their signals by communicating on a multi-cast bus called the admin bus. Source signals can be connected to destination signals by requesting a connection on the admin bus; when the devices responsible for these signals detect the request, the link is automatically configured. Rather than clutter this multi-cast bus with signal data, devices communicate directly peer-to-peer, with source devices sending data directly to destination devices. In order to prevent issues if multiple instances of the same device are present on the network (e.g. if two people use the same type of synthesizer), a simple algorithm is used which allows devices to negotiate unique names by appending an ordinal number to their device name. The same algorithm is also used to secure unique UDP ports for all devices to prevent them from competing for these resources.

A.2 Webmapper

One of the most interesting features of libmapper lies in its distributed topology. Any device can observe the entire network, which in and of itself is not that exciting, but any device may also request connections between any other two devices. This is in contrast with e.g. MIDI, in which only the recipient of data may open one of the MIDI ports offered by the operating system, or typical use of OSC, in which both sender and receiver must be manually configured to exchange data. This feature of libmapper's design allows multiple interfaces to monitor and edit the network simultaneously. These interfaces are informally known as monitor devices, since they only monitor and coordinate the network rather than contributing signals to it. Although any device, including an interface or a synthesizer, could perform this role, it has typically been performed by a dedicated GUI application. Since libmapper itself does not include any canonical monitor application, several have been developed over the years. Over time, many of these have coalesced into a single application called *Webmapper* that provides multiple different representations of the libmapper network, each facilitating different perspectives and workflows.

The original monitor application was a GUI implemented in Max/MSP, called Maxmapper in [13]. As the C implementation of libmapper grew more dominant, several other user interfaces were designed, including a command-line application and several other GUI implementations. At some point, Stephen Sinclair developed a web-based GUI using a python server as intermediary between a web browser and the libmapper network. This application, called Webmapper, was described by Aaron Krajeski in [47] as "a moderately-featured, little-used GUI for libmapper." However, although Webmapper initially did not implement the majority of the features exposed in the original Maxmapper GUI, its web-based architecture offered compelling advantages over this and other monitor applications. In particular, it would allow a single monitor device to run as a server, which could then serve multiple client web browsers, making it easier to deploy the GUI on various platforms and in collaborative contexts.

Several MA theses at IDMIL focused on providing new GUIs for libmapper making use of the Webmapper python server in order to leverage these web-based advantages. In [13], Vijay

Rudraraju developed Vizmapper, a GUI using a balloon tree visualization to provide useful representations of large libmapper networks with many nodes. In [47], Aaron Krajeski developed a web-based GUI which implemented all the features of the original Maxmapper GUI, with improvements to visual clarity and a versatile responsive layout. In [46], Jonathan Wilansky presented two new GUIs: a grid-based editor which excels at representing networks with a large number of connections, and a hive visualization which is useful for examining large networks. These last three interfaces by Krajeski and Wilansky were all integrated into a single application; by using a model-view-controller architecture, each interface could be implemented as a view-controller pair within the same application. In Krajeski's thesis, this application is called MapperGUI. Over time, it has become conflated with the original Webmapper upon which it was based.

The present day incarnation of Webmapper provides eight different view modes, extending Wilanksy and Krajeski's list, grid, and hive views with canvas, graph, parallel, console, and chord views. Each view offers a different representation of the libmapper network, offering multiple different perspectives on the state of the network, different ways of working, and different strengths and weaknesses. List view is the simplest view to understand and use when there is a relatively small number of signals on the network; this made it the obvious choice for use in this thesis project. Based on the original Maxmapper GUI, source signals are displayed in one column on the left of the screen, while destination signals are displayed on the right side of the screen. Signals can be connected by dragging from one to another, and edited using an inspector pane at the top of the screen which updates to reflect the parameters of the currently selected mapping.

A.3 Convergent Mapping with Libmapper

Earlier presentations of libmapper consistently note as future work that it is not possible to create convergent mappings in libmapper. This functionality was eventually added to the C library by changing how connections are represented. Originally, a connection was considered as a link from a single source signal to a single destination signal. This relatively limited model of a connection was extended to the current notion of a map, which is a link from any number of signals to one other signal. Notably, the "sources" of the map are no longer required to be source signals, nor is the "destination" of the map required to be a destination signal. The source and destination labels which were originally used to restrict the meaning and use of signals are now merely metadata. When one signal is mapped to another, the value of the latter is updated when the former changes, regardless of their source or destination labels.

Although this functionality has been available for some time before this thesis project began, it was never exposed through any monitor application, and was thus inaccessible to most libmapper users. During the pilot study phase of the present research, participants consistently mentioned this as a limitation of the mapping environment. Support for convergent mapping was thus added to Webmapper. This required the invention of a novel interaction workflow for making convergent mappings. The structure of the Webmapper source code was also modified to make it easier to represent convergent mappings in all views.

Previously the representation of mappings between signals has been part of the responsibility of each different view in Webmapper. This introduced some duplication of effort, since several views used similar methods to draw the mappings. In order to avoid increasing this duplication of effort as methods were added for representing convergent mappings, the work of drawing a mapping was abstracted into a module seperate from the different view-controller pairs which implement the eight views of Webmapper, essentially treating the mappings themselves as having their own distinct view-controller hierarchy. This allows the same code to be used to draw mappings in views which represent mappings similarly, as well as allowing the interaction used to create convergent mappings to be reused between views.

To create a convergent mapping, the user builds up the mapping bit by bit. First a simple one-to-one mapping is made by dragging from one signal (either a source or a destination) to another (either source or destination). Then another signal is added to the convergent map by dragging from the new signal to add to the existing mapping. This causes the interface to highlight the prospective convergent mapping. When the user releases the mouse button (the drop phase of the drag-and-drop interaction), a radial menu pops up offering the user several default options for how to combine the new signal with the old mapping.

When a convergent mapping is created, it is necessary to specify how the source signals should be combined. This is implemented in libmapper by requiring that a mapping expression be given. Essentially, the user is required to specify the combining function[2]. From an interaction standpoint, this raises questions about how to present this work to the user. The approach taken was to decide on a few likely default expressions, and allow the user to select from these defaults using a radial menu. Specifically, the user may request that the new signal is added to the previous mapping, that the previous mapping is multiplied by the new signal, that the destination be derived from the average of all the source signals, or that the default expression supplied by libmapper be utilized. These options represent an initial ad hoc guess at a few useful defaults; as more users engage with this new functionality, additional default settings can easily be added. Otherwise, skilled users are also in full control of the expression generated by their selection, and can modify it by hand to accomplish other combining functions.

With the addition of support for convergent mapping to Webmapper, this functionality is

---

[2] Notably, the user is not required to supply a combining function that describes a convergent mapping in the sense given by [3], so a convergent mapping in libmapper is not necessarily a convergent mapping in this strict sense.

now readily available to all users of the libmapper ecosystem. Although a few usability issues were noted by some participants regarding the interaction with this functionality, none of the participants felt that any important functionality was missing from the mapping environment. On the contrary, many participants were impressed with the wide range of possibilities presented by Webmapper and libmapper.

A.4 Summary

Of the wide range of features offered by libmapper+Webmapper, the present study makes use of a small subset: visualizing a simple network with two devices, each with a handful of signals, connecting these signals from one or more sources to a destination, and applying simple linear transformations or arbitrary C-style expressions to transform the signals. These features make up the heart and soul of libmapper. They also represent the essential operations in a systems-oriented mapping design process.

Compared to commercially available mapping systems, such as the MIDI mapping functionality available in a DAW, or the modulation mapping system in a synthesizer, libmapper+Webmapper provides greater possibilities by far. The ability to create convergent mappings and to provide arbitrary mathematical mapping functions are effectively unheard of in commercially available systems.

On the other hand, compared to creative coding environments such as Max/MSP or Pure Data, libmapper+Webmapper is relatively limited. Easy abstraction, audio-rate processing, Turing-completeness, and huge ecosystems of external objects and abstractions are some of the advantages that these environments provide over libmapper+Webmapper. Although in actual every day use of libmapper, it is simple to interoperate with these ecosystems, participants in the present study were limited to making their mappings within the constraints of libmapper alone.

The libmapper+Webmapper environment lies somewhere in between the offerings of commercial sound and music creation systems and full blown creative coding environments. In doing so, it effectively caters to the most highly-skilled programmers and the most technically naive musicians by providing the essential features required for making mappings in an efficient and easy to use package. This balance makes libmapper+Webmapper well suited to the current investigation into the mapping design process, since it allows a wide range of musicians and music technology researchers to participate in the study without having to learn a complex new system, while still allowing for advanced usage by those who are capable of it.