University of Southampton

Faculty of Faculty of Arts and Humanities

School of Music

Graphically Interpolated Synthesis Parameters for Sound Design: Usability and Design Considerations

DOI

[Volume 1 of 1]

by

Darrell Gibson

ORCID ID https://orcid.org/0000-0002-0658-5965

Thesis for the degree of Doctor of Philosophy

December 2021

University of Southampton

Abstract

Faculty of Arts and Humanities

School of Music

Thesis for the degree of Doctor of Philosophy

Graphically Interpolated Synthesis Parameters for Sound Design: Usability and Design Considerations

by

Darrell Gibson

This research investigates the use of graphical interpolation to control the mapping of synthesis parameters for sound design, and the impact that the visual model can have on the interpolator's performance and usability. Typically, these systems present the user with a graphical pane where synthesizer presets, each representing a set of synthesis parameter values and therefore an existing sound, can be positioned at user-selected locations. Subsequently, moving an interpolation cursor within the pane will then create novel sounds by calculating new parameter values, based on the cursor position and an interpolation model. These systems therefore supply users with two sensory modalities, sonic output and the visual feedback from the interface.

A number of graphical interpolator systems have been developed over the years, with a variety of user-interface designs, but few have been subject to formal user evaluation making it difficult to compare systems and establish effective design criteria to improve future designs. This thesis presents a novel framework designed to support the development and evaluation of graphical interpolated parameter mapping. Using this framework, comparative back-to-back testing was undertaken that studied both user interactions with, and the perceived usability of, graphical interpolation systems, comparing alternative visualizations in order to establish how the visual feedback provided by the interface aids the locating of desired sounds within the space. A pilot investigation compared different levels of visual information, the results of which indicated that the nature of visualisation did impact on user interactions. A second study then reimplemented and compared a number of extant designs, where it became apparent that the existing interpolator visuals generally relate to the interpolation model and not the sonic output. The experiments also provide new information about user interactions with interpolation systems and evidence that graphical interpolators are highly usable in general.

In light of the experimental results, a new visualization paradigm for graphical interpolation systems is proposed, known as Star Interpolation, specifically created for sound design applications. This aims to bring the visualisation closer to the sonic behaviour of the interpolator by providing visual cues that relate to the parameter space. It is also shown that hybrid

visualizations can be generated that combine the benefits of the new visualization with the existing interpolation models. The results from the exploration of these visualizations are encouraging and they appear to be advantageous when using the interpolators for sound design tasks.

Table of Contents

Table of Tables

Table of Figures

Table of Figures

Research Thesis: Declaration of Authorship

Print name:	Darrell Gibson
Title of thesis:	Interpolated Synthesis Parameters for Sound Design: Usability and Design Considerations

I declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:

Gibson, D. and Polfreman, R., 2022. Analysis and Evaluation of Visual Cues in Graphical Interpolators. Proceedings of the 19th Sound and Music Computing Conference, 5-11 June 2022, Saint-Étienne (France). DOI: 10.5281/zenodo.6573475

Gibson, D. and Polfreman, R., 2021. Analyzing journeys in sound: usability of graphical interpolators for sound design. Personal and Ubiquitous Computing, 25 (4), 663-676. DOI: 10.1007/s00779-020-01398-z

Gibson, D. and Polfreman, R., 2020. Star Interpolator – A Novel Visualization Paradigm for Graphical Interpolators. In: New Interfaces for Musical Expression 2020 21-25 July 2020 Royal Birmingham Conservatoire. DOI: 10.5281/zenodo.4813168

Gibson, D. and Polfreman, R., 2019. A Journey in (Interpolated) Sound: Impact of different visualizations in graphical interpolators. Proceedings of Audio Mostly 2019: A Journey in Sound, 18--20 September 2019, Nottingham. DOI: 10.1145/3356590.3356622

Gibson, D. and Polfreman, R., 2019. A framework for the development and evaluation of graphical interpolation for synthesizer parameter mappings. Proceedings of the Sound and Music Computing Conferences, 302-309. DOI: 10.5281/zenodo.3249366

Signature: Date: 25/11/21

Acknowledgements

An undertaking of this size would not have been possible without the love and support of all my family and friends, but particularly Jenny Bouquet who's help and support has got me to this point. I also wish to thank my supervisor Dr Richard Polfreman, for all the support and guidance in our supervision sessions which sent me in new directions that I may not have considered without his unique perspective on the work. I am grateful to Bournemouth University, Creative Technology Department for paying the PhD fees, providing facilities and time towards the research. I would also like to thank everyone that participated in my experiments.

Finally, I am hugely grateful to both Richard and Jenny for their help putting together this thesis and for proof-reading many pages written by a dyslexic fool. Thank you.

.

Definitions and Abbreviations

ADR	Automated Dialogue Replacement
ANOVA	ANalysis Of VAriance
CI	Confidence Interval
CIT	Critical Incident Technique
CSI	Creativity Support Index
DAW	Digital Audio Workstation
DMI	Digital Musical Interface
DSP	Digital Signal Processing
EC	Evolutionary Computing
EDM	Electronic Dance Music
FM	Frequency Modulation
GA	Genetic Algorithms
GIS	Geographical Information Systems
GP	Genetic Programming
GPU	Graphical Programming Unit
GRM	Groupe de Recherches Musicales
HCI	Human Computer Interaction
IDW	Inverse Distance Weighting
IQR	Inter-Quartile Range
IRCAM	Institute for Research and Coordination in Acoustics/Music
ISEE	Intuitive Sound Editing Environment
ISPW	IRCAM Signal Processing Workstation

ISSD	Intelligent System for Sound Design
IV	Interactive Visualization
LFO	Low Frequency Oscillators
LoM	Library of Mapping
MIEM	Multitouch Interfaces for Electroacoustic Music
MnM	Mapping is not Music
NHST	Null Hypothesis Significance Testing
NPS	Net Promoter Score
PSDep	Probability Score Depth
RST	Regularized Spline with Tensio
SD	Standard Deviation
SDD	Standard Distance Deviation
SUS	System Usability Scale
UX	User Experience

Chapter 1 Introduction

The phrase “sound design” originated in the film industry in the 1970s [1] and since that time it has been used in many different contexts; here sound design is taken to be the design of new sounds for specific purposes. This can be considered to be the generation, synthesis, recording (studio and location) and manipulation of sound to meet a given specification or brief. That is, creating and making sounds where there are constraints on the output. Therefore, all of the following can fall within the scope of sound design: synthesizer programming, generating and recording found sounds, Foley, applying effects during audio production, etc. [2].

Sound design is required in many areas including: music production, soundscapes, film, television, theatre, computer/video games, live sound, data sonification, sonic art, etc. One important area of Sound Design as a discipline is synthesizer programming, where the designer will configure a sound synthesizer’s available parameters to give a desired output [2]. Whether the synthesizer is implemented in hardware or software, in a standard model the parameters are typically accessed through controls such as dials, sliders, switches and buttons. Figure 1 shows examples of typical synthesizer interfaces, with a hardware Korg Minilogue (on the right) and the software Xfer Serum (on the left). As can be seen, although they are implemented using different technologies the parameter controls are essentially the same. The only real difference is that software-based instruments often provide additional graphical displays for things like waveshapes and envelopes.

Figure 1 Typical Synthesizer Interfaces - Korg Minilogue (Left) and Xfer Serum (Right)

This approach relies on the designer having extensive knowledge of the particular synthesis paradigm used by each synthesizer, the internal architecture and the sound design possibilities of each parameter. The sheer number of parameters that many synthesizers possess (often hundreds and sometimes more, for example, Native Instruments FM8 has over a thousand) further compounds the difficulties, while for some forms of synthesis (e.g. frequency modulation and wavetable) the parameters' relationship to the resulting sound characteristics are not always straightforward.

In addition to these problems, sound designers require creative and critical listening skills that take considerable practice to develop, in order to move the process towards a defined sonic goal. This combination of challenges means that it can be very difficult to learn how to design sounds with synthesizers and often places effective design outside of the reach of traditional musicians and casual users.

Historically synthesizer manufacturers have addressed this problem by supplying their devices with extensive banks of presets (also known as snapshots, programs, preset patches or just patches), each of which is a configuration of multiple parameter settings, designed to generate a specific sound. Although this is satisfactory for users that only want to use predesigned sounds, it detracts from the creative process and can be restrictive. It is also of limited value to those wishing to learn the intricacies of synthesizer programming. The best they can hope for is to audition presets until they find something close to the desired sound and then attempt to modify the sound by selectively "tweaking" the used parameters. However, as modern software synthesizers possess a large number of presets, it can be a huge task to just locate a suitable starting point. This situation is also not desirable for experienced sound designers, who will often have a good idea of the sound they are aiming to create, but without considerable synthesizer experience it may not be obvious how to go about either creating it from scratch or moving from a preset to the desired sound.

This is particularly evident for more complex synthesis systems and can result in a huge amount of time when a trial-and-error approach is used. In addition, there is normally no way of working between multiple target sounds so that designers can arrange the sound in different configurations and explore the sound space defined by multiple target sounds. These limitations of synthesizers, combined with the historical origins of sound design in Foley [3], are perhaps reasons why designers often work more with recorded sound [4] than with synthetic sound.

The issues outlined above raise three distinct questions: First, is there a way that sound design can be performed without an in-depth knowledge of the underlying synthesis technique? Second, can a large number of synthesizer parameters be controlled intuitively with a set of interface controls that relate to the sounds themselves? Finally, can multiple sets of complex synthesizer parameters be controlled and explored simultaneously?

The initial background research, presented in Chapter 2, provides more context by examining research into synthesis-based sound design, exploring areas that could potentially answer these questions. This starts with an investigation of synthesizer programming techniques and leads on to both automatic techniques and interface design, before focusing on graphical interpolation techniques. As part of this a detailed literature review of interpolated parameter mapping techniques is presented. Chapter 3 then presents the evaluation of five different areas that graphical interpolation systems possess and considers previous work that has been undertaken in each of these. Through this it became apparent that four of the five areas had been examined previously. For the fifth area, the visual representation, although a number of different graphical models have been presented over the years, no comparative investigation has been undertaken. This thesis aims to fill this deficit and the exact research methodology is presented in Chapter 4. This is done through the development of a graphical interpolation framework, presented and explored in Chapter 5, that aids the development and testing of different interpolation systems. The framework was initially used to investigate the impact that different visual models had on the interpolated output and so the resulting sounds it was possible to create. Having ascertained that the graphical model that an interpolator uses, and its layout has a direct impact on the sonic pallet that it is possible to produce, a couple of fundamental questions arose: given that the graphical interpolator's output is sonic, are visual cues needed? And do they aid the process of sound design with an interpolator? Chapter 6 presents a pilot study using the framework that was undertaken to answer these questions by employing a usability testing methodology. The results indicated a strong correlation between the number of visual cues an interface presents and the perceived usability of the interpolator for sound design. In Chapter 7 using a similar methodology further usability testing is presented to compare and evaluate different existing graphical interpolation models. The results from this testing again demonstrated that the more visual cues an interface presents the user, the higher its perceived usability. Therefore, in Chapter 8 a new interpolation paradigm is defined that attempts to provide the user with visual cues that none of the existing models provide, namely for the underlying parameters values. Initial findings are presented along with the results from informal bench-testing of this interpolation model. These appear to indicate that there are a number of potential benefits to using this model that require further investigation. Finally, Chapter 9 presents overall conclusions from this body of work and recommendation for further investigation.

Chapter 2 Background

Given that sound design is a highly practice-based discipline that covers many different topics, it means that many of the areas covered are only receiving formal definition now [5]. Moreover, while sound design requires excellent technical skills, at the same time the quality of the aesthetic output is of primary importance. As a result, this research is truly multi-disciplinary and requires a wide range of literature and media to be considered in order to define.

2.1 Music Interaction

Although this work is focused on the application area of sound design there is a great synergy to the music domain and often it is difficult to define hard boundaries between the two. In recent years a number of studies have been undertaken into the area of music and human-computer interaction (HCI) and it is often now referred to as just music interaction [6]. As such, despite the specific focus here on sound design the vast majority of this chapter is directly about music interaction and applicable to the wider NIME community[1].

In recent years, the wider HCI community has broadened its focus from the first wave of HCI, towards the second and third waves [7], [8]. As this has happened so we are beginning to see music interaction make a likewise shift broadening to include more consideration of psychology and cognitive science (second wave) and then the social context (third wave) [6]. A review has been undertaken where these waves have been related to the developments within the music community [9]. This review also linked the four stages of interaction (electrical, symbolic, textual, and graphical) to specific music technologies to demonstrate the appropriateness in a music interaction context. However, a study undertaken in 2017 reviewed conference papers published in the previous three years within the music technology community to assess which aspects of interaction design were considered [10]. They assessed all the papers using the user experience (UX) dimensions of usability, generic UX, aesthetics, emotion, enchantment, engagement, enjoyment, motivation, and frustration, to establish if there were areas for further consideration. They found that the majority of papers considering music interaction focused on the UX metrics of usability, aesthetics and generic UX. A minority of papers focused on engagement, emotion and enjoyment, and very little consideration has been given to motivation, enchantment, and frustration. The evaluation also showed that most of the papers used a specific task with questionnaires and surveys to gather data. The authors considered this to indicate that the sector

[1] https://www.nime.org/

has a focus on traditional HCI, but that there appears to be movement towards UX approaches. Further consideration of all the UX dimensions and the ability to explore the user's subjective experience will help to provide a fuller consideration of music interaction, even in a sound design context.

2.2 Sound Design

What is known as sound design today, evolved over a period of many years. The main driver came from the film industry when the "talkies" were introduced in the 1920's [11], resulting in advancement of what had been previously done in the theatre and radio industries. The marriage of image and sound, over the following decades, borrowed heavily from the live music aspects of the silent film era and also the sound from theatre [12]. Despite film being primarily a visual medium, the soundtrack can be used to augment the narrative, provide emotional signifiers and grammatical underpinning [13]. Therefore, the direction that the soundtrack takes can have a huge impact on the overall aesthetics of the film [14].

Film productions generally have five distinct phases [15], being:

1. Development
2. Pre-production
3. Production
4. Post-production
5. Distribution

The first four stages relate directly to the making of the film and as a result the soundtrack should be considered and worked on at each stage. On a typical film the final soundtrack will comprise of the following sound elements [16]:

1. Dialogue – spoken and narration
2. Music – diegetic and non-diegetic
3. Sound Effects – Foley, hard effects (shown actions and events), soft effects (off screen sounds) and designed sounds
4. Backgrounds – room tones, ambiences and atmospheres

Ideally all of these should be considered as early in the production process as possible, although in practice there is a tendency that sound does not get given the consideration that it deserves [17]. Moreover, although sound is recorded on the production set, the primary focus is on capturing dialogue and due the set being a noisy environment, often the quality less than ideal. As a result, most of the work on the soundtrack tends to be focused in post-production where dialogue is re-recorded as well as adding sound effects and music. As the years have progressed, more emphasis has been given to the manipulation of the audience through the use of sound. This is one of the reasons why the role, known as sound design, where sounds are created for a particular goal is considered to be so important in the modern film era [18].

2.2.1 Evolution of Sound Design

What can be considered as the modern age of sound design developed over several decades, from the 1970s to the present day [19]. Two sound designers that helped to forge this transition were Ben Burtt, who worked on the Star Wars films [20], [21] and Walter Murch who created the sound for Apocalypse Now [22]. These two not only designed specific sound effects for the films that they worked on, but they also started to take on an overriding supervisory role for the sound in films that spanned from pre-production, through to production and on into post-production. This allowed the individual elements of the soundtrack (dialogue, music, sound effects, room tones, ambiences, Foley, ADR, etc.) to be considered with respect to the whole soundtrack rather than being separate, unrelated elements that are combined in the final stages of the film production. In this way, the elements of the soundtrack can be blended together to create additional dimensions to the narrative. As a result, the soundtracks for the films that Burtt and Murch have worked on are renowned for their overall aesthetics that helped to add another level to the film viewing experience. This is backed up by the awards and nominations that they have received for their work since the 1970s [23], [24].

Another important aspect of the work from Burtt and Murch has been their use of synthesized sound and well as recorded sound. Burtt created a “voice” for a robot character, “R2-D2”, in the Star Wars films using an ARP 2600 analogue synthesizer and his own voice to apply vocal articulations [25]. Likewise, in the film Apocalypse Now, Murch deconstructed the sounds required, such as helicopters, into a number of individual synthesized sound components. These could then be combined together to create a realistic sound of the helicopters or they could be used individually to create different perspectives [26]. Murch has said that he did this this so that the sounds could be “positioned between realism and hyper-realism and surrealism”, to pull the audience in an intended direction. As well as the work of these two pioneers of sound design there are many further examples of the use of synthesizers in film sound [27].

Since these early days of sound design the discipline has matured and established itself in many fields as well as film, such as television and radio [28], theatre [29], music [30], effects [31], vocalisation [32], soundscapes [33], cartography [34], virtual reality [35], games [36] [37], website [38], product design [39], sonic interactions [40], robotics [41], electric automotive [42], user interfaces [43], audio-visual interaction [44], auditory displays [45], sports [46], etc. The huge diversity shown in this selection show that sound design has many different application areas and has truly become a multi-disciplinary field.

Although as mentioned, synthesis techniques are used in sound design in many areas there is still a heavy reliance on recorded sound content, particularly for synchronised sound effects that accompany visual content [17]. Nonetheless, there are many examples where synthesised sound has been used [27], [31], [47], [48], [49], [50], [51], [52], [53], [54]. Most of these examples used standard music synthesizers to generate audio which is exported to the target application, but with the advancement of computer programming the use of Procedural Audio is a realistic possibility. This is the use of real-time synthesis models that are typically under programmatic control [55], for example, to generate the sounds of rain, fire or other phenomena. As a result, the use of procedural audio is becoming more popular for the design of sounds in computer games as decisions for the sound can be left until run-time and can adapt to game-play scenarios, rather than being defined at the time of production [56]. This trend looks set to continue in the games industry and will possibly find a home in other areas where sound design is used. This is discussed further detail in Section 2.3.3.

2.2.2 Designing Sounds

Although in the film industry sound design is considered to be a supervisory role, it still has a lot to do with the actual design of new sounds [19]. This will usually be for a specific brief, specification or context defined by the visual requirements, rather than for purely aesthetic goals. As mentioned previously, both Burtt and Murch gained reputations for designing new sounds. For example, Burtt personally designed many of the sounds in the Star Wars Universe which have become recognised well beyond the scope of the original films. To give some context to this kind of practice-based process, a summary of the sound design of the lightsaber (a laser-sword) from Star Wars is given, as explained by Burtt in an interview [23]:

> *“The lightsabers are one of my favourite sounds, and in fact it was the very first sound I made for the whole series. For some reason after I read the script even though my assignment was to find a voice for Chewbacca, and then a voice for Artoo, and then, well maybe come up with some sounds of laser guns and other things. The lightsaber fascinated me at the time when the script had first come out, they had some* paintings *that Ralph McQuarrie had done. So that there were some concepts visually of what some of these things would look like, and those pictures were very inspiring because they gave an idea of the direction we were trying to go in the look of the film and it was inspiring to me to therefore think of sounds that might fit that kind of visual style.*

I could kind of hear the sound in my head of the lightsabers even though it was just a painting of a lightsaber. I could really just sort of hear the sound maybe somewhere in my subconscious I had seen a lightsaber before. I went to, at that time I was still a graduate student at USC, and I was a projectionist and we had a projection booth with some very, very old simplex projectors in them. They had an interlock motor, which connected them to the system when they just sat there and idled and made a wonderful humming sound. It would slowly change in pitch, and it would beat against another motor, there were two motors, and they would harmonize with each other. It was kind of that inspiration, the sound was the inspiration for the lightsaber and I went and recorded that sound, but it wasn't quite enough. It was just a humming sound, what was missing was a buzzy sort of sparkling sound, the scintillating which I was looking for, and I found it one day by accident.

I was carrying a microphone across the room between recording something over here and I walked over here when the microphone passed by a television set which was on the floor which was on at the time without the sound turned up, but the microphone passed right behind the picture tube and as it did, this particular microphone produced an unusual hum. It picked up a transmission from the television set and a signal was induced into it's sound reproducing mechanism, and that was a great buzz, actually. So I took that buzz and recorded it and combined it with the projector motor sound and that fifty-fifty kind of combination of those two sounds became the basic lightsaber tone, which was then, once we had established this tone of the lightsaber of course you had to get the sense of the lightsaber moving because characters would carry it around, they would whip it through the air, they would thrust and slash at each other in fights, and to achieve this additional sense of movement I played the sound over a speaker in a room.

Just the humming sound, the humming and the buzzing combined as an endless sound, and then took another microphone and waved in the air next to that speaker so that it would come close to the speaker and go away and you could whip it by, and what happens when you do that by recording with a moving microphone is you get a Doppler's shift, you get a pitch shift in the sound and therefore you can produce a very authentic facsimile of a moving sound. And therefore give the lightsaber a sense of movement and it worked well on the screen at that point."

From this example, there are a number of important aspects that can be drawn: First, the sound design for this sound effect clearly started at the very beginning of the production process, which allowed the sound element to potentially influence other aspects in the production; Second, this sound effect was designed for a specific application, in this case, defined by the conceptual paintings. Although there are aesthetic choices that will be made with such a creative task, this is different to a purely artistic process where there are no constraints; Finally, although a single sound element is being designed, it is noteworthy that it consists of three sound components: first the sound of the inter-locked motors, then the static interference from a television set and finally the Doppler effect with level changes that supplied the sense of movement. Although this is just a single example it has a lot in common with other sound design tasks and demonstrates the kind of complexities involved in the design of new sounds.

2.2.3 Sound Design Workflow

In the Star Wars example given above, it is clear that Burtt was involved in the project from the very early stages of development. However, this is not always the case and often a sound designer is not hired until the final stages of the production [57]. Equally a sound designer may be hired at any point in the production process and will be expected to pick-up the design process from whatever stage the sound production is currently at and in some cases, a sound designer may not be used at all. That is not to say that sound design does not occur, but that other members of the sound production team may be doing it or stock sound effects libraries are used. To fully understand the typical workflow of a sound design task, it is first important to understand other roles within the sound department.

The Production Recordist, often called the Production Sound Mixer, is the person on the "set" during the actual capture of a production (film, television, game, etc.), that records the on-set sound. They will record the dialogue, room tones, background and atmospheres and any sound effects that are available during the actual shooting. Depending on the budget and size of the production the Production Recordist may also have assistance from a Boom Operator that will position and move microphones during shooting and possibly a Cable Guy or Sound Technician that will help out generally [17].

The Sound Editor will work in a post-production studio and is responsible for assembling and editing all the sound components in the soundtrack. There may be a separate Dialogue Editor and/or Music Editor who would be responsible for their content in the soundtrack or these responsibilities may fall to a single Sound Editor to do them all. The process is usually begun with "spotting" sessions where the visual content is played with the sound recorded during the production, with the purpose of deciding where additional sounds, music and dialogue are required. As a result, other people maybe hired to provide any additional content, such as a Location Recordist, Foley Artist, Music Composer or ADR (Automated Dialogue Replacement)

Dialogue Recorder[2]. The role of the editor is to edit the recordings and to assemble them into the soundtrack and synchronise them to the visual footage [17].

The role of the Sound Mixer, sometimes called Re-recording Mixer, is to blend together all the different sound elements that make up the soundtrack. During the mixing, the edited sound elements, such as dialogue, ADR, backgrounds, room tones, atmospheres, sound effects, Foley and musical elements, are balanced. To manage the complexity of mixing, providing flexibility and keeping the mix from becoming overwhelming, individual sub-mixes, called "stems" are created. These sub-mix stems (dialogue, ADR, group walla[3], backgrounds, room tones, atmospheres, sound effects, Foley, music, adds, extras, etc.) are easier to manipulate and update during the mixing process. The sub-mixes are combined together to eventually create a set of Final Mixes for Dialogue (DX), Effects (FX) and Music (MX). From these the Full Theatrical Mix is created with discrete channels for all the relevant output formats. In the film industry, the music and effects (M&E) is often kept as a separate mix so that foreign language tracks can easily be added [17].

As mentioned previously, one of the roles of the Sound Designer is to take a supervisory role for the production's sound and would ideally be involved at all stages of the process, working with the Production Recordist, Sound Editor and Sound Mixer. Assuming a Sound Designer is appointed, the process would start with reading the script and meeting with the Director to establish aesthetic direction that will be taken on the project. Advice and suggestions can then be given to the Director about the sounds required for the production. In this way, it is possible for the sound to influence the picture choices, and vice-versa [57]. The Sound Designer would also be present during the filming to gather sounds and see what is going on so they can begin designing and make sounds that they anticipate will be required. They will also be on hand during the editing of the film to pick out sounds from the library of designed sounds and edit them as required and synchronize them with the screen action. Finally, they will be involved in the sound mixing so that they can ensure that the projects goals are met, usually in direct consultation with the director [58].

Historically all of these processes would have been carried out using magnetic tape machines, which allowed content to be recorded, played back and edited. Editing would have been performed by physically cutting and splicing sections of magnetic tape together and the results could be re-recorded with another tape machine. The mixing would have then been done using a

[2] Recording and replacing dialogue delivery from the actors
[3] Recorded background vocalisations

mixing console that allowed multi-track taped content to be played back, balanced, equalized and the results were then re-recorded onto a different tape machine. These days the processes are typically performed using a software-based Digital Audio Workstation (DAW). These applications allow digital recording, editing and mixing of the audio content all within a single software application. There are many DAWs on the market that offer similar functionality, but Avid's Pro Tools has established itself as the de facto market leader in the film/television production industry [59].

The workflow that is of particular interest here is the point where the Sound Designer actually engages in the process of designing new sounds. As with any practice that has a creative output the exact procedures used to develop and produce the content are not set-in-stone and can be highly individual. Nonetheless, certain common characteristics can be established. There is no defined point in the production process where the design of new sounds occurs. Sometimes this will happen at the very inception of the project, where the sounds can then start to have an impact on the visual production. Equally, it is not unknown for it to suddenly be decided during the final mixes that certain sounds are not working with the visual content and then either additional work is required on the existing sounds or completely new sounds may be needed. Wherever in the production process that the design actually takes place, today it is likely that the workflow will centre around a DAW. That is not to say that other devices and software will not be used as sources, but content will invariably be brought back into a DAW for editing and mixing.

As mentioned previously, a lot of designed sounds will be made up from a number of individual sonic components. Historically these are often captured as audio recordings, frequently processed in some way and combined and edited in within a DAW. Audio recordings have the advantage of being highly accurate and realistic representations of a particular sound, with complex and varied sonic textures, but once the sounds are recorded it is difficult to change individual aspects of the sound, as the sounds are not under full parametric control. Many of the formal definitions in this area have been derived from the principles of *Musique concrète* and *electroacoustic* music [14]. In these areas, the idea of describing the morphology of sounds, that is, the properties of a sound with respect to time, has been well established and refined [60]. Based on this idea, it is then possible to define morphological concepts that allow the manipulation of these properties. Many technologies have been defined that implement such morphological concepts and these have been categorised [61]. As each category defines a different type of operation that modifies the sound properties with respect to time and as recorded sound itself time-based, it is reasonable to also consider them sound design operations. Therefore, the following operations are defined [61] for how sound design can be performed with recorded sound:

1. Sound isolation and observation – recording, listening and viewing sounds
2. Sound editing – cutting-out, loops, time inversions, substitutions, incrustations (augmenting with fragments of different sounds)
3. Dynamic modifications – amplitude modulation, envelope changes, noise reduction or elimination, compression, expansion
4. Speed modifications – speed variations, phase variations, Doppler effect
5. Time modifications – time stretching, time contraction, time freezing, looping, reversing
6. Spectral modifications – filtering, resonant filtering, harmonisation, ring modulation, spectral interpolation, analysis-resynthesis, distortion, formant shifting, pitch shifting
7. Density modifications – shuffling, feedback, multiplication
8. Order of events modifications – shuffling, editing, brassage
9. Spatial modifications – panning, circling, Doppler effect, reverberation
10. Sound combination – mixing, layering, sound interleaving, vocoding

Although many of these operations can be performed within a DAW there are often situations where external systems (hardware and software) will be used. Following such an operation the resulting audio would be reimported into the DAW.

Sound design is often experimental in nature where a large number of possibilities may be tried. The exact way in which the sound components are captured, selected, processed and combined will invariably be different every time and this is what provides the diversity and variety required. As a result, the design process is often very time consuming and may last months, before the desired sounds are created. For example, Burtt was given a year to create sounds for the original Star Wars film [62].

The experimental nature will also mean that a sound designer will often create a large number of alternative sounds for a single sound effect. These will then be placed into a library so that during the post-production sound editing, where the sounds are edited and synchronised with the visual content, an appropriate sound selection can be made based on the context given by the visuals and the intended direction for the content. This means that creating hundreds of sounds is not uncommon and sometimes it can even stretch into the thousands.

2.2.4 Sound Design in Electronic Music

Another arena where sounds are designed is in Electronic Music and the many different genres of Electronic Dance Music (EDM), which are prevalent within popular culture [63]. This kind of music is generally made with electronic instruments and as such, each of the sounds within the music will require designing. Although there may not be a definite brief or specification, for each of the sounds being designed it is important that the individual sound elements being created will work together with the composition and meet the creative and aesthetic goals [2]. As a result, many artists of electronic music will also be sound designers, particularly with synthesizer technology. However, unlike the sound design in film production this is on a much smaller scale and typically will only involve a couple of people at most. Moreover, sound design in this area will tend to possess more of a focus on musical aspects than sound design for moving image. For example, a snare or kick hit may have similar characteristics to film sound design, but when designing melodic or harmonic sounds further consideration would also need to be given to musical aspects such as articulations, note transitions and expression, such as vibrato, bends, note dynamics, portamento, etc.

2.3 Sound Synthesis

Sound synthesis is the process of creating sounds using signal processing techniques to construct or build audible outputs. Originally this was done using mechanical techniques, such as Luigi Russolo's mechanical synthesizers, called Intonarumori [64]. The introduction of electrical systems permitted electro-mechanical instruments, such as the Helmholtz Sound Synthesiser, to be created and these led to the design of entirely electrical instruments, like the Theremin and Ondes Martenot. The 1960s and 1970s saw the emergence of commercial synthesizers that were implemented using analogue-based circuitry, resulting in new techniques and many different realizations. In the 1980s and 1990s the use of digital electronic technology for the implementation of synthesizers was pervasive and today many synthesizers are built with software-based realizations [65]. Despite the advancement in technology there are many examples of older sound generation techniques still being used in contemporary sound design. For example, many original hardware synthesizers are highly sought after and even mechanical synthesis devices, such as rain sticks, thunder sheets, door slams, bird whistles, etc. are still used today.

One of the unique features of synthesizer technology compared with many traditional instruments is that they usually present two different interfaces to the user, one for the programming of the sound generator and the other for the actual musical input. Despite much research into the design of new performance interfaces the use of a keyboard-based interface is still very popular. This is because it has a fixed arrangement of notes and is familiar with anyone that has learned to play the piano or organ. The sound from early analogue electronic synthesizers was programmed by the user directly controlling variable component values in the electronic circuits via dials, sliders, switches, and buttons. Setting the controls into various different configurations would then program a particular sound.

Although synthesizers do have two interfaces there is not a complete separation between their uses. For example, it is not uncommon for a performer to change the synthesis parameters and so create new sounds as part of the performance. Equally when programming a new sound there is a relationship back to the performance where the sound will be used. That is, a sound being designed is a function of a particular performance where it will be used and what can be performed is a function of the sound being designed. Therefore, while programming a sound it will often be played or performed. This can either be through direct interaction with the interface or could be driven by a sequencer that allows the generation of an arrangement of musical notes. Nonetheless, a performance is supplied to allow the sound to be programmed. This may even extend so far as playing accompanying sounds so that it is possible to ascertain if the sound being programmed will work within the context of the other sounds it will be used with in a particular composition.

Although over a number of years the synthesis technology has changed, this programming interface paradigm still remains popular, especially with commercially available devices. Often users are still presented with individual controls that will allow the modification of a single parameter of the sound generation implementation. For example, as shown in Figure 2 Native Instruments' Massive's programming interface consists of controls directly mapped to individual sound parameters. In terms of actually designing sounds, it is this interface that is of primary importance. How does a sound designer go about designing a sound for a certain specification or brief using this interface? It generally relies on experience and understanding of the underlying synthesis architecture.

Figure 2 Programming Interface on Native Instruments Massive

As can be seen from this example it is not uncommon for synthesizers to possess a large number of parameters, resulting in a huge sound space. For example, assuming even a modest number of parameters, say ten, each with a MIDI range of 128 possible values results in a sound space with $128^{10} = 1.180591621 \times 10^{21}$ possible outputs, although some of these may be difficult to distinguish. Clearly even for a highly experienced sound designer it would be impossible to be fully aware of the entire parameter space and its relationship to the sound space that it defines. This is particularly the case where there is not an obvious relationship between the parameters and the sound generation characteristics, as is the case with some methods of synthesis.

As already mentioned, even though manufacturers supply their instruments, particularly software ones, with a large number of pre-designed sounds, this presents a further issue of identifying a particular sound from those available. This is further compounded by the use of creative names for the preset files that bear little relationship to the sound generated. Some manufacturers use a category-based system to aid location and others use a sound attribute system, as shown in Figure 3. However, in both of these cases the sounds available in a particular selection can be hugely different.

Figure 3 Preset Attributes System on Native Instruments Massive

2.3.1 Sound Synthesis Methods

As mentioned, there are many different types of synthesis and also many realisations for each with varying architectures. The result is that the type of synthesis and its precise implementation will have a direct impact on the sounds that it is possible to create. For example, on a subtractive synthesizer the number of oscillators and their waveshapes will directly impact on the sounds that it is possible to create.

It is not possible to cover every type of synthesis here, but the following broad categories exist: Additive, subtractive, frequency modulation, sampling, wavetable, granular and physical modelling [66], [67]. Although there are numerous other types of sound synthesis that have been implemented over the years, they tend to either be variations of these major themes or hybrid solutions that use various combination of these techniques. More details on the on the theories of these can be found in the book by Curtis Roads [66] and practical realisation for many of these techniques are in the book by Martin Russ [67].

2.3.2 Synthesizer Architectures & Implementations

The exact sound space defined for a particular synthesizer is not only a function of the type of synthesis technique that is used, but it is also a function of precisely how a synthesizer has been realised. For example, over the years there have been many commercial and academic realisations of subtractive synthesis, but each sounds different and offers a different range of sound possibilities. This is a function of two factors: first, the exact architecture that has been used, that is, which sound generation and manipulation components are available. For example, on a subtractive synthesizer the number of oscillators available and their exact waveforms will directly affect the range of sounds that it is possible to create with it; secondly the precise realisation of each of the sound generation and manipulation components will result in different sound possibilities. For example, oscillators implemented digitally will sound different to those built with analogue circuitry. Moreover, different analogue circuit implementations will sound different and even just the use of different analogue components in the same circuit can result in different sounds being generated[4]. This is why certain vintage synthesizers are so revered for their particular sound.

2.3.3 Procedural Audio

With the advent of faster and faster computers there has been more of a move to use Procedural Audio. There have been many different definitions for procedural audio, but the one by Andy Farnell has been widely adopted and states: "Procedural audio is non-linear, often synthetic sound, created in real time according to a set of programmatic rules and live input" [55]. This is actually a very broad definition and Farnell himself has used a more succinct definition that "any sound can be generated from first principles, guided by analysis and synthesis" [68]. Nonetheless, what is being considered here is the real-time creation of sound using procedural models or algorithms. Given the broadness of the definition these models are not specific to sound effects and do not solely deal with sound synthesis techniques. However, both of these do constitute a major element of the work being undertaken in this area. Moreover, unlike the synthesis types mentioned already (Section 2.3.1), that have fixed architectures, these models tend to be custom designed and specific to particular types of sounds. There have been many models proposed for different types of sounds, including, solid object [69], water [70], rain [71], fire [72], cloth [73], birds [74], insects [75], mammals [76], footsteps [77], motors [78], engines [79] and many more [68].

Using such synthesizer models offers a number of advantages over recorded sound. First, the sound can be completely isolated from all other background and ambient sounds. Moreover, individual components of the overall sound can be isolated as previously mentioned in the earlier (Section 2.2.1) Walter Murch example. Second, as every aspect of the sound is under parametric

[4] Example of comparison between three analogue subtractive synthesizers: https://www.youtube.com/watch?v=L89eqV_BlB8

control the Sound Designer has greater flexibility: if a director were to say the sound needs more "impact" or has to sound "darker", provided the designer knows which parameters will affect these adjective terms, the sound can be changed. In addition, the parameters give fine control of the sound so different expressions and articulations can be applied so every sound can be unique. Although this can be done with recorded sound, it is more complex and it is not possible to adjust the sounds with the same fine detail. In addition, although the procedural models given as examples earlier, have each been designed for specific sound effects, they do each offer sound design possibilities that could be manipulated, modified and combined to create designed sounds.

The use of procedural audio principles also has a number of advantages for interactive systems, such as games and web-sites. First, with this type of media the interactive element means that the soundtrack maybe non-linear. Therefore, being able to trigger sound events, based on the user's interaction offers a better solution. Second, the parametric control of the models means the soundtrack can be changed based on the user's interaction. For example, the music tempo could be directly linked to a player's gameplay or the sound effects could be given more impact depending on previous user inputs. Finally, with most computer-based systems memory storage and processor load are often constraining factors. Generally procedural audio models will use significantly less memory than a corresponding audio recording and potentially less processing load depending on the exact architecture and implementation [56].

Although these procedural audio factors do not directly relate to sound for traditional linear film-based content, the flexibility and creative possibilities of synthesised sounds seems to be attractive to some sound designers. For example, Harry Cohen the Sound Designer for the film Django Unchained (2012) has said that procedural audio "was very useful for dialling in just the right character for our wind ambiences, and for creating the missing colours we needed in the backgrounds for Django" [80]. As a result, there are an increasing number of commercial software products becoming available [81], [82], [83].

2.3.4 Synthesizer Programming

The basic problem with programming a synthesizer is how to setup the parameters of the synthesizer to create the desired sound output. Hitherto there has been significant research in the area of synthesizer programming, which can be separated into two foci. First, the automatic programming of a synthesizer to try to replicate target sounds or meet a given specification. Second, improving the programming interface so that the details of the underlying techniques are not visible, but can be controlled.

2.3.4.1 Automatic Programming

The automatic programming interfaces tend to exploit resynthesis techniques, where a "target" sound is supplied and the system attempts to replicate the target sound with a synthesis engine. These techniques are either used for recreating sounds without having to understand the synthesis engine or to populate a search space for sound design. Resynthesis approaches can be separated into two categories: analyse the target sound and then based directly on the results, the synthesis engine is programmed from the analysis parameters [67]. The other category uses Evolutionary Computing (EC) methods to program the synthesizer based on analysis of the supplied target [84] or some form of user interaction [85].

Over time these techniques have been applied to virtually every possible synthesis technique, mainly to try to synthesise the sound of acoustic musical instruments [67]. However, the suitability of these techniques for sound design purposes is limited as they assume that the target sound is available or can be specified. If this were the case, it would not be a true design scenario and would not satisfy the exploratory requirement. Nonetheless, in the following two sections a brief overview is presented of some work that has focused specifically on sound design in these two areas.

2.3.4.1.1 Sound Design with Analysis-Based Resynthesis

As already mentioned in Section 2.3.4.1, the idea of analysis and resynthesis is not new and has been implemented many times using frequency analysis of the target sound and additive synthesis to build a representation of the target spectrum [86]. A popular technique for the implementation of the analysis stage has been the use of a Phase Vocoder [87], although other techniques exist. In practice this basic premise has been refined many times. A recent example of this was presented by Kreutzer, who proposed an efficient additive-based resynthesis engine that claims to provide larger flexibility for the user and reduces the number of synthesis parameters compared to traditional methods [88]. In addition to the work being done to refine the synthesis process, others have also examined how the process is driven. An example is PerceptSynth, which is controlled with perceptually relevant high-level features, such as pitch and loudness [89]. The developers then defined a framework for training, tuning and evaluating their system in various audio and musical application areas. Sethares also presented tools for manipulation of the spectral representations of sounds between analysis and resynthesis [90]. This then gives a mechanism to dynamically change the tonality of the sound and create morphing effects.

Using similar additive resynthesis principles, TAPESTREA, is a complete sound design framework that facilitates the synthesis of new soundscapes from supplied audio recordings, through interactive analysis, transformation and resynthesis [91], [92]. The system first analysed the recordings using sinusoidal analysis to isolate and extract deterministic sounds. Then transients are also isolated and extracted, leaving just the stochastic background sound, which is then parameterized by wavelet tree analysis. The individual sound events were then saved as templates so that various transformations could be applied to the individual events. It was then possible to build new complex environmental audio scenes, using a graphical interface, constructed from the resynthesized templates. It was also shown that it could be used as a "workbench" for performing Musique Concrète or Acousmatic compositions and other sonic sculpting tasks [93].

Klingbeil demonstrated another interesting resynthesis system, called SPEARS [94]. This system performs analysis based on the McAulay-Quatieri technique and represents the sound as lots of individual sinusoidal partials, each corresponding to a time varying frequency and amplitude. The system then offered the flexibility to edit and manipulate the sinusoidal model to create new sounds. It was then shown that these principles could be applied to a number of different musical compositional applications, with particular attention to the needs of spectral composition [95].

Much work has also been published on the analysis of acoustic audio features, of the sort used in music information retrieval and other sound analysis applications [96], [97]. These techniques are now being applied to a resynthesis paradigm. In this manner, Hoffman presented a framework for synthesizing audio with sets of quantifiable acoustic features that have been extracted from supplied audio content [98]. This permitted the synthesis of sounds based on the features and allowed the creation of novel musical timbres as well as applications in other areas.

Although not technically resynthesis, similar analysis has been applied to a corpus-based concatenative synthesis technique by Schwarz, called CataRT [99]. A corpus is built from input audio that is granulated, analysed and descriptors allocated for specific sound characteristics. These then populate a descriptor space so that grains that possess similar descriptors are in close proximity. It then allowed user-driven parameter settings to be generated and new sounds could be created as the user navigated the space and the grains are concatenated. This allowed the creation of new sounds based on the input audio, selected analysis and the user's interaction with the descriptor space. This system has been used in a wide variety of different musical contexts [100].

2.3.4.1.2 Artificial Intelligence Techniques in Sound Design

As already mentioned in Section 2.3.4.1, the idea of using Artificial Intelligence has become increasing popular in the area of synthesizer programming. An early knowledge-based system by

Miranda, 1995, called ISSD (Intelligent System for Sound Design), represented sounds in terms of their attributes (brightness, openness, compactness, acuteness, etc.) and how these attributes mapped to subtractive synthesis parameters for formants [101]. In 1998, Miranda further expanded this idea and implemented a system called ARTIST [102] and applied it to different synthesis algorithms. This system used Machine Learning to infer which sound attributes should be considered by making analogies with other known sounds, which have similar constituents.

More recently there has been much work on the use of EC techniques for the programming of synthesizers. In 2001 Garcia developed a system where Genetic Programming (GP) was used to design a population of synthesis topologies, consisting of oscillators, filters, etc. The sounds generated by individuals in the population were then evaluated to establish how closely they matched the target [103], [104]. In this way, it was possible to “grow” and design new synthesis architectures and so the resulting sounds.

Another AI technique that has been employed for synthesizer programming is the use of Genetic Algorithms (GA). These have been used to search large parameter spaces for target sounds, based on user interactions [105]. Then in 2003 Johnson refined this so that the new population was generated based on a fitness proportionate selection, where the higher the fitness rating given, the more likely it is to be selected as a parent [106]. GAs have also been used with fuzzy logic to allow the user to make explicit associations between twelve visual metaphors presented by a particular sound [107]. McDermott, 2005 – 2008 proposed a new interface for the design for interactive EC, which allows faster evaluation of large numbers of individuals from the population, based on user interaction [108], [109], [110].

As well as these interactive systems, in 2008 Yee-King presented an unsupervised synthesizer programmer, called SynthBot [111]. This was able to automatically find the subtractive synthesis parameter settings necessary to produce a sound similar to a given target, using a GA. In addition, in a more recent study by Dykiert, 2011, GAs were suggested as a mechanism to reduce the size of the parameter search space [112]. Finally, it should be noted that as well as synthesizer programming, in 2004 Miranda has also shown how EC can be applied to the compositional process [113].

2.3.4.2 Synthesizer Programming Interfaces

The programming interface that a synthesizer presents to the user is often a direct mapping of the synthesis parameters rather than related to the output sound, and in many cases, follows the interface of vintage hardware synthesizers [114]. As a result, the user is invariably presented with individual controls that directly vary individual parameters of the synthesis engine. This is known as a one-to-one mapping between the control and the synthesis parameter. More sophisticated mappings can be produced if one-to-many, many-to-one, or many-to-many strategies are used [115]. In this application domain, it would be desirable to have a one-to-many or few-to-many mapping. That is, a small, easily manageable number of control parameters on the programming interface for the user to interact with, being used to change a large number of synthesis parameters on the synthesizer [116], [117]. Therefore, it becomes a dimensionality reduction challenge [118], [119].

Simple examples use a two-dimensional space where a locator can be positioned within the space and used to control two parameters simultaneously, such as is available on the Korg Kaoss Pad [120]. In actual fact, this is still a one-to-one mapping as the position is a function of the X-Y location within the space. However, it does allow the user to simultaneously explore the sound space defined by the two parameters or express trajectories within the space that represent specific sounds or expressions.

Various proposed solutions [121], [122] have examined more complex mappings between the synthesizer parameters and the synthesis engine. However, it should be noted that this has invariably been done with respect to mapping performer expressions or articulations on the synthesis engine, rather than as a sound design mechanism. However, as several people have noted, using these techniques it is possible to create new sounds through exploration of the parameter space [123], [119]. This kind of exploration is ideal for sound design as it is possible to define a pallet of sound components or textures and then use these to create new sounds. Another important aspect is the possibility to discover or create unexpected results that turn out to fit the specifications. These requirements can be met through the use of interpolated parameter mapping systems.

2.4 Interpolated Parameter Mappings

The basic problem with programming a synthesizer is how to set the parameters to create a certain sonic output. As already stated, many synthesizers have a large number of parameters and although having direct access to every parameter (*one-to-one* mapping) gives very fine control of the sounds, it can complicate the sound design process. However, it is possible to map a smaller number of control parameters to a larger number of synthesizer parameters (*few-to-many* mapping) to reduce the control complexity. The sound can then be modified by changing the control parameters and interpolating between the synthesis parameters. That is, the synthesis parameters for “known” sounds will be associated with different control values. Then as the controls are changed from these associated values, new values for the synthesizer parameters will be generated by interpolating between the values for the known sounds. In this way, it is possible to create sonic changes that are constrained by the known sounds and the changes of the control parameters. This will provide a mechanism for exploring a defined sound space.

A number of such interpolation systems have been proposed and these can be categorised based on whether the control mechanism is via some form of visual graphical interface or some other medium.

2.4.1 Non-Graphical Interpolation Parameter Mapping

Several non-graphical interpolation techniques have been employed to map to synthesizer parameters. These have not necessarily been proposed as a mechanism for designing or creating new sounds, but as a way of mapping controller-based expressions or articulations to sounds. This is important when considering the design of new instrument controllers so that musically useful outputs are generated.

Given an instrument-based controller with a number of control parameters, a generally smooth mapping must be constructed for the degrees of freedom, defined by its control parameters (or dimensions, geometrically speaking), to a probably larger number of synthesizer parameters. To produce musically useful results suitable mappings between the control values and the synthesizer parameters will be required. These can be built up from a pointwise mapping, where particular combinations of input values are associated with specific output parameters (sounds). An interpolator can then be used as a mechanism for producing new output sounds for intermediate control inputs [118]. In most cases this will be a situation where a small number of control values is being mapped to a larger number of synthesis parameters, in other words a *few-to-many* mapping [117]. As this is a dimension reduction problem, and a high-dimensional interpolator is required. Several authors have highlighted the importance of such mappings in the design of new musical instruments [117], [122], [124].

Although, these mapping and interpolation techniques are not necessarily a mechanism for programming new sounds on a synthesis engine, these techniques are included in this literature review for their close relationship to graphical interpolation parameter mappings. In fact, graphical interpolation systems are often a 2D version of the same principles, but moving to higher number of dimensions in the control space can be difficult to represent visually. Given that these systems have been proposed for use with instrument-based controllers they tend to require greater dimensionality within the control space. In addition, a number of the actual interpolation calculations are the same, all be it with different, non-graphical, control interfaces. These interpolation techniques could be used for sound design and controlled with a visual interface in future systems.

2.4.1.1 Grid-Based Interpolation

In the early 1990's a grid-based interpolation scheme was proposed that provided a general way to map *N* articulations of performance parameters to *M* synthesizer parameters through interpolation [125]. This was done by placing the *M* synthesizer parameters on a geometric grid in an *N* dimensional performance space. Hence, creating a lattice arrangement where the control points are associated with points in synthesis space. This results in exact, pointwise mapping between some collection of points in controller space and some corresponding points in synthesis space. To interpolate to other points within the grid, a scheme was developed that partitions each hypercell into simplices. When a point is put into the lattice the controller outputs a parameter stream and an algorithm determines which subdivision the point lies in. If the point is not at a vertex of the lattice, that is, a known data point, the vertices of the simplex, which contains the point is determined. This scheme differed from multi-linear interpolation (see next section) in that it reduces the number of necessary operations and produces a flat response that allowed real-time control of a synthesis engine, at a time when computer memory and processor speed were very different to those today [126].

2.4.1.2 Multi-Linear Interpolation

Later in the 1990s, work was done on a sound synthesis environment called ESCHER [127]. This was a modular system that provided synthesis-independent prototyping of gesturally-controlled instruments by dividing the system into two components: gestural controller and synthesis engine. To achieve this, mapping between them took place on two independent levels, coupled by an intermediate abstract parameter layer. This created a multi-layer mapping hierarchy with an abstract parameter layer between the control space and the synthesis space. The separation of the components and their mappings allowed flexibility in choice of controllers and sound synthesis methods so either could be changed independently of the other.

In this system, the abstract parameters of D-dimensions were placed between the controller space and synthesis space. The first mapping layer is described as an adapter between controller parameters and the abstract parameters. The known points in the synthesis space are stored in a D-dimensional geometric lattice. The second mapping layer, between the abstract layer and synthesis parameters, used a multi-linear interpolation based on the 2^D points in the hypercell that contains the input point. This method is continuous and is differentiable, but results in discontinuities at the joins between hypercells. Another problem with this method is the computation required increases exponentially with the dimension of the space [126].

2.4.1.3 Simplicial Interpolated Mappings

In the late 1990s and early 2000s more work explored interpolated mapping between performance controls on an instrument interface and the parameters of a synthesis engine. Given the number of controls and synthesis parameters a *few-to-many* mapping was required, high dimensional interpolators were used as a means of interpolating between a set of control values (defined by the "degree of freedom"), called a "query point". The interpolator will then interpolate between synthesis presets to give an output sound, called the "image point". The author discounted many of the "classical interpolation" methods as they may produce unmusical results in a performance context [118]. Like several other systems it had two phases: an initialization phase where the sounds were defined and a running phase where the system performed the interpolation. During the initialization phase here, a genetic algorithm (GA) or Sammons algorithm was used to define the control values (query points) for desired sounds (image points). In the running phase the interpolation was performed using a geometric technique, called simplicial interpolation, where a continuous mapping was created with an n-dimensional convex hull made up of n-simplices in the control space, mapping to another convex hull, with a higher dimensionality in the parameter space. There was then a pointwise mapping between the control space and the presets in the sound space. When the controls define a position within a simplex in the control space the coordinates are mapped to the parameter space and their values provide the interpolate weightings between sounds.

This technique extends the grid-based mappings by allowing the collection of known points to be scattered rather than fitting them to a grid, as has been widely used for spatial interpolation [128]. Geometrically the simplex hull in the control space induces a similar simplex hull in the sound space, giving a mesh embedded in the higher-dimensional space. The barycentric coordinates of the interpolation point can be determined, which is mapped to the higher dimension in sound space. The corresponding barycentric coordinates in the higher dimension then provide the weightings for the interpolation. The author noted that this was a *one-to-one* in regards to mapping of the simplices, but that the parameter mapping was potentially a *few-to-many* mapping, depending on the orientation of each simplex in space. This method is more flexible than Bowler's grid-based approach (section 2.4.1.1), as it allowed scattered data points, can be edited and expanded, and the number of points is not constrained by the structure of a grid [126].

2.4.1.4 Regularized Spline with Tension

In the mid 2000s the use of Regularized Spline with Tension (RST) surface was proposed for interpolating sounds from controller values so that greater smoothness could be obtained [126]. This was a radial function approach that computed coefficients based on a matrix of linearly independent equations for known data points. For the purpose of real-time control, these coefficients were pre-computed. As with other spline-based approaches, unwanted overshoots could occur due to fluctuations of data points and sparse data sets. However, these were alleviated by a modification of the standard method to include parameters that could adjust smoothness and tension, essentially tuning the effect of higher derivatives. Varying these parameters allowed the mapping to move between a rigid or flexible model, while varying the amount of approximation. Not only is this technique continuous and smooth, but it can also be modified so that it is as smooth as required. The ability to tune the mapping to relative levels of approximation was shown to be beneficial when working with different data sets possessing varying amounts of noise, from a controller.

This technique was evaluated with respect to the other techniques covered in section 2.4.1 of this thesis [126], and was shown to encompass many of the desirable properties that the other techniques possessed. However, with the adjustable smoothness and tension it had greater flexibility, at the cost of the computational complexity. Nonetheless, it was shown that for low dimensionality it could be implemented efficiently. The authors created a low-dimensional example and tested the multi-linear interpolation verses RST approach. They then argued that by changing the mapping interpolation, the instrument could be given a completely different “feel”, which allowed an appropriate choice of mappings to be made.

2.4.1.5 Library of Maps

Given the findings from the test of the RST approach, subsequent work was done to create a Library of Maps, called LoM, that contains a set of Max objects that allowed the exploration of different mapping strategies, using a geometric representation in the control space to map to the synthesis parameter space [129].

The LoM implemented the following previously presented interpolated mapping: piecewise linear techniques relative to a triangulation of parameter space [118], a lattice constructed in this space [125], [130], a multi-linear interpolation between points spaced in a grid [127] and a regularized spline-based technique that generates variable smoothing between points [127]. The paper also makes the following comments on the MnM (Mapping is not Music) [131] library:

> *"An existing toolbox (MnM) for mapping within Max/MSP was presented in (Bevilaqua, Muller, and Schnell 2005). It is based on multiple linear regression techniques: given a set of control/sound parameter presets, the "surface" which represents all traversable regions of parameter space is a hyperplane that is situated near the preset points relative to some best-fit criteria. Whereas the aforementioned techniques are made up of one or many surfaces that pass through or very near each preset, this regression approach creates a single linear control/sound surface that may not pass through any preset value. This drawback is traded off with the ability to draw on vast resources from matrix algebra and linear systems theory, and to deeply utilize the matrix processing available in packages such as Jitter and FTM (Schnell, Borghesi, Schwarz, Bevilacqua, and Muller 2005). Therefore, rather than recreate any of the work put into the MnM toolbox, this current library of mapping strategies seeks to add to the available options by providing linear, piecewise linear, multilinear (hyperbolic) and spline-based strategies for interpolation and extrapolation."*

LoM was designed to allow the rapid exploration of sound spaces and mappings for instrument design. The library allowed various combinations and visualizations of three different interpolation strategies. As well as the interpolation objects, the toolbox contained abstractions that allowed multi-layer mappings. In this way, it is possible to create hybrid mappings, such as, mapping from control space into an intermediate parameter space using RST to provide a smooth transition through the space. These trajectories could then be mapped into a high-dimensional sound space, via simplicial interpolation. The first mapping layer determined the part of control space that was to be accessed as well as the nature of the trajectory through this space [132]. The second mapping layer defined the sub-region of sound parameter space that was to be explored. In this way, the intermediate perceptual control space could be treated separately from the synthesis parameter space.

Subsequently LoM was used to evaluate control strategies for the navigation of complex sonic spaces [133]. To test the toolbox the authors recreated a system, defined by Momeni, that consisted of a bank of resonant filters that were controlled by geometric models [119]. The model was first setup and preset sounds were identified when the resonant filter bank was being driven with a noise generator. They then designed mappings from the X-Y position of a tablet, to the parameters: spectral slope, spectral corner, global decay, global gain, as well as location, spread, attenuation and decay of the "clustering" modes. Because this was a relatively high-level space, points in the space were associated with steady-state sounds and the controls allowed morphing between the identified sounds. In addition, the user could design and edit new control structures by moving the presets to new locations on the tablet, thereby changing the tablet response in neighbouring regions. In this system, the preset points were triangulated on the tablet surface, which in turn induced a piecewise-linear interpolation of the high-dimensional sound parameters. This was not truly control of a perceptual space as the degrees of freedom do not correspond to perceived sound qualities, and linear changes in sound parameters did not result in perceived linearity of sound transformations. However, the high-level nature of the chosen parameters led to a situation where it was possible to create new sounds from repeatable musical gestures. Moreover, these musical gestures were not constrained to an absolute path of physical gesture: while the parameter mapping itself was fixed, the perceived mapping varied due to the hysteresis present in the modal synthesis model (due to the inherent memory of the resonant filters employed). Because of this, the process of constructing a particular sound space and a predictable coupling of physical/musical gesture meant varying the speed and ordering of the different pen/tablet trajectories, resulting in different sonic gestures at the same tablet location, which in turn prompted the movement or insertion of different sound models in a design feedback loop. The authors noted that the mapping choice as well as the mapping design process itself were determined by the interpolation model, the complexity of the synthesis parameters and the time-based behaviour of the chosen model.

In a different approach to the same material the authors laid out the preset sounds in a grid around the tablet boundary, and utilized a non-linear mapping function to generate the sound space. Then instead of moving preset points around in control space they tuned the weighting of each sound. Given the multi-linear quality of the mapping, this amounted to warping the geometric “shape” of the sound space. With this technique, it was more difficult to define a sound to occur at a precise location and it proved to be easier to define regions that had a certain general character. With this approach, it became easier to construct regions of the tablet having a global feel, but more difficult to construct repeatable musical gestures. However, the ability to “tune” the mapping technique compensated for the inability to define sound presets at specific locations in control space. Further, the globally smooth nature of the mapping made it easier to create long, smooth musical gestures, which worked well within the slow-moving and dense sonic space. The authors noted that the trade-offs might be seen as beneficial if, for example, designing a system for improvisation rather than for composing music.

In both of these examples the input device and sound synthesis method were the same, as were the underlying preset sounds, and the only difference was the control structures from the parameter mapping. The design of the mappings was a function of the different musical control context for each example, which was both determined and informed by the design process as well as the choice of mapping strategies.

The authors called the resonant model "high-level" and state it is appropriate for interpolating between known presets in a user-defined perceptual space. However, for examining immediate gestures they used a granular synthesis engine that allows access to the "low-level" synthesis parameters. They noted one approach would be to find interesting trajectories in sound parameter space, and to constrain a mapping to only produce these sounds. However, this limits the exploration and expansion possibilities, and would not make for an interaction design with a very high ceiling on virtuosic use. The authors' solution was to use two mappings: one that controls the sounds and a second that modifies the responsiveness of the control data in a feedback loop. They noted that this needed some "adjustment" of the mappings, but some interesting dynamics were possible from the tablet.

This system was then modified so that the responsiveness was mapped to tilt values of the tablet pen. From a geometric standpoint, this resulted in a four-dimensional control space (two separable two-dimensional planes of control) rather than a single two-dimensional surface as in the previous system. In this way, the pen mapping acted as a meta-control that affected the responsiveness, before being mapped to the high-dimensional sound parameter space. It was found that this was harder to maintain, but offered a more diverse response to gestures.

In conclusion the authors say these two examples underscore the complex role that mapping plays in the structuring of subtle and articulatory control, including issues such as the potential importance of time-variant mappings through meta-control and/or feedback control [133]. The authors have continued this work and have subsequently defined a holistic conceptual framework for further exploring mapping techniques [134].

2.4.1.6 Timbre Space and Perceptual Mappings

Although the interpolation systems examined in the previous sections do provide a way of managing complex synthesizer programming control structures, they do not necessarily relate to the perception of the sound produced. In 1975 Grey defined "Timbre Space" based on a 3D space using a three-way multidimensional scaling algorithm called INDSCAL to position 16 timbres in the space. The first axis is interpreted as the spectral energy of the sound, the second dimension is temporal behaviour in the attack stage between the upper harmonics, and the third is the spectral fluctuation, which relates to the articulatory nature of the instrument [135]. These principles were expanded on in 1979 by Wessel who showed that a 2D timbre space could be used to control the mapping of synthesizer parameters [136]. Later in the mid 1990s a system called, Intuitive Sound Editing Environment (ISEE) developed by Vertegaal used a hierarchical structure for timbre space, based on a taxonomy of musical instruments. This allowed changes in timbre that require numerous parameter changes to be generated by relocating the sound within the timbre space hierarchy [137], [138].

Although not directly related to timbre space, in 1996 Rolland developed a system for capturing the expertise of sound designers, programming a synthesizer, by using a model of knowledge representation. This was not based on the attributes of the sound structures themselves, but on the manipulations or variations that can be applied to them. These transformation procedures were then defined using adjective terms such as “brighter'” or “warmer”. This means classification of a sound according to the transformations that can be applied to it, rather than the properties of the sound itself. This resulted in a hierarchical network of sounds and connections between them, which define the transformations that are required to modify between them [139]. Seawave, developed in 1994 by Ethington, was a similar system that allowed an initial synthesizer preset to be modified using controls that are specified using timbral adjectives [140]. More recently, in 2006, Gounaropoulos produced a system that used a list of adjectives to provide an input, which were mapped via a trained neural network [141]. The user could then adjust the sound using controls allocated to the timbral adjectives. Aramaki in 2007 then showed that a similar mapping process could be applied to percussive sounds, based on different materials and the type of impact [142].

Nicol in 2005 was the first to propose the use of multiple timbre spaces, with one being generated from a listening test and another that is drawn from acoustic parameters [143]. In a comprehensive body of work, Seago expanded this idea and presented a synthesizer interface that allows the design and exploration of a timbre space, using a system of weighted centroid localization [144], [145].

Work has continued in generating more accurate representations of perceptual adjectives and hence definition of timbre space, recent examples being [146], [147], [148]. Potentially this will result in a more controllable mapping between a synthesis engine and timbre space.

2.4.1.7 Performance Expression

As already mentioned, one of the unique features of synthesizer technology compared with traditional instruments is that they present two interfaces to the user, one for the programming of the sound generator and the other for the actual musical input. However, during a performance the user can potentially interact with either or both interfaces. Therefore, the mapping between these two interfaces will ultimately affect the expressiveness of the synthesizer as an instrument. With both interpolated parameter mapping and timbre space mapping systems, the parameters mapped to the performance interface will ultimately affect the expressiveness of the instrument [117]. As a result, the expressive control of both systems has been considered extensively and has been included here as in the area of non-graphical interpolation as they often act as the input.

Winkler in the mid 1990s considered the mapping of different body movements as expressive gesture control of Interactive Computer Music [149]. Although the mapping to a synthesis engine was not considered, it demonstrated the notion of capturing movements for the control of performer expression. Along similar lines, in 2001 Camurri presented a framework for capturing and interpreting movement gestures [150]. This framework was built around the premise that a "multi-layer" system is required to take physical input signals captured from movement sensors, and map them to interpreted gestures. The framework allows different formats for the input signals, such as, time variant sampled audio signals, sampled signals from tactile, infra-red sensors, signals from haptic devices, or events such as MIDI messages or low-level data frames in video. Around the same time, Arfib highlighted not only the need for gestural control, but also a visual feedback mechanism from the expression so that the performer can learn to use the expressiveness available [151]. This work has then been expanded with a multi-layer mapping strategy based on the definition of a "perception space" that allowed multi-modal feedback [152] and in a subsequent paper specific examples were given [153].

In 2002 Hunt defined a "many-to-one" mapping that uses fewer layers, but claims to offer more expressiveness [116]. Then in 2004, Wanderley reviewed gesture control of sound synthesis and presented simulated results of the various constituent parts of a Digital Musical Interface (DMI) that are mapped to digital audio effects and computer synthesized sounds [122]. Next adaptive control was added [154] and trajectories were used as the input stimulus [132]. This work highlights not only the importance of piecewise mappings, but also the mapping of entire regions of control and sound synthesis space. As a result, the authors define a general framework that provides a creative environment to electro-acoustic music composers, performers and sound engineers to explore and define their mappings [155].

Work was undertaken by Caramiaux to look at synthesizing sounds that had a direct similarity to the gesture used to generate it [156]. In this way, specific sounds could be accessed with specific gestures in an intuitive way [157].

2.4.1.8 Morphing

Being able to morph a synthesizer between multiple sounds in real-time is not a new concept, but often it is just created as a simple cross-fade between two or more different presets. However, some more complex ways of morphing a synthesizer with an interpolator have been proposed where points in the parameter space representing desirable sounds can be controlled in time, usually with some form of instrument controller. Equally the mappings between the control parameters and the synthesis parameters can also be changed over time and at varying speeds. In this way, a path or trajectory can be defined in the parameter space so it is possible to morph the multiple sets of parameters in a specific time order.

Ssynth was developed by Verfaille in 2006 at McGill University and is a real-time additive synthesizer that allows “additive frames” to be arranged as a 3-D mesh [157]. The morphing is done in two stages: first, additive frames are generated as a weighting of pitch-shifted additive frames – these are generated using interpolation of a set of neighbour notes from the same instrument with the fundamental frequency and dynamics. Then in the second stage interpolation is performed with frames from several different instruments [158]. Trajectories within the 3-D mesh can then be used to morph between different sounds in time and change the mappings, based on the control inputs [157].

Also in 2006 Pendharkar suggests another form of parameterized morphing where desired parameters can be selected from the parameter spaces and a control signal can be used to modify the interpolation function itself. Interpolation can then be performed between multiple sets of parameters in a specific order [159], but allows points in the parameter space representing desirable sounds to be parameterized with high-level controls. The choice of end points of the morph and the extent of the morph can be used to control the synthesis parameters. Aramaki also used a similar process in 2007 to morph between different sounds (materials) in a percussive synthesizer [142].

In 2010 Wyse built on this principle and proposed a system called Instrumentalizer that allowed synthesis algorithms to be controlled with traditional instrument controls for things such as pitch and expression. The system mapped these controls to the synthesis parameters and allows morphing to permit typical instrumental expressions [160]. This was achieved by defining “morphing lines” in the parameter space that allow musical expressions to be generated if the lines were navigated and these were mapped to the instrument controller.

Another example presented by Brandtsegg in 2011 is a dynamic modulation matrix that enables dynamic interpolation between different tables of matrix coefficients [161]. This permits the morphing of the modulator’s mappings, allowing the sound produced to be morphed.

2.4.2 Graphical Interpolation Parameter Mapping

As mentioned in Section 2.3.4.2, the Kaoss Pad provides simultaneous control of two parameters, via a representation of a two-dimensional space [120]. Moving on from this idea of the *one-to-one* mapping of synthesis parameters in a two-dimensional space, the model can be expanded so that presets of parameters, that represent sounds, can be positioned in the space. Then interpolation can be used to generate parameter values in-between the locations. Interpolating between sets of parameters can facilitate non-discreet transitions and the discovery of new "custom" settings that blend the characteristics of two or more existing parameter presets. In this way, it is possible to map a small number of control parameters to a large number of synthesis parameters. This allows the user to explore the interpolation space defined by the parameter presets and discover new sounds that are a function of the presets, their location within the interpolation space and an interpolation point [162]. It is also possible to define positional movements of the interpolation point that result in new sonic gestures. Hitherto there have been a number of interpolation systems that have been built for use with synthesizer technology. These will be considered in chronological order so that the development can be seen and considered.

2.4.2.1 SYTER

Work in this area was first completed at GRM in the early 1980's, where the SYTER system was developed [163], [164]. This system was a hardware workstation that was developed to allow real-time audio processing and synthesis. As well as offering real-time processing, work was also undertaken to develop a control interface that would allow musicians to experiment with the workstation and use it for live performance. A two-dimensional graphical interface was proposed by Daniel Teruggi and the GRM team, which offered a real-time control window, called INTERPOL [165]. This provided an X-Y visual plane to control the relationship between different parameters of the real-time sound-processing engine. The positions of points on the visual interface were mapped to a set of up to 16 parameters, referred to as a "snapshots". Each snapshot is given a circular representation (a "planet") in the interpolation space, as shown in Figure 4. Clicking in one of the circles would recall the corresponding snapshot of sound-processing parameters, allowing a mechanism for loading a particular preset sound.

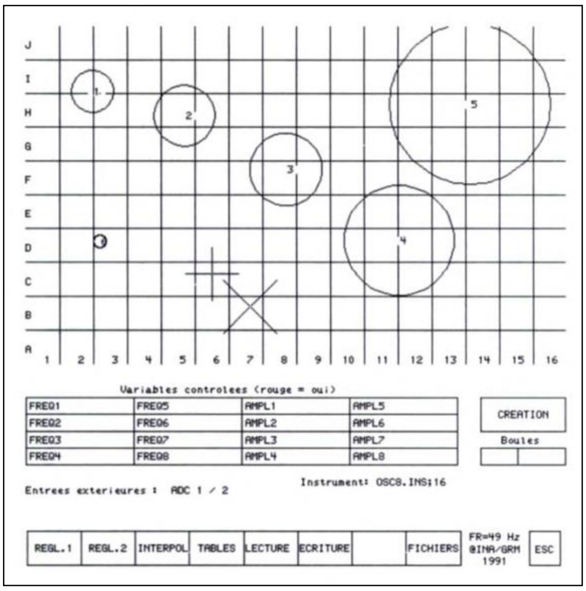


Figure 4 INTERPOL SYTER Control Screen Showing Planet Locations and Mapped Parameters

However, the system also allowed interpolation between the snapshots using a gravitational model by moving the cursor to different locations between the presets. This is based on an Inverse Distance Weighting (IDW) function, first proposed by Shepard [166], which is used to calculate the interpolation values. In Shepard's original version, an exponent value can be defined for the distance weights that allow different gradients to be created between the parameter sets. In the SYTER realisation the exponent value is set to a value 2, which results in a representation of a gravitational model [167]. This provided the influence of each snapshot within the interpolation space, with larger planets generating a stronger gravitational force and so influence, than smaller planets. The interface provided continuous interpolation between the defined preset sounds. By adding new planets, it was possible to expand the number of snapshots controlled by the X-Y plane and defining new positions for the planets allowed the interpolation space to be modified. When an interesting sound had been located it was possible to take a new snapshot of the parameter values and define a new planet in the interpolation space with these values and so further changing the distribution of values in the two-dimensional plane. The INTERPOL system offered a mechanism for musicians or non-expert users to empirically explore the parameter space [61] and this concept has subsequently been referred to as "intuitive" [119], [167].

It is also worth noting that in 1993 a Max/MSP graphical interpolation object, called Vect VTboule, was created that was inspired by the SYTER model [168]. This has been subsequently superseded in later versions of the software with an object named "nodes", which is covered in Section 2.4.2.9.

Another application area where IDW interpolation has been used is within Geographical Information Systems (GIS) to estimate new values based on geographical distributions of known values. A review of these techniques is available [169], but the visualizations often tend to be geographical maps and the interpolation is typically performed on either a single or few parameters, rather than the relatively large number used in synthesis.

2.4.2.2 GRM Tools

By the 1990’s many of the audio-processing algorithms developed on GRM’s hardware system were being converted to host-based software implementations. From this the GRM Classic plug-in bundle was created that could be used within the software DAWs that were becoming popular at the same time. The classic bundle offered eight adaptations of the audio-processing algorithms that ran on the SYTER system, with new interfaces and controls [61]. These interfaces were not as easy to use as the original system and the two-dimensional interpolation space was replaced with a one-dimensional interpolation slider with presets located at fixed intervals along its axis [170]. Figure 5 shows the Comb Filters plug-in from the GRM Classic bundle, but each of the plug-ins in the bundle offers the same interpolation functionality. This allows 16 parameter presets to be saved and then allows timed linear interpolation between the parameter sets by clicking on the appropriate preset button. The time it takes to perform the interpolation could be set using a slider so that either slow or fast transitions can be defined.

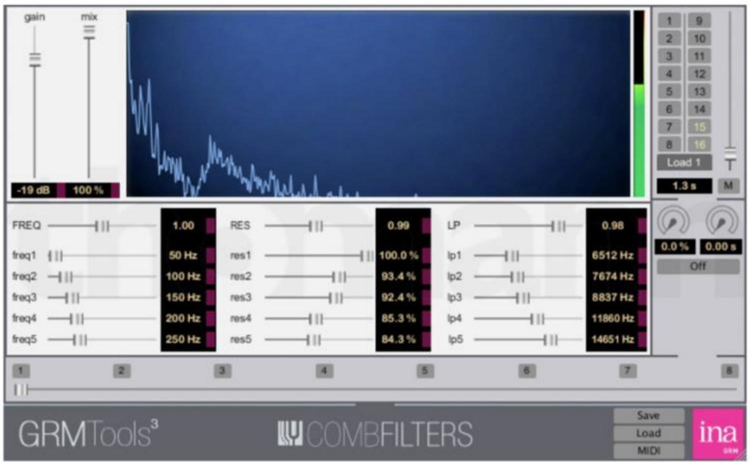


Figure 5 GRM Tools Plug-ins offer 1-D Interpolation Sliders

Where interpolation between multiple presets is required, there is a control called SuperSlider that allows up to eight presets to be placed at locations along a horizontal slider. Moving the slider then performs linear interpolation between the two closest presets and so the SuperSlider can create interpolation between a sequence of presets. This offered some interesting sound design possibilities as movement of the slider does not have to be in the same direction and the speed of movement could be continually varied. However, when compared to SYTER, due to the reduced dimensionality, it is only possible to simultaneously interpolate between two presets at a time and the slider must pass through the preset sound before it is possible to move to a subsequent interpolation.

2.4.2.3 overSYTE and Rectilinear Interpolators

overSYTE was a real-time software application developed in the mid 1990's allowing live granulation of instrumental or vocal performances. The interface gave users access to parameters via sliders, but also had a mechanism for saving and loading preset parameter values. It then featured a two-dimensional preset interpolator that allowed four presets to be positioned at the corners of a space. The system then allowed continuous interpolation between all of the presets at once, based on the position of the mouse within the interpolation space [171]. This was done using bilinear interpolation of the parameter values that are then applied to the granular synthesis engine. The overSYTE preset interpolator is shown in Figure 6.

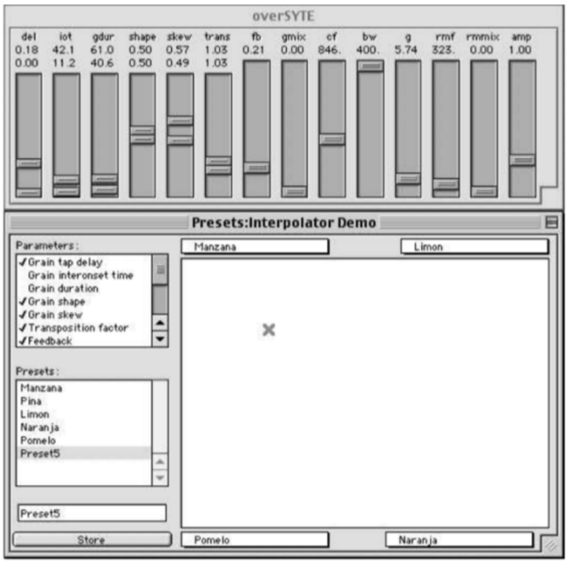


Figure 6 overSYTE with 2-D Preset Interpolator

The author subsequently noted that although designed for performance an unexpected feature "is that novel parameter combinations are often found at intermediate locations on the surface" [123]. However, the author also notes that the disadvantages of this system are that it limits users to a maximum of four presets and they can only have fixed locations within the interpolation space. It is also noted that when a number of "disparate" presets were used, it could generate fierce modulations within the parameter space. These could lead to some interesting sounds, but was less satisfactory for live performance. As a result, in live performance situations, either static parameter settings or slow-moving modulations were used to allow the performers to feel in control of the sonic outcome.

It is also worth noting that Korg have produced a hardware-based effects controller, the Kaoss Pad Quad that offers the same two-dimensional, corner-based parameter interpolation [172]. Similarly, this functionality is also offered on Native Instruments FM synthesizer, FM8, where four presets can be loaded into the corners of a square interpolation space and real-time interpolation between a manufacturer restricted set of parameters is allowed [173].

This principle has also been expanded upon on Apple's Alchemy synthesizer [174]. This instrument has performance controls called Transform Pads, which allow up to eight presets arranged as 2x4 grid as shown in Figure 7.

Figure 7 Alchemy's Transform Pads Arranged in a Grid

This allows the parameters of eight presets to be interpolated between, giving the ability to morph between the sounds. However, as the presets are positioned at the centre of the corresponding pad it means the cursor can either be positioned between four pads or only two. Therefore, it can either interpolate between four presets at a time, like overSYTE or interpolate between two presets, like the GRM SuperSlider.

2.4.2.4 Three-Dimensional Gravitational Interpolation

At the end of 1997 the two-dimensional SYTER style interpolation was further expanded to a three-dimensional model [175], [176]. This was developed as an interface for a hardware DSP (Digital Signal Processing) platform developed by IRCAM, known as IRCAM Signal Processing Workstation (ISPW). The interface allowed users to size and place interpolation spheres in a visual representation of a three-dimensional space. Users could then define trajectories in the interpolation space that could be recalled on-the-fly so that dynamic interpolation could be performed. The trajectories could be recorded, played, scaled and even reversed. Figure 8 shows an example of the three-dimensional control space, developed by Todoroff.

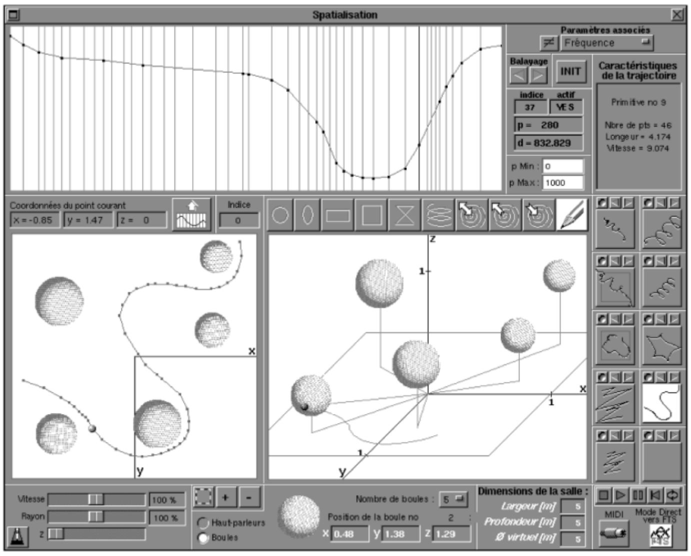


Figure 8 3-D Interpolation Control Screen Showing Arrangement of Spheres and an Interpolation Trajectory

The same interface could also be used to spatialize sounds within an equivalent three-dimensional space, with specified speaker arrangements. By combining the parameter interpolation and spatialization it allowed the composer to explore spatio-timbral relationships. In practice the author has implemented the same system using different programming environments [160], [167], [175] [176].

2.4.2.5 Interpolator

The SYTER style planetary model, where a gravitational paradigm is used for interpolation between sets of parameters, gives no visual representation of the "field of influence" for each planet. As a result, accurate navigation of the space could be difficult with the output sound generated being the primary feedback to the user. In addition, the planets themselves occupy an area in the interpolation space that cannot be used for interpolation, as when the cursor is positioned on the planet, only the associated preset will sound. The result is the effective interpolation area is reduced by the total size of the planets. In an attempt to resolve these issues, in the early 2000's, a system called "Interpolator" was developed in collaboration between GRM and University of Hertfordshire [177].

This system used a light model, where parameter presets were represented as "lamps" that had an angle, aperture and extent for the light source. In this way, the light beam gave a visual representation of the field of influence and its extent. In addition, if the angle of a lamp's aperture is opened up to 360 degrees then it becomes similar to the planetary system, except the lamp shows field of influence and the planet's area is not lost from the interpolation space. The user could define up to four sets of parameter mappings, represented by four beam colours, and each lamp represented a preset of values for the parameters assigned to the lamp's beam colour.

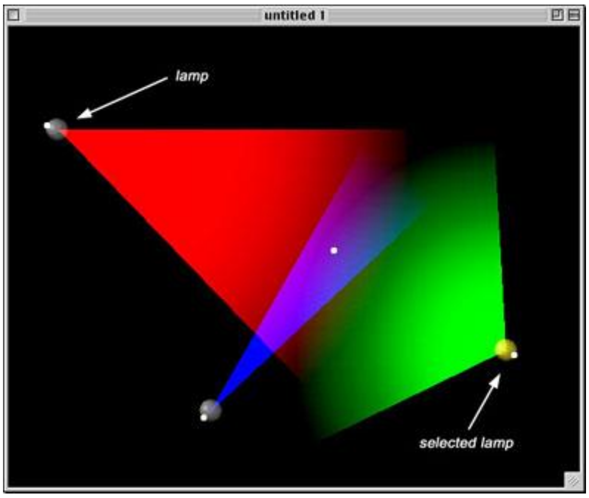


Figure 9 Interpolator's Light-Based Interpolation Model with Three Lamps having Different Beam Apertures

The system allowed up to four light colours to be defined as well as a set of values for the background that is not covered by light beams. The background stored a value for every parameter defined for every lamp colour (parameter mappings). The background allowed interpolation for a single preset (lamp) and a set of default values (background). Therefore, the background became like a preset made up of all four lamp colours, which is all parameters within the interpolation space. This background function could be turned on and off and when off, interpolation only took place when at least two lamps were active and only where the light beam colours overlapped. The lamps had no influence outside of their light extent, whereas in a gravitational model the influence exists through the full interpolation space.

The system had two operation modes: an edit mode where the preset parameters could be defined and the lamps setup, and a performance mode where the interpolation space could be navigated. In addition, to free hand movement of the space, the system also allowed trajectories to be specified and played back, where the trajectories could be performed at specified speeds. There was also an influential curvature utility that allowed either an inverse-square or exponential function to be specified for the interpolation space. The interpolator then used the distance of the cursor (or trajectory) from the lamp centre to calculate a weighted value (expressed as a percentage), based on the extent of the beam and the curvature. If the percentage is less than 100%, (it could be more as multiple lamps may have the same colour and therefore will be controlling the same set of parameters, but may have different values), the value of the background was taken as the remaining value to make a 100% total. If the combined weight of the lamps with the same colour is over 100% then the background was "washed away" and had no effect. Using this weighting system and providing trajectories it is possible to morph between different presets.

In user testing it was found that users preferred the light model of interpolation and found it more useful than the gravitational model. The light model offered better visualisation when performing interpolation tasks and gave more accurate navigation through the interpolation space. The Interpolators authors also noted that the interpolation space provided an intuitive mechanism for users looking to create new sounds, without having to spend a large amount of time adjusting individual parameters.

2.4.2.6 Geometric Models

In 2003, Momeni built on the earlier work of “timbre space” (covered in more detail in Section 2.4.1.6) of Grey [135] and Wessel [136], to define a generalized system that allowed the spatial layout of objects that related to musical material [119]. This could then be used for the control of live performance. The system permitted the objects, called a “one-points” to be a recorded sample, a single number, or a list of numbers, such as a preset of synthesis parameters. Each of these could be placed at a user-defined location within a two-dimensional space, known as “space-master”. Each of the objects were represented as a Gaussian kernel, whose value at any given point in the space-master indicated the weight of its associated data point in an interpolated mixture. The result was a space that allows weighted interpolation among all the sources based on the values of the Gaussian kernels at each point in the space. The user could specify centres, amplitudes and standard deviations for each Gaussian kernel within the space [119]. An example of the space-master is shown in Figure 10. Each Gaussian kernel could be associated with a list of numbers that represented a synthesizer preset and hence the space offered high-dimensional interpolation. The space, as shown in Figure 10, is visualized in two dimensions by an image that is a bird’s-eye projection of all the Gaussian kernels onto a plane.

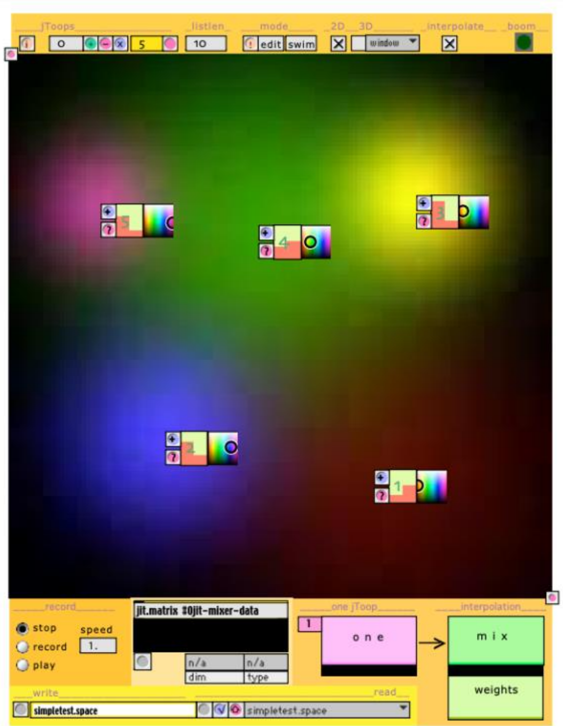


Figure 10 Space-Master Showing Five Gaussian Kernels Viewed From Above

To design a space the user first defined the number of parameters associated with a kernel and placed the desired number of kernels into the space using the graphical interface. When the system was in edit mode, the user could move each point to a desired location in the space, select an amplitude and standard deviation for the kernel and define a colour.   While designing a space it was also possible to audition each preset by clicking on it to initiate playback, synthesis or calculation that was appropriate to the space.

The space-master would then show a two-dimensional visual representation of all the kernels in the space by mapping the height of the kernels to a brightness scale applied to the selected colour of that kernel. For more accurate visualization the Gaussian kernels could be viewed as a 3D image as well.

Once the space-master was configured, the user could put the system into “swim” mode where a 2D slider allowed interaction using the mouse or incoming control values so the space could be explored. Selecting a location in the space generated interpolation between the parameter presets based on the selected X-Y position and weighting based on the kernels. The Gaussian kernels provided not only a mechanism for interpolation, but also for extrapolation beyond the perimeter of the points specified in the space.

The system has been tested with a number of different examples where the interpolation space was used. The authors noted that the described method for creating spaces was found to be extremely useful as they could design a space based on their intuition and perceived subjective similarities, but also suggested this could be impractical for very large sets of data. However, they said that as the system allowed users to employ their musical intuition to define the spaces, it made performing with these spaces intuitive and rewarding. They also stated that interesting musical results were quite often found when exploring the space between the sounds and in a live performance context, it provided a way for countless new sonic possibilities derived from those designed by the composer. The use of Gaussian kernels, as opposed to Euclidian distance, was found to be crucial in these musical applications as often the interpolation results in frequency-scaling. Using weighted functions that are not adjustable by way of a parameter, like standard deviation, results in a timbre space that is filled with glissandi. They state that in their experience, continuously changing pitch adds a transparent synthetic quality to the sound which detracts from the effectiveness of the instrument, resulting in a drum-like entity. They found that using Gaussian kernels resulted in the glissandi being localized to one region of the space. However, this would be less of an issue in a sound design context where glissandi can produce interesting sonic effects.

It was also shown how a multi-layer approach could be applied, where geometric arrangements in performance environments could involve multiple perceptual spaces in one instrument. The same method of dimensionality reduction that was applied to an individual perceptual space was also applied to the entire system in order to organize specific arrangements of the individual characteristics. Specifically, a space of spaces was created by using a parent-space to interpolate between sets of coordinates in a number of child-spaces.

In subsequent years this research group have continued their work by examining different visualisation and interactions for this style of interface [178]. In this work they have examined visualisation, placement and interaction of large numbers of points in a 2D interpolation space. They used two approaches based on the previous work where points could be placed anywhere in the space based on the user's performance strategy and a second approach where they were constrained by regular placement on a triangulation-tiled manifold. The work showed that although the user placement offered user flexibility, for a large number of points the interpolation space could result in unreachable outputs. This was different to the triangulation scheme, which ensured that all outputs are reachable. However, in practice, some output points were judged to be similar to others in the space. In order to maximize the variation of reachable mixtures through interpolation it was found the distance between similar presets in the space should be inversely proportional to the similarity.

2.4.2.7 Metasurface

In the mid-2000s the creator of overSYTE (Section 2.4.2.3) then developed a new interpolation solution, known as Metasurface. This is realised in a software application for live performance, audio processing, sound design and music composition [171], called AudioMulch Interactive Music Studio. The Metasurface took the principles of its predecessor and expanded on them [123]. The surface allowed any number of parameter presets, referred to as “snapshots”, to be defined and placed in the interpolation space. The placement could be done by the user and could be refined at any point by going into placement mode (as opposed to interpolation mode). When the presets are placed in the interpolation space a Voronoi tessellation is constructed where each convex polygon contains one preset point and any position in the polygon is closer to the preset location of that polygon, than the preset of any other polygon. When the Metasurface is then put into interpolation mode, a crosshairs cursor appears and natural neighbour interpolation is calculated. This is effectively a new polygon inserted at the current cursor position and a weighted sum is calculated for the presets that are natural neighbours, adjacent to the cursor position. The weight of each neighbour is calculated as the area of each cell “stolen” from the neighbours by the polygon centred at the cursor position [179].

The advantage of this method is that it gives continuous, smooth interpolation across the whole space. This can also be differentiated for any location, except the polygon centre points, offering smoother interpolation than if other techniques had been used [123]. Figure 11 shows an example of the Metasurface when in interpolation mode and as shown it gives a visual representation of the current neighbour polygons.

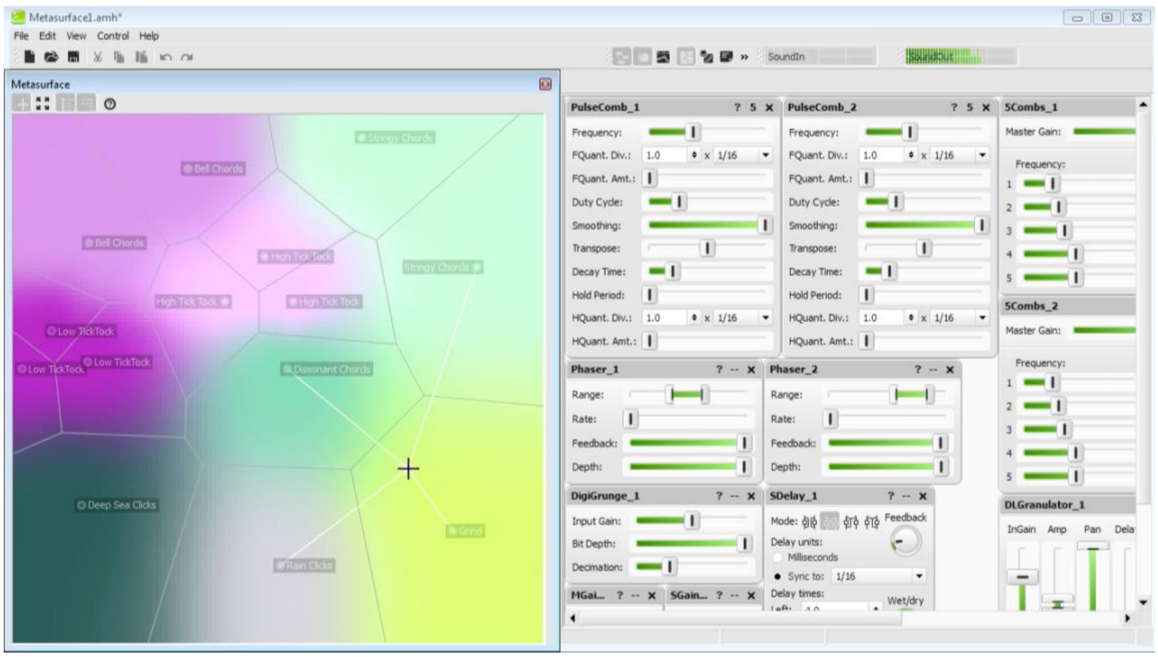


Figure 11 Metasurface offering Local Neighbour Interpolation with Each Preset as a Polygon

The Metasurface was presented as the successor to overSYTE and it provided the same baseline functionality, but with the improvement that it supported the arbitrary placements of any number of presets, anywhere on the surface. As a consequence, detailed surfaces with user-defined geometry could be specified. New preset points could be inserted on the surface using drag-and-drop and their locations could be altered using direct manipulation. Each preset also had an associated colour that was interpolated to provide a visual cue about the sonic properties of different areas of the surface. Initially the colours are randomly allocated, however they could be altered by the user to express specific associations with the corresponding sounds. Although the coloured representation may sometimes have low perceptual correlation with the sonic output, it was deemed adequate given the difficulties of visualising an arbitrary N-dimensional parameter space.

The paper also evaluated areas for further work and particular attention was given to the work of Momeni et al., saying that it probably offered an even more intuitive solution, where multiple two-dimensional interpolation spaces are offered [119]. The author noted that it is possible that Metasurface could be reimplemented using the simplical spatial interpolation method [118], but that it was unclear whether the computational requirements of this method would make it unsuitable for the design of an interactive surface.

It is worth noting that a paper by van Wijk et al. [162] that examined the visualisation of preset interpolation systems also used a Voronoi tessellation as a way of visualising the preset space. They used the tessellation to show the users the closest preset to the current interpolation and the nearest neighbours. They said that the diagram allowed the user to answer various questions about the parameter space: which preset has most influence, and at which locations two or three presets have the same influence.

2.4.2.8 INT.LIB

Work in the mid 2000s saw the SYTER style gravitational model revived, updated and expanded. INT.LIB was an extension for Max/MSP that offered the control of multiple presets using a gravitational model. This was done by taking a modular, multi-layered approach to preset interpolation and provided real-time visual feedback of the resulting interpolation. With INT.LIB multiple layers of interpolation could be created simultaneously within the interpolation space. Each layer was colour-coded and had its own set of crosshairs that indicated the current point of interpolation for that layer [180]. Optionally the interpolation points could be linked so that all layers are controlled simultaneously. An example of a three-layer interpolation space is shown in Figure 12.

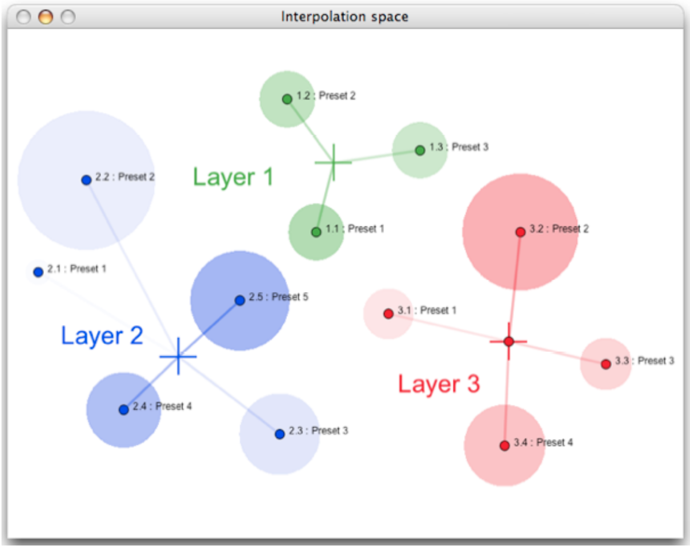


Figure 12 INT.LIB Multi-layer Interpolation Space Uses Colours for Each Layer

Each layer had its own instance of a synthesizer or signal processing plug-in and allowed interpolation between a number of presets. The layer colour-coding offered quick identification of which presets and crosshairs were linked to each layer. The layers and their preset's could also be named and labelled in the interpolation space. With this system, as well as the auditory feedback given by the resulting interpolated sonic output, it also provided the user with visual feedback so they could see which presets are included in the interpolation and their relative influence. This was achieved by linking the weighting coefficient of each preset to the transparency of the associated ball. Should a preset have zero weight, only the "handle" at the centre of the ball would be displayed, if the preset has full weight the ball is a solid colour and any weight in between has a proportionally transparent ball.

It is worth noting that the interpolation positions of different layers could be controlled by any two-dimensional input controller, such as joysticks, trackerballs, tablets, etc. The author noted therefore, that the system could be used as a rapidly configurable two-to-many mapping layer for gestural interfaces.

2.4.2.9 Nodes

The nodes object in Max [181] was first created by Andrew Benson, a visual artist [182], and proved so popular that is has subsequently been included in the official Max distribution. Although this graphical interpolation system uses a distance-based interpolation function, it uses a different model. Each preset is represented as a circular node within the interpolation space. With this model, the interpolation is performed where the nodes intersect and the distance to the node centres is used as the weights for each preset in the intersectional area [183].

Figure 13 Interpolation Space Created with nodes Objec shows an example of an interpolation space created with the Max nodes object.

Figure 13 Interpolation Space Created with nodes Object Showing Preset Intersections

Although the nodes object offers a different graphical interpolation paradigm it does have an issue when the nodes do not cover the whole interpolation space. In this situation, a discontinuity of interpolated parameters maybe generated when the cursor moves between the node boundaries and the background of the interpolation space. As a result of this, audible jumps maybe generated, rather than smooth, continuous changes in sound.

However, the nodes object does offer the benefit of being able to edit the layout of the interpolation space while still being able to interpolate the sonic output. It is worth noting that due to the nodes realisation being in the Max visual programming environment, it will also be possible to achieve this benefit with other interpolation systems built using it.

As this object has been included in the Max distribution it has been used in a number of different documented application areas [184], [185], and is likely to have been used in many others.

2.4.2.10 Spike-Guided Delaunay Triangulation Interpolation

In 2009 a different graphical visual interpolation scheme was created. This was a synthesis sound design front-end for a physically-based Sound Design Tools package that was developed as part of the Sounding Objects project [69]. This possessed two interesting features: the user was given a representation of the sound and the interpolation scheme itself [186]. The representation was called a “spike” code that allows efficient visual illustration of time relevant signals. This is done by non-linear decomposition of the signal using a method that consists of optimizing the kernel function scaling coefficients and time positions to form a shift-invariant representation [187]. In this case, the kernel function is a fourth order Gammatone filter bank, where the filter centre frequencies were distributed across the frequency axis in proportion to their bandwidth. This was chosen as it approximates the magnitude characteristics of the human auditory filter [186]. Each bank generated a spike representation consisting of the time, the centre frequency and the amplitude. The spikes are plotted on a graph of time verses frequency and colour for the amplitude of each spike. Figure 14 show two spike representations for two different sounds synthesized with the Sound Design Tools package.

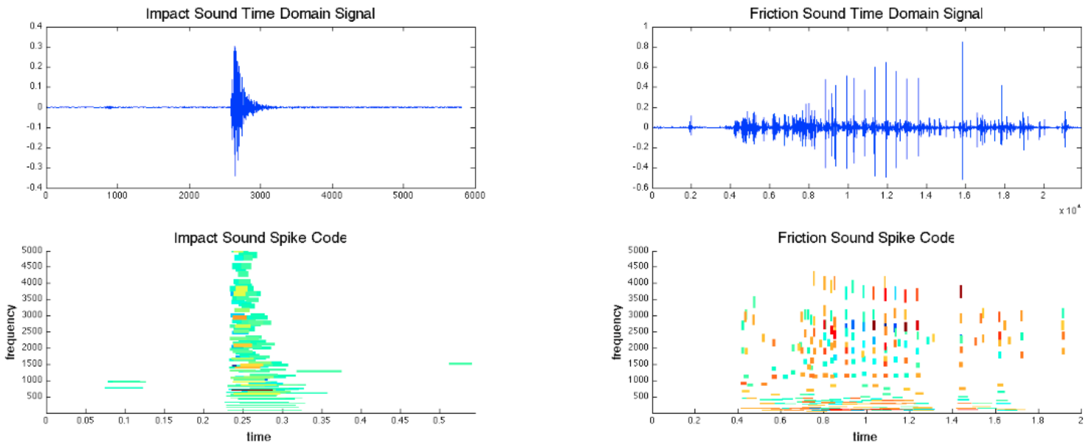


Figure 14 Spike Representations of Impact and Friction Sounds

The spike representations for synthesized presets, called sonic landmarks by the author, can then be positioned in an interpolation space. The positioning of the presets could be chosen arbitrarily by the user, derived from statistical analysis, or based on some form of perceptual characteristics. The presets in the space form a scatter of points and interpolation was performed based on a Delaunay triangulation of the points. The user can then select a new point in the space and the synthesizer parameters are calculated through linear interpolation of the three presets within which the new point is positioned within the triangulation [188]. This is based on the barycentric coordinates of the point within the triangle and can be efficiently calculated with matrix algebra [189]. Figure 15 shows an example of the Delaunay triangulation of the preset sounds with the spike represented for each being shown.

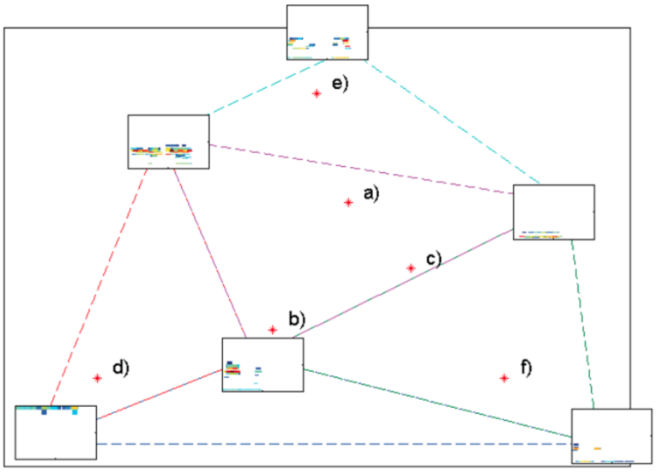


Figure 15 Delaunay Triangulation of Presets and Spike Representations

In this way, the user is given a visualisation of the sonic characteristics of specific locations in the interpolation space. The authors showed examples that they say perceptually have good agreement with the user's expectations, based on the neighbouring presets within the interpolation.

2.4.2.11 Intersecting N-Spheres Interpolation

More recently in 2012 a new method for interpolating between presets was described: Intersecting N-Spheres Interpolation. It is simple to compute and it is claimed that its generalization to higher dimensions is straightforward so that it could be used as a mapping strategy for interfaces that include multiple continuous sensors [190]. In this case the input device is a musical interface "the Sponge", a flexible interface that resembles a cushion [191]. It transmits eight continuous sensor signals from two force sensing resistors and two three-dimensional accelerometers, as well as seven buttons. From these sensors, more than 50 expressions (tilt, twist, fold, shocks, pressure, vibration, etc.) could be extracted and mapped to sound parameters, via interpolation [190]. The author first evaluates the merits of the Natural Neighbour Interpolation used in the Metasurface, including: localised influence, continuity and the ability to create flexible parameter spaces [123]. However, it also highlights some drawbacks. First, it was not possible to move the data points around while sound was generated. There were two modes: one that allowed the user to design the interpolation space by moving the preset points and another where the cursor could be moved and the interpolation was performed. The second drawback was the fact that the Metasurface was limited to *two-to-many* mappings and this author required the possibility of *many-to-many* mappings for use with the number of control signals from the sponge. This meant that higher dimension interpolation was required [190].

This technique used a two-dimensional space where a circle is drawn around the interpolation point, its radius being equal to the distance to the nearest preset data point. Circles are also drawn around each preset point, with the radii of these circles being equal to the distance to the nearest preset location or the interpolation point, whichever is nearest. All of these circles are also redrawn every time the interpolation point or data points are moved. Any preset point circles that intersect the interpolation circle are considered neighbours and will influence the value of the interpolation point. Having established the neighbours, the value of the interpolation point is calculated as a weighted average of the value of its neighbours, where the weight of each point is equal to the ratio of intersecting circles area [190]. An example of the two-dimensional version of the algorithm is shown in Figure 16.

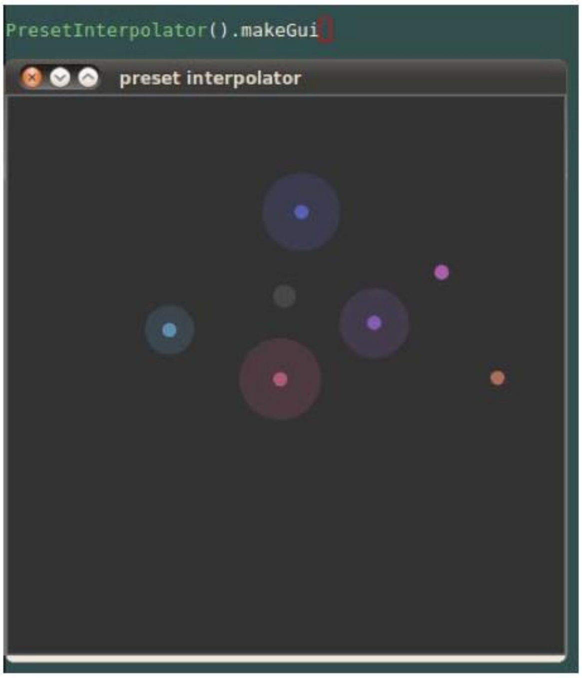


Figure 16 2-D Example of Intersecting N-Spheres Interpolation

This two-dimensional version works like the other graphical interpolators, in that, each preset point can be edited and positioned using a graphical user interface. However, the author states that with this interpolator it is possible to edit and navigate the space simultaneously without interrupting the sound. It is also worth noting that due to its realisation being in the SuperCollider environment it shows this benefit is possible in other modern audio programming environments.

It is pointed out that this model can be expanded for higher dimensions by using the N-Sphere volume, instead of an area, and calculate volume ratio instead of an area ratio. However, this approach would require significantly more computational power, that the author states, "is beyond what is required for a creative or musical application". Therefore, the data structures were modified so that they can represent n-dimensional data and once the intersecting n-spheres were found, weightings were obtained using area ratios. This is the method that has been used with The Sponge Interface [191].

2.4.2.12 Sound Maps

It is also worth noting that a graphical interpolation method was briefly (2012 – 2016) implemented by Arturia on some their software synthesizers, called sound map. It offered a way to explore and locate different presets visually, where the presets were shown as small icons in a multi-coloured map [192]. Presets were categorised by different shapes and colours for various types of sounds. It was possible to interpolate between four presets [193], either within the map itself or within a separate window, called Compass. An example of the Arturia sound map interpolation is shown in Figure 17.

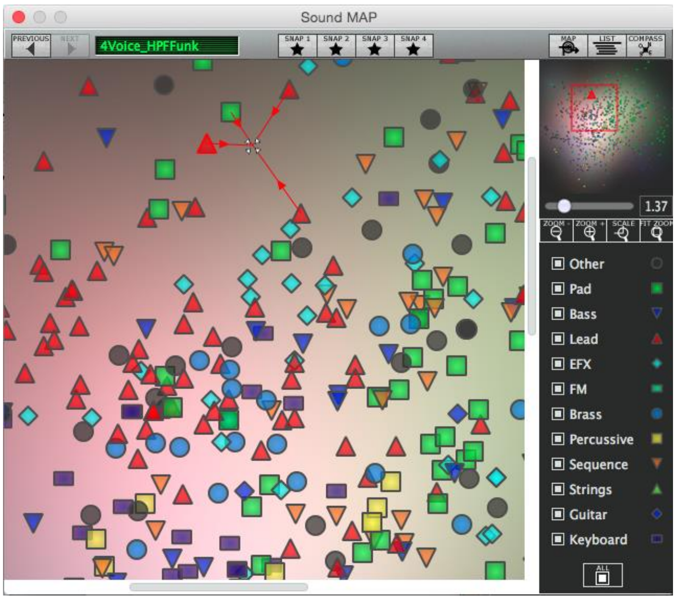


Figure 17 Arturia Sound Map Interpolation with Different Sound Categories

The interesting thing about the sound maps was that the presets were organised within the map based on their own audio characteristics [194]. Therefore, presets that possess certain sonic characteristics were placed close together and those that did not were placed further apart. Although the maps offer a mechanism to explore and modify the presets, there was no way to include new sounds within the maps or modify the maps layout, beyond selecting the type of sounds to include.

2.4.2.13 Multi-Touch Parameter Mapping

The principle of using a two-dimensional X-Y plane for synthesizer parameter mapping was further advanced with the use of multi-point touch screen interfaces. With the previous systems, controlled with a computer mouse or other two-dimensional controllers, only one point can be moved at a time. However, modern multi-touch devices facilitate the capture of multi-point control gestures [195]. Although not strictly being used for parameter interpolation the multi-point control data was mapped to synthesizer parameters. Interestingly some attempt has been made to move beyond the use of arbitrary grids, onscreen object interaction, and data streams typical of single-point, two-dimensional interfaces. That is, not as a polyphonic implementation of an existing X-Y interface approach, or as a vehicle to interact with virtual onscreen controllers [196]. Here the author focused on the extraction of useful data unique to a system with multiple points, such as used in performance of an instrument. It was hoped that mapping a combination of multiple relationships to synthesizer parameters would give the multi-point interface the depth and versatility of a physical instrument, where simple combinations of interactions could lead to complex results. As a result, points were treated generically, rather than as objects with specialized functions, as with interfaces such as the reacTable [197]. The multi-point data was analysed as the mode of interaction and the instrument was designed to use data streams that were dependent on the relationships between points. The analysis of point relationships became the focus of the research and new axes of interaction were created based on the comparison of individual points to other points [196]. The point's coordinates, creation times and motion were analysed to find relationships such as the distance between points, angles to other points, velocity as compared between points, and the time added and removed. Individual points could then be compared to all other active points, a specific sub-set of points, or points that had already been removed. Figure 18 shows an example of the on-screen representation that displays the global relationships, angles and distances of multi-touch points.

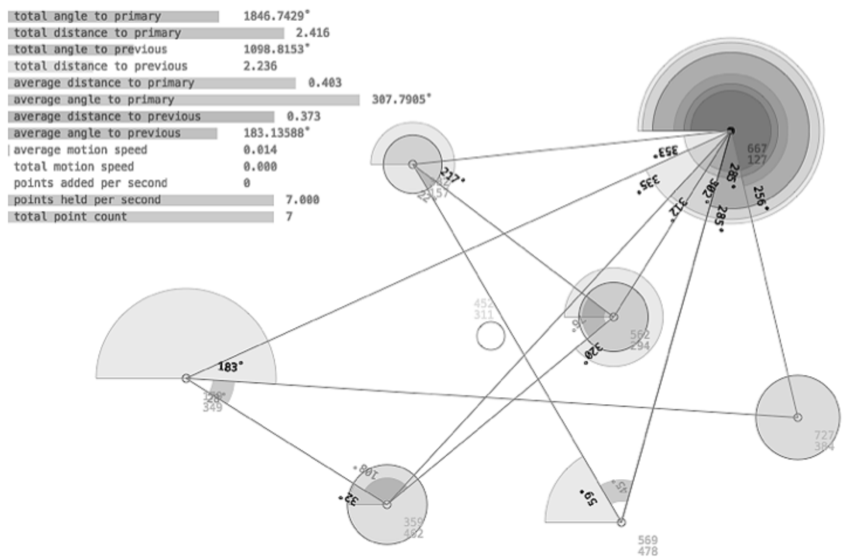


Figure 18 Multi-Point Relationship Data Display Showing Global Relationships, Angles and Distances

Each new relationship adds an "axis" of interaction to the multi-point control, making a two-dimensional (three-dimensional including time) surface a multi-dimensional interface. There are varying degrees of independence between relationships. For example, a group of points moving parallel to one another changes the average position, total velocity and average velocity values without altering the angle and distance relationships between points. In another example, points created in the same positions, but in a different order will create different distance and angle relationships, but the same final average position. However, some relationships are always linked. For example, a change in distance between points will also result in a change in velocity for at least one point.

This was then used as a performance interface for two different synthesizers that were then evaluated by a Laptop Orchestra [196]. The results were extremely encouraging and corroborated the finding of Hunt, Wanderley & Paradis, that more complex mappings actually result in more expressive outputs [117]. The author has then expanded on these principles in the development of a full iPad-based realisation that is now commercially available [198]

2.4.2.14 Polygon Interpolation

Since the initial research was undertaken for this project another interpolator has been created that provides a multi-touch interface for controlling the interpolator, known as MIEM (Multitouch Interfaces for Electroacoustic Music) [199]. It consists of two separate applications: the first of which allows the design of the interpolation space and the association of the mappings and the second allows the use of the interpolation space and provides OSC messages for the transmission of the interpolation output. The interpolator uses intersecting polygons to represent presets and the user can choose the shape of each from the options triangle, square, pentagon, hexagon, heptagon and octagon. These are all regular shapes when initially placed within the space, but the user can edit the exact position of all of the vertices so that new regular or irregular polygons can be created. It is also possible to add new vertices to any polygon so that new polygonal shapes can be created. The user can then configure the mappings to be associated to each of the polygons within the space. Due to the use of the multi-touch platform the user can also place multiple interpolation points, which they call exciters, within the designed space. These can either be constrained to just allow movement of the point(s) within the polygon enclosed area or to allow completely free movement. This is all done within the editor application that allows an interpolation space to be constructed.

The second application allows the space to be used for interpolation and as the interpolation point(s) are moved OSC messages are then generated and transmitted to any sound engine that has been setup to receive them. An OSC to MIDI bridge is also provided so traditional MIDI-based technologies, that do not support OSC can also be controlled. An example of the interpolator space with a regular polygon layout and three interpolation points is shown in Figure 19.

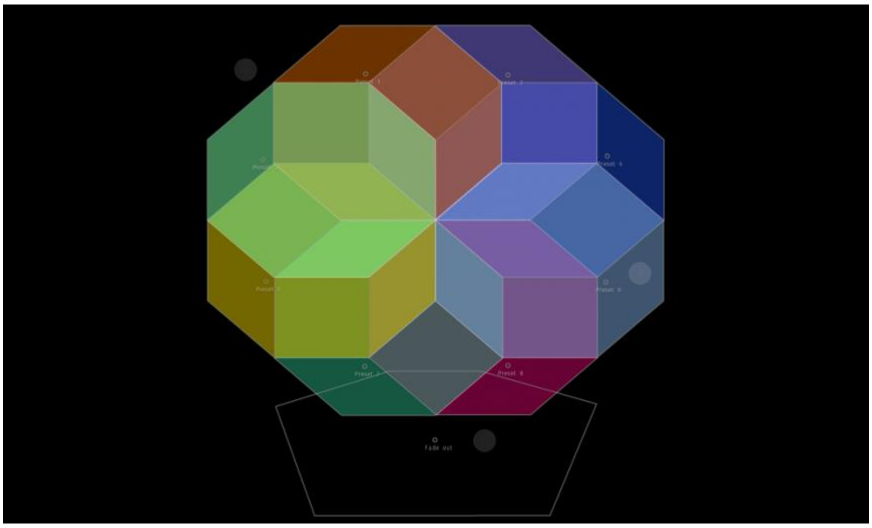


Figure 19 Polygon Interpolation using MIEM with a Regular Polygon Layout

This interpolator was initially designed for spatialization for electroacoustic music [200], but has subsequently been expanded out to sound parameter control [199]. These authors have also undertaken user testing with the developed platform where they compared participant performance when searching for a sound with the interpolator verses individual parameter sliders. This is similar to the experiment run previously by Hunt et al. [117], where individual parameter control was compared to a “multi-parametric” interface, but was done with a multi-touch technology. In this new study, four different parameters on a number of different synthesizers were used without the participants being told the exact parameter or particular synthesizer. In addition, the test order and the colours used to represent the presets within the space were randomized to remove any bias. The results were generated by comparing the parametric distance between the supplied reference sound and the results generated with both interfaces. The results concurred with those from the previous study in that overall the performance was better with the interpolator than the individual sliders and this was most pronounced for users that were beginner and intermediate users. As the experience level increased there was a narrowing between the two interfaces although the use of the interpolator still came out on top. The participants were also asked for their opinion on the speed, precision, intuitiveness and preferred interface. In all cases the interpolator interface was considered the best solution.

It is worth noting that this work is similar to an initial investigation that was undertaken in this body of work where the intersecting nodes model was expanded upon, detailed in Chapter 5, by constructing several nodes implementations using different shapes. As detailed, although this author did create a multi-shape interpolator using an intersecting model, it was never expanded beyond a limited range of shapes (circle, square, rectangle and triangle). It also did not allow the users to be able to define their own shapes.

2.5 Other Sound Design Systems

Over the years there have been a number of other sound design systems developed. A comprehensive review of these was undertaken by Misra, in 2009 [201], and many of these have already been considered. This review paper very much concentrated on the sound design of environmental scenes and as extensive as it was, since publication there has been some additional work.

2.5.1 Sound Design of Environmental Scenes

Following on from Misra's TAPESTREA system, already covered in Section 2.3.4.1.1, Verron designed a spatialized additive synthesizer for simulating environmental sounds. The additive engine is used in a hierarchical sound design process where environmental sounds (such as water drops, wind, fire, etc.) can be constructed from "basic sounds" (also called "atoms"). The use of these elements is controlled using high-level descriptors (such as size, intensity, trajectory, etc.) that are directly linked to the way basic sounds are combined and adjusted. This allows a complete three-dimensional auditory scene to be created from the additive engine [202].

The idea of controlling the creation of the auditory environmental scenes was expanded further with the Ambience Table [203]. This was directly inspired by the ReacTable [197], already covered in Section 2.4.2.13. However, in this case the ReacTIVision technology was applied as a control and interaction interface for the synthesis of ambient environmental soundscapes. This allowed the design, spatialization and manipulation of environmental scenes using an intuitive and tactile interface.

2.5.2 Physically Informed Sound Design

The idea of using physical parameters to control synthesized sounds is not new [204]. However, as computer-based technology has become more powerful in recent years it has become possible to create virtual worlds and use physical representations in these to control the synthesis models. This began with the Sound Design Tools package that was developed as part of the Sounding Objects project [69]. Work was then undertaken to consider possible physical interactions that could occur between the physical models of these sound objects. To aid this exploration a new tool kit was designed that allowed sound design to be explored through the interaction of different sound objects [205]. At the same time, similar work was being done with FoleyAutomatic, which described further algorithms for the synthesis of sounds based on interactive simulation [206].

Another system for synthesizing environmental sounds for virtual senses was presented in 2010 by Menzies. However, this system used physics-engines, associated to the virtual world, to control a Library of Physically Motivated Audio models, known as Phya [207]. Each sounding object model starts with a "body", which can be associated to "surface" and "resonator" models. The surface specifies how a "collision" will be generated on that surface and could be "impacts" or "contacts". On a surface, any number of collisions with other body surfaces could occur at any time. The resonator models generate the resonating state and so each body has a different resonator. The models can then be controlled by the physics-engines in the virtual worlds to generate sounds that correspond to what is happening in a particular virtual world. For example, the speed of a body contacting with a surface can be used to parameterize the sound generated [208].

Further work has continued into different aspects of physically informed sound synthesis [209], [210] that are leading to more accurate models for the sound objects and their interactions. However, a slightly different approach has been taken by Gohlke in 2011, which used motion-sensing technology to allow sound designers to apply gesture control to physical sounding objects. Hence permitting the sound designer to create the physical parameters for the sound objects using motion captured actions. This allowed them to use motor memory and motion skills to mimic generic and familiar interactions with everyday sounding objects, rather than requiring profound technical knowledge of the physical models. This allowed the user to focus on the expressive act of sound creation and when tested with sound designers it was accepted as a viable means of creating sounds in a fast and intuitive manner [211].

2.6 Research Summary

Having reviewed the subject domain in this chapter, it appears that interpolation systems and in particular graphical interpolation provides a solution that appears to meet the research questions raised in Chapter 1. Namely they provide a level of abstraction over the underlying synthesizer technology, allowing many parameters to be control simultaneously. The visual display fits with one of the four stages of interaction (graphical), defined in Section 2.1, and allows synthesis-based sound design to be done in an intuitive manner where moving the cursor towards a preset makes the output sound more like that preset and away less like it. This can be done with as many presets as necessary, and the user can determine how many and which parameters will be controlled. Moreover, there appears to be a natural synergy between the experimental nature of sound design and exploration of a graphical interpolation space. Based on this a more in-depth evaluation will be undertaken in Chapter 3 that will consider the different components of a graphical interpolation system.

Chapter 3 Design Considerations for Interpolation Systems

As defined in Chapter 1, the area of interest for this work is the act of “designing” new sounds with synthesizer technology and although a significant amount of research into the control of synthesizer parameters has been published, there are still areas that need further consideration. As professional sound designers tend to have extensive experience, when undertaking a sound creation task, they are likely to have ideas for the sound elements that they require or the direction they want to take the design. However, the designer will probably want to be considering the high-level “sound qualities” [16], rather than the fine detail of individual synthesis parameters, unless it is absolutely required. Sound design is a creative process and as a result there is a desire to remove or minimise any technical hindrances and allow sound designers to concentrate on the creative side. It has been shown from research into general creative processes that a large part of it comes down to generation and exploration [212], so it will be desirable to provide a platform that supports such a model.

Although many of the systems examined in Chapter 2, provide interpolated control of parameters, they represent different realizations, are implemented with different technologies and have a number of different application areas. Nonetheless, there is a common thread, in that interpolation allows the generation and exploration of sound parameters between defined presets, often via some form of visual model.

From the systems examined it is apparent that they can be separated into five different, but dependent areas that should be considered when developing such systems. These are:

1. Interpolation – calculating the interpolation values
2. Visual Metaphor – the visual interpolation model and how it is represented graphically
3. Control – input controls of the interpolation model
4. Synthesis – type/architecture/implementation of the sound engine
5. Mappings – the synthesis parameters that are interpolated

Each of these areas will be considered separately, however, for a number of the systems examined in Chapter 2 there was not a clear partitioning between them. In this body of work the separation between them is maintained so the individual impact of each area can be assessed. In addition, in this work they will be considered solely in a sound design context.

3.1 Interpolation Methods

Across the systems examined in Section 2.4, there are a wide range of processes for calculating the interpolation weightings between the presets. These can be as mathematically simple as linear interpolation used in systems such as the GRM Superslider [170] or as complex as the RST interpolation used in Van Nort's mapping experiments [133]. The method for calculating the interpolation will affect the sensitivity and "feel" of the interpolation system, as was demonstrated with LoM [129]. However, the method of calculating the interpolation can be dependent on the interpolation system's representation, control and parameter mapping (e.g. the SYTER gravitational model) or completely independent (e.g. as in the case of LoM). The non-graphical interpolation methods presented in Section 2.4.1, have generally been created to be utilised with some form of instrument or performance controller. Whereas with the graphical systems in Section 2.4.2 they have normally been created with the intension that the display is used as the input interface for either sonic exploration or performance. This being the case, the graphical systems offer a natural synergy to sound design, but the underlying interpolation methods from the non-graphical solutions may also work in a graphical context.

The graphical interpolation systems presented in Section 2.4.2 allow the generation of sonic outputs, based on a visual arrangement of presets and placement of the interpolation point(s). With many of these systems the user can define and freely change the layout of the presets within the space (scattering) as opposed to regular layouts, such as positioning on a regular grid, where the layout will result in regular influence being assigned to the presets. Scattering interfaces offer greater control over the topology of the interpolation space and therefore provide additional flexibility and if a regular layout is required the user can arrange the presets accordingly. As identified by other authors [119], [123], such graphical interpolators offer a platform that allows sonic exploration, based on the interpolation space defined by the chosen presets. The nature of the changes when moving between presets will be strongly influenced by the interpolation function used as well as the spatial arrangement. New presets can also be added to the interpolation space and existing presets removed, to change the constraints for the sonic output. In addition, the graphical interface allows multiple parameters to be controlled simultaneously. This could be as few as one parameter, or as many as all the parameters in the presets. Moreover, the graphical interface masks the synthesis process and details from the users, allowing them to concentrate on the sonic exploration, without worrying about the synthesis details. All of these factors match well with what is required for a sound designer's process.

As noted by a number of authors [118], [123], [124], [166], [173], it has been clearly identified that the interpolation model should produce a "smooth" sonic output that does not possess discontinuities or overshoots [118]. This will allow smooth changes and variation of the synthesis parameters and so the sonic output. Discontinuities would result in step changes in parameter values and as a result the sound would "jump" from one sonic output to another. Although in some situations this maybe what is required, this is something that should be under user control and not occur unexpectedly.

3.2 Visual Representation

As can be seen by the range of systems examined in Section 2.4.2 there have been many proposed visual representations for graphical interpolation systems. This visual representation provides the user with feedback on the interpolation, primarily the proximity of the interpolation point(s) to the preset locations in the space, but in some cases further details are provided. These visual cues are delivered in addition to the auditory feedback generated by the synthesizer output. However, for a sound design task it is not clear if the visual representation is needed and actually aids the process. Moreover, is there specific information that would aid a design task and what is the best way to provide cues visually?

In most cases these systems use a two-dimensional representation for the interpolation space and presets are placed within the interpolation space. There have been different visual representations for the presets within the interpolation space, often with geometric shapes: circles, triangles, polygons, etc. However, there is often some form of visual linkage between these representations and the actual method of interpolation. For example, circles have been used to represent presets in a number of different interpolation systems. However, the circles representing the presets are considered differently depending on the interpolation model being used. For example, with SYTER the circles represent planets in a gravitational based representation. Then the interpolation is performed, based on the cursor position between the planets and their gravitational force [165]. In the Max nodes object, the circles represent the extent of influence for the preset and the interpolation is performed where the nodes overlap, based on the interpolation point's distance from the centre of each overlapping node [183]. In Intersecting N-Spheres the circles represent the distance between the presets or the interpolation point. Then the interpolation weightings are calculated as the ratio of area for each preset intersecting the interpolation point [190]. In each of these examples, spatial interpolation is being performed and although in each the presets are represented by a circle, the way they are interpreted is directly linked to the interpolation metaphor being used.

In other cases, the shapes used in the visualisation of the interpolation space are linked to those presets that will be included in the interpolation calculations. For example, where triangulations are generated between the preset locations it provides the implication that the interpolation is being performed between the three presets of the enclosing triangle [186], a rectilinear grid implies interpolation between the four local presets [125] and using polygons implies interpolation between the closest presets forming a convex hull around the interpolation point [123]. Even a straight line (slider) can be used to imply interpolation between two presets [172]. Whereas with an intersecting paradigm, where the interpolation is performed when preset objects overlap in the space, the intersection itself implies which presets are included in the interpolation. In other cases, the presets included in the interpolation are implied by guidelines that connect between the interpolation point and the included presets [180], [194].

As well as different geometric representations for presets in the interpolation space, colour is also used in the majority of the systems examined. In most cases the colour is used to differentiate between the presets within the interpolation space. However, in some cases the colours or shadings are visually interpolated to give a visual cue for the weightings between the presets [119], [123], [162]. The systems that permit the creation of multi-layer mappings use colour to distinguish between the different layers in the interpolation space [177], [180], but with the INT.LIB realisation the visual transparency of the preset changes real-time to provide feedback on the influence of each preset. As a result, when a preset is displayed with a solid colour it has high degree of influence and it will become more transparent as the influence decreases [180]. This is also the case for the guidelines that show which presets are included in the interpolation, although the base colour already provides this information.

It is also worth noting that for the majority of the systems examined the representation relates to the interpolation of the parameter space and does not directly relate to the sonic output. As seen through the work on timbre space [135], it is possible to use this as a representation for the control of synthesis parameters [136]. Although work has continued in this area [143], [145], for the sound design application area being considered here, the use of timbre space will potentially restrict the possibilities as the sounds are normally defined as having musically pleasing tonal colour. Moreover, the resulting sound design will be a function of the timbre representation chosen for the space. This is clearly demonstrated with Arturia's Sound Maps where the presets have first been categorised based on type of sounds: lead, pad, bass, percussion, FX, etc. Each preset has then had its sonic output analysed and is positioned in the space, based on the audio characteristics of the output [194]. However, there can still be huge sonic differences between presets that are closely positioned in the map. In addition, the analysis of the sonic output from each preset and the calculation for its position in the map must be completed before the interpolated sound design can be performed. In the case of Arturia's Sound Maps, this has been done by the vendor and cannot be changed by the user. This means that it is not possible for new presets to be added to the sound maps. As a result, most of the visual interpolator systems do not use automatic placement and positioning of presets within the interpolation space. Instead the user is allowed to define the positional relationships between presets in the interpolation space. Therefore, the user can both define the presets that will be used in the interpolation, and also the relationships between them. The user can also determine the influence of each preset, but this will be dependent on the interpolation model. For example, in the gravitational model used in SYTER, the size of the planet determines its gravitational force, and its position will determine its influence on other planets [165]. With Interpolator, the extent of each lamp's light beam gives a direct visualisation for the influence of each lamp [177]. For the Max nodes object the user can change the size of each node and so its field of influence [183]. Metasurface only performs the interpolation between neighbours so the positional relationship solely determines the influence [123]. The other systems directly use the distance between the presets to determine their influence [119], [166], [180], [186], [170].

Finally, with most of the visual interpolation systems, individual presets can be recalled by positioning the cursor of the interpolation point directly on the preset's handle position. However, with the SYTER gravitational model the gravitational force remains the same while on the planet's surface so any position on the planet will recall that preset [165]. As a result, the area of each planet effectively reduces the potential size of the interpolation space. Also, with the nodes-based intersecting model, the preset sounds can be recalled by placing the cursor on a node area where no intersection exists. However, based on the layout of the nodes within the interpolation space, this may not always be possible.

From this analysis it can be seen that previous systems use some or all of the following visual cues in the interpolation space:

1. Preset handle (location of preset sound in the space)
2. Preset field-of-influence
3. Interpolation point(s)
4. Number of presets included in the current interpolation calculations
5. Preset strength at the interpolation point
6. Navigable interpolation space (region-of-interest)

3.3 Interpolation Control

As has been seen, there have been a number of methods for controlling interpolation systems. In Section 2.4, these have been categorised as either offering a visual interface that corresponds to the interpolation or those that use interpolation using some other form of control that does not require the visualisation. As many of these systems have been designed for performance the choice of controller is often linked to the performance requirements and is often some form of physical controller. Joysticks and drawing tablets are the standard spatial controllers used with many interpolation systems as well as instrument-based controllers [113], [127], [138]. Then there have been more unique controllers that have been designed for specific performance requirements, such as Dancing Viola [167], The Sponge [191] and voice-control [193].

Nonetheless, even for the graphical interpolation systems the control mechanism is an important consideration. With many of the older systems it was not possible to calculate the interpolation values in real-time so they were often pre-calculated offline. As a result, a number of these systems had two modes: one for the creating and editing the interpolation space and another for actually performing the interpolated sonic output [118], [123], [163], [171], [177]. This meant that the interpolation space could not be changed in the middle of a sonic exploration, without changing mode. With the computer processing power that is now available it is possible to perform the calculations in real-time for some of these interpolation models, dependent on the system complexity. This extends the control options to perform sound interpolation either by changing the position of the interpolation point within the space or by modifying the space: moving preset locations or adding and deleting presets. Moreover, with the layered interpolation systems presented, Interpolator and INT.LIB, it is possible to have multiple interpolation points and these could either be moved individually or linked so they could all be controlled simultaneously [177], [180].

It is also worth considering the input devices used for controlling a graphical representation of the interpolation space. Using traditional computer-based spatial control devices (mouse, drawing tablet, joystick, etc.) generally only allows one 2D value to be controlled at a time. This would mean that only one preset position or the interpolation point can be moved at a time. Whereas with multi-touch screen technology, simultaneous control of multiple points in the interpolation space is possible [195]. This potential creates new modes of operation:

1. Move the interpolation point(s)
2. Change the field-of-influence for one or more presets
3. Simultaneously move one or more preset locations, while the interpolation point remains static

4. Simultaneously move the interpolation point and one or more preset locations

As mentioned in Section 3.1, discontinuities in the interpolation model are not normally desirable for this application domain. However, if the user does want to produce audible jumps either a new instantaneous position for the interpolation point can be selected or if real-time mode is available the position of the presets could be instantaneously changed. This could cause a sonic jump, depending on the interpolation method.

It is also important to note from the sound design examples examined that most designed sounds would require the addition of performance expressions to bring them to life. For the lightsaber example given in Section 2.2.2, the base sound of the motors and hum was designed and then a Doppler effect was added through performance to provide the sense of movement. It has already been shown that interpolation methods provide an opportunity to apply expressions to synthesized sounds [150], [151], [154], [156]. However, it does not necessarily follow that expressive control of the interpolation should be performed at the same time as the design of the base sound. It may be the case that more traditional avenues for applying expressions will still be preferred, for example, through physical actions [211].

3.4 Synthesis

From the range of systems examined in Section 2.4, it can be seen that interpolation has been used with virtually every kind of synthesis. Nonetheless, the vast majority of these interpolation systems have been directly integrated into the same platform as the synthesis engine. This means that in many cases although the synthesis can be changed within the remit of the given engine, it is not possible to use the same interpolation platform with a completely different engine. Exceptions to this have come from those systems developed through visual programming environments [129], [131], [167], [180], [183]. The flexibility of using the programming environment means that it is possible to build new synthesis engines to be used with the interpolation model. Moreover, as the software Max also supports use of common audio plug-in formats [214], it is possible to use many commercially available software synthesizers with interpolators built in Max [180].

As mentioned in Section 3.1, the use of an interpolator interface for sound design allows the details of the synthesis and the associated parameter manipulation to be masked from the user. This will allow the designer to concentrate on the design process, without having to worry about the details. This hiding of the synthesis details makes the design of a system where the engine can be changed for different types of sound synthesis attractive, as it increases the sonic potential of the system without requiring the user to learn new technical details.

3.5 Interpolation Parameter Mappings

Although the use of interpolation gives the user a mechanism to adjust multiple parameters simultaneously between preset values, the sonic changes possible is defined by which parameters are mapped to the interpolation points. Interpolating all the parameters within a set of presets can create large sonic changes, depending on the chosen sounds, whereas a mapping that contains fewer parameters will allow more subtle variations. Moreover, with some forms of synthesis there is not an obvious link between the synthesis parameters and the sonic output. Therefore, how will the sound designer know which of the synthesis parameters will give the desired sonic outputs? Again, this will come down to either experimentation or the expertise of the sound designer to know the relationship between the synthesis parameters and the sonic output. Although some investigation into the mapping of parameters has been done, these have very much focused on musical outputs [116], [117], [118], or gestural control [149], [152], [154], of instruments. As well as being a different application area, the outcomes of this work have been fairly broad and have not considered specific relationships. In addition, multi-layered mappings have been proposed, where intermediate abstract parameters can be used [117].

As already mentioned in section 3.1, a number of authors have identified the desire for continuity from the interpolation method. However, for interpolation systems that are being used in musical instruments, a number of other desirable characteristics have also been identified for the mappings, such as, differentiability, linearity, range space, exactness, extensibility and editability [118], but it is not clear if all of these are of importance and desirable in a sound design context.

With the graphical interpolation systems examined in Section 2.4.2, the mapping between the synthesis engine and the interpolation point is controlled by the user. This is generally done by presenting the user with a list of parameters and allowing them to select the desired parameters to map between the preset handles on the visual interface and the synthesis engine. Although this process gives control to the sound designer, completely different sonic outputs will be generated depending on which parameters are selected and which are not. With the majority of these systems one set of mappings is controlled by the graphical interface. However, both Interpolator [177] and INT.LIB [184] allowed multiple interpolation layers to be presented simultaneously in the interpolation space permitting the interpolation of different parameters. As highlighted in the previous section, with the INT.LIB system each layer can either be controlled individually with its own interpolation point or the points for the different layers can be linked so that all the layers can be controlled simultaneously, as was the case with Interpolator. However, with INT.LIB each layer was mapped to a different synthesis module so different sounds could be layered and controlled separately. Whereas with Interpolator the different layers were on the same synthesis engine which permitted different aspects of a sound to be changed through multiple mappings.

As outlined in the literature review, sound designers within the film industry currently work largely with recorded sound, rather than synthesized sounds [4]. Working with sound this way means that the designers are limited to morphological operations [61], as presented in Section 2.2.3. Potentially working with synthesized sound would offer sound designers much finer control and greater sonic possibilities, provided it is possible to design the sound in the first place. Moreover, with the wider adoption of procedural audio, particularly in the games industry [56], there is clearly a potential for this to see wider use in film production. There may be several factors for slow uptake of these techniques. First, traditional film is a linear medium, whereas modern games are often non-linear and potentially a variety of different sounds maybe required. Given this factor there is greater synergy between the linear nature of the film medium and linear recorded content. Conversely the non-linear nature of game content is easier to generate with synthesis techniques so the sound can be generated to match the live requirements. Secondly, having looked at a wide range of material on sound design in the film industry there does appear to be a *it has always been done this way* attitude to the use of recorded sound. This maybe driven by time and cost constrains, as these are often major factors in the film industry [17]. In order to promote the uptake of sound design with synthesized sound in the film industry, it would be advantageous to provide a mechanism that allows the same morphological outcomes to be created, with which designers are already familiar. The key to being able to do this will be working-out the relationship between each operation and the mappings of the synthesis parameters.

Chapter 4 Methodology

Although a number of graphical interpolation systems have been created and documented, they were developed over a forty-year period, using different implementation platforms, different synthesis architectures and designed for different application purposes. Consequently, many of the realisations used technologies that are now obsolete and so no longer available. As a result, it is not possible to do any back-to-back evaluation between the original systems.

Although previous work has been undertaken into the five different areas of graphical interpolation systems identified in Chapter 3, it is clear from the evaluation that some have received more attention than others. In addition, much of the work undertaken in each of these areas has been from the perspective of digital musical instrument design, and not sound design. Nonetheless, four of the five areas have received investigation from a number of different parties and comparisons and evaluations have been published (interpolation [126], control [153], synthesis [122], mapping [117]) relating to digital musical instruments. Although as covered, a number of different visual representations have been offered, until now no formal evaluation has been undertaken for the different visual representations. In previous work, the visualisations seem to be determined by the interpolation and/or control paradigm with little consideration of what is required from a user perspective.

As covered in Section 2.1 many authors have used Human Computer Interaction (HCI) techniques for the design and evaluation of musical instrument systems, for example [215], [216], [217], [218], [219]. These have been applied to both the physical controllers associated with musical instruments and computer-based instrument interfaces. Moreover, formal frameworks have been developed for the design [220] and evaluation [221] of digital musical instruments, using adaptations of HCI techniques. Although sound design may not be considered a musical performance in the traditional sense, the concept of designing sounds is closely linked to the idea of sound objects in Musique Concrète [222]. Given that Musique Concrète is defined as part of the development framework [220] for digital musical instruments, this framework will be used as the starting point for development and evaluation of the different visual representations for the interfaces. The existing representations will be compared using dimension space analysis to try to understand the design of each interface and the differences between them [223]. Although a dimension space has been defined for musical devices this will require modification for the sound design context of interest, based on an analysis of the design goals and practices. The design space plots will then be used to directly compare the design characteristics of each system's visual representation. For example, the following characteristics that are specific to these interfaces will be considered for the graph axis: field-of-influence, number of patches, visualisation of patches included, etc. These may then be combined with some of the general instrument interface metrics [223].

In order to be able to evaluate the suitability of graphical interpolation systems for the purpose of sound design they require reimplementation so that comparative testing can be undertaken. This permits direct comparisons to be made between the different interpolation systems through sound design tasks and usability tests. Moreover, if there is clear partitioning between the different areas identified in Chapter 3, it will be possible to isolate the different aspects (interpolation, visual representation, control, synthesis and mappings) of each system and evaluate each separately. In this way, it will be possible to directly compare these aspects for each system. The results can then be evaluated to determine the suitability of each aspect for the sound design application area of this study.

The systems will be evaluated for sound design based on HCI approaches [224]. For the design and evaluation of new systems this would be based on a five-stage approach consisting of:

1. Identification of Users and Tasks
2. Selection of Data Visualisation Metaphor and Appropriate Interaction Styles
3. Prototyping
4. Design of Usability Tests
5. Evaluation of Results

As the systems being reimplemented have already been designed, albeit for different contexts, the stages will be undertaken with respect of sound design as the goal. The prototyping stage will look to replicate the original functionality as faithfully as possible so that the original designs can be tested and evaluated given the new context. The usability testing will be based on the strategy for musical instruments [221], [225], but will be modified in accordance with the outcomes from stages 1 & 2. Any modifications will be done with the desire to generate a repeatable collaborative evaluation process [226]. Also, when stages 4 & 5 are designed they will be completed in compliance with standards of evaluation and testing in the sector [227]. As already highlighted in Section 2.1, in this sector when systems are considered from a design perspective the use of quantitative analysis and evaluation is more widespread [10]. As a result, authors of this study identified there is evidence that not all the work considers the full user experience. As qualitative approaches are more widely used for assessing a user's perspective, a hybrid approach will be employed where both quantitative and qualitative methods will be used for the evaluation of the graphical interpolators. This will be achieved through task-based activities where quantitative user data can be captured, but questionnaires and critical incident technique (CIT)[5] will also be used get qualitative data and gain insight into the full user experience.

When undertaken the testing and evaluation will be centred on answering the research questions raised in Chapter 1. As a result, the primary focus for the testing will be on the visual representation, the relationship between the visualisation and the sonic output, the application of expressions and morphological operations. When this has been completed the results can be used to develop further graphical interpolation interfaces, based on the outcomes. Full details of the testing are given in the rest of this chapter.

4.1 Framework Development

In software development, "frameworks" are often used to provide structure for the development of applications with certain functionality and to promote rapid development through reuse [6]. Frameworks can be used to provide an abstraction and generic functionality that can be customised with user-generated code, so that core functionality can be achieved quickly, but yet offering the ability to create bespoke solutions. The notion of a framework matches with the desire to reimplement a number of graphical interpolation systems, test different aspects of them and then develop new systems in response to the results obtained. With all of the graphical interpolators that have been reviewed in Section 2.4.2, there is certain core functionality that is common to all. Through further analysis these will be extracted and refined into a framework that can be used in the creation of such systems and a template architecture.

The framework will initially be developed through the creation of a graphical interpolation system built in Max/MSP using the *nodes* object detailed in Section 2.4.2.9. This will act as a proof-of-concept and allow the infrastructure of the interpolation system to be built using the only currently available model. Moreover, as this graphical interface is already available it offers the most rapid development times. This prototype will then be used to undertake initial testing and evaluation to establish the general suitability and limitations for using interpolation as an interface for synthesized sound design. The framework will then be broadened out to allow the implementation of some of the different graphical models and will be tested by reimplementing a number of different graphical interpolators. Full details of the methodology and the work undertaken can be found in Chapter 5.

[5] A background overview of critical incident technique is provided for reference: https://www.nngroup.com/articles/critical-incident-technique/
[6] Overview of software frameworks is provided for reference: https://www.sciencedirect.com/science/article/abs/pii/S0920548998000245

4.2 Pilot Study

Once the framework has been developed and defined it will be used to test the impact that different levels of visualisation have on the usability of a graphical interpolator. The aim of this testing, as well as to investigate different visualisations, is to act as a pilot study for a larger usability test that will be subsequently undertaken. The pilot study will allow further testing of the framework usage in the development of interpolators. It will also allow the appraisal of a usability testing strategy for these systems that will form the basis of wider testing for different types of graphical interpolators that will be undertaken afterwards. The exact rationale, methodology, experiment design and results for the pilot study are given in Chapter 6.

4.3 Interpolator Usability Study

Having used the pilot study as a proving ground for both the framework and the testing methodology, a wider study will be tackled. In this, different interpolator models will be compared and evaluated in a series of back-to-back sound design tests. The framework will be used to develop the interpolators and the testing undertaken in the pilot study will provide the foundations for the usability testing that will be done. However, in both cases lessons learned through the pilot study will be fed in to make refinements and improvements to this testing. Full details of these and the exact methodology are given in Chapter 7.

4.4 Response Development

The results from both the pilot study and the wider usability testing will be used to build a new paradigm for the visualisation of graphical interpolators. This looks to keep the benefits of each interpolator model while attempting to provide additional visuals that may be of benefit for sound design tasks. Full details of the development and evaluation of this is provided in Chapter 8.

Chapter 5 Graphical Interpolation Framework & Exploration

From the work that has been previously undertaken it can be seen that the graphical interpolation systems satisfy the questions of interest defined in Chapter 1. Although a number of graphical interpolation systems have been created and documented, they were developed over a thirty-five-year period, using different implementation platforms, different synthesis architectures and were designed for different application purposes. Consequently, many of the realizations used technologies that are now obsolete and no longer available making it impossible to run back-to-back evaluations between the original systems. In order to evaluate the effectiveness of these graphical interpolation systems for sound design, they needed reimplementation on contemporary platforms. This allowed direct comparisons to be made between the different interpolation systems.

It is important to also consider the characteristics that a sound design graphical interpolator should ideally possess. The following summarizes the most important factors from the assessment undertaken in Chapter 3:

1. Synthesis independent interpolation – the same interface can be used with different synthesis engines
2. Clear relationship between interpolation control and the sonic output – sound space defined by the populated parameter presets
3. Constrain the navigation and exploration of the parameter space – user selecting and positioning presets in the interpolation space (scattering)
4. Control a number of parameters simultaneously – reduce the control complexity of many parameters
5. Changeable parameter mappings – provide users with control over the parameter mappings
6. Exploration of the sound space with both coarse and fine levels of detail – change resolution and precision
7. Smooth interpolation – no discontinuities unless user selected
8. Real-time interpolation (not different edit/interpolate modes) – allow either preset points or cursor to be moved to change sounds
9. Support the design of base sounds and the application of performance expressions
10. Usability, repeatability, predictability and playability – user can design a sound based on the supplied preset sounds

To evaluate different aspects of graphical interpolators, a hierarchical framework is proposed that compartmentalises each of the system elements common to graphical interpolators. This can be thought of as a processing pipeline, where the inputs control the interpolated sonic output, via the pipeline. It starts with the control input at the top-level moving to the sonic output at the bottom, as shown in Figure 20. Although the final output, sound, is at the bottom level it is worth noting that the visual representation typically also gives the user visual feedback on the current configuration of the interpolation system and therefore the sound. Equally the user may be given inputs that allow the parameter mappings to be modified. However, what the framework shows is the interdependencies of the different elements of an interpolation system and the relationships between them. For example, the sonic output from the synthesizer is dependent on the control inputs, the visual model, the interpolation function, the parameter mapping and the synthesis engine used.

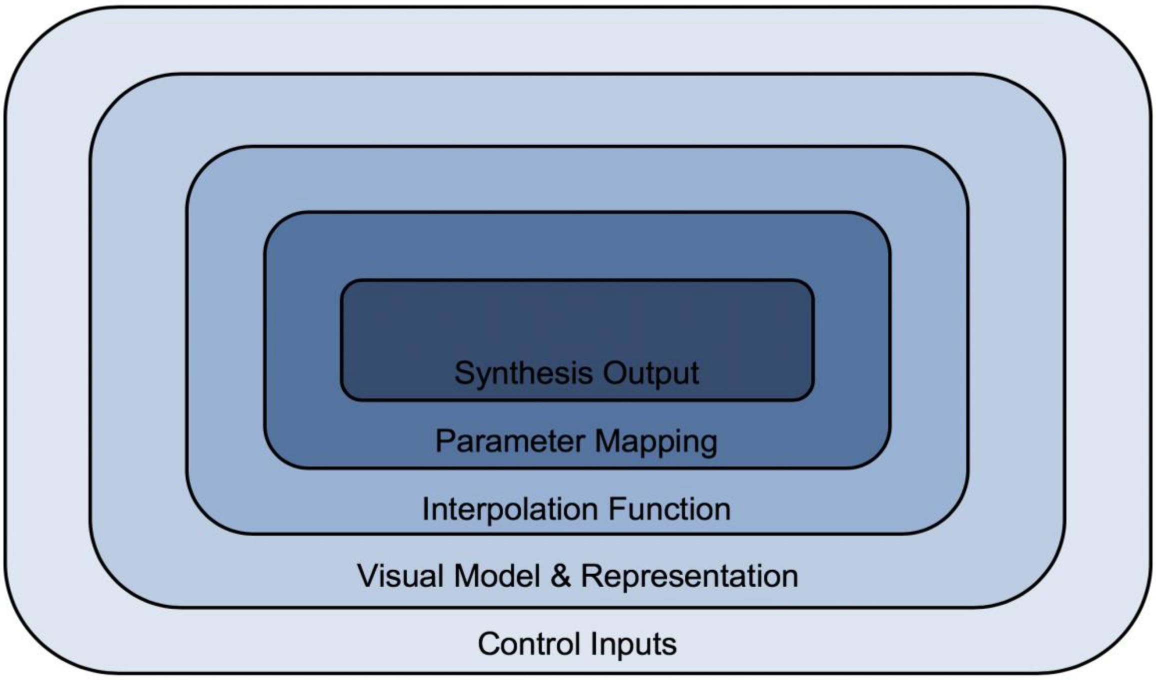


Figure 20 Graphical Interpolation Framework with Hierarchical Layers from Control Inputs at the Top to Synthesis Output at the Bottom

5.1 Framework Implementation

Having formalized the framework, the next stage was to consider implementation. Using the defined framework, it was possible to structure the different levels (control, visual model, interpolation, mappings and synthesis) into separate modules and test them individually. In this way, it was possible to directly compare these aspects of each system and evaluate their impact on usability through comparative user tests, where only one element is changed at a time. To facilitate this the framework has been implemented in the Max environment using the architecture shown in Figure 21.

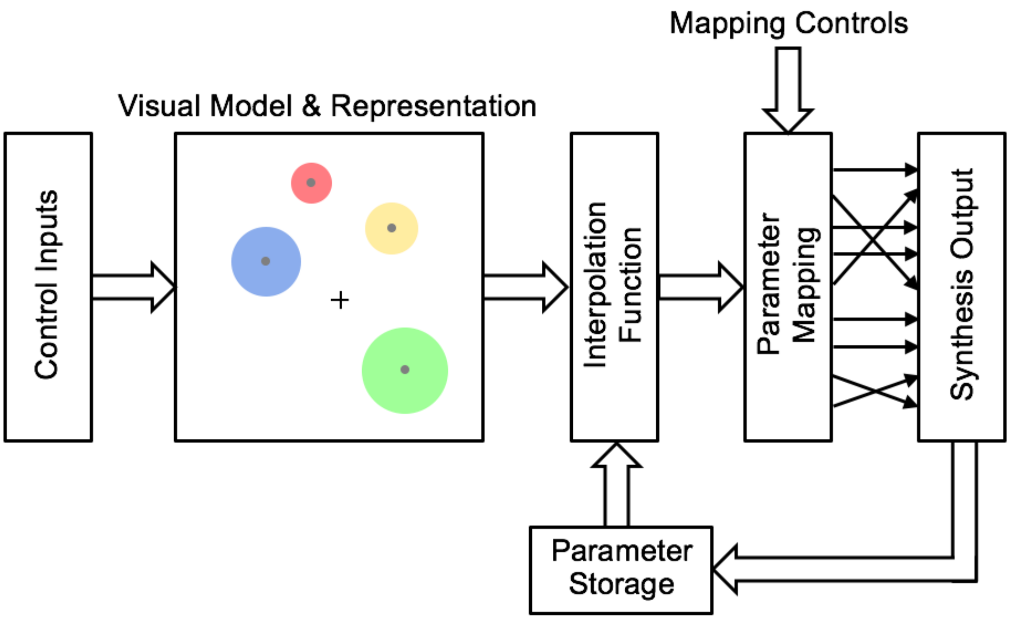


Figure 21 Framework Architecture in Max Showing the Structure of the Different Modules

As a proof of concept, a graphical interpolation system was built in Max using the *nodes* object detailed in Section 2.4.2.9. When this interpolator system was implemented, care was taken to develop each of the five elements of the interpolation framework into separate modules so that each one can be modified independently of the others.

The first implemented was the **interpolation function** module, which is storage that holds the parameter values and performs the interpolation. The parameter values for each synthesis preset are stored as a new data set and it then interpolates between the parameter data sets, generating interpolated values for all the individual parameters. The interpolation is performed based on the modules input which is the relative weightings for each preset. By default, the calculation performed is linear interpolation, but it is possible to change the mode so that any interpolation function can be realised.

As the *nodes* object has been specifically designed as a graphical interpolator, the object has been created with certain functionality for the **visual model** and the **control inputs**. The **control inputs** realized in the *nodes* object are standard computer-based spatial controls. However, it is also possible to send the object positional input data from other sources and so other input devices could be used to control the interpolation space. When the interpolation point on the *nodes* object is moved within the space, an output weighting for each node is generated. The **visual model** generates node weighted distances (0.0 – 1.0) which are proportional to the interpolation point's distance from the circumference of a containing node to its centre. That is 1.0 for the node centre and 0.0 for the point just beyond the node perimeter. Therefore, when the nodes in the interpolation space overlap and the interpolation cursor is placed in an overlapped region, a weighting is generated for each node. For the layout of nodes and cursor position shown in Figure 22, the nodes have the following values - 1 = 0.1696, 2 = 0.0000, 3 = 0.3180 & 4 = 0.5724. All the weightings are summed and the individual weightings are normalized relative to this total to give each as a percentage, as shown in Figure 22.

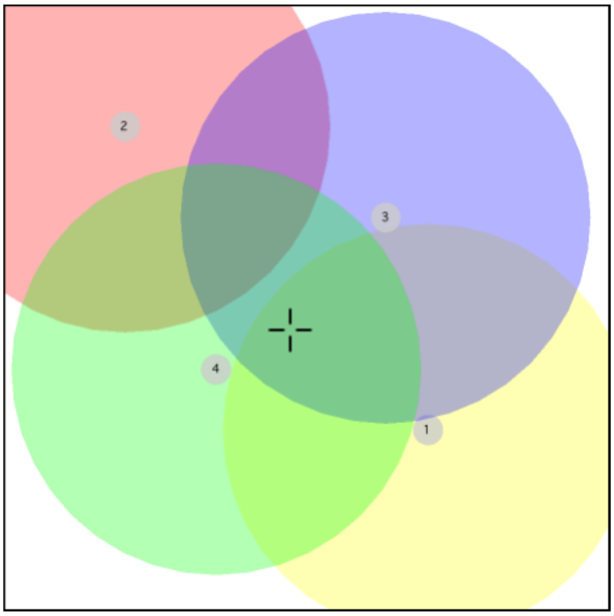


Figure 22 Nodes Outputs Weighted Distances Resulting in – 1 = 0.1696 (16%), 2 = 0.0000 (0%), 3 = 0.3180 (30%) & 4 = 0.5724 (52%) For Shown Cursor Position from the Interpolation Function

These weightings are used as the input to the **interpolation function**. As the visual interpolation model is encapsulated by a single object (*nodes*) it is possible to replace it with different implementations for the visual model.

The **synthesis engine** has been constructed to be separate from the interpolation platform by using software plug-ins, allowing different (and including commercially available) synthesis engines to be loaded and tested. However, the framework would also allow bespoke synthesis engines to be used, provided the access to parameters is the same. When a new synthesizer is loaded, it is interrogated to determine all the parameter values for the selected presets. Each preset is associated to a node in the interpolation space and all of the preset's parameter values are sent to the **interpolation function** storage.

By default, all of the parameters for the presets are associated to the corresponding node and so every aspect of the sound's synthesis is controllable. However, the **parameter mappings** between the **interpolation function** and **synthesis engine** can be changed by user selection. This allows the user to select which parameters are included in the interpolation and which will remain "locked" at their non-interpolated/last values. This is important as generally some plug-in parameters are not suited to interpolation, while limiting the interpolated parameters can focus the system on particular sonic attributes.

5.2 Framework Testing

The prototype nodes-based interpolator was initially tested to ascertain if each module built in the framework could be changed independently of the others and to establish the impact on sound design tasks. Through exploratory testing, where the nodes-based interpolator and its parameter space were left the same (shown in Figure 23), it became apparent that changes to each of the other modules in the framework leads to the system generating different sonic outputs and results in a different user experience with each realisation.

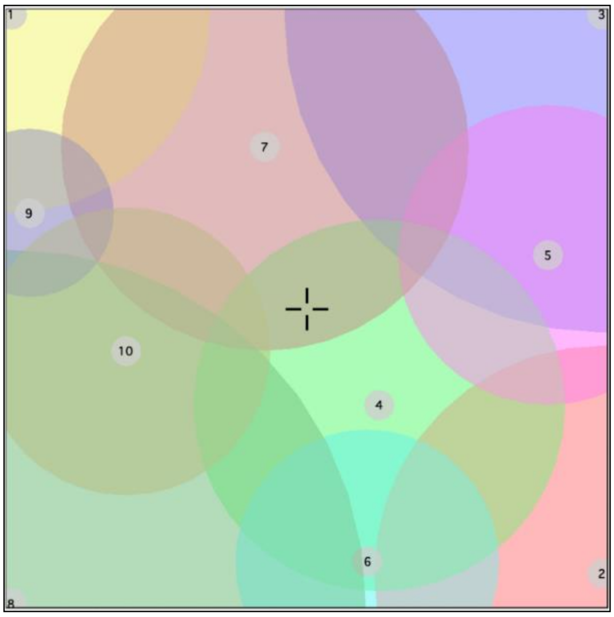


Figure 23 Nodes Prototype Test Layout

From testing with different **synthesis engines**, it was found that changes to the engine (preset changes, synthesis realisation or changes of synthesis type), were the main determinants of the sonic output. Moreover, with some forms of synthesis, changes to a single parameter can produce large sonic variations, but for others, more subtle alterations resulted. Changes to the **control inputs** allowed different mechanisms for interacting with the sonic manipulation, and potentially changing the usability of the interpolator. Modifications to the **parameter mappings** permitted the refinement of the sonic changes that it is possible to generate with the interpolator. Mapping lots of the synthesis parameters to the nodes resulted in big sonic changes, whereas mapping a few parameters permitted more subtle variations to be generated. Changing the **interpolation function** resulted the subtlest differences. The chosen function affects how the sound transitions as the interpolation point is moved between preset locations.

5.3 Graphical Interpolator Implementation

The prototype nodes-based interpolator was used as the basis for the subsequent development of different graphical interpolation systems. As the source-code for the *nodes* object was not available, it was reimplemented so that the visualisation of the model could be customised for comparative testing, allowing the impact of different visual models to be considered. For each visual model and its control, the nodes object was replaced with an interactive user-interface built using OpenGL for the interpolation model's visual representation and JavaScript to create the control mechanism and calculate the preset weightings. This separation allowed the testing of different visualizations or control with the same interpolation model to be carried out, while also facilitating implementation of alternative interpolator models in further studies. Each model was constructed and integrated with the other elements of the framework for testing.

5.4 Exploring the Nodes Reimplementation

Using the OpenGL/Javascript architecture the exact functionality of the original nodes object was realised. This was then functionally tested by undertaking back-to-back tests between it and the original nodes object, ensuring that both implementations gave identical results. This ensured that any results that were obtained with the reimplementation of the nodes model would be consistent with the original version.

The concept of the nodes model is an intersecting model, where circular nodes are used to provide weighted distances to each node's centre. However, as the interpolation is only performed in regions where the nodes overlap it means that the interpolation space is localised to specific regions and smaller than the whole parameter space. This can be seen in Figure 24 where the intersections between the nodes is shown by the shading.

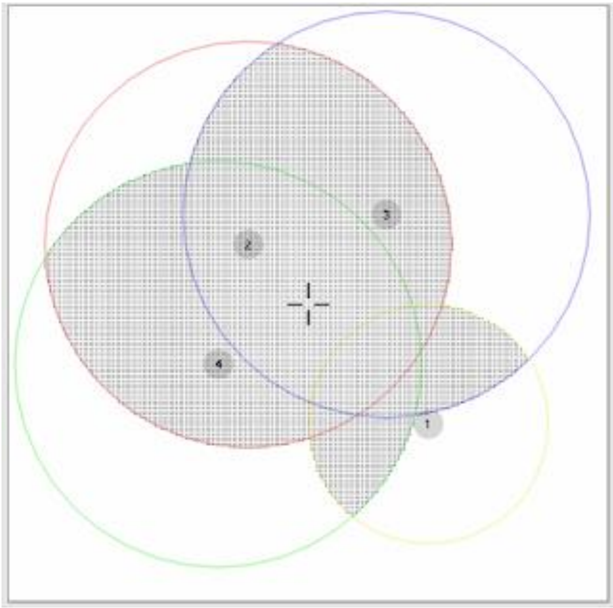


Figure 24 Nodes Model with Interpolatable Space shown by Shading

As can be seen for this shown layout the areas where interpolation is performed are smaller than the available parameter space. Even when there is a much denser layout of nodes it is often difficult to provide interpolation between all the presets and cover the whole space as shown in Figure 25, which has the same node layout as shown in Figure 23.

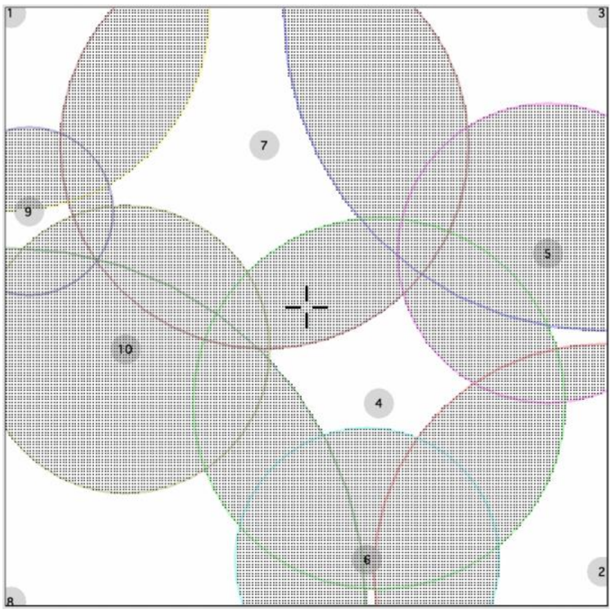


Figure 25 Nodes Model with Denser Layout of Nodes - Interpolatable Space shown by Shading

In the original nodes implementation, a transparency component in the visualization allows the intersection to be identified and shows the “regions-of-interest” where new interpolated sound will be generated (see previous Figure 22 and Figure 23 for examples). Although the visualisation shown in Figure 24 and Figure 25 are useful for clearly showing where the interpolation will be performed it does not give any indication of how the sounds will actually be interpolated. That is, the relative weightings of the interpolation. As already identified, in the past, some interpolation systems have visually interpolated the colour or shading to give a visual cue for the interpolated values between the presets [119], [123], [162]. An example of this was realised and is shown in Figure 26.

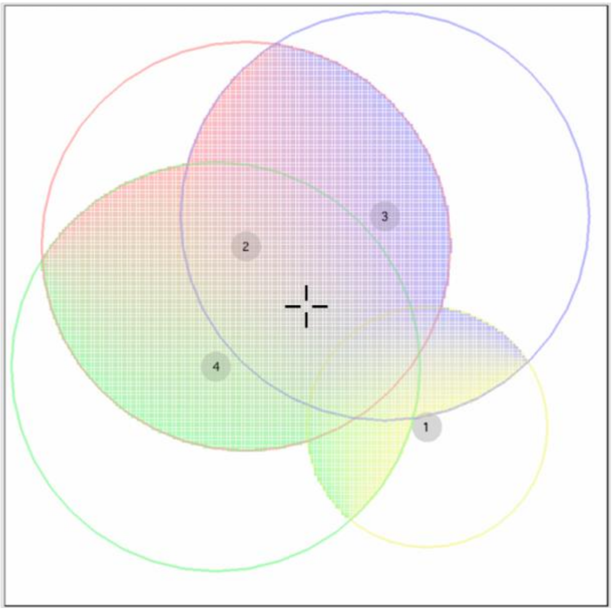


Figure 26 Nodes Model with Interpolation Values shown by the Interpolated Colour Shading

This provides not only a visualisation of the regions-of-interest, but also the relative interpolation level as an interpolated colour. The colour interpolation is performed using an RGB palette as it provides direct translation between colours [215]. For example, from Figure 26, the intersecting area between node 1 and node 3 is shown as an interpolation between the node colours *yellow* and *blue* and so only shades between these two colours result. If other colour palettes had been used, such as HSV or LCH [215], interpolation would have created shades of green, which in this example would actually represent node 4. Although this colour visualisation is useful for seeing how the interpolator will blend in-between the preset sounds, it is extremely expensive in terms of the processing required to generate the visualisation. For example, the image in Figure 25 took approximately ten minutes to generate when running on a 2018 MacBook Pro, with a 2.9 GHz Intel Core i9 processor and 16Gb of RAM as it was calculated on a pixel-by-pixel basis. Clearly with such a time overhead this visualisation is not practical for a real-time interface and although more efficient implementations could be created the Javascript engine will always be slow for such computationally intense operations. However, it has proved useful for comparing the interpolation space for different positional layouts of the presets or making comparisons of different interpolation spaces. However, as can be seen in Figure 26, as the colours become closer it becomes harder to distinguish between them within the space. For example, in Figure 27 as several different shades of green/blue are used for nodes 4, 6 and 8, it becomes harder to distinguish between them when they are in close proximity and the colours are interpolated.

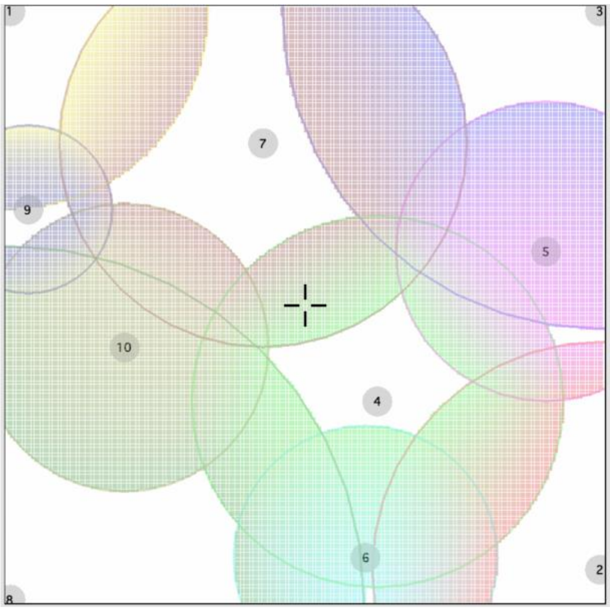


Figure 27 Nodes Model with Ten Preset Test Layout and Interpolated Colour Shading

This gives a good indication of how changing the layout results in a very different interpolation space being created in terms of the area and level of interpolation. This can be seen in Figure 27, which has a different layout of the ten presets.

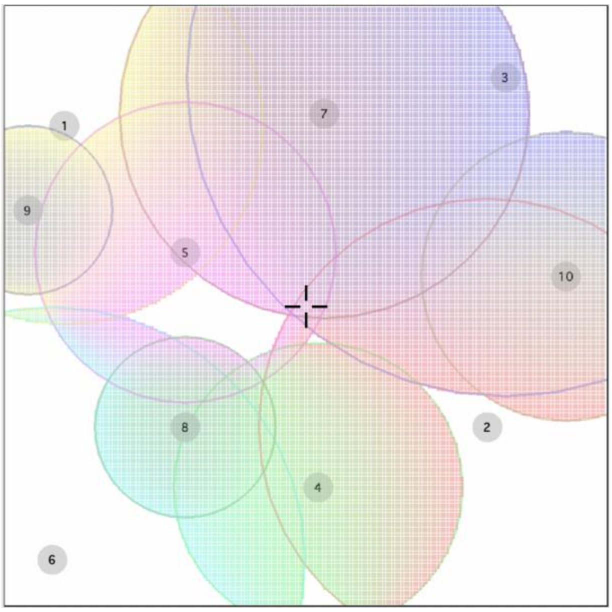


Figure 28 Nodes Model with Interpolation Values shown by the Interpolated Colour Shading

As can be seen, the different arrangement of ten presets within the parameter space results in new colour combinations being generated and as a result potentially different sonic outputs will be generated from the interpolator. In addition, it should be noted that with the nodes model the sound of the nodes preset is only recalled when the cursor is placed on a non-intersecting region of the corresponding node. With the layout shown on Figure 27, note that for preset 10, this node has no non-intersectional areas, so it is not possible to recall this preset. All the other nodes have some non-intersecting areas so their corresponding presets can be recalled. If this is compared to preset layout shown in Figure 28, it can be seen that presets 7, 8, 9 & 10 cannot be recalled.

To better understand the effect that the node layout has on the interpolation, different simple layouts were explored, both visually by examining the interpolation space and the sonic output to understand the resulting sonic palette. In the first instance the two nodes of the same size were positioned to create a small intersection. The interpolation was then explored between the two sounds. The nodes were then moved to increase the size of the intersecting region and the experiment was repeated. The layout of different positions between the nodes is shown in Figure 29 with layout colour interpolation to illustrate the relative weighting between the sounds.

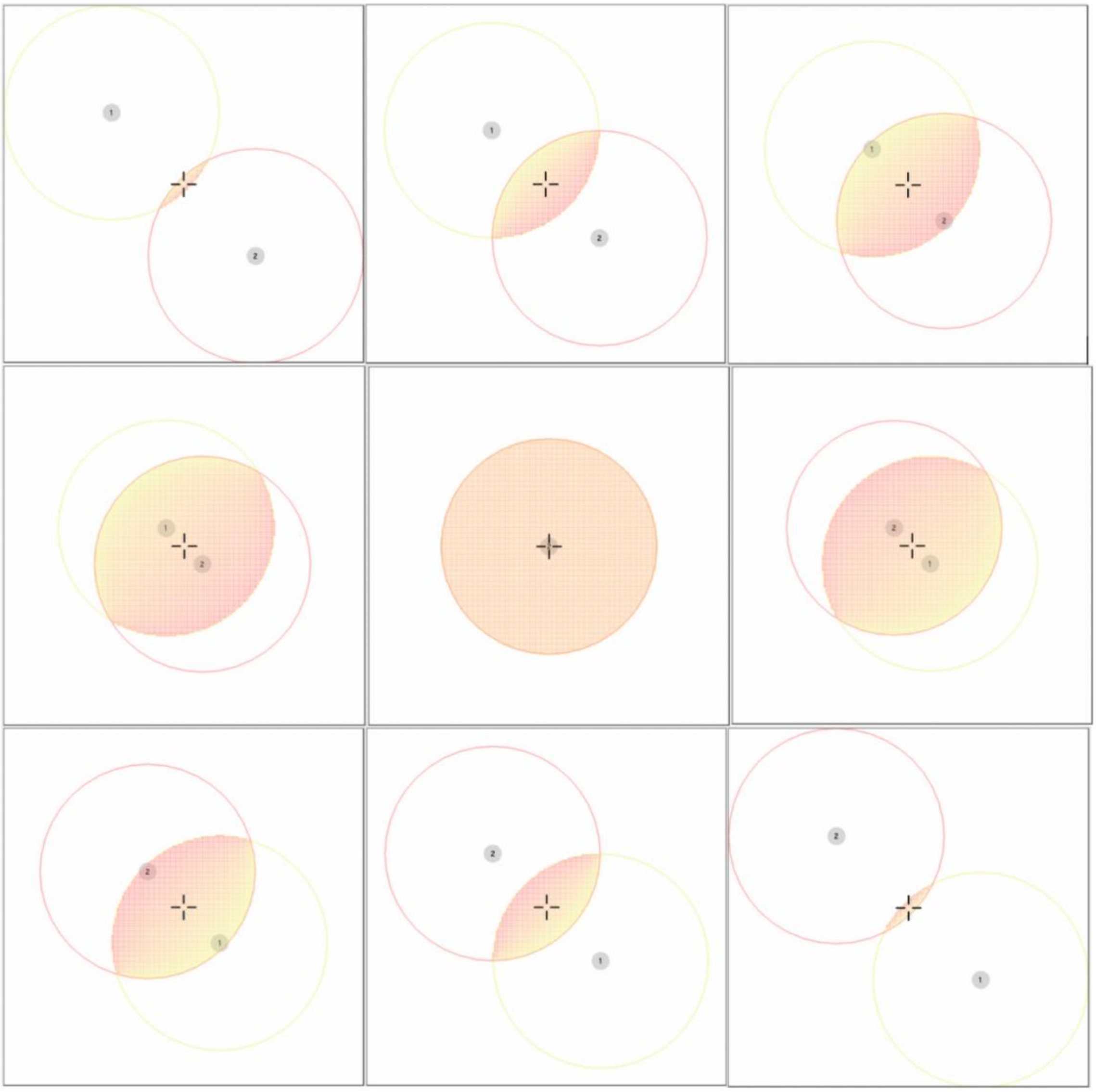


Figure 29 Interpolated Colour to Show Different Sized Intersections Between Two Nodes of the Same Size

As can be seen, both nodes have equal influence, due to both being the same size so the intersection between the two remains the same as the size of the intersection increases. This means the sonic outputs are the same. However, as the size of the intersection increases it provides a bigger area to explore the sounds and as a result, due to the positioning resolution, the cursor has more possible positions that it can be placed in. This provides a much finer adjustment of the sounds and so access to sounds that it was not possible to locate with a smaller sized intersection. In this way, increasing the size of the intersection provides a "sonic zoom in" which can be very useful when trying to explore interesting loci in the space. This trend continues until the nodes occupy the same position when interpolation will stop being performed and just the intermediate sound is generated. This can be seen in Figure 29 where both nodes are at the centre position and the intersection just shows a constant colour across the whole area. As the nodes are then moved past the central position the size of the intersection decreases and a "sonic zoom out" results. It should also be noted that as the nodes are moved past the central position, the node positions have swapped so do the resulting sounds that are being generated. This can be seen in Figure 29, as the region where the "yellow" sound is generated and moved to the opposite side of the intersecting region.

Next a similar experiment was performed, but this time with nodes of different size so each has a different influence. Again, the results were examined by considering the visual interpolation space, and also the sonic output generated. Figure 30 shows the colour interpolation for different intersecting positions between the two nodes. As can be seen, as node two is moved for each position the colours within the intersecting region are different, indicating that different weightings will be produced, resulting in different possible sounds. This is the same for all positions up to the centre location and then the weighting proportions repeat themselves, except each nodes influence is on the opposite side of the node. This can be seen clearly in the colour shading with the yellow and red regions of the intersection swapping position. Not only was this seen in the visual results, but it could also be heard in the interpolators synthesized sonic output. It should also be noted that the weightings between the two nodes are symmetrical around the centre line between the two nodes. However, weightings are clearly different along the length of this centre line axis. This can be seen in Figure 29, where if an intersecting line is drawn between the two nodes the colour shading is identical either side of the centre line, but different along the length of the line. The only position for which this is not true is when the two node centres are at identical positions and in this case, there is a smooth variation in all directions from the node centres.

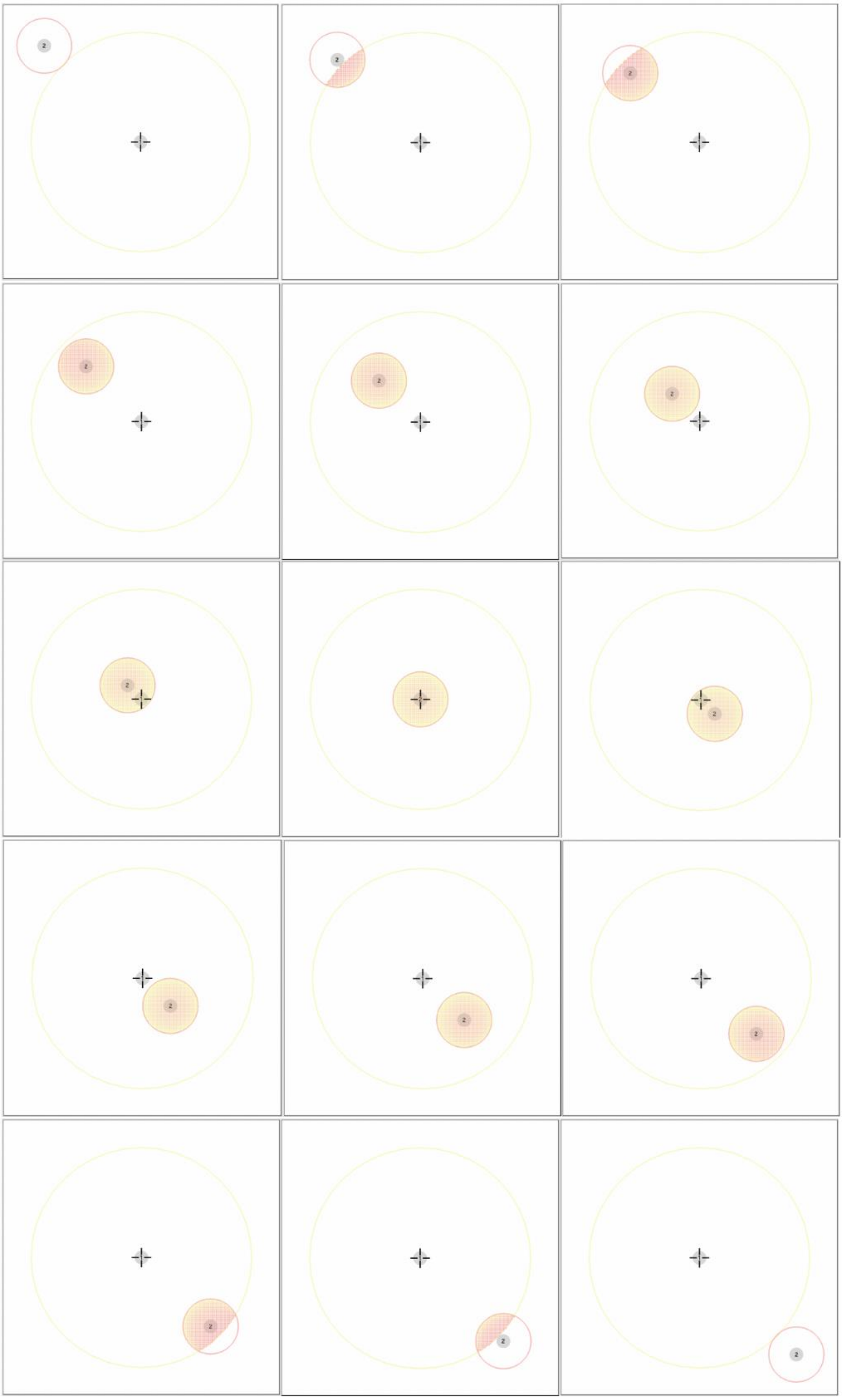


Figure 30 Interpolated Colour to Show Different Sized Intersections Between Two Nodes of Different Size

This process was repeated with many different sized nodes and it was found, as expected, that the bigger a node the greater it's influence over the weightings. Moreover, the bigger the size difference between them, the bigger the weighting variation produced for different intersection sizes. Until, as already seen in Figure 29, when the nodes are the same size and the same weightings are produced regardless of the size of the intersection. It should also be noted that when one node is entirely within another, as in some of the positions in Figure 30, when node two is entirely within the area of node one, it is not possible to entirely recall node two's preset sound, even if the cursor is positioned directly over node one's centre. (As previously identified, the only way to recall a nodes preset sound is to position the node so that a non-intersecting region is available.) As node two's position gets closer to the centre of node one's so its influence becomes stronger meaning that the sound generated at node one's central position gets further away from the preset sound.

Through this experimentation it is clear that use of circular nodes provides an even manipulation of the weighing's in all directions, with the relative positions of the centres affecting the interpolation and not a directionality that different shapes might introduce.

5.4.1 Alternative Node Geometries: Squares

Having seen in the exploration and evaluation of the nodes interpolator, detailed in the previous section, the interpolation space and the relative interpolation weightings are directly related to the shape of the nodes, which were circles in the original nodes object. Consideration was then given to the possibility of nodes that are represented by different shapes. The first to be implemented was a version of nodes that used squares to represent the preset sounds. This was created by modifying the code for the circular nodes reimplementation and this new version of the nodes model was integrated into the rest of the framework so the impact of this change could be evaluated. An example of the square nodes interpolator can be seen in Figure 31 with an identical layout to the original nodes interpolator shown in Figure 23.

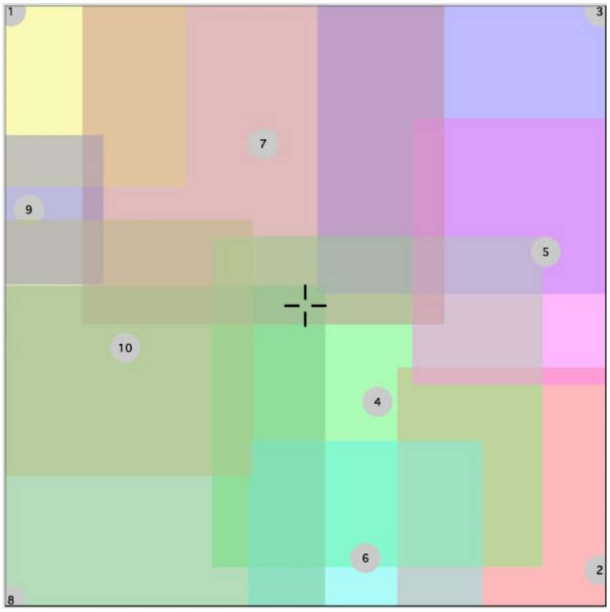


Figure 31 Square Nodes with Test Layout of Nodes

To provide comparability between the two node versions, the square nodes were implemented so that the squares had an identical area to the circular nodes. As a result, the total node area remains consistent. Despite this and as expected, the square nodes results in a different interpolation space being created even when the nodes are positioned at identical locations. This can be seen by comparing the same layouts with the original nodes, shown in Figure 23 and that shown in Figure 31, the two interpolators create a different interpolation space due to the different node shapes that are being used to create the intersections. The exact interpolation area is shown in Figure 32 where the intersecting areas are shown for the square node interpolator with the same layout as shown in Figure 23 and Figure 31.

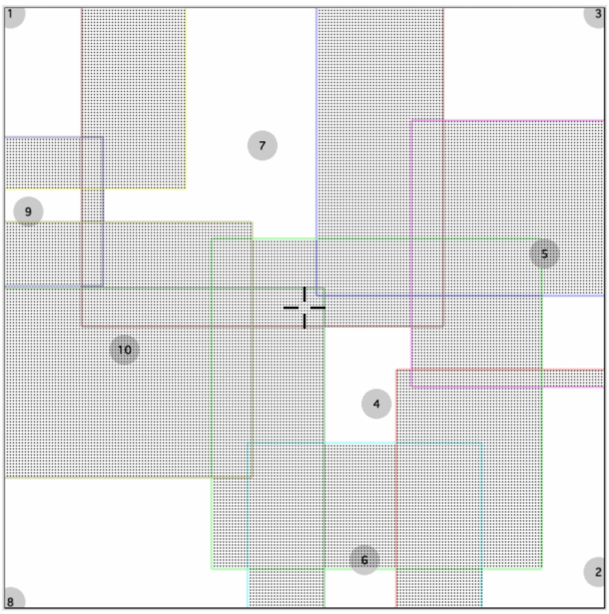


Figure 32 Square Node Interpolator with Test Layout - Interpolatable Space shown by Shading

Directly comparing this to the interpolatable space for the original nodes it can be seen that although there is a large amount of area that is common between the two, the different shapes do result in some areas that are different with new intersectional regions being created for the cursor position. For example, for the shown cursor position in Figure 32, the cursor is in an intersection created by nodes 4, 7 & 8. If this is compared to the identical cursor position shown in Figure 23, the original nodes creates an intersection for just nodes 4 & 7 at this position. As a result, the two interpolators will produce different weightings at this location and potentially different sonic outputs. To try and further understand the impact of the differences in the regions-of-interest between the two, colour interpolation was used to get a visual representation of the interpolation weightings. Figure 33 shows the square nodes interpolator with the same preset layout.

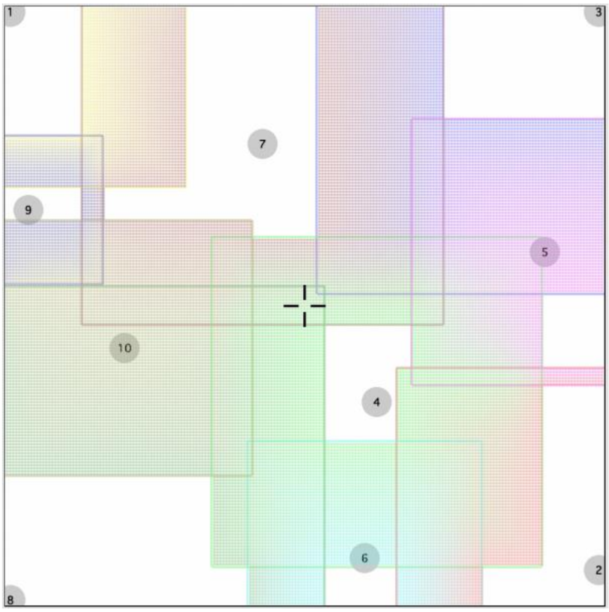
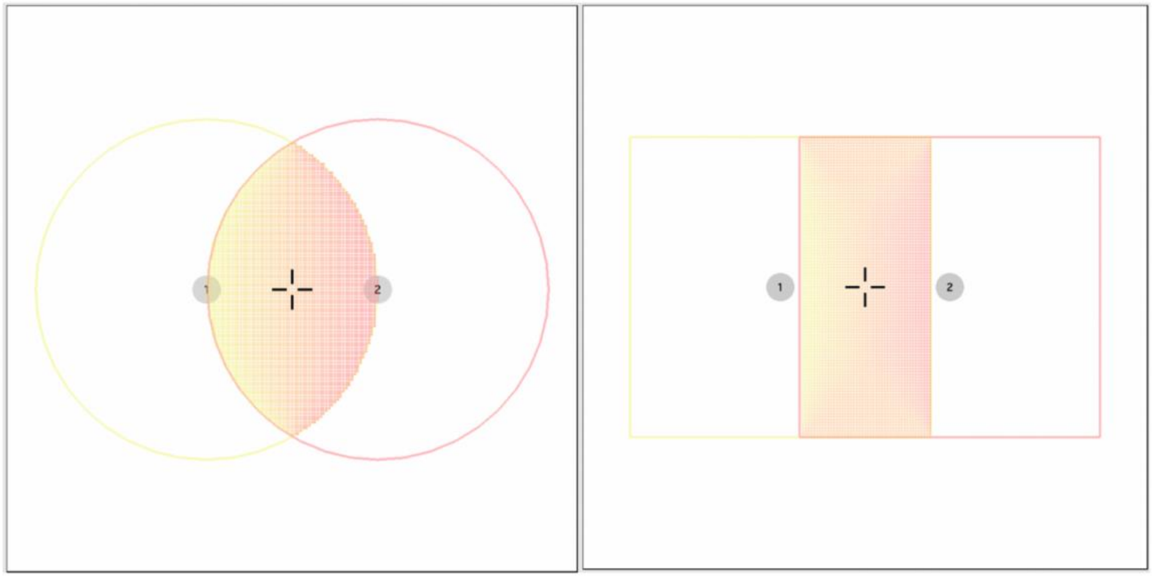


Figure 33 Square Nodes Model with the Ten Preset Test Layout and Interpolated Colour Shading

This can be compared to Figure 27, where the same is shown for the original nodes interpolator. As can be seen, although the shape of the intersecting regions is different, the colour interpolation appears to be fairly close. However, on closer inspection where the colour interpolation was directly compared there are some areas where unique colour shades are produced that are not present in the other. Again, this will result in generation of different weightings and the potential of generating different sonic outputs. To further understand the implications of using squares for the nodes as opposed to circles some simple comparisons were made for the same positioning of the nodes. Figure 34 shows a simple comparison between just two intersecting nodes with the two different models.



(a) (b)

Figure 34 Comparison of Intersections Between Circular (a) & Square (b) Nodes

From this it can be seen that with the circular nodes, because it is based on a radius function, an even shading is produced across a particular arc. However, for the square nodes as the distance to the node centre is not constant, the diagonal has a greater distance to the square axes, and as a result in the corners of the intersection take on the opposite colour. For example, in Figure 34(b) the intersection along the edge closest to node one is yellow representing node one's larger weighting but in the very corners an orange shade is produced as node two has an increasing impact on the weightings.

To further understand the difference between the intersecting regions of the original nodes and square nodes the same experiment as shown in Figure 29 was repeated for the square interpolator. That is, two identically sized nodes are moved to create different sized intersections. The results from this are shown in Figure 35 for all the same node positions as used in Figure 29. As can be seen, the results are very similar to those generated with the original nodes where the same shading is generated for all intersection sizes. Again, this is because the two nodes have the same influence so as they are moved across each other the size of the intersection increases, but the weightings remain the same, although over a larger area.

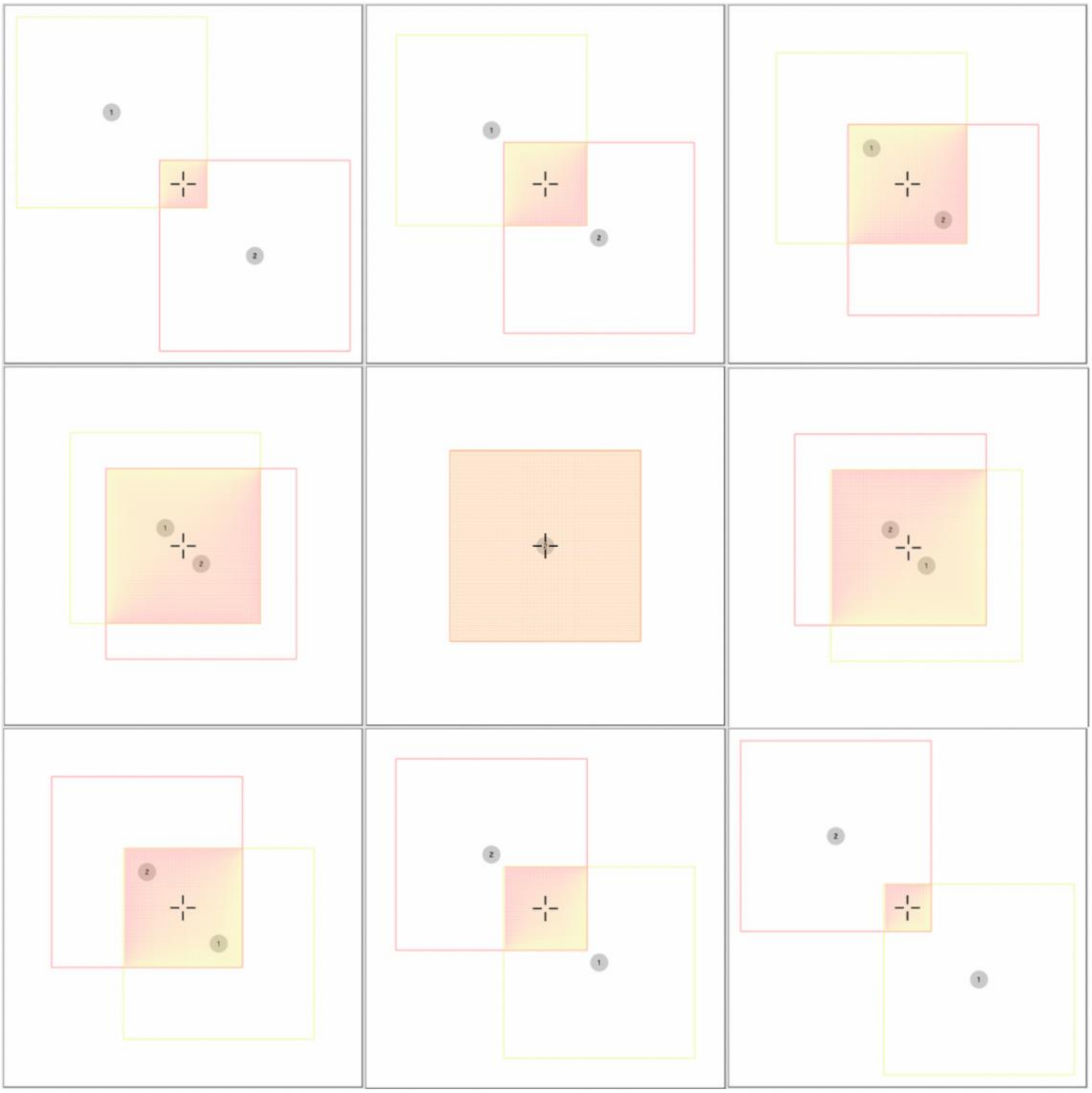


Figure 35 Interpolated Colour to Show Different Sized Intersections Between Two Square Nodes of the Same Size

The experiment shown in Figure 30 with intersections between two different sized nodes was also repeated with the square nodes. The results are shown in Figure 36 for all the same node positions as used in Figure 30. The results between the two are very similar with the weightings being different along the centre-line axis between the two nodes. However, as seen with the node intersection comparison shown in Figure 34, there is a bias in the node corners, but this time in the opposite corners to the centre-line axis, whereas for the circular nodes the radial effect smooths the results being seen and heard in the corners.

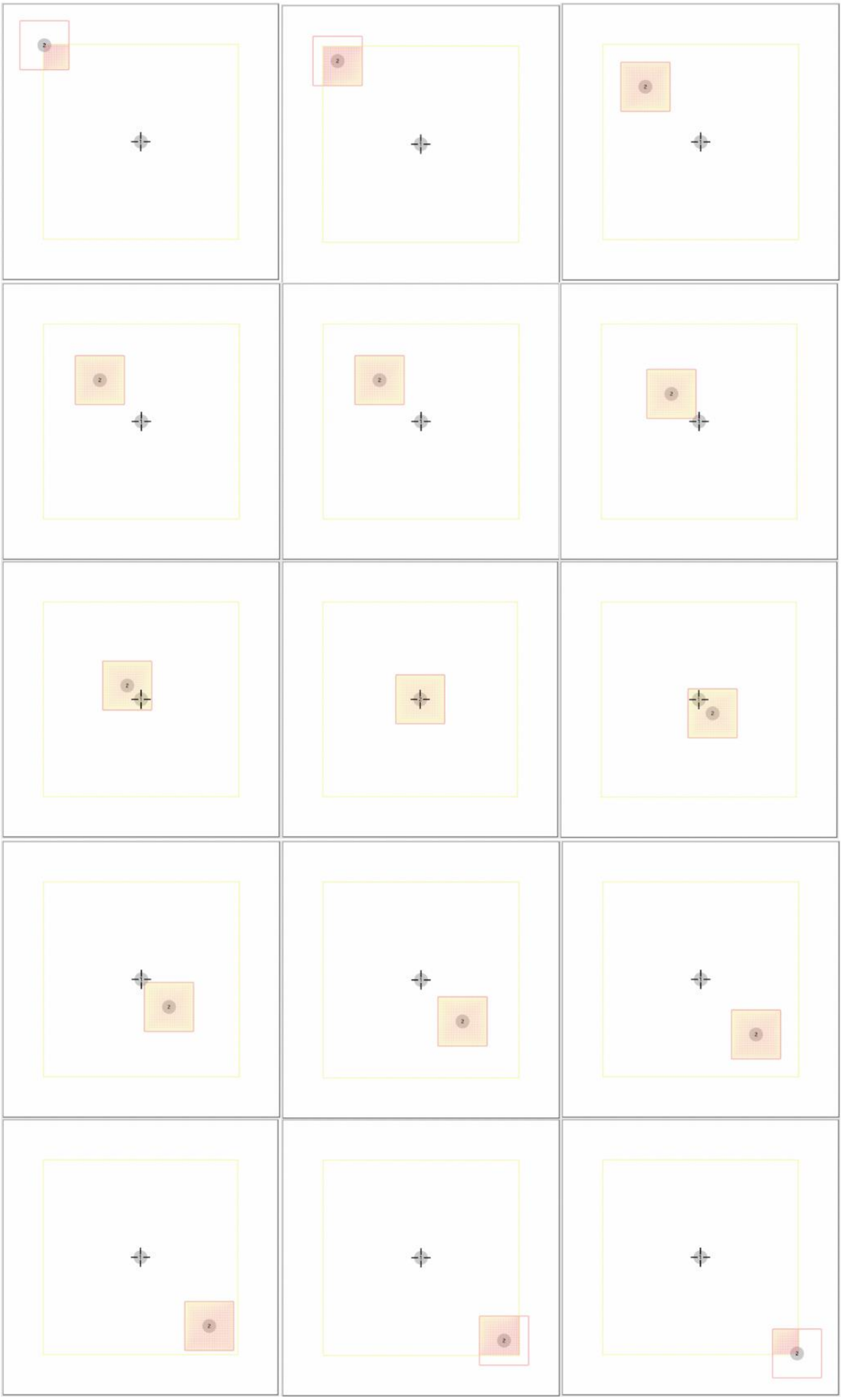


Figure 36 Interpolated Colour to Show Different Sized Intersections Between Two Square Nodes of Different Size

5.4.2 Alternative Nodes Geometries: Rectangular

From the square nodes interpolator, a further modification was made to create a rectangular node interpolator. As with the construction of the square nodes it was desirable to ensure that the nodes maintain the same node area so that the nodes occupy the same area within the interpolation space. This was intended so comparisons would be made between the interpolators. However, with the rectangular nodes it was important to consider the ratio between the sides. As a starting point, a 3:4, height to width ratio was chosen for ease of area conversion from the circular nodes implementation. Figure 37 shows the rectangular nodes interpolator with the same test layout of ten nodes as used in the original and square nodes.

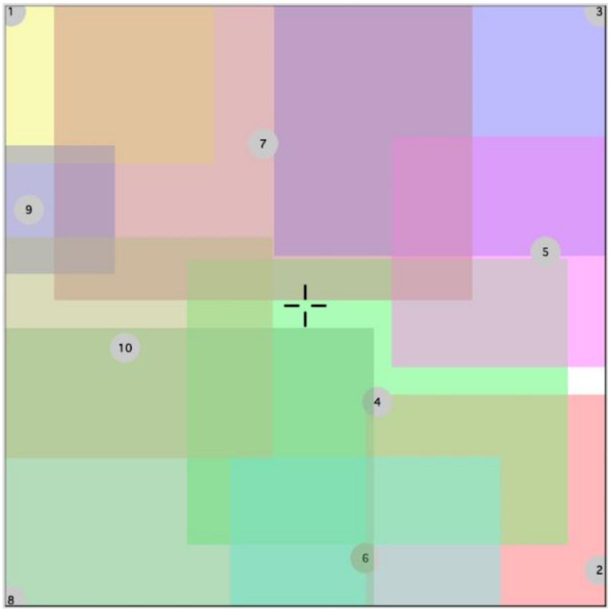


Figure 37 Rectangular Nodes with Test Layout of Nodes

The change of node shape results in the interpolatable space being different again with the interpolation cursor position shown now not being in a region of intersection. To be able to fully compare the interpolation space for the rectangular nodes with that of the other node versions the space is shown as the shaded area in Figure 38. By comparing this with the layouts for the other node interpolators shown in Figure 25 and Figure 32, it can be seen that the resulting interpolation space is different again.

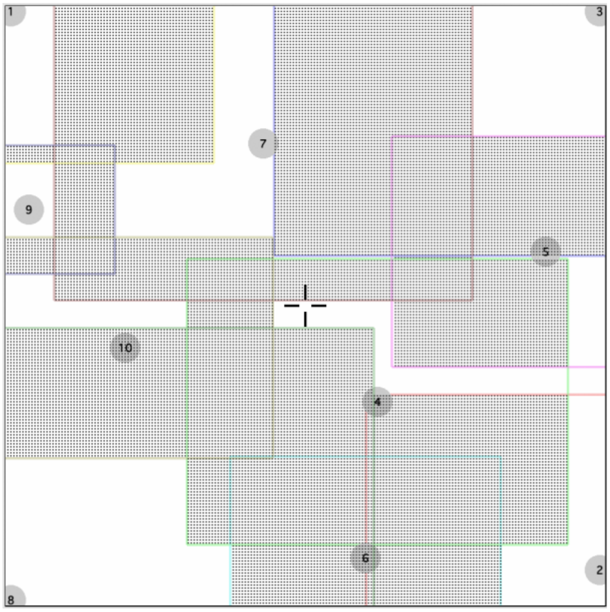


Figure 38 Rectangular Node Interpolator with Test Layout - Interpolatable Space shown by Shading

Similarly, by viewing this layout with the colour interpolation provided between the nodes, shown in Figure 38, it can be seen that this interpolator provides some areas of unique colour which shows that it provides a different sonic palette to either the circular or square node models. Albeit that the differences between the rectangular and square nodes interpolators are less obvious than those that separate the rectangular and circular nodes examples. Figure 39 shows the same layout of the rectangular interpolator with colour interpolation.

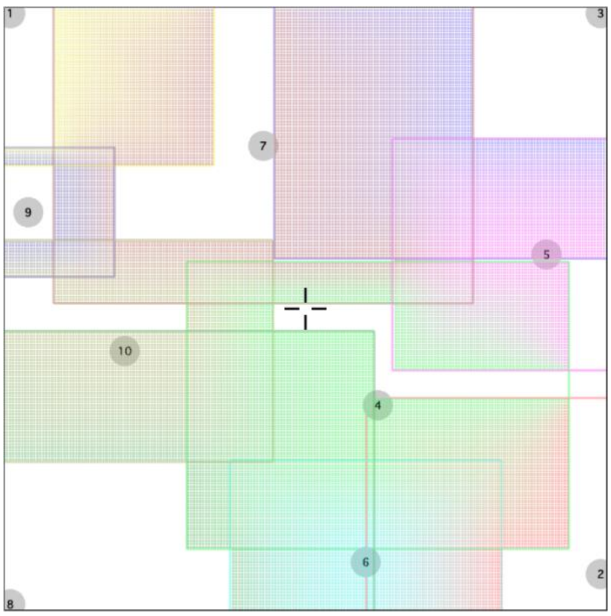


Figure 39 Rectangular Nodes Model with Ten Preset Test Layout and Interpolated Colour Shading

5.4.3 Alternative Nodes Geometries: Triangular

Having seen in the exploration and evaluation of the nodes interpolator, detailed in the previous sections, the interpolation is dependent on the shapes used in the intersecting model. To corroborate this observation a triangular nodes interpolator was created. This was based on equilateral triangles and for comparability, the triangles were sized to occupy the same area as the shapes in the other node implementations. Figure 40 shows the intersecting regions for the triangular interpolator with the same layout as used with the others. Again, it can be seen that the interpolatable space is different to the other interpolators, with a higher number of unique intersections created. For example, with the triangular interpolator there is a region where nodes 2, 3 and 5 intersect that is not produced with any of the other interpolators.

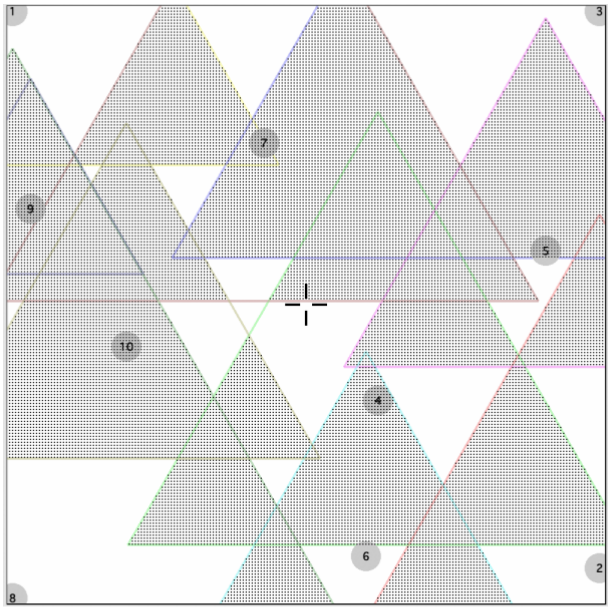


Figure 40 Triangular Node Interpolator with Test Layout - Interpolatable Space shown by Shading

Again, by using colour interpolation it is possible to get a visual representation of the interpolation weightings from this interpolator. This is shown in Figure 41. As can be seen, the intersecting regions that are unique to this layout result in new colour permutations. These can also be heard when using the interpolator to generate synthesized sounds.

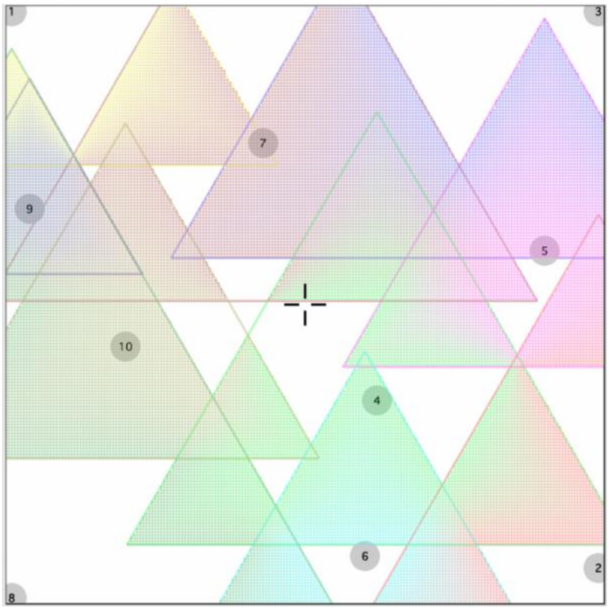


Figure 41 Triangular Nodes Model with Ten Preset Test Layout and Interpolated Colour Shading

5.4.4 Alternative Nodes Geometries: Multi-Shape

Having seen by the exploration and evaluation of the nodes interpolator, detailed in the previous sections, it is evident that with the intersecting model the shapes used in the space have an impact on the size and shape of the interpolation space and the output that each interpolator provides. Given that the interpolators usually allow the user to define the layout of the interpolation space a similar strategy could be taken with the choice of shape for each node. In this way, the user could have complete control over the design of the interpolation space. To this end, a multi-shape nodes implementation was generated that allowed the users to select the shape for each node from the choices: circle, square, rectangle and triangle. An example is shown in Figure 42 for the same layout of the nodes as before, but the shape selection cycles through the four choices available. Here the shading shows the regions of interest, where the interpolation will be performed in the intersectional areas of the nodes. By changing even one shape it will result in changes to these regions offering the user even finer control for the layout of the interpolation space.

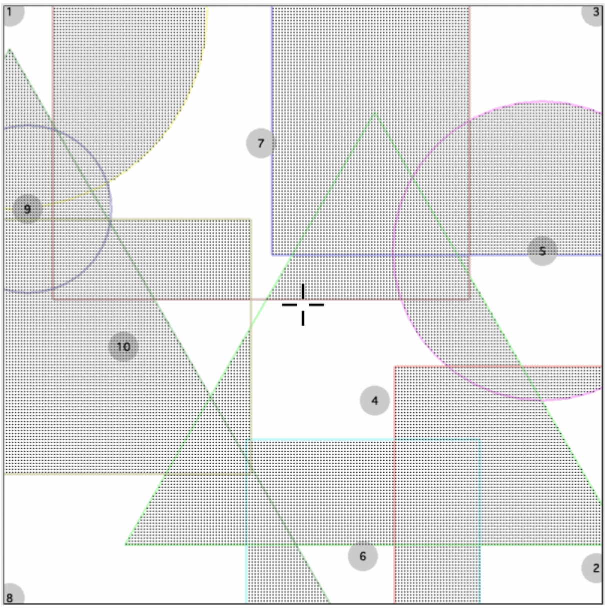


Figure 42 Multi-Shape Node Interpolator with Test Layout - Interpolatable Space shown by Shading

Once more, by using a colour interpolation it is possible to see that as a result of the shape changes new colours are generated where this interpolation will provide unique sonic outputs. This is shown below in Figure 43.

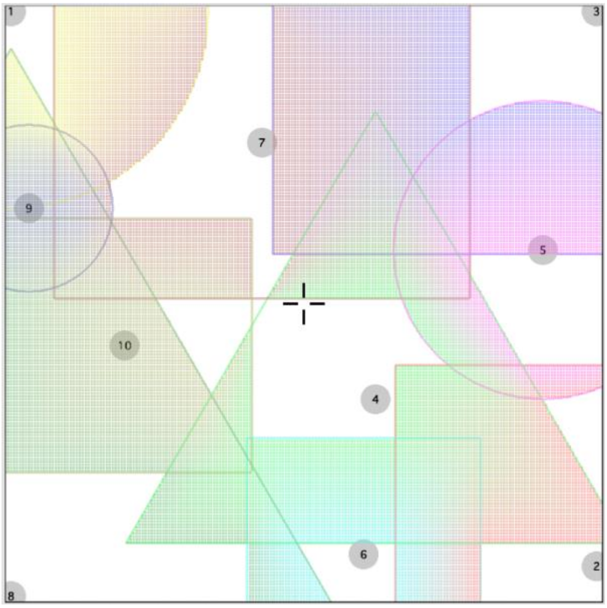


Figure 43 Multi-Shape Node Model with Ten Preset Test Layout and Interpolated Colour Shading

Moreover, changing shapes within the choices available will equally change the interpolation space and hence the outputs that are possible to achieve with the interpolator. For example, changing the shapes of two nodes in this layout makes subtle changes to the intersecting areas around them. Figure 44 shows this where the shape of node 9 is changed from circle to a square and node 10 is changed from a square to a rectangle.

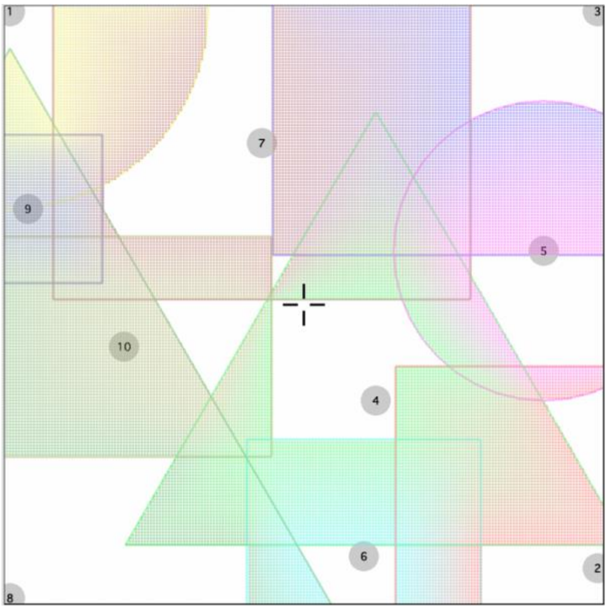


Figure 44 Multi-Shape Node Model with same Test Layout and Shape Change for Nodes 9 & 10

As can be seen, new colour shades and gradients are generated which are representative of where the interpolator will have new sonic outputs. Providing the users with a mechanism to be able to change the shape of individual nodes refines the interpolation space. In this way, the user has full control over the position, size and shape of all the nodes offering more customisation options for the layout of the space.

This idea could be further expanded by letting the user have more control over the selection of the shape. For example, the rectangles are limited to 3:4, but the model could be expanded to allow any different ratio to be selected. Equally the triangles are currently restricted to equilateral, but this could be modified to include all the different types of triangles, by allowing the user to select the lengths and angles for the triangle. Moreover, the model could be expanded still further by allowing the user to select different regular and irregular polygons.

Since the initial development of the framework, a research team at University of Mons developed an interpolator using this strategy [199], [200]. As detailed in Section 2.4.2.14, the polygon interpolator expands on the intersecting model explored with the framework. This provides the user with complete flexibility when designing the layout of an interpolation space to be used with the intersecting model, through allowing the user the ability to be able to choose the exact shapes to be applied. The only limitation with the present version of the Mons interpolator is that it does not currently allow the use of non-polygon shapes, such as ellipses and circles, but this may be expanded in future versions.

5.4.5 Nodes Implementation Conclusions

From the exploration of the nodes interpolation model it was observed that the topology of the interpolation space is directly impacted by the position, range or “field-of-influence” and shape of each preset. This is true even when the allocated preset sounds remain the same. Changes to any one of these, alters the regions-of interest and will result in the interpolator generating different sonic outputs. The original nodes object allowed the arrangement of presets to be changed and their fields-of-influence, but did not allow alternative shapes to be utilised. This work, and the parallel work undertaken at University of Mons [199], [200], have both shown there are benefits to providing users with the additional control of being able to change the interpolation area, via selection of different shapes.

This work has also demonstrated that the visual representation of an interpolator’s interface can directly impact on the systems performance and the sound design possibilities. Moreover, these could potentially affect the systems usability and suitability for particular applications. Given that many of the different models presented in Section 2.4.2, used geometric alternatives, these may affect their performance and will require investigation to establish their influence and whether they alter the systems usability.

Finally, through this proof-of-concept it has been shown that the graphical interpolator framework offers a mechanism that facilitates the rapid prototyping of alternative interpolators. Here this has been illustrated for alternative visual representations, but in wider informal testing undertaken by this author, it has been used to rapidly prototype different realisations for the other four areas of the framework (control inputs, interpolation function, parameter mappings and synthesis engine). Therefore, the framework will continue to be used to develop other interpolation models.

5.5 Expanded Interpolation Models

To explore other interpolation models and to allow back-to-back evaluation to be performed between them, it was decided that a number of the different interpolation models required implementation. In the first development cycle six interpolators were built, integrated with the framework and functionally tested. These were chosen to represent the main interpolation types with desirable characteristics for sound design identified in Section 2.4.2 and all offer uniquely different visual representations:

1. Nodes (Overlapping Circles) [183]
2. Gravitational (Planets & Space) [163]
3. Radius-based IDW (Scattered Points & Interpolation Cursor Circle) [166]

4. Light (Lamps) [177]
5. Delaunay (Triangulation) [186]
6. Voronoi Tessellation (Polygons) [123]

The original visual displays for each of the chosen interpolators is shown in Figure 45.

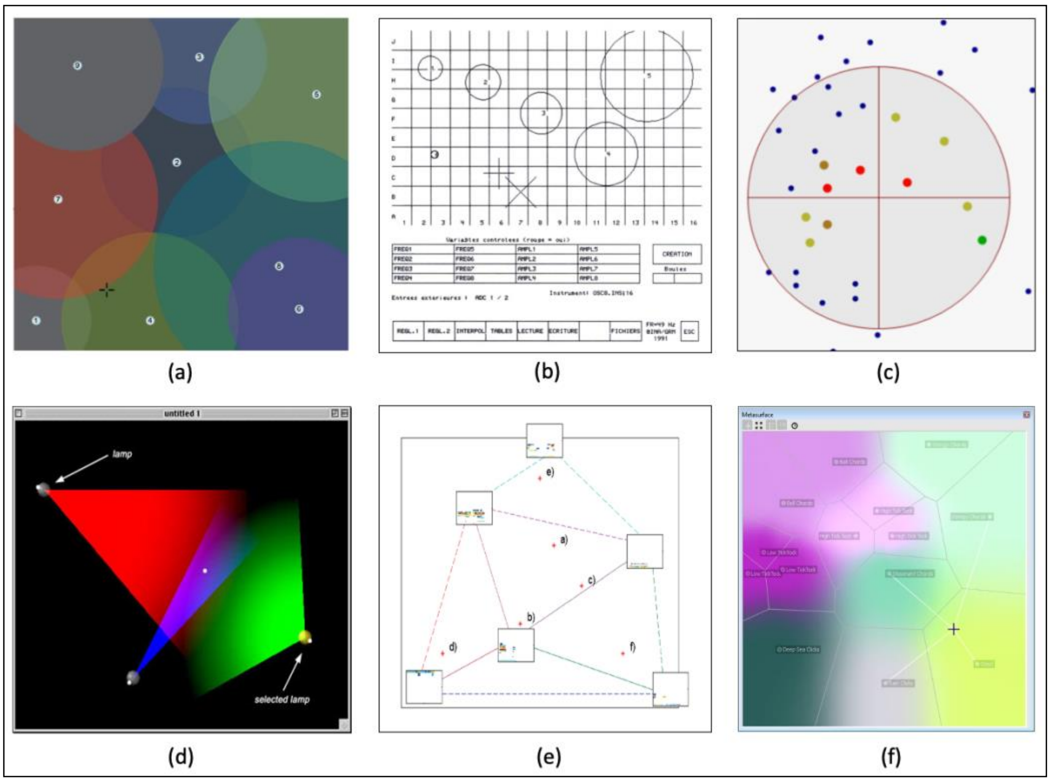


Figure 45 Original Interpolator Visuals (a) Nodes, (b) SYTRE, (c) ArcGIS, (d) Interpolator, (e) Spike-Guided Delaunay Triangulation and (f) Metasurface

The nodes interpolator reimplementation, detailed earlier, was chosen not only to act as a benchmark for the other interpolators, but also enabling the visual representation to be changed to assess the influence of different visualisations using the same interpolation model. The other interpolators were chosen to represent the key traits of the interpolation systems that have been previously created. However, as these systems were created by different people, at different times, using different technologies and for different applications there will inevitably be some disparity between the originals and the reimplementation's. These only exist in the visual models and representations as the use of the framework allowed all other aspects to be kept the same so that only their impact could be measured in the testing. The exact difference between the visuals in the originals and their reimplementation's are detailed in the following section.

5.5.1 Interpolator Reimplementations

Nodes was the first interpolator constructed and with the original freely available for comparison, this one has a virtually identical visual representation and functionality. An example of the visual display for the nodes reimplementation is shown in Figure 23. All of the remaining implementations are based on the same premise of control, so each uses a crosshair marker to show the position of the interpolation point cursor and each preset's position is indicated with a circular handle that can also be used to move the presets position. In addition, as nodes offer the ability to change a sound through the real-time editing of preset locations this is also included in all of the subsequent implementations.

For the *gravitational* model the same colour scheme was adopted as used in the nodes interpolator. However, the transparency element was removed to help the user understand that when the interpolation point is positioned on the planet's surface the corresponding preset will be recalled. Like the nodes implementation, each of the planets was given a central handle to aid the movement and positioning of the planets within the interpolation space. The handle, when combined with a hotkey, also allows the resizing of each planet, matching the functionality found in nodes. An example of the visuals for the gravitational model reimplementation is shown in Figure 46.

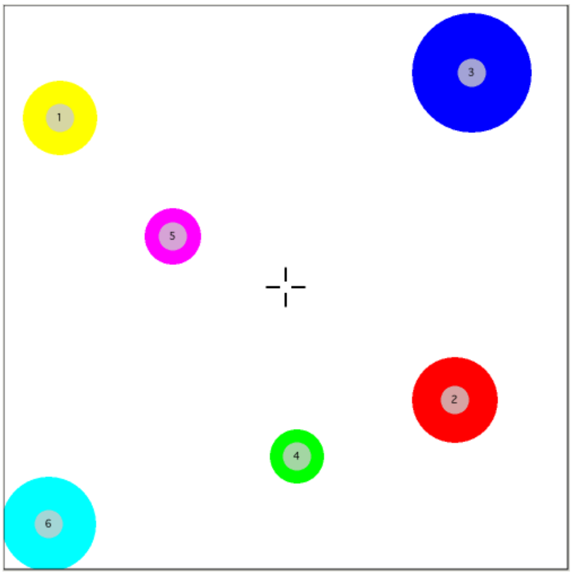


Figure 46 Visuals from the Gravitational Model Reimplementation

For *radius-based IDW* the interpolation radius is shown as a shaded circle that is centred at the interpolation cursor. The containing circle defines which presets are included in the interpolation and can be resized with a hotkey. Therefore, as the cursor is moved so the circle is moved and provides visual feedback to the user of which presets are included in the current interpolation output. Also, the position of each preset is shown by a handle that has the same functionality as the other interpolators, but has been colour coded to help distinguish which handle corresponds to which preset. An example of the visualisation for the Radius-based IDW is shown in Figure 47.

In this example it can be seen clearly that presets 2, 4 & 5 will be included in the current interpolation output.

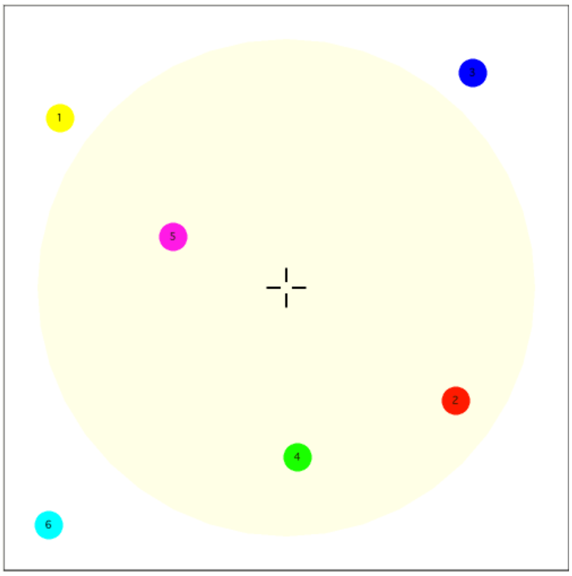


Figure 47 Visual Representation for Radius-Based IDW

The functionality for the *light* model has been kept the same as the original, but each lamp has a different colour and the angle, aperture and extent of each can be changed with the use of hotkeys. As with the original, the intensity of the light beam changes with the distance from the lamp, but the model does not include any diffraction or diffusion of the light at the beam edge. The resulting hard edges of the beams can create discontinuities in the interpolation. The original model also allowed a preset to be assigned to the background so that the interpolation was not constrained to just the areas of the light beam intersection and could be performed across the whole of the interpolation space area [177]. This feature was not included as it was not available in any of the other interpolators and it was only the model's visualisations that were of interest. Figure 48 shows an example of the light model's visual interface.

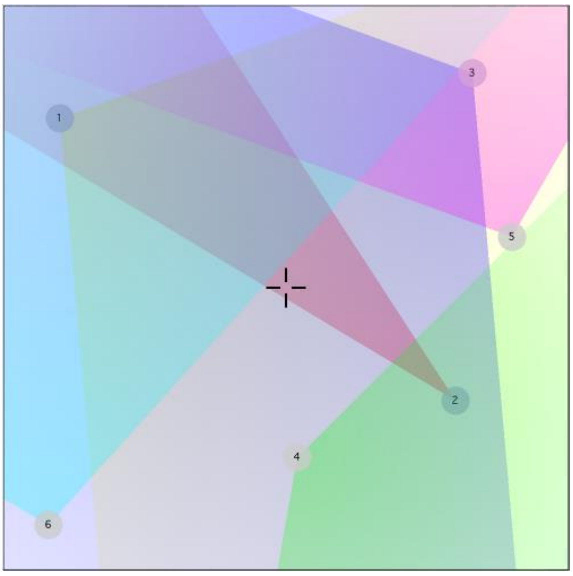


Figure 48 Visual Display for the Light Model

For the *triangulation* model a Delaunay triangulation mesh was drawn with the vertices being the defined positions of the presets [229]. This mesh explicitly shows the containing triangles that will be used in the interpolation at any point within the space. This can be seen in the example shown in Figure 49. In one of the original versions of this interpolator model, at the preset locations a visual representation of the interpolator's sonic output was provided in the form of a Spike plot, which provides an indication of frequency content, with respect to time [186]. These can be difficult to interpret and as it is not a feature offered in any other interpolator it has not been implemented here. However, as with the radius-based IDW reimplementation, the preset handles have been colour coded to correspond to the presets. To provide the user with additional visual cues for this interpolator the interpolation weightings are indicated by colour coding the barycentric coordinates used for calculating the weightings to match the corresponding patches. In this way, the relative areas between the three triangles of the barycentric coordinates provides an indication of the weightings [189].   This can be seen in Figure 49, where for the shown cursor position it can be seen that preset 5 has the biggest weighting, then preset 2 and finally preset 3 only has a small contribution.

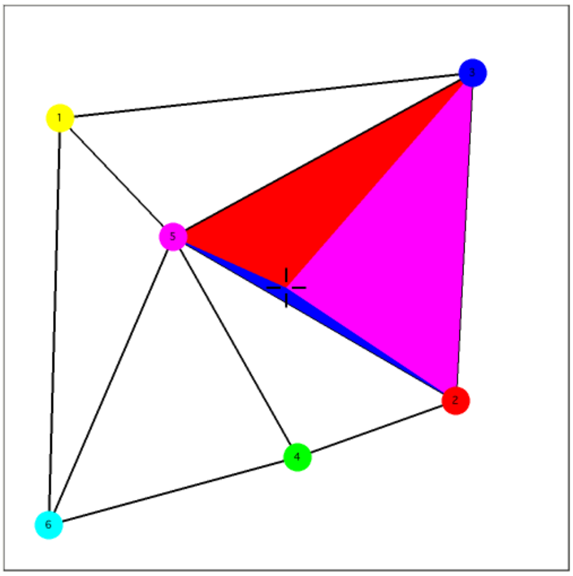


Figure 49 Visual Display for the Delaunay Triangulation

For the final interpolator a *Voronoi Tessellation* is used to give a visual representation of each preset's natural neighbours. Here the extent of each preset is shown as an irregular polygon and any point within the polygon is closer to its centre (the preset handle) than any of the neighbouring presets [123]. Each of the polygons is colour coded to visually represent the different presets as shown in Figure 50.

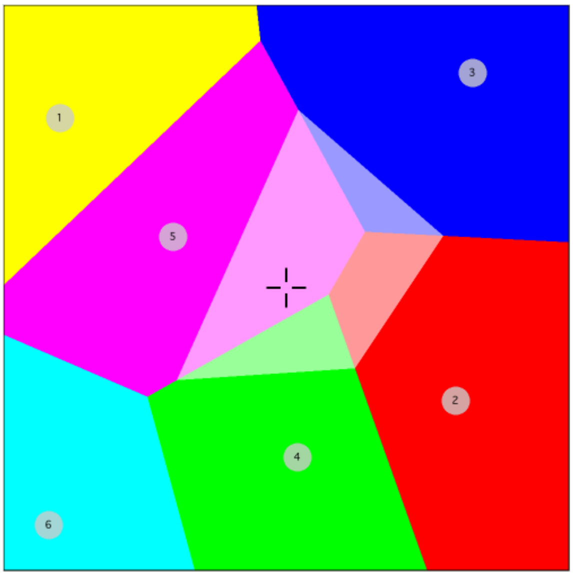


Figure 50 Visual Display for the Voronoi Tessellation

In the original implementation the polygon colours are shaded to give a visual cue of the interpolator's output for the corresponding cursor position. However, this is a computationally intensive calculation to perform real-time. In the original this is done via an edit layout mode where the presets' positions are defined and the shading is precalculated. Then afterwards it can be put into an interpolation mode to actually generate a sonic output. However, as no other interpolator offers shading and the goal was to allow real-time editing of preset locations on all implementations, it was decided not to include this feature from the original. Similarly, the presets included in the current interpolation output were originally indicated with the use of visual guidelines that connected the interpolation cursor and the included preset handles. Here they have been replaced by the inclusion of a “ghost” polygon centred at the cursor position. This polygon shows the area that would be “stolen” from each neighbour if a new polygon were inserted at this position. The cursor’s ghost polygon not only shows the presets included in the current interpolation output, but their relative weightings from the areas stolen from each. Therefore, the reimplementation offers the same visual cues as the original. The ghost polygon is shown in Figure 50 as the semi-transparent white polygon centred at the cursor position.

5.5.2 Graphical Interpolator Comparison

Following functional testing the different interpolators were back-to-back tested by placing the same ten presets at identical locations in each. The nodes interpolator was populated first with the same layout that was used to evaluate the nodes implementation with different shapes, detailed in this Chapter. In this layout, although the size of the individual nodes were selected randomly, they were chosen to ensure the whole space was covered so there was no non-interpolatable area within the space. For the gravitational interpolator, while the same locations were used, this model requires space between the planets, where the interpolation is performed. However, so that each preset has the same relative influence as they do in the node interpolator, the sizes were scaled by one tenth of those in the nodes interpolator. For the radius-based IDW the interpolation point’s radius was chosen to cover approximately 50% of the interpolation space so for all interpolation positions, multiple presets are enclosed by the radius. For the light interpolator although the same locations were used, as each lamp has an angle and aperture, it results in each lamp having a specific directionality. To try to give coverage over the whole interpolation space the extent of each lamp was scaled to four times the nodes size. Despite this each lamp’s angular directionality also needed to be selectively chosen to ensure the whole interpolation space was covered, whilst still giving a good spread of intersecting light beams. For the two remaining interpolators the presets do not have different influences or directionalities, so the locations were kept the same as the nodes layout. The test layouts for the six interpolators are shown in Figure 51.

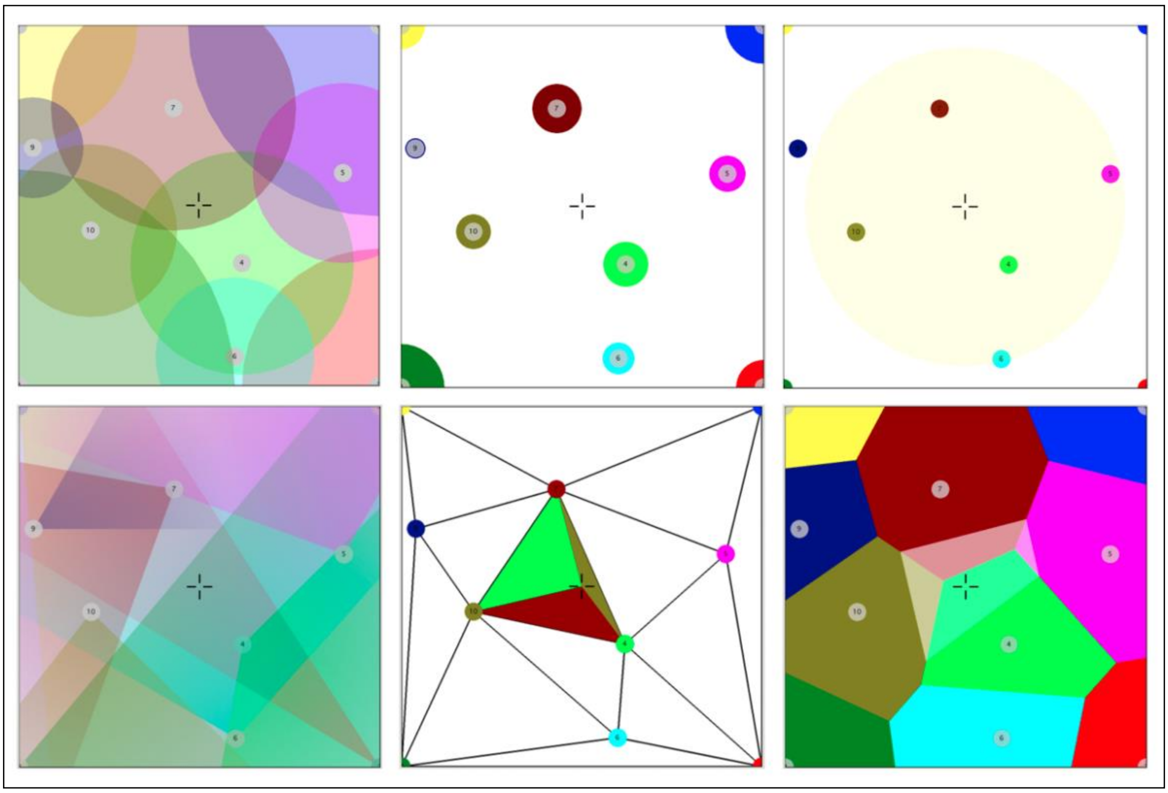


Figure 51 Test Layout for Reimplemented Graphical Interpolators

These layouts were used to perform back-to-back tests where outputs from the different graphical interpolators were compared. For the tests the control inputs, interpolation function, parameter mappings and synthesis output, all remain the same, as detailed:

1. Control Inputs – Fixed 2-D movement of interpolation point only
2. Interpolation Function – Linear interpolation
3. Parameter Mappings – All synthesis parameters mapped to the corresponding preset location
4. Synthesis Output – Native Instruments Massive with ten presets loaded

This ensured that any differences between the different interpolators were purely from the use of different visual interpolation models and not from other factors. The tests compared the sonic output from the different interpolation models for the same interpolation positions. This was first done by instantaneously moving the interpolation cursor to ten different locations and comparing the sound generated with each system.[7] From these tests, it was evident that each visual interpolator generated significantly different sonic results, despite being populated with the same preset sounds. To try and get a better understanding of each system's sonic nature, another

[7] Comparative example of the sonic output generated from ten identical locations with the different interpolation models: https://youtu.be/KiT2wXujrv4

comparative test was created, where the interpolation point was moved through a fixed trajectory path around the defined interpolation spaces. The path began at the centre of the space, moved diagonally towards the left-top corner until the mid-point and then moved around parallel to the outside edge of the space.[8] It was found that each interpolator gives a very different range of sonic outputs across all interpolation positions. The fact they were different was not necessarily surprising, but the diversity of the sonic differences was not anticipated. Moreover, each interpolator results in a completely different sonic palette that it can generate, meaning it is very difficult to create the same sound with each interpolator. This is because each interpolation model results in different preset weightings for the interpolation function. An example is shown in Figure 52 which shows preset weightings for just the centre position (initial position in the comparative tests) of each interpolation space, as shown in Figure 51.

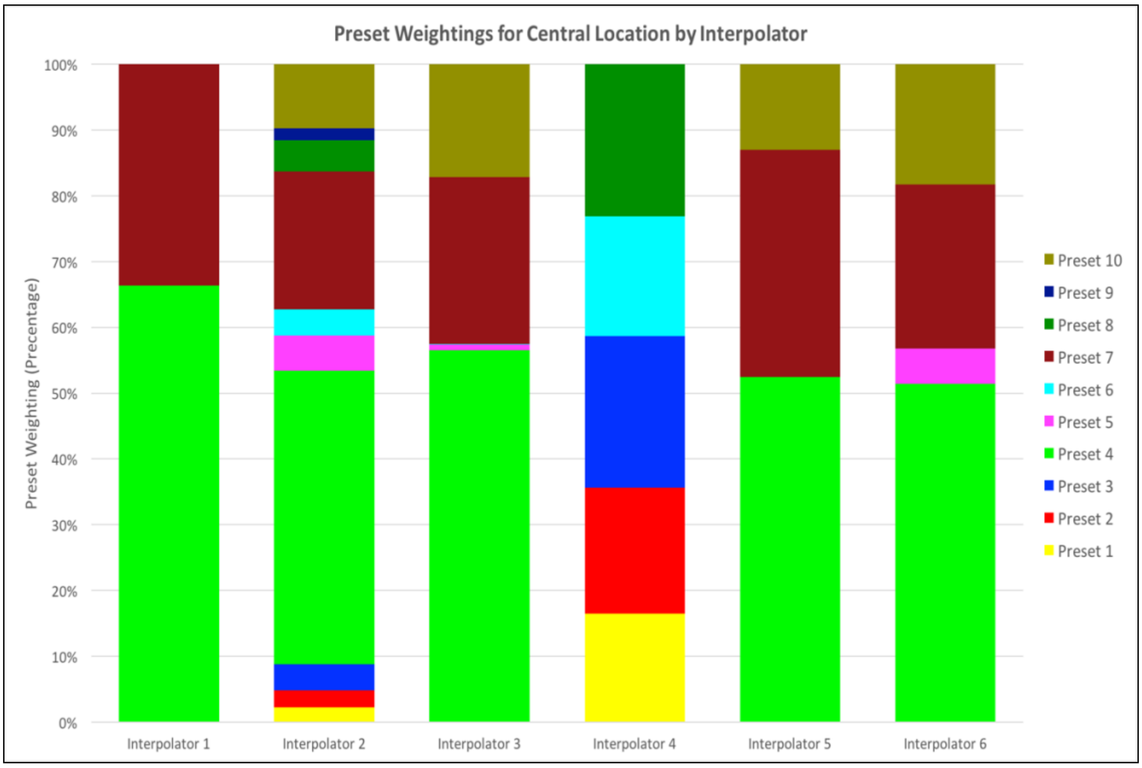


Figure 52 Comparison of Interpolator Preset Weighting's for the Centre Location[9]

As can be seen, each interpolator generates very different weightings for the same positions and as these determine the parameter values there is a significant impact on the sounds generated. Comparing the weightings for another position from the comparative test (position 2), where the

[8] Comparative example of the sonic output generated from the different interpolation models for identical traces through each space: https://youtu.be/E_l1XdX-E80

[9] Example of the sonic output generated at centre location for the different interpolation models: https://youtu.be/KiT2wXujrv4?t=7

cursor was positioned in the top left corner of the space resulted in the weightings shown in Figure 53.

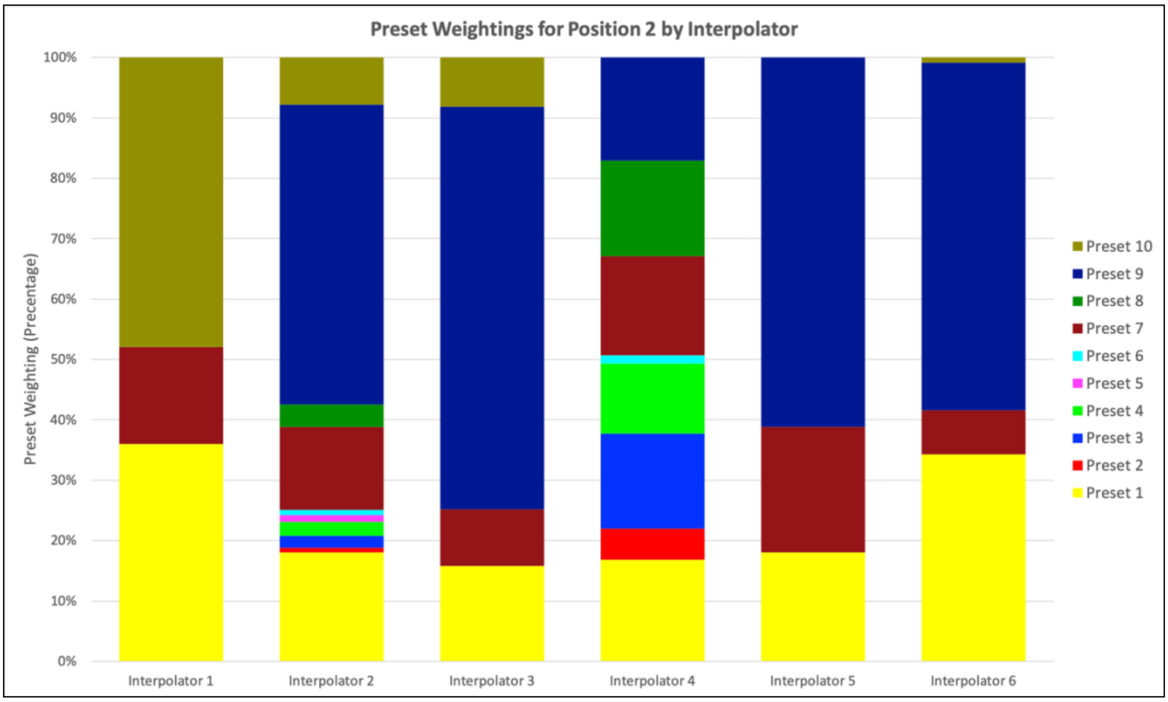


Figure 53 Comparison of Interpolator Preset Weighting's for Position 2[10]

Again, it is evident that each interpolator will give different sonic outputs for the same position even when populated with the same presets because each model generates different weightings. For completeness the weightings are shown for another location within the comparative tests. This time the weightings are compared for a location in the bottom right quadrant (position 5) and the results can be seen in Figure 54.

[10] Example of the sonic output generated at Position 2 for the different interpolation models: https://youtu.be/KiT2wXujrv4?t=196

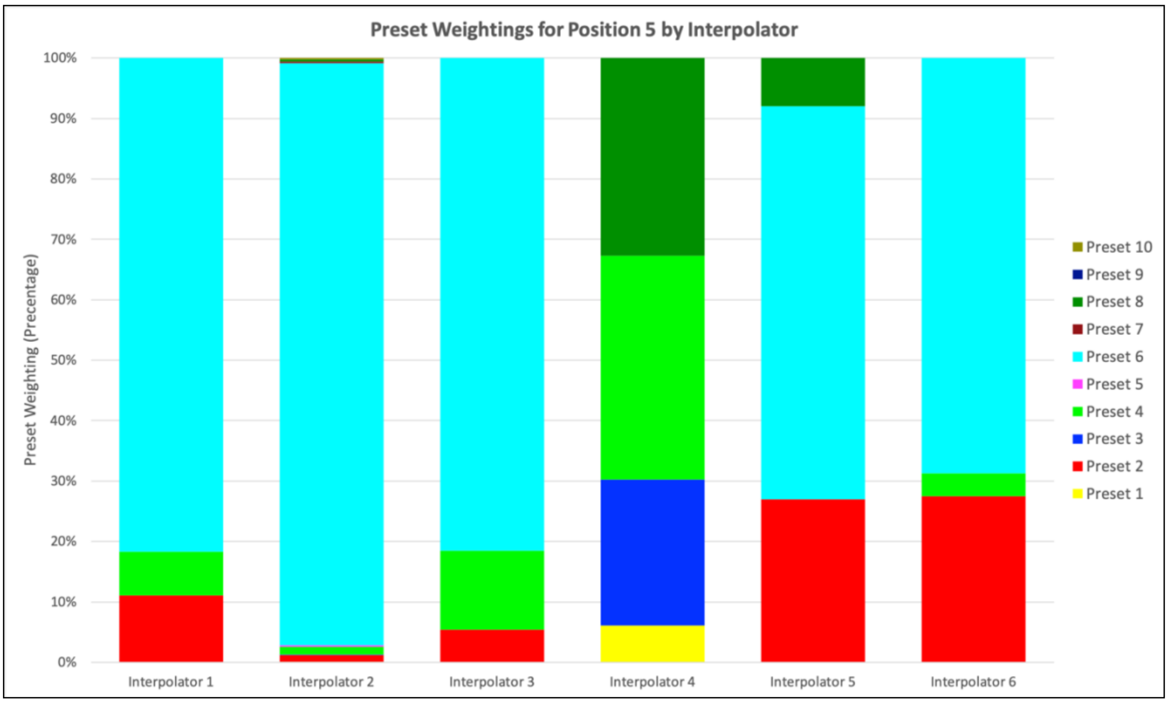


Figure 54 Comparison of Interpolator Preset Weighting's for Position 5[11]

Similarly, there is a wide variation between the weightings that the interpolators produce and hence the sonic outputs that it is possible to generate with each model. The trend was largely seen across the whole interpolation space. Colour interpolation was used to give an overall impression of the difference between the interpolation spaces created by each model. The results of this can be seen for each interpolator with the same layout of presets in Figure 55. As can be seen, each interpolator provides a different distribution of possible outputs that can be generated. In addition, each interpolator has a unique sonic palette as each one has regions where distinctive colour shades are generated and this will result in the generation of unique sounds. It should also be noted that in this representation the white areas signify regions where there is no interpolation performed and the sound output will not change when the cursor is positioned in these areas.

[11] Example of the sonic output generated at Position 5 for the different interpolation models: https://youtu.be/KiT2wXujrv4?t=466

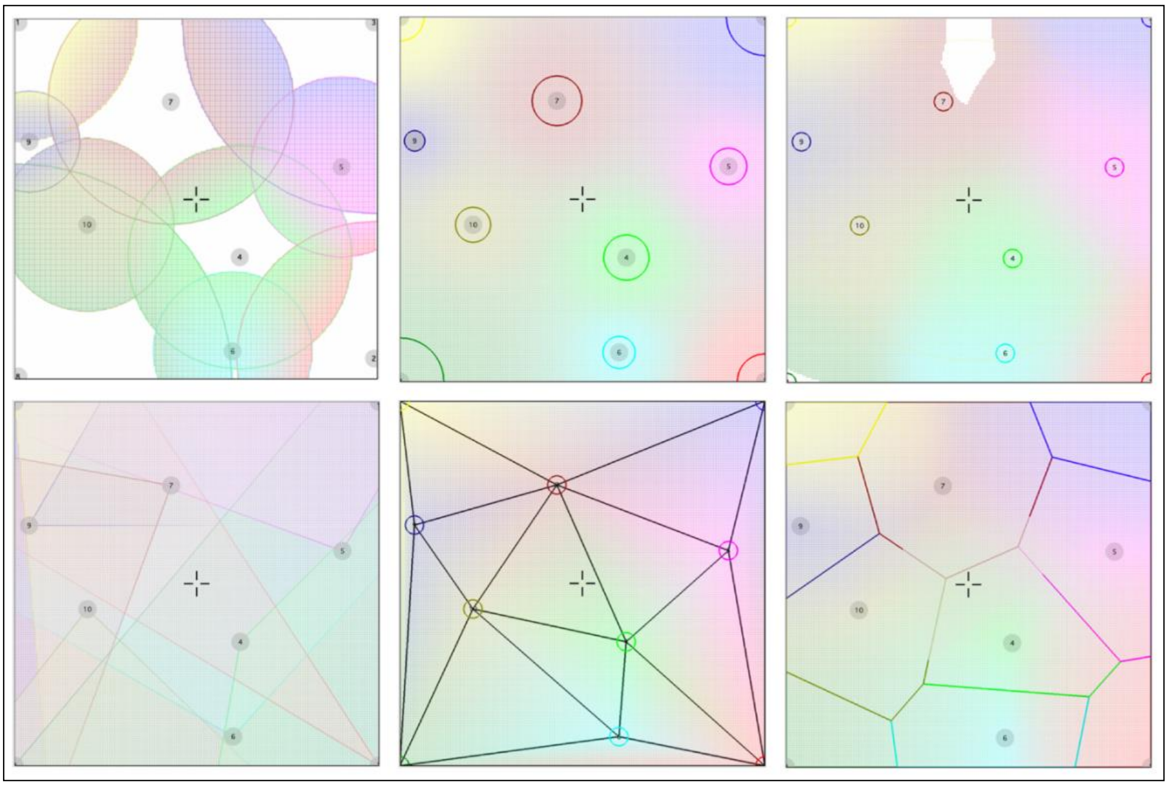


Figure 55 Test Layout for Graphical Interpolators with Colour Interpolation of the Space

In all cases, the relative positioning (layout) of the presets determines the interpolated outputs. Different layouts of the same presets result in different outputs being obtained. It was also noted that for interpolators 1, 2 & 4 the extent (size/range) of each preset, further changes the interpolation space and has an impact on the preset weightings. Also, the directionality of the lamps in interpolator 4 gives an added element for further modifying the interpolation space. For example, in Figure 56, changing the directions of just two lamps within the original layout (Figure 51) creates very different weightings compared to the original (Figure 52) even though all the locations and ranges remain the same. In this example, the angle of lamp 2, present in all output weightings in the original layout (Figure 52), was increased by 30 degrees and the angle of lamp 6, in the original layout only present in the output weighting for Interpolator 2 (Figure 52), was reduced by 20 degrees. As a result of these relatively small adjustments to the interpolation space, the weightings change substantially, with preset 2 dropping to zero and preset 6 being included with a relative weighting of 32%.

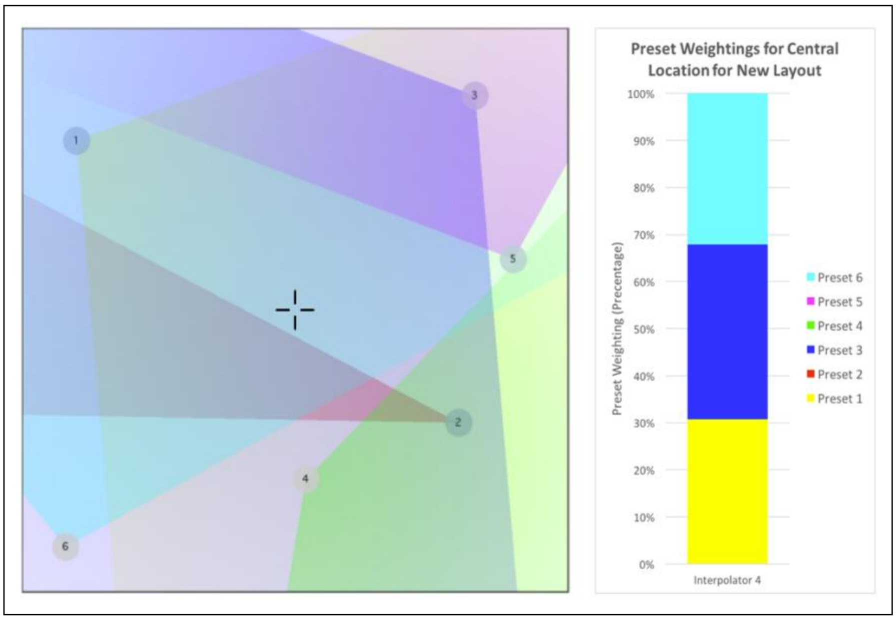


Figure 56 Comparison of Interpolator Preset Weighting's for Identical Presets and Cursor Positions

For interpolators 3, 5 & 6 the influence of each preset is potentially identical, but the layout determines the relative strengths. For interpolator 3 this is constrained by the interpolation point's radius which determines which presets will be included. If the radius size is changed, corresponding presets will be added or removed from the interpolated output. Whereas interpolator 5 uses only the three closest presets and interpolator 6 uses the natural neighbours.

5.6 Interpolator Exploration Conclusions

From this assessment it is evident that the alternative geometric models of the different interpolators have a significant impact on the sonic pallet generated. This is independent of the presets used, their layout within the space and chosen fields-of-influence. In essence, the interpolator model is also having a large effect on the topological surface and as such changing the sonic potential of the interpolation system. From this there are two areas of interest: First, given the variety of visual models and the alternative representations that these could take, do the visuals actually aid the exploration of the space or could it be explored blind (without visuals) and still achieve the same results; Second, of the alternative models do any of them provide a better platform for sound design than others and is it possible to identify desirable characteristics that maybe provided. The first will be explored in an initial investigation that will attempt to answer this question and will also act as a pilot study to establish a procedure for undertaking usability testing of graphical interpolators (Chapter 6). A wider study will then be undertaken that will evaluate the usability of alternative interpolators and will attempt to identify any traits that will benefit sound design (Chapter 7). The framework will be used for the development of all subsequent interpolators as through the exploration undertaken in this chapter it has provided a consistent platform for rapidly-prototyping a range of interpolators.

Chapter 6 Pilot Study

From all the systems that have been reviewed in Section 2.4.2 and the exploration undertaken in Chapter 5, a number of questions arise regarding the utility of the visual information provided by the interpolator interfaces, and whether this in fact aids the user in the identification of desirable sounds. Moreover, given that a sound design task is sonically driven, do the visual elements aid or distract from the goal? If the visualizations do aid the process, how (much) do they help, and which visual cues will best serve the user when using the interface for sound design tasks?

Although, there have been comparative studies relating to interpolated parameter mappings in musical instrument design, none of these have focused on the visual aspects [129], [131]. Another area where visuospatial interpolation has been used is within Geographical Information Systems (GIS) to estimate new values based on geographical distributions of known values. A review of these techniques is available [169], but the visualizations tend to be geographical maps and the interpolation is typically performed on either a single or few parameters, rather than the relatively large number used in synthesis. In an even wider context, general guidelines exist for the design of multimodal outputs [230], in our case audio and visual, but these are fairly broad and not specific to this class of system. To attempt to provide this missing link an experiment was designed that specifically focused on interpolator usability in sound design, based on the visual display and its combined effect with the sonic output. The intention was that this experiment would also act as a pilot study for a subsequent wider experiment that would involve different interpolator visual representations.

6.1 Experiment Design

Using the interpolator framework detailed in Chapter 5 and the reimplementation of the nodes interpolator, an experiment was designed to evaluate user interactions with a graphical interpolation system, where different levels of visual feedback would be supplied. The aim was to evaluate the user interactions to determine the impact of the visualisation on the systems usability. To assess the usability of the interface five metrics were identified for investigation:

1. Time – do different visualizations encourage users to spend more time exploring the interpolation space?
2. Speed – do the visualizations presented support users to move faster in the interpolation space?

3. Distance – do the presented visualizations facilitate the travel of longer distances of the interpolation space?
4. Accuracy – does the visualization allow users to be more accurate when locating sounds in the space?
5. Satisfaction – do the visuals presented affect the perceived operation of the system?

The ISO 9241-11 (1998) standard, defines the usability of systems in terms of efficiency, effectiveness and satisfaction [231]. The metrics time and speed were chosen to measure efficiency, distance and speed for effectiveness and then satisfaction to capture user perceptions. To permit the examination of these metrics, three different visualizations were created for the nodes interpolation model. These were:

1. Interface 1 – no visualizations (i.e. an empty 2D display).
2. Interface 2 – only preset locations displayed.
3. Interface 3 – the original nodes interface.

These different visual representations for the interface are shown in Figure 57, Interface 1 – 3, left to right. In each case, the underlying nodes interpolation model remained the same so that the impact of different visualizations alone could be assessed. This included the layout of the preset sounds within the interpolation space and the target location. However, so that this was not obvious to the participants, for each interpolator the interface was rotated through 90° clockwise.

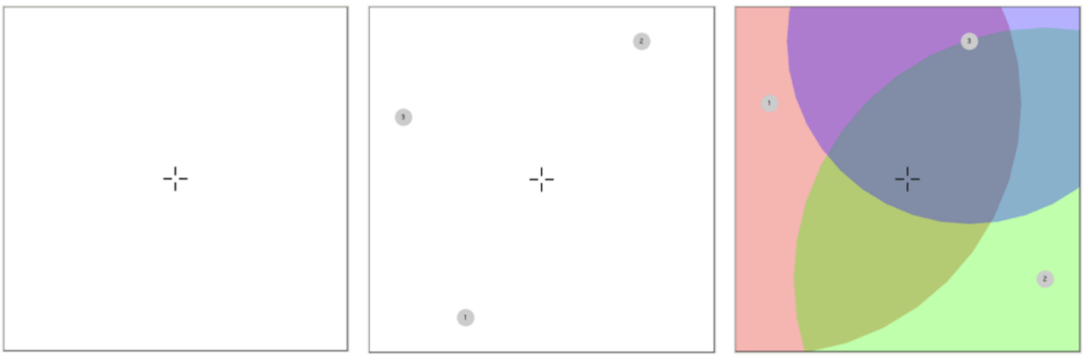


Figure 57 Different Visualizations for Nodes Interpolator, Left to Right – No Visual Cues, Preset Locations and Full Nodes

The user testing took the form of a sound design task, where the participants were asked to match a given sound which on the interpolator had a fixed, but unknown to the participants, target location in the interpolation space. A subtractive synthesis engine (Native Instruments' Massive) was chosen for the experiment as it was found to provide a rich sonic palette and predictable transitions between sounds. All of the available continuous synthesis parameters (149 in total) were mapped to the interpolators so every aspect of the preset sounds could be modified. This resulted in an interpolation space with a vast range of distinguishable outputs, with little overlap, that would be very difficult to explore with Massive's own synthesis user interface. Each interpolator display was populated with different preset sounds, with all of the presets being created from the same base patch, resulting in some sonic commonalities between them. In addition, for each interface, the relative position of the target sound location was the same, meaning at the target location in each interface generated identical preset weightings[12].

To simulate a real sound design scenario, the participants were given only three opportunities to hear the target sound before commencing the test. Once the test was initiated there was no further opportunity to hear the target. In this way, similar to a sound design task, the participants had to retain an idea of the required sound in their "mind's ear". All participants completed the same sound design task with each graphical interface, but as each interface was set up with different presets, the resulting sonic outputs for each of the three interfaces were different. To minimise any bias through learned experience of using an interpolator, the order in which each participant used the interfaces was randomised. Each test lasted a maximum of ten minutes with the participants being able to stop the test beforehand if they felt the task had been completed. If the participants did not complete the task within the allotted time the test would automatically end. All of the user's interactions with the interfaces were recorded for analysis. When the participants felt that they had matched the required sound they were asked to press a "Target" button so the location could be registered. All other aspects of the interpolation system – inputs, interpolation calculation, mappings (all parameters) and synthesis engine – remained identical between the three interfaces. From this experiment, the raw interaction data could be analysed to examine differences between the journeys made with the three interfaces.

To assess the user experience of the interfaces, the participants were asked to complete a usability questionnaire. The questionnaire was divided into two parts – the first part completed following the use of each interface and the second part filled-in after all three interfaces so they could be compared. The first part utilised one of the most commonly applied and well-tested usability models, the System Usability Scale (SUS) [232]. SUS has been used many times in over a twenty-five-year period, with a wide range of different systems and has been found to be highly reliable and robust [233], even with small numbers of users. The SUS questionnaire comprises of ten, 5-point Likert items providing a quick assessment of the usability of a system on a scale from 0 to 100. As SUS has been used in many studies it has been possible to establish norms which give an indication of a system's perceived usability [234]. It has also been shown that benchmarks for perceived complexity, ease-of-use, consistency, learnability and confidence-in-use can also be

[12] Target sounds for each interpolation model: https://youtu.be/IdiPBLqb3pU

extracted from the same survey [235]. The completion of the SUS questionnaire following the use of each interface was to try and establish if the perceived usability changed based on the visual elements of the interpolator.

In the second part of the questionnaire participants were asked which of the three interfaces they preferred and were then asked to rate it on a scale 1-10. To try and understand each participant's level of experience they were asked how many years they have been using music technology and to rate their sound design experience on a scale 1-10. Finally, they were asked to note down any critical incidents that occur during the test, both positive and negative. This allowed qualitative data to be captured to provide a clearer view of more aspects of the user experience.

The relevant ethical approval was sought and granted for the experiment and all the relevant paperwork was generated before participants were recruited and the test was undertaken. All paperwork used for this experiment can be found in Appendix A.

6.2 Pilot Study Results

The desired number of participants for the experiment was set at fifteen, based on a power assumption of 0.8 and the desire to observe a large effect size (0.5) [236]. However, when the experiment was undertaken sixteen participants were actually recruited, all with some degree of sound design experience. For each participant, their interactions with the interfaces were captured via the recording of mouse movements. This then allowed traces of the movements to be visually compared between the different interfaces. The trace gives a pictorial representation of the journey that each user made through the interpolation space. An example is shown in Figure 58 for participant 1 who had the following interface order – 1, 3 & 2[13], participant 2 with interface order 1, 2 & 3 and participant 9 with order – 1, 3 & 2. Here they are shown Interface 1 – 3, left to right with participant 1 on the top row, participant 2 on the middle row and participant 9 on the bottom. Remember when comparing the traces between the different interpolators the interfaces are rotated clockwise 90° for each.

[13] Interpolator pilot study mouse traces for participant 1: https://youtu.be/ZcQoxI1YCf4

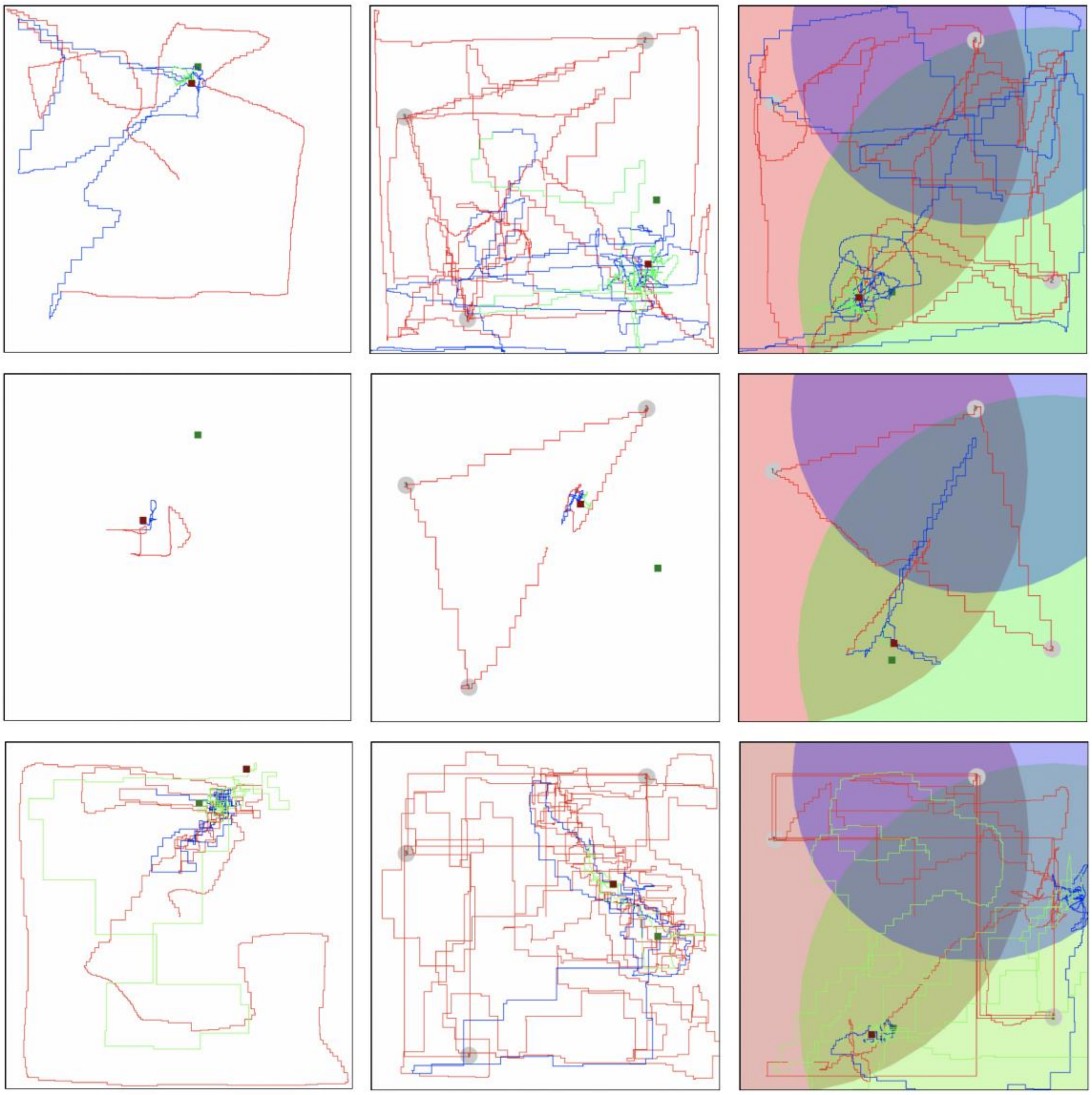


Figure 58 Mouse Traces for Participant 1 (Top Row), Participant 2 (Middle Row) & Participant 9 (Bottom Row) - Showing the Location of the Target Sound (■) and the Participants Chosen Location (■)

The cursor mouse position was sampled at a rate of 10Hz (every 100mS) as it was found to provide enough detail on the cursor movement without overloading the data analysis stage. Also, with this sample rate it is possible to visually see the relative speed of movements, where slow movements can be seen as smooth lines and fast movements appear as step changes in the trace. In addition, the location of the target sound is shown as a green square and the participant's chosen location is indicated by the position of the red square.

It was observed that at the start of the test, while exploring the space, users tended to make large, fast movements. In the middle of the test the movements tended to slow and become more localised, but a few larger, moderately fast movements were often made. Towards the end of the test movements tended to slow and become even more focused towards the intended target location. To visualize these aspects, in Figure 58 the first third of the trace is shown in red, the middle third is shown in blue and the final third is shown in green. This was also corroborated when the mouse movement speed and distance to target were plotted on a graph, using the same colour coding. Figure 59 shows an example for participant 13, with interface 3.

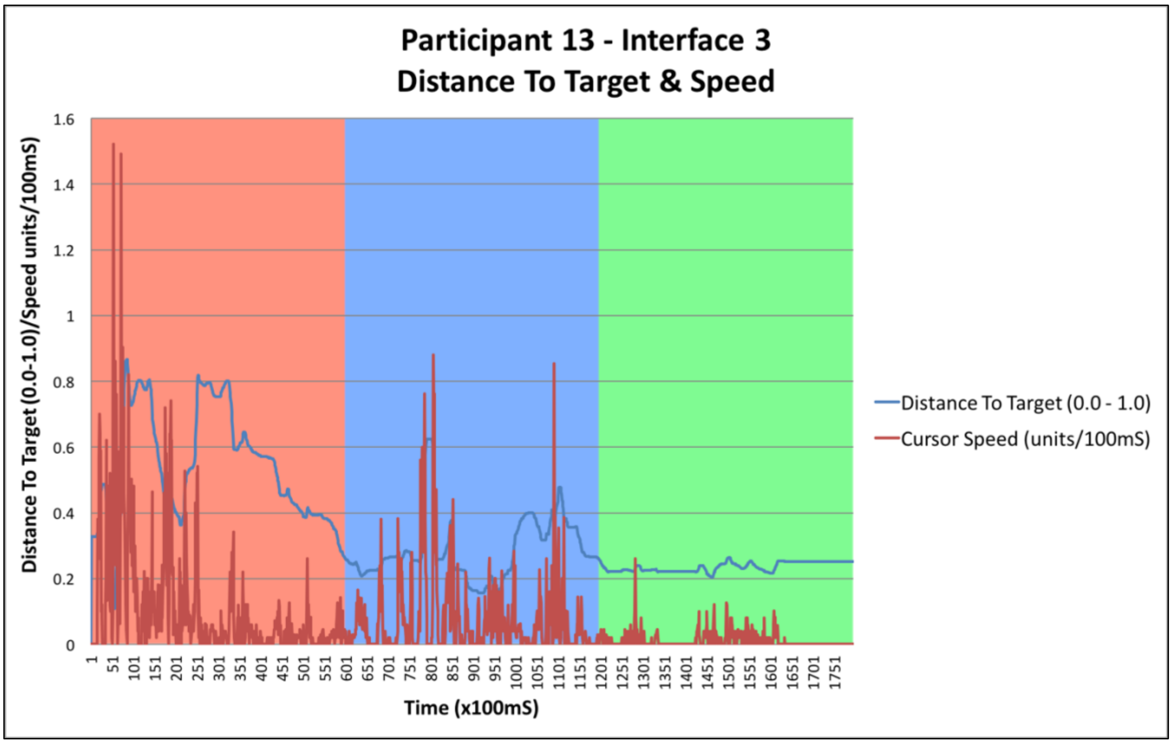


Figure 59 Mouse Distance to Target & Speed - Sampled Every 100mS for Participant 13 with Interface 3

Broadly these trends were seen in fifteen of the sixteen participants, although it did not always evenly divide into thirds of the test time. Nonetheless it appears to indicate that there are three distinct phases during the use of a visual interpolation interface:

1. Fast space exploration to identify areas of sonic interest
2. Localise on "regions of interest", but occasionally check that other areas do not produce sonically better results
3. Refinement and fine tuning in a localised area to find the ideal results

These three phases can be summarised as exploration, localisation and refinement. Interestingly these phases were present regardless of the interface being used, showing that the phases must be associated with exploration of the space and not the interface.

From the traces, it was also observed that as the detail of the visual interface increased so did the distance travelled within the interpolation space. This was despite the fact that the participants were given no information with regard to what the visuals represented. It seems that giving the participants additional visual cues encouraged them to explore those locations. To demonstrate this effect, the mean location for each trace was calculated and the deviation in the form of Standard Distance Deviation [237]. These were calculated with respect to the 'unit-square' dimensions of the interface (height and width) and were then plotted back onto the traces to give a visual representation for each interface. An example is shown in Figure 60 for participant 6 who took the test with interface order – 3, 2 & 1, although they are shown Interface 1 – 3, left to right.

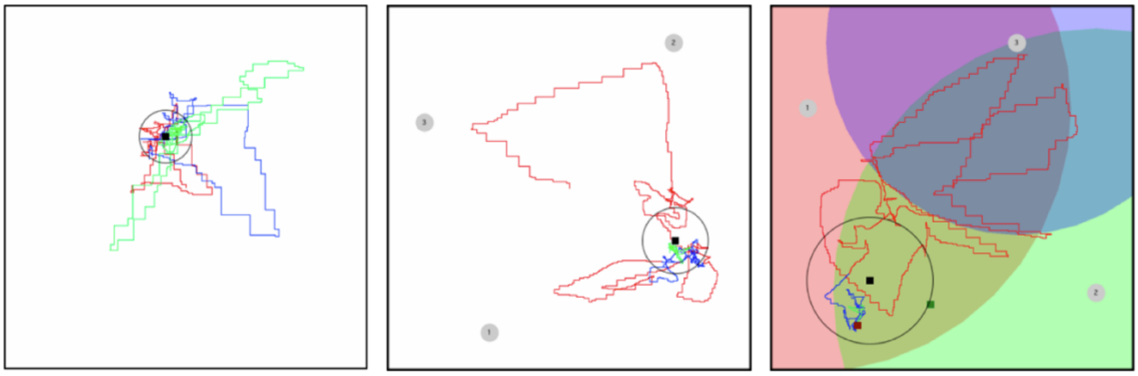


Figure 60 Mouse Trace, Mean Location (■), Standard Distance Deviation Circle, Target Sound (■) and Chosen Location (■) for Participant 6

It was noted that out of the sixteen participants thirteen showed an increase in the Standard Distance Deviation when more visual cues were provided by the interface. The normalized standard distances were examined by interface and it was found for Interface 1 the mean Standard Distance Deviation was 0.131 units (SD = 0.23), for Interface 2 the mean Standard Distance Deviation was 0.146 units (SD = 0.19) and 0.180 units (SD = 0.21), for Interface 3. These results appear to indicate that the presence of more visual cues on the interface tends to encourage wider exploration of the interpolation space, regardless of the fact that the system's output and the goal of the test is sonic.

The locations that the participant actually selected as their target sound were also plotted to see if there were any trends resulting from the different interfaces. Figure 61 shows the selected target locations for all the participants, by interface. It can be seen that with each interface the users have primarily selected points within the regions of interest (that is, node intersections), where the sonic output changes. This can be seen as clear localisation on the principal axis of variation, albeit rotated 90 degrees for each interface. As this is present with the three different interfaces it would suggest that the participants used the sonic output to identify the regions of interest.

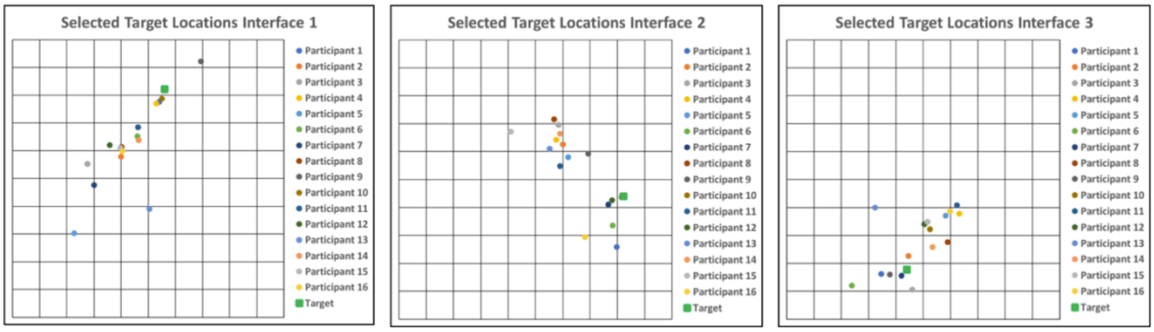


Figure 61 Participants Selected Target Locations by Interface and the Location of the Target Sound (■)

The results in Figure 61 show that for Interface 1 (no visualization) there is a fairly wide distribution of locations selected as the target[14]. The Standard Distance Deviation from the correct target location was calculated and found to be 0.300 units for Interface 1 (relative to the height and width). For Interface 2 (preset locations), there is a tighter placement of selected target locations with the Standard Distance Deviation reducing to 0.267 units[15]. Finally, for Interface 3 (full nodes) there is an even tighter localisation with the Standard Distance Deviation reducing further to 0.187 units[16]. This appears to show that as the interface provides more detail it improves users' ability to identify the intended target.

Given that the three interpolators used in the experiment were populated with different presets they each produced a range of sounds. Therefore, the sonic results for each selected target location were directly compared to the auditioned sound. In all cases there were sonic differences, but as might be expected, as the selected locations got closer to the true location the differences became less distinguishable.

6.2.1 Significance Testing

As mentioned previously, there were four areas that it was hypothesized that the visual representation might affect: time, speed, distance travelled and accuracy. Therefore, NHST (Null Hypothesis Significance Testing) was undertaken to establish if there was a significant difference between the interfaces in each of these areas. In all cases, it should be noted that the data was tested for normality (Skewness/Kurtosis, Shapiro-Wilk and visual inspection) and found to lack a normal distribution so non-parametric statistical methods were used. This is most likely due to the exploratory nature of the task being tested. For example, from the start of the test a

[14] Interface 1 target sound and selected locations for all participants: https://youtu.be/GBhENQBmCGw
[15] Interface 2 target sound and selected locations for all participants: https://youtu.be/tHxLr6j2tdY
[16] Interface 3 target sound and selected locations for all participants: https://youtu.be/jVqCEG3iK8w

participant might immediately move in the right direction, whereas others might move in the opposite direction. Therefore, a Wilcoxon signed-rank test was used to compare the median difference between the interface with no visuals (Interface 1) and the full nodes (Interface 3) visuals [238]. The calculation of effect size for non-parametric statistical methods is less clearly defined than for their parametric equivalents [239]. However, two approaches have become fairly widely adopted for calculating an effect size from the results of a Wilcoxon signed-rank test: correlation coefficient (*r*) [240] and probability score depth (*PSDep*) [241]. For completeness, both of these effect size parameters were calculated from the results obtained.

6.2.1.1 Time

From the captured mouse data, it was possible to establish the total amount of time that each participant moved the interpolation cursor within the space. It was hypothesized that using the full interface would result in an increase in the cursor movement time over the interface with no visualization (H_A: Median3 > Median1). Thus, the null hypothesis was that the different interfaces had no effect on the time the cursor was moved (H_0: Median3 = Median1).

Of the sixteen participants, Interface 3 (full nodes) elicited an increase in the cursor movement time for thirteen participants compared to Interface 1 (no visualization), whereas one participant saw no difference and two participants had a reduced cursor movement time. The difference scores were broadly symmetrically distributed, as assessed by visual inspection of a plotted histogram.

The Wilcoxon signed-rank test determined that there was a statistically significant increase in cursor movement time (Median Difference = 18.0 sec, Inter-Quartile Range (IQR) = 56.32 sec – 2.85 sec) when subjects used Interface 3 (Median3 = 66.5 sec, IQR = 129.00 sec – 31.72 sec) compared to Interface 1 (Median1 = 48.4 sec, IQR = 75.32 sec – 13.87 sec), $Z = -2.669$, $p < 0.008$). The effect sizes ($r = 0.462$, PSDep = 0.867) showed a medium to large effect, where 86.7% of the participants taking the experiment saw an increase in the time they moved the cursor with the full visual interface.

6.2.1.2 Speed, Distance & Accuracy

A similar methodology was used to compare the median difference for the average cursor speed, the total distance moved by the cursor and the distance from the selected target to the true target location, between Interfaces 1 and 3. Table 1 shows a summary of all the results obtained for the NHST. Note that all distance-based measures are with reference to the unit-square which is used to plot the visuals within the interface pane (height and width).

Test: Wilcoxon Signed-Rank	Median (IQR)	Significance	Effect Size
Time Z = -2.669	$Median_3$ = 66.5 secs (IQR = 129.0 secs – 31.72 secs) $Median_1$ = 48.4 secs (IQR = 75.3 secs – 13.9 secs)	p < 0.008	r = 0.462, PS_{Dep} = 0.867
Speed Z = -2.689	$Median_3$ = 0.569 units/sec (IQR = 0.788 unit/sec – 0.283 unit/sec) $Median_1$ = 0.297 units/sec (IQR = 0.464 unit/sec – 0.181 unit/sec)	p < 0.007	r = 0.475, PS_{Dep} = 0.937
Distance Z = -2.430	$Median_3$ = 5.85 units (IQR = 17.56 units – 3.63 units) $Median_1$ = 4.06 units (IQR = 7.87 units – 1.12 units)	p < 0.015	r = 0.429, PS_{Dep} = 0.867
Accuracy Z = -2.068	$Median_3$ = 0.177 units (IQR = 0.248 units – 0.080 units) $Median_1$ = 0.264 units (IQR = 0.367 units – 0.176 units)	p < 0.039	r = 0.366, PS_{Dep} = 0.750

Table 1 Results of the Statistical Testing To Compare Participants' Performance with Different Interface Visualizations

From examination of the data in Table 1 it can be seen that with the full interface the participants spent longer exploring, moved faster and travelled further within the space. It is worth noting that when using the visual interface, the average speed was fairly high at over half the interface width every second. In addition, the interface afforded participants greater accuracy when selecting the target sounds location. As can be seen in all cases there is a significant difference between the two interfaces. Using the normal conventions [236], the effect size appears to indicate that in all cases it is a medium effect size although a number are approaching the large effect size threshold of 0.5.

6.2.1.3 Interface 2 Results

The plots and the descriptive statistics generated (shown in Table 2) suggest that Interface 2 generates an intermediate effect between the other interfaces. However, when undertaking NHST it was not possible to show significance (with 95% confidence interval) between Interface 1 & 2 or Interface 2 & 3. Therefore, these results are considered inconclusive although, from the descriptive statistics it appears to be a smaller effect size and so it is likely that the small sample size has not allowed the significance to be shown. Given a larger sample size it may be possible to show significance for Interface 2, with the intermediate interface visualizations (preset locations).

Test:	Median (IQR)
Time	$Median_2$ = 48.5 secs (IQR = 112.3 secs – 30.8 secs)
Speed	$Median_2$ = 0.374 units/sec (IQR = 0.778 unit/sec – 0.233 unit/sec)
Distance	$Median_2$ = 5.72 units (IQR = 22.51 units – 3.18 units)
Accuracy	$Median_2$ = 0.251 units (IQR = 0.342 units – 0.130 units)

Table 2 Descriptive Statistics for Interface 2

6.2.2 Usability Questionnaire

The SUS scores for the three different interfaces were evaluated and the resulting descriptive statistics are presented in Table 3.

	Interface 1	Interface 2	Interface 3
Mean	82.81	80.62	83.59
Standard Error	2.425	3.295	3.251
Median	82.50	82.50	87.50
Standard Deviation	9.699	13.182	13.005
C.I. (95%)	5.168	7.024	5.930

Table 3 Descriptive Statistics for SUS Scores by Interface

As can be seen from these results there is very little difference between the mean and standard deviation for the different interfaces. For completeness a one-way repeated measures ANOVA was performed, which showed that there was no significant difference between the SUS scores for the three different interfaces ($F(2, 30) = 0.378$, $p = 0.688$), given that the sphericity assumption had not been violated ($\chi^2(2) = 1.914$, $p = 0.384$). It appears that the perceived usability of the interpolators is not affected by the visualisation. This may be attributed to the fact that the systems functionality and operation is identical with the only change being the visuals presented on the interface. However, in all cases the SUS scores were extremely high and equivalent to an "A" grade in the 90-95 percentile range, which is equivalent to the top 10% of scores from a database of over 10,000 previous SUS scores from a wide range of computing systems [234], [241], with both simple and complex interfaces [243]. From this database, it has also been shown that the average SUS score is 68 and for all of the interpolators the mean scores are much higher (82.34%, *SD* = 11.68). To assess if the average SUS scores of the interpolators is significantly different to the average SUS score a one-sample t-test was performed. This showed that the SUS score was statistically significantly higher by 14.34 (95% CI, 10.90 to 17.79) when compared with the average SUS of 68, $t(47) = 8.367$, $p > 0.0005$. This appears to indicate that the users found the use of interpolators for sound design to be positive. It has been previously suggested that users that give a SUS score of 82 (±5), tend to be "Promoters" and likely to recommend the system to other users [242]. Although these norms [236], [242] are from a wide range of general computing systems, the obtained SUS scores still appear to perform well when compared to other music/sound systems that have also been tested with SUS [244-247].

The results for each SUS item were also compared to the defined benchmark values, based on an overall SUS score, in this case the average (68). These have been computed based on data from over 11,000 individual SUS questionnaires and offer a mechanism to examine whether specific items have any bias in the overall SUS score [235]. Again, it should be noted that this database consists of results from a wide range of general computer systems, simple and complex. The score for each item in the interface tests were compared to the corresponding benchmark values using one-sample t-tests. In all cases, the results show the interpolators perform significantly better than the SUS average, confirming the previous result. The benchmark tests were also repeated for a 'good' system with a SUS score of 80. These results did not show significance so it is not possible to say the interpolators are perceived as being better in some areas of usability than others.

In the second part of the questionnaire participants were asked which of the three interfaces they preferred using. The results are shown in the frequency table in Table 4.

	Frequency	**Percent**	**Cumulative Percent**
Interface 1	2	12.5	12.5
Interface 2	4	25	37.5
Interface 3	10	62.5	100
Total	16	100	

Table 4 Frequency for User Preferred Interface Choice

Of the participants, 62.5% preferred using the interpolator with the full nodes visualisation and only 12.5% preferred the interface with no visuals. These results appear to indicate that most users preferred to use interfaces that provide visual cues. To understand the users experience they were also asked “How many years have you been using music technology?” This data was evaluated with respect to their choice of preferred interface giving the results shown in Table 5.

	Interface 1	Interface 2	Interface 3
Mean	7.00	6.75	10.00
Standard Error	1.000	1.250	3.528
Median	7.00	6.50	4.00
Standard Deviation	1.414	2.500	11.155
C.I. (95%)	12.71	3.980	7.980

Table 5 Descriptive Statistics for Music Tech. Experience by Interface

These results appear to show users that preferred the full nodes interface had an average of 10 years’ experience using music technology, as opposed to 7 years and 6.75 years for interface 1 and 2, respectively. However, given the small sample size between the groups, there appears to be a wide deviation in the means and as such the results should be considered inconclusive. When all the participants responses were analysed together it did confirm that the participants had a range of different experience levels and there appears to be little bias, although the small sample size (16) should be noted as a larger sample may show other trends. Table 6 shows the results of this analysis.

	Music Tech. Experience
Mean	8.810
Standard Error	2.216
Median	5.500
Standard Deviation	8.864
C.I. (95%)	4.730

Table 6 Descriptive Statistics for Music Tech. Experience

Analysis of the critical incidents noted by participants largely mirror the statistical results presented. However, this information did allow an insight into why the participants made the choices they did. The participants that chose Interface 1 (no visualization) made comments that the lack of visuals required them to use their “ears” to locate sounds without decisions being “influenced” by the visuals. Similar remarks, that the sound was primarily used, were also made by those that chose Interface 2 (preset locations), but that the locations were “helpful” and guided them to "pinpointing" sounds. Statements were also made that this interface "matched [their] experience level", but that Interface 3 was "simplest to pick up quickly". Comments made by those that chose Interface 3 (full nodes) related to it "guiding the user how to use it". This appears to be corroborated by multiple participants as comments were made that the visual showed "sound range", "areas that played different sounds" and "where [the sounds] overlap". One participant also stated that the visuals allowed them to "explore sounds without being too lost from [another] sound". It should also be noted that the two participants that did not choose Interface 3 as their preferred choice, stated that the visuals could be "distracting".

In the general comments, ease of use came up repeatedly, with one participant actually stating that the interpolation systems "did not have a steep learning curve". Another theme that appeared continually was that the participants found the activity "fun" and/or "enjoyable". These aspects appear to support the findings of the SUS and appear to encourage further use of interpolators for sound design activities. In fact, one participant stated they would "like to see it in sound design tools". However, there might be other application areas, as one comment identified that the interpolators maybe useful for "making sounds for EDM (Electronic Dance Music)" and another considered their use for "composition".

6.3 Pilot Study Discussion

Although the use of sixteen participants in the testing is a small number when compared to usability testing for general computer applications, it is a fair number in the music/sound technology area. Here formal usability testing is not often undertaken, as with the other interpolators [123], [162], [163], [177], [180], [183] or where it is done, it is often done with smaller numbers [244-247]. Nonetheless, with sixteen participants some interesting results were obtained, and significance was shown between user interactions with the interface that had no visuals and the one with the full visuals. In addition, to the relatively small number of participants, other limitations of the experiment were that it only tested one specific sound design task and the users were not permitted to change the layout of the interpolation space. These restrictions were included to constrain variability in the experiment, creating consistency between the different interfaces and focusing the participants on a single activity. This appears to have been successful and further testing can be undertaken in the future to corroborate the results.

As identified in Section 6.2, the testing showed there appears to be three phases to the identification of a sound with a graphical interpolator system (exploration, localisation and refinement). In the first phase the users make large, fast moves as they explore the space. During the second phase the speed tends to reduce as they localise on specific regions of interest. In this phase, though, confirmatory moves have been observed when the user seems to quickly check that there are no other areas that may produce better results. These are inclined to be made at a moderate speed, often in multiple directions. Then in the final phase the user refines the sound with small, slow movements as they hone-in on a desired location. These phases appear to be apparent regardless of the visual display that is presented to the users, with similar phases being observed with all three of the interfaces tested. However, the frequency of movements, scale and locations did vary with each participant. This is to be expected as this was an individual task and the participants possessed different skill levels.

From examination of all the journeys (mouse traces) for the different interfaces, the visual feedback presented by each affects how the users interact with the systems. When no visualization is provided, the users were effectively moving “blind” and tended to just make random movements within the space. When the preset locations were provided, although the users were not aware of where or how the interpolation was being performed, the provided visual locations encouraged the users to investigate these points and so explore the defining locations. The full interface not only shows the location of the defining sounds, but also focuses the exploration and appears to indicate to the users’ regions of interest (node intersections for this interpolation model), where there may be interesting sounds. This was also supported by the user’s feedback on the questionnaire, where specific comments were made about the identification of overlaps between nodes.

The results from analysis of the interpolation paths appear to be corroborated by the NHST, where significant differences were found in the total time taken to complete the test, the average speed of movements during the test, the total distance moved during the test and the accuracy in locating a target, between the interface with no visuals and the full visual display. Although the primary output from the interpolator is a sonic one, it appears that the feedback provided by the visual display is also of importance. Given the increase in the time taken and distance moved, it appears that the visuals encouraged the participants to explore more of the space and this is also supported by the increase in the standard distance deviation. It also seems that the visual feedback gives users the confidence to make faster movements with the interpolator. This may be similar to the way a blindfolded person may take longer to explore a space compared to a person without a blindfold, making slower movements and with minimal travel. Finally, given the same activity was being undertaken with each interface, with identical controls and the goal was a particular sonic output that did not directly relate to the visuals, the increase in accuracy was not foreseen. This could be a secondary effect from exploring more of the space meaning participants were then more likely to locate the correct target. However, it could also be that the full interface provided visual cues so that when a region-of-interest had been located during the exploration phase the visual cues then made it easier to return to the same area during the localisation phase, in the same way that a map might aid navigation when trying to discover an unknown location. This was also highlighted by one of the participants who said that the visual stopped them getting lost after finding a sound of interest.

Interestingly, although the visual representation was not explained to the participants it is clear from the participant feedback that they were able to work-out from the visuals what they meant and their function in terms of the system's operations. This appears to imply that the system with the full visuals is intuitive and guides the user in its operation.

Given that no significant difference in the SUS scores was observed between the interfaces it may be said that there is no difference in the perceived usability, despite the fact that the subjects' behaviour differed between interfaces 1 and 3. However, as the overall average SUS score is so high it appears to suggest that just the concept of using a graphical pane to control interpolations is considered highly usable in itself. This seems to be supported by the general comments made by the participants that they enjoyed using the interpolators and found the experience to be fun. Moreover, given that there seems to be little difference in the perceived usability, based on the visualisation, and some participants stated they preferred fewer visual cues, there is a case for giving the user control of the level of detail provided by the interface. In this way, users that just want to concentrate on the system's sonic output can do so with no visual guidance and those that find the visual useful, could customise the level of detail for their particular needs or preference.

6.4 Pilot Study Conclusions

The identification of three distinct phases of use during the testing of the graphical interpolators is of significant interest as it suggests that users interact with the interfaces differently at different stages during their journey through the interpolation space. Better understanding of the user behaviour with these systems will allow further evaluation of different interface visualizations. Moreover, this information could be used in the design of new interfaces that provide users with visuals that facilitate the different phases of an interpolated sound design task. There is also no reason why the visuals have to remain static, and they could perhaps change as the users enter the different phases of the process using an Interactive Visualization (IV) paradigm [248]. This could either be through allowing the user to select different visualisations or automatic, based on their interactions with the space.

From the results obtained it appears that the visual display of an interpolator's interface has a significant impact on the sonic outputs obtained. The resulting journeys made with the full interface show a wider exploration of sonic outputs, faster speed of movement and improved accuracy at locating a specific sound. The effect sizes are relatively large, giving greater confidence in the validity of the results. At the same time however, it appears the system is still perceived by the users as highly usable and unaffected by the change in visualisation. This again adds further strength to the idea of using IVs that can be adapted to the user needs.

A number of different visual models have been previously presented for graphical interpolators [123], [162], [163], [177], [180], [183], each of these using very different visualizations. In Section 5.5, it has been shown that these different interpolation models generate very different sonic palettes, even when populated with identical sounds. Given now the suggested importance of the visual feedback provided by each interface, it will be important in future work to evaluate the suitability and relative merits of each through further user testing. Based on the results from this study there are two areas to be refined. Although for this experiment the directive nature of the task, asking all the participants to locate the same sound, worked well as it allowed direct comparisons between the different levels of visualization, it is perhaps not truly representative of a real sound design task. Sound design is a highly creative and individual practice so typically there are likely to be different choices made between designers. As a result, in the testing of the different interpolators, a more typical sound design scenario will be adopted. It is also anticipated that this will help to identify not only the positives of using interpolated interfaces for sound design, but also potential limitations.

Another area that will be examined further is the use of the standard SUS questionnaire to compare the usability. While SUS showed that these interpolators are considered highly usable, it did not give enough detail to show any perceived differences between the visualizations, even though the quantitative data suggests otherwise. For comparison purposes, SUS will be used in subsequent interpolator testing, but to probe this area further, the standard SUS questions will be augmented with additional items, more specific to the interface. This strategy has been shown to work successfully in other evaluations of audio technology interfaces [249].

Finally, given that it appears graphical interpolators are perceived as highly usable in this application area, further consideration should perhaps be given to their wider use in other domains. This was even highlighted by participants taking part in this study, who made suggestions relevant to their own practice, as detailed in Section 6.2.2. In the music/sound area they could be further utilized for generation, composition, performance or musical expression, as well as sound design, while in a wider context, graphical interpolators could be beneficial for the control of graphics, animation, texturing, image-processing, database transactions, avatar generation, game-level design, etc. In fact, graphical interpolation lends itself to any situation where new states require exploration and/or identification, based on a set of known states, particularly within dense parameter spaces.

Chapter 7 Interpolator Usability Study

The results from the pilot study presented in Chapter 6, demonstrate that visual cues provided by a graphical interpolation system can have an impact on usability. As identified in Sections 2.4.2 and 5.5, a variety of distinct graphical models have been used in the past for parameter interpolation [123], [163], [166], [177], [183], [186], each of which presents the user with different levels of visual feedback. As detailed in Section 5.5, six of these have been reimplemented with the Graphical Interpolation Framework and the range of different visualisations are shown in Figure 51 (page 104). Through this work it has been shown that each of these models provides a unique sonic palette, even when populated with the same preset sounds at identical locations. This chapter describes a wider evaluation undertaken to see if there is any relationship between the specific visual characteristics of each interface and the suitability of the interface for sound design. Moreover, to establish the impact the visual representation has on the usability of each interface.

7.1 Interpolation Visualisation

From the six different interpolator reimplementations detailed in Section 5.5 it was possible to analyse and compare the visual cues provided by each interface, a summary of which is presented in Table 7. This is based on the cues identified in Section 3.2, but as the preset handles and the interpolation point cursor are provided in every case they have not been included in the table. From this comparison it became apparent that the Nodes and Light interpolators are very similar intersecting models, except for the addition of the angular settings available in the light model. From the pilot study investigation, Chapter 6, it appears that one of the most important aspects for a graphical interpolator is the identification of regions-of-interest where interpolation is actually performed. As can be seen in the comparison table, most of the interpolator visualizations explicitly show this with some form of visual cue, either with an intersecting region or as free space. However, for both the Radius-Based IDW and Voronoi Tessellation the graphics only imply where the interpolation is performed rather than explicitly showing a defined area. For the Radius-Based IDW, it is areas of free-space where more than one preset can be contained within the cursor radius. Similarly, for the Voronoi Tessellation it is areas of free-space, but this time restricted to areas where the cursor has more than one natural neighbour. Closely aligned to this is the display of which presets are contributing to the current interpolation. For the intersecting models the presets included are shown by those intersecting the cursor position. The

Gravitational and Triangulation models result in a constant, fixed number of presets (all and three, respectively) being included in the interpolation.

	Interpolator 1	**Interpolator 2**	**Interpolator 3**	**Interpolator 4**	**Interpolator 5**	**Interpolator 6**
Visual Model	Nodes	Gravitational	Radius-Based IDW	Light	Delaunay Triangulation	Voronoi Tessellation
Minimum Interpolation Requirement	Two Nodes (Overlapped)	Two Planets	Two Presets (within Cursor Radius)	Two Lamps (Overlapped)	Three Presets	Two Presets
Presets Included in Interpolation	Intersecting Nodes	All	Presets within Cursor Radius	Intersecting Light Beams	Presets at Cursor Containing Triangle’s Vertices	Natural Neighbours
Field-of-Influence (Preset Range)	Shown by Node Size	Implied by Planet Size (Strength) Across All Free-Space	Implied by Cursor Radius Size	Shown by Extent of Lamp Beam	Implied by Area of Adjacent Triangles	Implied by Area of Polygons Between Natural Neighbours
Region-of-Interest (Interpolation Space)	Area of Node Intersections (Overlapped Node Colours)	Free-Space minus Planet Surface (White Space)	Free-Space with More Than One Preset within Cursor Radius (Shaded Cursor Region)	Area of Light Beam Intersections (Overlapped Light Beam Colours)	Free-Space within Triangulation Mesh (Triangles within Mesh)	Free-Space with More Than One Neighbour (All Polygon Surfaces Between Neighbours)
Preset Weightings (When Included in Interpolation)	Implied by Cursor Position in Intersection and Relative Distance to Node Centre	Implied by Relative Distance to Planet and Size	Implied by Relative Distance to Presets within Cursor Radius	Implied by Cursor Position in Intersection and Relative Distance to Lamps	Shown by Relative Area of Triangles Between Cursor and the Containing Presets	Shown by Relative Area Covered by Ghost Polygon
Preset Recall	Non-Intersecting Area of Node (If Available)	Planet Surface	One Preset Marker with No Other Presets within Cursor Radius (If Available)	Non-Intersecting Area of Light Beam (If Available)	Preset Marker	Preset Marker

Table 7 Interpolator Visual Cue Comparison for Reimplemented Interpolators

The Radius-Based IDW explicitly shows the presets included in the interpolation as those within the cursor radius. Finally, the Voronoi Tessellation includes all presets that are natural neighbours of the cursor position. That is, the preset polygons that are adjacent to the cursor's polygon. In the past, others have provided a visual cue for exactly which presets are included in the interpolation by drawing guidelines that connect all included presets to the cursor [123], [180]. This technique provides a direct representation of all presets included in the interpolation and could potentially be included in any interpolation system. However, there may be certain situations where the use of guidelines would detract from other aspects of the visual display.

Linked to being able to understand where the interpolation is performed is being able to interpret a preset's range and so its relative influence within the interpolation. With the first four interpolators, the weightings are implied primarily through a distance component and this may not always be obvious, especially when there is a secondary component, such as size, to consider. The other two interpolators show a visualization that directly relates to the relative proportions of the contributing presets. In some of the other original implementations shading and/or colour interpolation have been used to imply the weightings of presets through the interpolation space [123], [124], [162] & [180]. Although this could be added to any of the interpolation systems it may not always be desirable as it might detract from some other visual aspects, such as being able to clearly see regions-of-interest. An alternative solution that maybe better is the area-based representation as occurs with Triangulation and Tessellation systems as it provides direct visual feedback to the user of each preset's proportion in the weightings. The fact this is possible with both is no surprise as geometrically the two are the duals of each other, because for every Triangulation a unique corresponding Tessellation can be constructed and vice-versa [229]. However, the Voronoi Tessellation does have the benefit of extending to the full area of the space, whereas Triangulation creates a boundary defined by the outermost presets. A similar solution could be provided for the other interpolators if an additional bar-graph display is included that shows the relative weightings of the current interpolation output.

To help interpret and spot trends in the results presented in Table 7, dimension space analysis was used to give a representation to the visualisation that each interpolator model provides. This will then be used to identify commonalities between models and visual cues that each model may offer.

7.1.1 Interpolator Dimension Space

Dimension space analysis has already been defined and applied to the design of digital musical devices [223], [250], as a means of visualizing the differences between systems. These define a space with seven and eight dimensions respectively, but they have been created for slightly different contexts, based on phenomenological versus epistemological factors. Although a graphical interpolator could be used as a musical device it does not fit with the sound design application area of interest in this work. In addition, all the interpolators have similar basic functionality so it is unlikely that it will be possible to identify significant differences using the dimension spaces previously defined. Given that the aspect of interest here is the visual cues that each model provides, a new six-axis space is defined. Figure 62 shows the dimension space used to analyse the interpolator visualisations. Each of the axes is marked with a representative range and are described in detail in the remainder of this section. It should be noted that although the space has been primarily created to analyse the six interpolator visualisations that have been reimplemented, care has been taken in the definition of the space to ensure that it is flexible enough to analyse any of the graphical interpolators identified in Section 2.4.2.

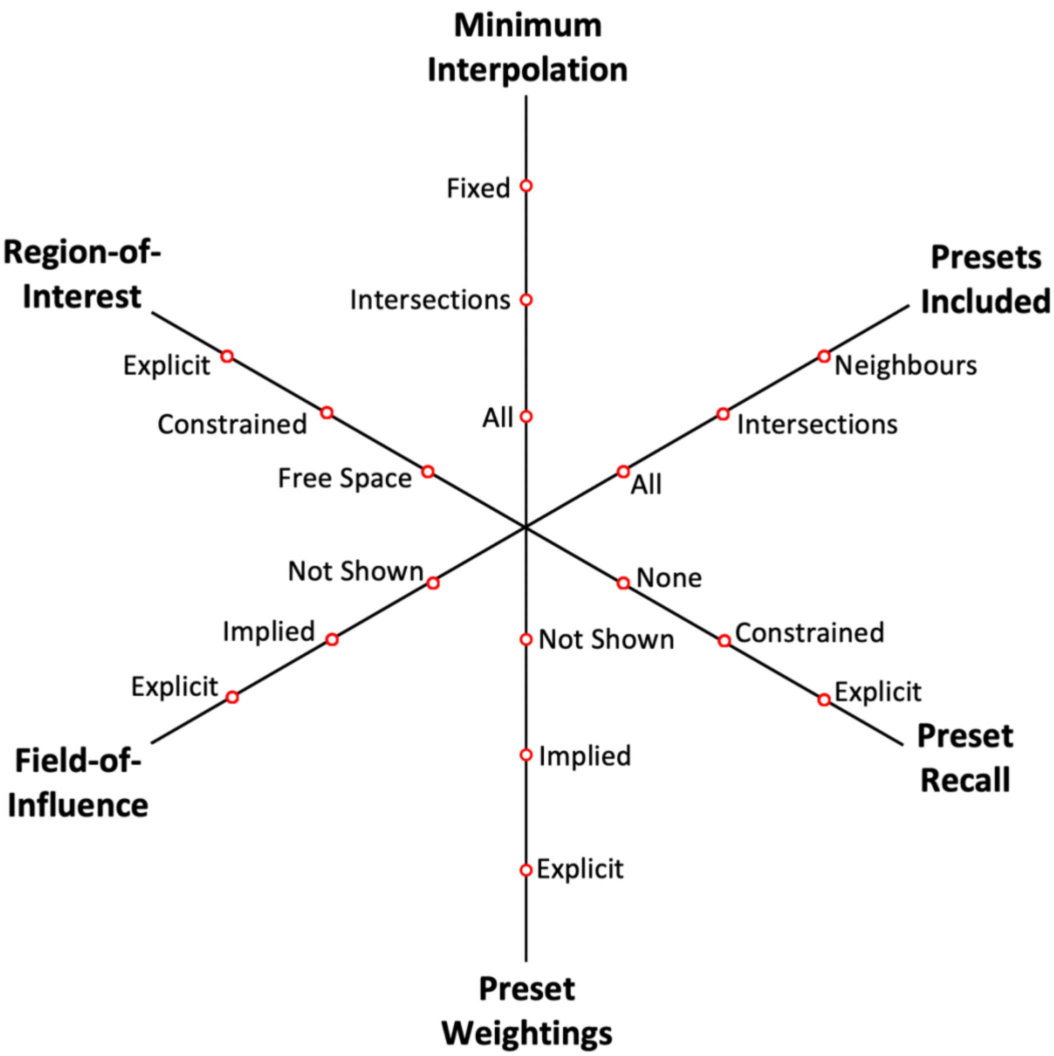


Figure 62 Six-Axis Dimension Space for Interpolator Visualisations

The six dimensions are derived from the information presented in Table 7 and each is made up of three discrete possible values that describe the requirements of the visual model or how the interpolator model handles each visual cue. Note that where there are properties that are common to all of the interfaces (such as, preset locations, handles, etc.), these have not been included in order to highlight the differences between the interfaces. Where there are common discrete values on different axes, they have been positioned the same and they have been arranged in order of increasing desirability, based on the findings from the testing already undertaken and the evaluation of previous systems. In this way, the larger the radar plot is, more visual cues are displayed on the interface. However, the evaluation of the cues actual desirability will need to be established through testing. As much as possible, the axes have been arranged to provide grouping between the common values on them. The individual axis details are given below and work from the origin outwards:

- The *Minimum Interpolation* axis specifies the minimum requirement before the interpolation can be performed. The axis contains three discrete points for this requirement: *all, intersection* and *fixed*. The first value, *all*, is used for those interpolators that perform interpolation between all of the presets within the space without any restriction, such a SYTER [163]. The *intersection* value is used where the interpolator needs the presets to be arranged within the space so that intersections occur, for example as seen in nodes [183] or Interpolator [177]. The *fixed* value is used for interpolation models where a specific number of presets is required in order to perform interpolation. For example, a triangulation model requires three presets to perform the interpolation [186] and a quadrilateral grid requires four [125].
- The *Presets Included* axis indicates how the interpolators visual model shows the user which of the presets placed within the space are included in the current interpolation output. The first value again represents the interpolation models where *all* presets shown within the space are included in the interpolation. The middle value, *intersections*, similar to the previous axis is for those systems that use some form of intersectional model and the presets included in the current interpolation are those that intersect the current interpolation point's position. The final value, *neighbours*, represents those interpolators where the interpolation is performed with the presets that are neighbours of the current interpolation point, as is the case for Metasurface [123].
- The *Preset Recall* axis represents that model's ability to recall just the original preset sounds. The first value, *none*, is for models where interpolation is always being performed and so it is not possible to hear the original presets. This is the case for models such as an unconstrained IDW [166] because the interpolation is always performed as all

presets are always included in the interpolation. The *constrained* value is for interpolation models where the layout of the preset may constrain the recall of preset so that it is not possible. For example, with the intersecting models, such as nodes [183] the preset sound can be recalled if the interpolation point is positioned in a non-intersecting region of the presets area. However, if the current layout does not offer a non-intersecting region, then it will not be possible to hear the original sound. The final value, *explicit*, is for those models that have a specific location where the interpolation results in recalling the preset sounds. For example, for the SYTER model this is when the interpolation point is positioned on a planet's surface [163].

- The *Preset Weightings* axis defines how the interpolator model shows the weighting's of the individual presets included in the current interpolation output. The first value, *not shown*, is for interpolators that have no visual cue to represent the individual weightings of the interpolation presets, as was the case with Interface 1 (no visualization) in the pilot study detailed in Chapter 6. The *implied* value is where the weightings are implied through the model. For example, with IDW the individual weightings are implied by the distance between the interpolation point and the included presets [166]. The *explicit* value is where the interpolator provides an explicit visual cue showing the individual weightings. For example, with INT.LIB weightings are shown visually by linking them to the colour transparency of the presets within the space [180].
- The *Field-of-Influence* axis shows the interpolators ability to display the range of each preset. The point *not shown* is used for interpolators that give no indication of a presets range, as would be the case with interface 1 (no visualisation) or 2 (preset locations) from the pilot study detailed in Chapter 6. The next value is *implied*, and is used where the range is not directly shown, but is implicit by the presets position relative to the other presets. This would be the case for interpolators such as those that use some geometric arrangement of the presets [162]. This leaves the final value of *explicit* which is used for the interpolators that plainly show the extent of each preset, such as Interpolator [177].
- The final axis of *Regions-of-Interest* represents how the visual model shows the areas where the interpolation is being performed. The first value, *free space* is used for interpolators that can perform the interpolation across any area that is not a preset location, as is the case for SYTER [163]. The next value *constrained* is used where the free space is restricted in some way such as being contained as is the case for the radius-based IDW or proximity of neighbours as with Metasurface [123]. The final value *explicit* is where a region is explicitly shown either by a containing polygon as with triangulation [186] and quadrilateral [125] interpolation or as an intersectional area as with nodes [183].

7.1.1.1 Analysis of Interpolator Dimension Space Plots

Having constructed a suitable dimension space for examining the interfaces of graphical interpolators this was then applied to the interpolators that have been reimplemented using the Graphical Interpolation Framework, detailed in Chapter 5. Each of these was analysed against the six dimensions defined in the previous section and a plot was generated for each. For comparison these are shown in Figure 63. The first thing to become apparent is that the gravitational model (Interpolator 2) resulted in the plot with the smallest area with most of the axes getting the minimum score. The exceptions being the preset recall axis which notably got the highest value for explicitly allowing the source presets to be recalled and the preset weighting as this is implied by a distance function. As the values have been arranged along each axis with increasing desirability, plots that are focused on the origin could be considered less ideal than those that are wider. In this case, the gravitational model appears to be the least suitable as it does not provide the user with as many visual cues as alternatives.

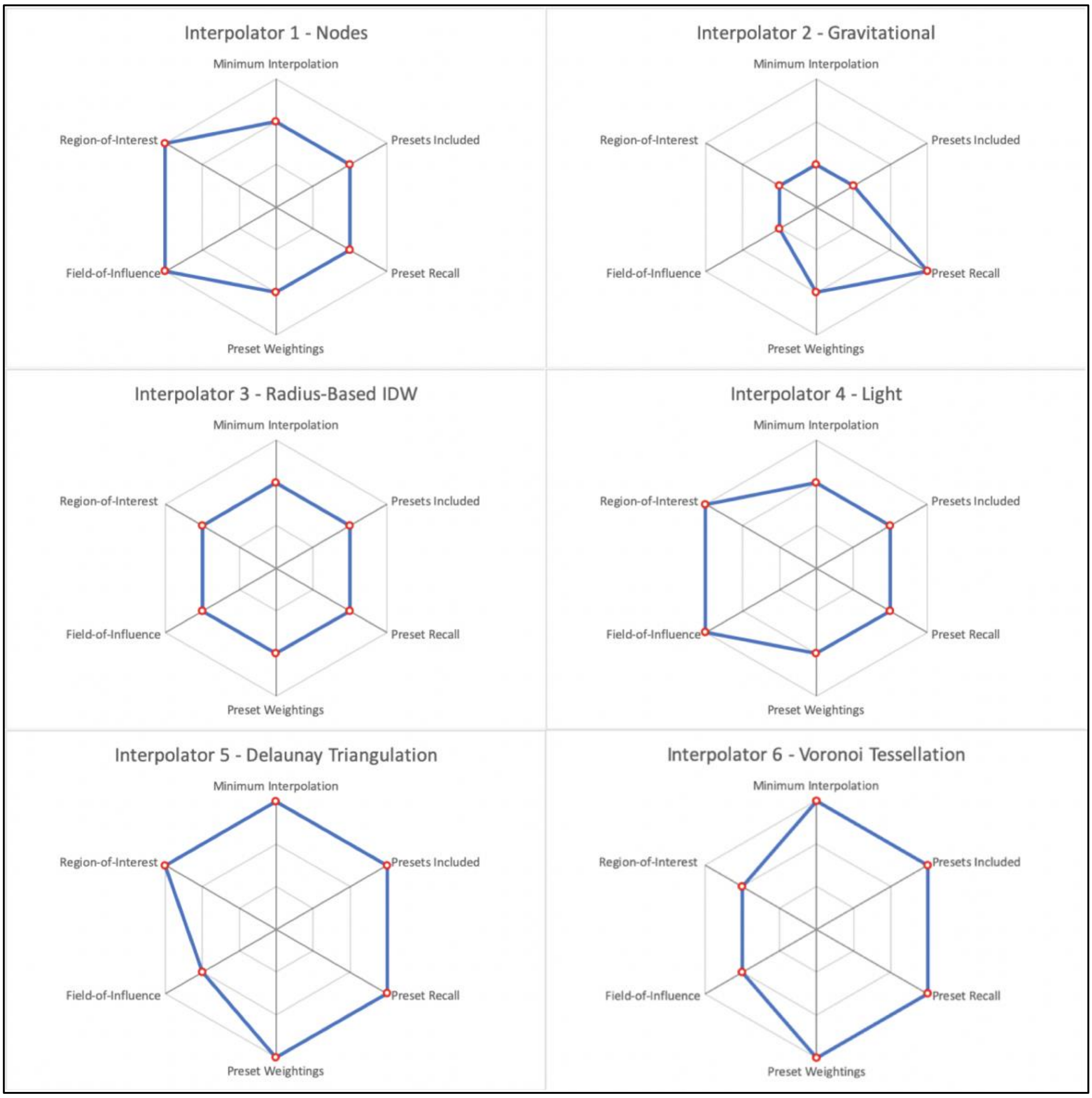


Figure 63 Dimension Space Analysis for the Reimplemented Interpolators

The next widest plot is for Radius-based IDW (Interpolator 3) which got the middle value on all of the axes, indicating it is preferable to the gravitational model, but not as favourable as the others. The two interpolators that both use intersecting models, nodes (Interpolator 1) and Light (Interpolator 4), both produce identical plots for the defined dimension space. This is perhaps not surprising as the light model's angular component is the only significant difference between these two intersecting models. Both got the highest score on two axes for explicitly showing each preset's *field-of-influence* and the *region-of-interest* created. The Voronoi Tessellation (Interpolator 6) produced the next widest plot and got the highest value on all of the axes apart from two: *field-of-influence* and *region-of-interest*, which are the two axes that the intersecting models achieved the highest values on. Finally, the triangulation (Interpolator 5) produced the plot with the widest area, achieving the highest value on five of the six axes, but only achieving the middle value for showing the presets *field-of-influence*. The fact these two models that achieved the widest plots are the geometric duals of each other [229] may be of importance.

7.1.1.2 Evaluation of Dimension Space Results

The dimension space analysis has proved useful for gaining a pictographic representation of the characteristics of each graphical interpolator and has allowed their ranking against a scale of desirability. It has also forced the consideration of which characteristics might be advantageous when designing new visual interfaces for the control of an interpolation system. However, the scales of the plots are based on the perceived desirability, as a result of the author's bench testing and evaluation undertaken in Section 5.5 and not on any empirical data. Also, with this method of dimension space analysis there is an assumption that each axis is of equal importance in the radar plot [251]. However, in terms of the systems' usability this may not be the case. Nonetheless, they do offer a good way to directly compare the differences between the multiple interpolators that all have the same base functionality. While in this analysis each axis was defined with three possible values, in the future it may be possible to define more granularity for the scales to provide greater detail.

To try to verify the outcomes from the dimension space analysis and to gather some quantitative data, usability testing was undertaken, based on that completed in the pilot study, detailed in Chapter 6.

7.2 Interpolator Usability Experiment Design

Using the interpolator framework detailed in Chapter 5 and the reimplementation of the six interpolators, an experiment was designed to establish if there was any difference between them in the way that users interact with the interfaces. The aim was similar to that of the pilot study in evaluating the user interactions with each interface to determine if the different visual cues influence a system's usability. The same metrics used in the pilot study (time, speed, distance, accuracy and satisfaction) were used again due to their success in the pilot study. To examine these metrics comparative testing was undertaken with six reimplemented interpolators using the same layout of presets as was utilised during the previous bench testing and evaluation (detailed in Section 5.5.2 and shown in Figure 51). To recap, the six different interpolators have an identical layout of presets within the interpolation space and the only difference is the visual model, and so the visual cues already identified. As already recognised through the earlier bench testing, although populated with the same sounds, each interpolator generates a different sonic palette. Therefore, as well as examining how users interact with each visual interface, the experiment will also aim to see if the visual models have an impact on a sound design process. To this end, rather than locating a target sound, as was the case in the pilot study, this time the participants were given a more realistic sound design task. For each interpolator they were given a written "brief" detailing the type of sound that required designing, a visual context of where it is intended the designed sound will be used and an aesthetic requirement for the sound. To ensure some comparability between the different sound design tasks for each interface it was decided to stick with one type of sound design task, allowing the same preset to be used with each interpolator. Then for each interface a different context was defined so the individual tasks had some unique aspects. To provide a diverse range of contexts and potential sonic solutions, the sounds chosen were background ambiences for spacecraft in a science fiction film context. Science fiction was selected as the genre, as it requires a diverse range of sonic outputs [242] and because it is not real there should be less preconception of how it "should" sound. Spacecraft were chosen as there have been many different depictions over history and media of different types of spaceships: motherships, fighters, cargo freighters, shuttles, etc. All of these different types of spacecrafts require unique sonic identities [253], not only based on their type, but also the contents narrative and aesthetic goals. For example, the sound of the Nostromo[17] from the film Alien (1979) [254], sounds very different to the Millennium Falcon[18] from the film Star Wars: Episode IV - A New Hope (1977) [21], although they are similar types of vehicles created around the same time. This being the case the following rubric was created for the sound design tasks and was given to all the participants:

> *"You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a "soundbed", over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:"*

Having decided on the type of sounds and the design rubric to be used, the different characteristics for each task then needed to be established. Six different goals were required so they could be mapped to the six different interpolator interfaces. These were chosen to be as varied as possible so that each scenario was distinct:

Interpolator 1 - Soothing and healing sound for a medical hospital spaceship.

Interpolator 2 - Manic and chaotic sound for a spaceship owned by a psychopath.

[17] Recording of the Nostromo Ambient Engine Noise: https://youtu.be/U4p1mZnKkhc

[18] Recording of the Millennium Falcon Ambient Engine Sound: https://youtu.be/P93kbL0G0ww

Interpolator 3 - Calm and tranquil sound for a spaceship owned by a battle hero.

Interpolator 4 - Threatening and scary sound for a spaceship where a killer is hunting the crew members.

Interpolator 5 - Despair and despondency for the sound of a spacecraft that is stranded in deep space with no engines and dwindling life-support systems.

Interpolator 6 - Sombre and gloomy sound for a dying spaceship that is being eaten by parasitic space slime.

For each of the different scenarios a visual representation was also supplied to give the participants a particular target aesthetic to make the sound design task as realistic as possible. These are shown in Figure 64 and provide a visual linkage to the sound characteristics that were defined for each interpolator.

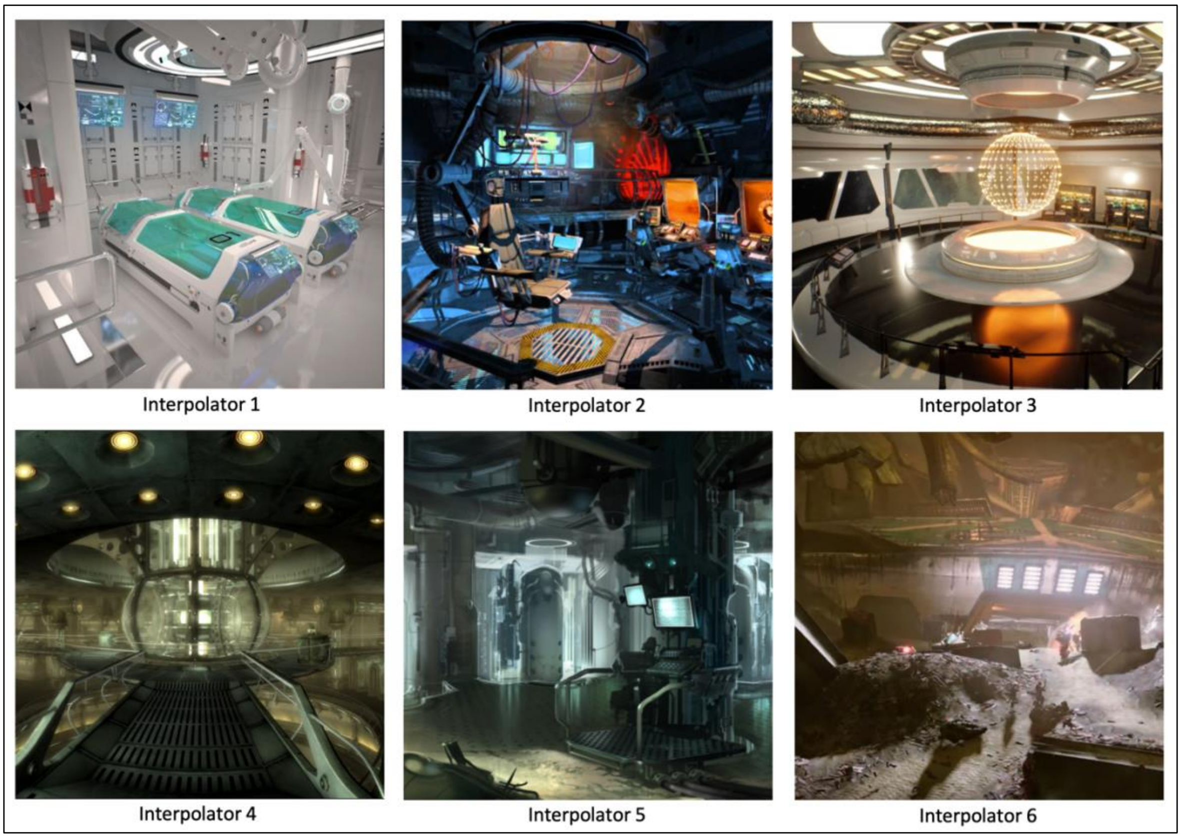


Figure 64 Visual Representations for the Sound Design Tasks

Although it is quite likely for a subjective sound design task that all participants undertaking the task will interpret it differently and hence create unique resulting sounds, the aim was to provide a focus for each scenario. This was done by making one of the preset sounds within the interpolator a sound that this author believed fully met each design brief. This was considered to be a perceptual ideal solution, but only based on the author's estimation. This will clearly be subjective, but it will ensure there is at least one possible solution for each scenario within the parameter space of each interpolator. This provided a possible target for geographic calculations to be performed and allowed the measurement of any possible spread between participants, which would allow some insight into commonalities and differences between them. It will also be possible to analyse if the interpolated sounds could provide subjectively better results than the author's selection. The allocation of the perceptual ideals is given here:

Preset 1 = Manic and chaotic (perceptual ideal for interpolator Interface 2)

Preset 2 = Soothing and healing (perceptual ideal for interpolator Interface 1)

Preset 3 = Threatening and scary (perceptual ideal for interpolator Interface 4)

Preset 6 = Despair and despondency (perceptual ideal for interpolator Interface 5)

Preset 8 = Calm and tranquil (perceptual ideal for interpolator Interface 3)

Preset 10 = Sombre and gloomy (perceptual ideal for interpolator Interface 6)

The remaining four presets were created to provide generic spacecraft engine ambiences that were chosen for their rich sonic nature. For the distribution of the presets within the interpolation space, the same test layout (Figure 51) was used as detailed in Section 5.5.2, where analysis was already undertaken for this layout. The allocation of presets as the perceptual ideals for the specified interfaces were chosen to try to provide an even distribution across the interpolation surface, but without a discernible pattern that the participants might pick up on and interpret incorrectly. However, when the presets were mapped into the interpolation space it was discovered that due to the blending characteristics of the systems that better options existed. Again, although these maybe subjective, the author undertook the defined sound design tasks with the allocated interfaces and the author selected locations were used as the target locations. These locations are shown for each interpolator in Figure 65. Note that unlike the pilot study (Chapter 6), the layouts of the presets were not rotated as these locations are all different and there is no pattern.

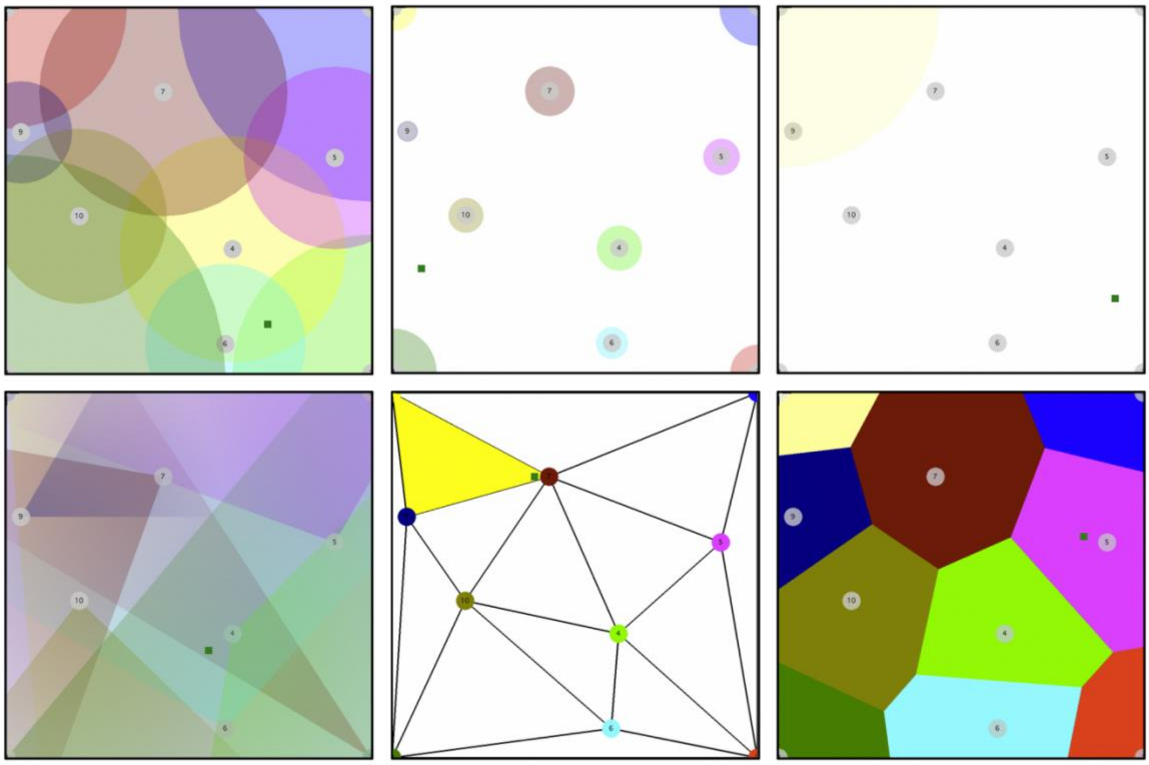


Figure 65 Interpolated Perceptual Ideal Location (■) for each Interpolator[19]

The user-testing took the form of giving the participants the sound design tasks in a random order, where the participants were asked to create a sound that met each supplied brief. The participants were not made aware of the fact that there was a perceptual ideal sound with a fixed location in the interpolation space. The same subtractive synthesis engine (Native Instruments' Massive) was used here as was employed in the pilot study due to the success of the previous experiment. Each interpolator was populated with all ten of the preset sounds defined, (six perceptually ideal sounds, and four generic spacecraft engine sounds).

When the data from the pilot study was analysed, there was no evidence of learned bias being exhibited by the participants when using the interpolators. However, given that the intention of this experiment is to design a sound using an interpolator there was a need to remove any bias from the users learning the principle of interpolator use. So, before the experiment was commenced the participants were given the opportunity to undertake an interpolator training session where they were introduced to interpolator functionality and operation. They were then allowed to use an interpolator until they felt confident they understood its functionality and were

[19] Sonic outputs generated at the perceptual ideal locations for the different interpolation models: https://youtu.be/yQyN2ghkFdQ

secure in its operation. To avoid showing them any of the interfaces being used in the upcoming experiment and creating a bias, Interface 2 (Figure 57) from the pilot study was used for the interpolator learning phase. This interface only showed the preset handles and had no other visual cues.

To simulate a realistic sound design task, before using each interface the participants were given the written scenario for the task, along with the corresponding visual representation. Participants were then free to choose when to initiate the test once they were happy they understood the intention of the task. All participants completed the same sound design tasks with the allocated graphical interface so that comparisons could be made between the participants. As the interfaces were populated with the same presets any resulting sonic differences between the interpolator outputs were purely a function of the different visual models.

Each test lasted a maximum of ten minutes with the participants being able to stop the test beforehand if they felt the task had been completed. If the participants did not complete the task within the allotted time, then the test automatically ended. When the participants felt that they had matched the required sound they were asked to press a "Target" button so the location could be registered. All of the user's interactions with the different interfaces were recorded for analysis. All other aspects of the interpolation system – inputs, interpolation calculations, mappings (all parameters) and synthesis engine – remained identical between the six interfaces. As with the pilot study, the raw interaction data could be analysed to examine differences between the journeys made with each interface.

To assess the perceived usability of each interface, the participants were asked to complete a usability questionnaire. Similar to the previous study the questionnaire was divided into two parts – the first part was completed following the use of each interface and the second part filled-in after all interfaces had been used so they could be compared. However, as detailed in Section 6.4, although SUS did provide confirmation that interpolators were considered as highly usable by the participants, it did not provide enough detail to show perceived differences between the visualizations. Therefore, the standard SUS items were used again, but they were augmented with the addition of four extra items to try and provide greater detail on the interface's different visualisations. The standard ten items were provided first so the results of these could be viewed separately for comparison with either the results for the pilot study or the wider benchmark values and scales (as was done in the pilot study). The additional four items then followed on directly afterwards and were constructed with the same alternating positive and negative wording of the questions [231]. These new items were designed to provide the participants with an opportunity to answer specifics related to the application domain, using the same 5-point

Likert items as the rest of the scale. The questions were derived from those that have been used successfully in other audio technology applications [249]. The new section of the questionnaire is shown in Figure 66. In this way, the standard SUS questions could be employed to provide the usual scale, but it could also be combined with the interpolator specific item scores, or the interpolator specific items could be considered completely separately. In the case of the latter two, the scaling can be adjusted to provide a score from 0 to 100, which is the same as the standard SUS.

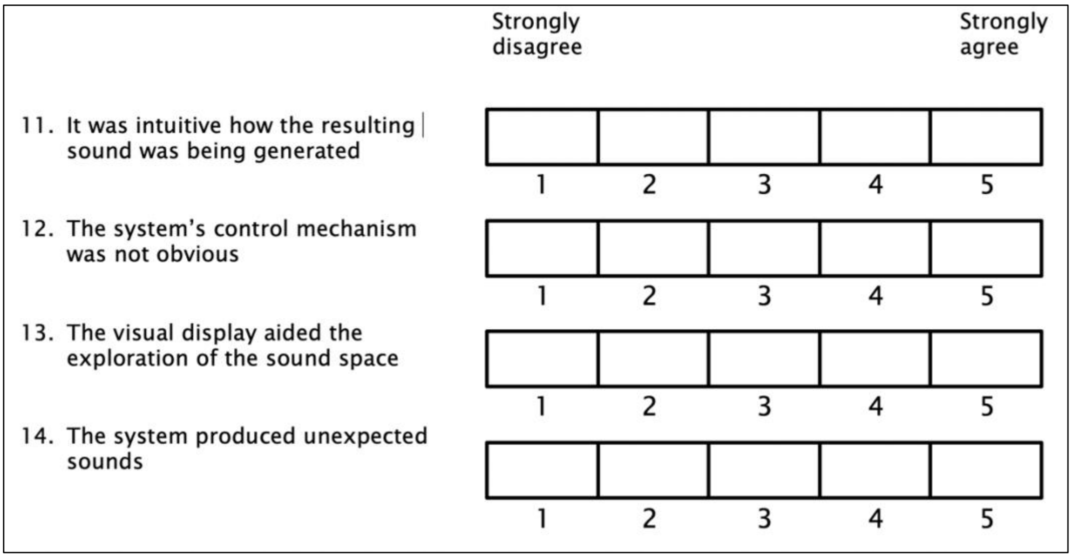


Figure 66 Additional Questionnaire Likert Items Relating to Interpolator Interfaces

Following the completion of the questions the users were asked to write down any critical incidents that occurred while using the interface, both positive and negative. This provided qualitative data and an additional level of detail that could be investigated if the scores from the questions indicated an area of interest with a particular interface.

In the second part of the questionnaire participants were asked to rank the six interfaces with the best being 1st and the least favourite being 6th. They were then asked to rate their preferred interface on a scale 1-10. To try and provide more detail of this rating, they were asked how confident they were they had designed the best possible sound with that interface. They were also asked, to what extent did the preferred system allow them to create sounds that they would not ordinarily have discovered. Both of these questions were answered on the same 1-10 scale. As with the pilot study, to try to understand each participant's level of experience they were asked how many years they have been using music technology and to rate their sound design experience on a scale 1-10. Finally, they were asked to note down any final comments, both positive and negative having completed the experiment.

The relevant ethical approval was sought and granted for the experiment and all the relevant paperwork was generated before participants were recruited and the test was undertaken. All paperwork used for this experiment can be found in Appendix B.

7.3 Usability Experiment Results

The desired number of participants for the experiment was set at thirty-six, based on a power assumption of 0.8 and the desire to observe a medium effect size (0.1758631) [255], as it was believed that given the similarity between the interface's visuals the results were likely to be closer than observed in the pilot study. However, given the 2020/2021 Covid-19 pandemic and subsequent restrictions that occurred mid-way through the experiment, it was not possible to reach this number of participants and the current number of participants stands at twenty. Given that to observe a large effect size (0.2294157) with the same power assumption would require twenty-two participants, it is not an ideal situation as even this number has not been met, but it is unavoidable given the situation. All of the participants recruited had some degree of sound design experience and all their interactions with the interfaces were captured via the recording of mouse movements. As with the pilot study, this allowed traces of the movements to be visually compared between the different interfaces. The trace gives a pictorial representation of the journey that each user made through the interpolation space. An example is shown in Figure 67 for participant 1 who had the following interface order – 6, 2, 3, 5, 1 & 4. Here they are shown Interface 1 – 3, left to right on the top row and Interface 4 – 6, left to right on the bottom. The traces have been colour coded again so the first third of the trace is red, the next third is blue and the final third is green.

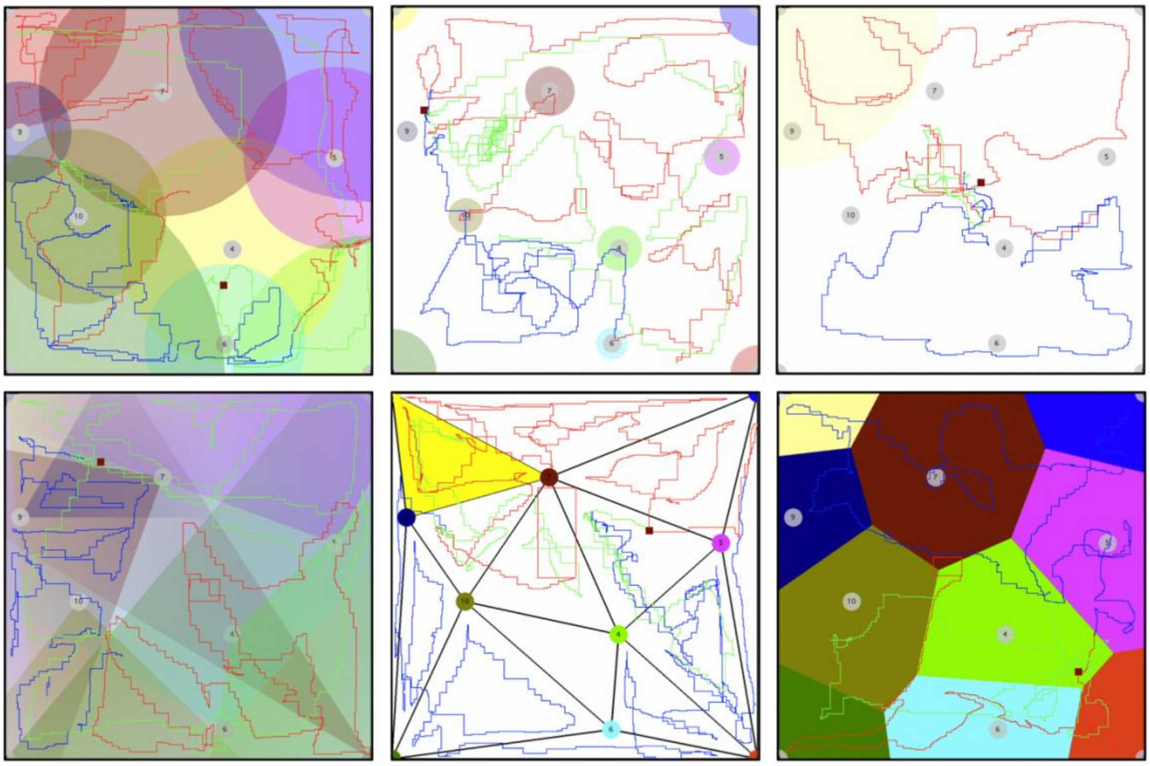


Figure 67 Mouse Traces for Participant 1 - Showing Top Row Interface 1 – 3 and Bottom Row Interface 4 – 6 and the Participants Chosen Location (■)[20]

From observation of the traces, it was seen that while exploring the space the participants appeared to follow the same trend of three distinct phases of interpolation as was identified in the pilot study: making large fast moves at the beginning while exploring the space, localising on regions-of-interest, but occasionally checking if better options exist and then refining the sound through slow small movements in the space. Although this trend does not always split evenly into thirds of time used for the depiction many of the participants appear to follow this trend. In addition, it was noted from the traces that sometimes confirmatory moves are made in the refinement phase or slower moves are made in the exploration phase when interesting results are found. This was also observed by viewing a plot of the cursor speed over the duration of the exploration. An example of this is shown in Figure 68 for participant 7 for the first test they undertook within the experiment.

[20] Interpolator usability study mouse traces for participant 1: https://youtu.be/ZcQoxI1YCf4

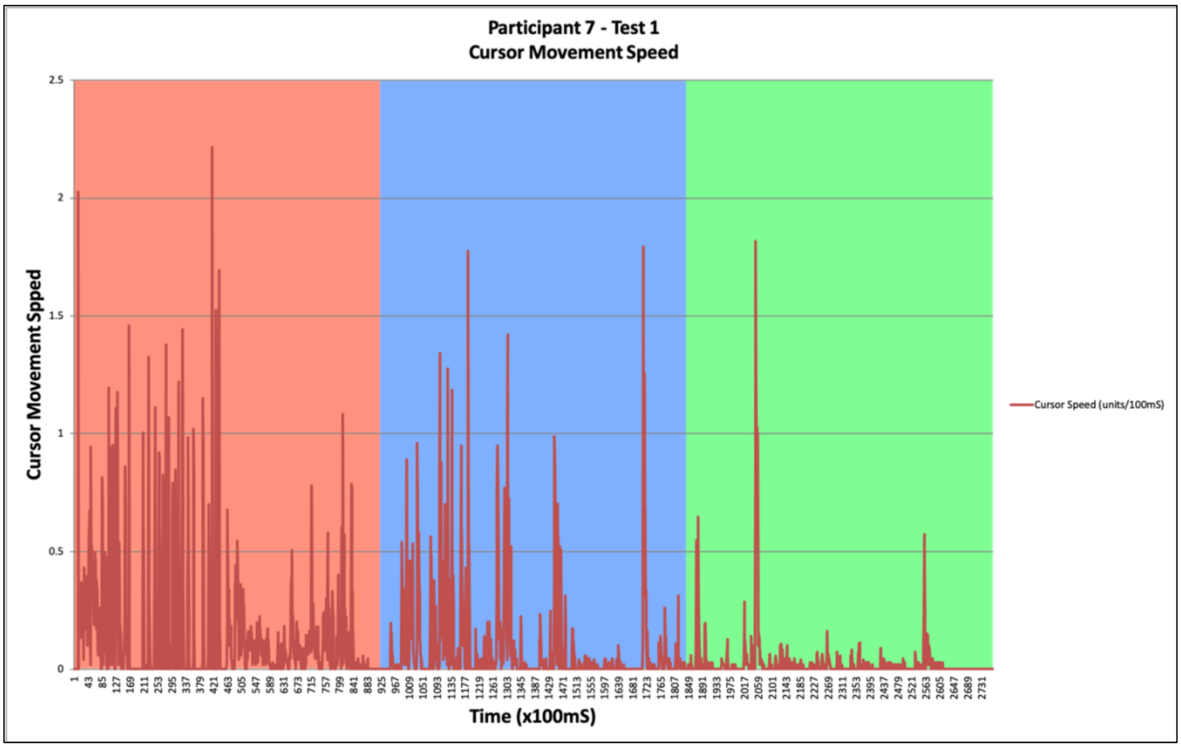


Figure 68 Mouse Speed - Sampled Every 100mS for Participant 7 with Interface 5

As can be seen in this plot during the first phase this participant did find an area that caused their movements to slow so that smaller distances were travelled. Also, during the final refinement phase, a couple of medium speed moves were made as are normally seen when the participant is localising on regions-of-interest. Despite these anomalies the trend of three phases when using the interpolators appears to hold. To confirm this observation, for every test undertaken by all participants, the mean cursor movement speed and mean number of high-speed moves were calculated for the three phases by dividing the total test time by three. Note that a high-speed move was defined as greater than 0.5 units/100mS. This value was chosen as these moves showed as medium spikes on the mouse speed plots, such as in Figure 68 and Figure 69. The results of these calculations are shown in Table 8.

	Mean Cursor Speed (Standard Deviation)	**Mean High-Speed Moves (Standard Deviation)**
Exploration Phase	1.373 units/sec (SD = 0.607)	40.28 mvs (SD = 38.76)
Localisation Phase	0.981units/sec (SD = 0.433)	23.94 mvs (SD = 17.04)
Refinement Phase	0.565 units/sec (SD = 0.422)	13.62 mvs (SD = 14.92)

Table 8 Mean Cursor Speed and Number of High-Speed Moves for The Three Interpolation Phases

As can be seen from these results there is a difference between the interpolation phases with the mean cursor speed and number of high-speed moves decreasing as the participant gets closer to their selected location.

It should also be noted that a few participants would exhibit different modes of operation when using the interface. For example, participant 13 adopted a strategy where they undertook their exploration of the space and then afterwards they would click on different locations, causing the cursor to jump and audition the sound at these alternative locations. An example of this is shown in Figure 69 for participant 13 on the second test, but by observing all of this participants test's it was seen they continued this strategy for all of the interfaces, to a greater or lesser extent. However, aside from the cursor jumps at the end of each test, the participant still appeared to explore the space in the same manner. Occasionally other participants also exhibited their own unique modes of operation when using the interpolators, although they were not always as prominent as this example. However, it adds strength to the notion that some users may have distinctive search strategies that they tend to use with all of the interpolator interfaces regardless of the visual display.

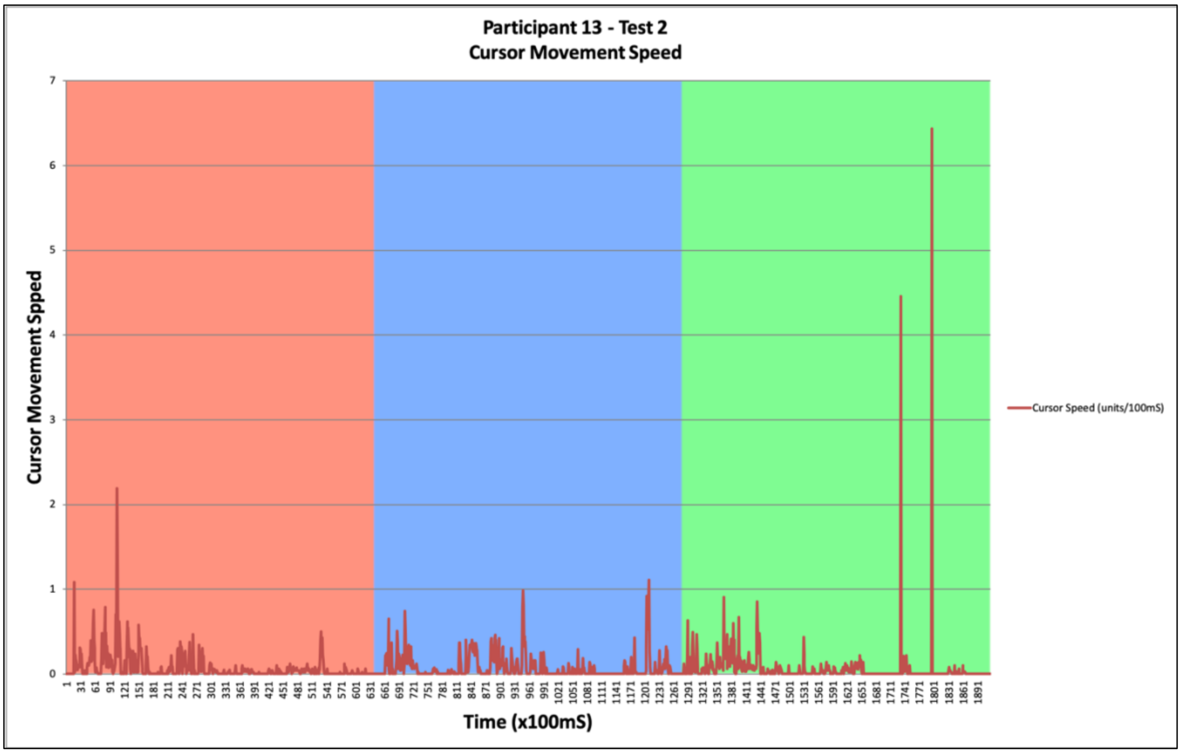


Figure 69 Mouse Speed - Sampled Every 100mS for Participant 13 with Interface 3

As with the pilot study, for each interface the mean cursor location of the mouse traces and the standard distance deviation was calculated. This was examined by interface, with the mean and standard deviation of the standard distance deviation calculated, based on the unit square size of the interfaces and are shown in Table 9.

	Mean Standard Distance Deviation (Standard Deviation)
Interface 1	0.397 units (SD = 0.051)
Interface 2	0.387 units (SD = 0.042)
Interface 3	0.412 units (SD = 0.053)
Interface 4	0.380 units (SD = 0.057)
Interface 5	0.434 units (SD = 0.067)
Interface 6	0.467 units (SD = 0.066)

Table 9 Mean Standard Distance Deviation by Interface for Mouse Trace Movements

As can be seen, all of the interfaces generated similar values with little obvious difference between them, both in terms of the mean value of standard distance deviation and its standard deviation. This is different to the results from the pilot study as all the interfaces resulted in a wider distribution for the interpolation space traces. This adds further weight to one of the conclusions drawn from the pilot study that the visual cues provided by the interface result in the participants exploring a wider area. It was also noted that for the nodes interface which was common between both this experiment and the pilot study there was a large difference. In the pilot study the nodes interface (Interface 3) resulted in a mean Standard Distance Deviation 0.180 units (SD = 0.21), whereas in this experiment (Interface 1) the mean increased to 0.397 units (SD = 0.051). Moreover, all the interfaces in this experiment resulted in higher means. This maybe the result of the experiment's having different goals for the participants. In the pilot study the task was essentially sound identification, where the target sound had a specific location within the space. The task in this study was deliberately made to be more like a real sound design activity where the participants explored the space to see if they could find a location that resulted in a sound that they thought met the brief. As a result, in the pilot study as soon as the participants thought they had located the sound they were more likely to end the test and cease exploration. Whereas in this experiment as the participants made the judgement on the suitability of the sound, there may have been a greater likelihood that they would explore more possibilities before making a final decision. This can be seen in the traces (shown in Figure 67) where often the chosen location or close proximity to it have been visited a number of times during the experiment. It is also noted that the two interfaces with the highest desirability from the dimension space analysis, tessellation (Interpolator 6) and triangulation (Interpolator 5), got the two highest scores for the standard distance deviation. Similarly, the interface with the lowest standard distance deviation was one of the two interfaces that the dimension space analysis showed to be the least desirable. This appears to show that the interfaces that possess more visual cues for the participants to use for navigation result in a larger area being explored.

The locations the participant selected as their chosen sounds were also plotted to see if there were any trends resulting from the different interfaces. Figure 70 shows the selected sound locations for all the participants, by interface. Given the subjective nature of the sound design task it is no surprise that the task resulted in a much wider distribution of locations than was the case in the pilot study. Nonetheless, from inspection it does appear that there is some clustering of selected locations within the space. This may indicate that despite the subjective nature of sound design, there are common sonic traits that the participants identified for each scenario. However, given the low number of participants this could just be natural variation.

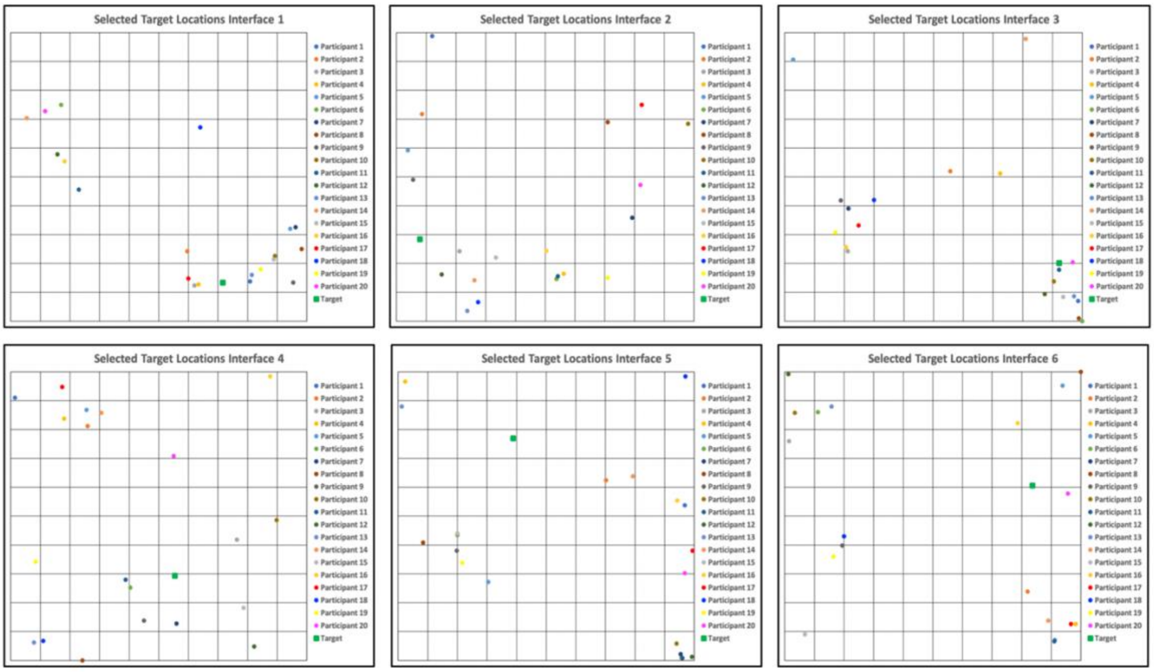


Figure 70 Participants Selected Target Locations by Interface and the Location of the Target Sound (■)

From the results shown in Figure 70 the standard distance deviation was calculated with respect to the mean selected location. Hence this provides a basic measure for the distribution of selected locations. The calculated standard distance deviation values, with respect to the unit square of the interface, are shown for each interface in Table 10.

	Standard Distance Deviation
Interface 1	0.385 units
Interface 2	0.394 units
Interface 3	0.437 units
Interface 4	0.441 units
Interface 5	0.488 units
Interface 6	0.523 units

Table 10 Standard Distance Deviation of Participant Selected Locations by Interface

It is interesting to note from these values there is again an apparent correlation to the dimension space analysis that was undertaken in Section 7.1.1. Interface 1 resulted in the lowest distribution of selected locations (Table 10) and, as was shown in Table 9, the participants also explored less of the space. The dimension space analysis showed that this interface provided the participants with fewer visual cues in contrast to Interfaces 6 and 5 which both provided more visual cues and respectively resulted in a larger distribution of selected locations and the users explored a larger area of the space. These results could be taken to mean that the interfaces that provided more visual cues encouraged the participant to explore more of the sonic space, providing more choice of sonic results.

It was also noted from the plots that there did not appear to be much commonality between the selected location with each interface, showing the participants clearly found multiple sounds that were unique enough to meet the brief. Also, in most cases the locations were not considered to be suitable to meet multiple briefs. This is shown in Figure 71 where all the selected locations were plotted by interface.

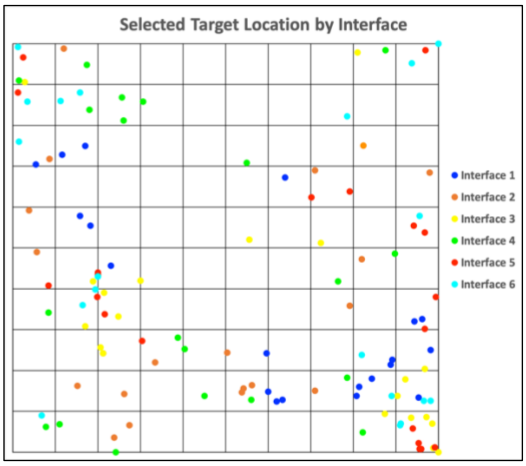


Figure 71 Participant Selected Locations shown by Interface

Moreover, where the locations selected with different interfaces are in close proximity of each other, as each interface generates its own unique sonic pallet (as highlighted Section 5.5.2) it still results in the individual locations generating distinctively different outputs. This was confirmed by checking the sonic result for each target location and comparing it with other selected locations found with that interface and also locations within close proximity (a radius < 0.05), chosen with different interfaces. In all cases there were audible sonic differences, particularly between interfaces, adding further confidence to the observation that each interface has a unique sonic identity. It is also worth noting from the plot shown in Figure 71 that there were areas where no sounds were selected with any interface. These areas have not been selected as they were deemed by the participants to not meet any of the sound design briefs. Interestingly these areas correlate strongly with the locations for the generic engine sound presets (4, 5, 7 & 9).

7.3.1 Mouse Movement Significance Testing

As with the pilot study, NHST was undertaken to establish if there was a significant difference for time, speed, distance and accuracy with each interface. Like the pilot study, in all cases, the data was tested for normality (Skewness/Kurtosis, Shapiro-Wilk and visual inspection) and found again to lack a normal distribution so non-parametric statistical methods were used. As this has occurred in both studies it adds further weight to the idea that this is related to the exploratory nature of sound design tasks, rather than being a random anomaly. As it was found the data did not possess a normal distribution, the same methodology was used as in the pilot study and a Friedman test was used to determine if there were significant differences between the ranks of the interfaces. If a significance was shown, then post-hoc testing was undertaken with pairwise Wilcoxon signed-rank tests to identify where the significances were present. The effect size was then calculated with correlation coefficient (*r*) [240] and probability score depth (*PSDep*) [241].

7.3.1.1 Time, Speed & Distance

The Friedman test was undertaken for the total cursor movement time, average cursor speed and total distance cursor moved. In all three of cases the Friedman test showed that there was no significant difference in these variables between the interfaces. The results of these tests are shown in Table 11 with the median value for each interface.

Variable	Test: Friedman Test	Median (IQR)	Significance
Time	$\chi^2(5) = 4.886$	$Median_1$ = 100.70 secs (IQR = 132.75 secs – 35.80 secs) $Median_2$ = 49.45 secs (IQR = 85.27 secs – 41.00 secs) $Median_3$ = 80.00 secs (IQR = 132.67 secs – 52.30 secs) $Median_4$ = 81.55 secs (IQR = 109.45 secs – 48.45 secs) $Median_5$ = 68.00 secs (IQR = 128.47 secs – 37.60 secs) $Median_6$ = 63.90 secs (IQR = 142.350secs – 37.02 secs)	p = 0.430
Speed	$\chi^2(5) = 6.714$	$Median_1$ = 0.879 units/sec (IQR = 1.244 units/sec – 0.713 units/sec) $Median_2$ = 1.076 units/sec (IQR = 1.270 units/sec – 0.814 units/sec) $Median_3$ = 0.955 units/sec (IQR = 1.326 units/sec – 0.711 units/sec) $Median_4$ = 0.900 units/sec (IQR = 1.046 units/sec – 0.638 units/sec) $Median_5$ = 0.873 units/sec (IQR = 1.015 units/sec – 0.715 units/sec) $Median_6$ = 0.915 units/sec (IQR = 1.234 units/sec – 0.736 units/sec)	p = 0.243
Distance	$\chi^2(5) = 4.429$	$Median_1$ = 19.30 units (IQR = 28.75 units – 10.36 units) $Median_2$ = 15.29 units (IQR = 21.10 units – 11.93 units) $Median_3$ = 15.45 units (IQR = 25.91 units – 10.86 units) $Median_4$ = 14.39 units (IQR = 19.67 units – 9.63 units) $Median_5$ = 15.23 units (IQR = 23.83 units – 11.77 units) $Median_6$ = 15.45 units (IQR = 23.31 units – 11.43 units)	p = 0.489

Table 11 Results of the Statistical Testing of Differences in Time, Speed and Distance for Each Interface

Although the interfaces present the participants with different visual cues, in these tests there is an absence of evidence that they had a significant impact on the time, speed or distance when using the interfaces.

7.3.1.2 Accuracy

The same methodology was used to compare the ranks for the distance to the mean target location with each interface. Using the Friedman test there was a statistically significant difference between the interfaces during the experiment, $\chi^2(5) = 20.114$, $p < 0.001$. As a result, post-hoc testing was undertaken in the form of pairwise comparisons using Wilcoxon signed-rank test with a Bonferroni correction for multiple comparisons (fifteen). These allowed the identification of where there was a statistically significant difference between the interfaces. This correction method was chosen as it allowed the same tests to be applied as were used in the pilot study, aiding comparisons to be made between the results from both studies. Making the Bonferroni correction, statistical significance was accepted as $p < 0.0033$. The Wilcoxon signed-rank tests determined that there was only one statistically significant difference in the distance to target (Median Difference = 0.223 units, Inter-Quartile Range (IQR) = 0.316 units – (-0.006) units), when subjects used Interface 6 ($Median_6$ = 0.534 units, IQR = 0.597 units – 0.399 units) compared to Interface 1 ($Median_1$ = 0.319 units, IQR = 0.400 units – 0.231 units), $Z = -2.949$, $p < 0.0032$). The effect sizes ($r = -0.466$, $PS_{Dep} = 0.75$) showed a medium effect, where 75% of the participants taking the experiment saw an increase in selecting a target location between Interface 1 and Interface 6. This appears to indicate that there was more convergence between participants when selecting a target location with Interface 1, compared to Interface 6. Moreover, when using the normal conventions [236], it appears to indicate that this is a medium effect size, but approaching the large effect size threshold of 0.5. For completeness the median values for all interfaces are presented in Table 12.

Variable	Test: Friedman Test	Median (IQR)	Significance
Accuracy	$\chi^2(5) = 20.114$	$Median_1$ = 0.319 units (IQR = 0.400 units – 0.231 units) $Median_2$ = 0.367 units (IQR = 0.440 units – 0.258 units) $Median_3$ = 0.416 units (IQR = 0.454 units – 0.333 units) $Median_4$ = 0.444 units (IQR = 0.515 units – 0.350 units) $Median_5$ = 0.410 units (IQR = 0.548 units – 0.377 units) $Median_6$ = 0.534 units (IQR = 0.597 units – 0.399 units)	p = 0.001

Table 12 Results of the Statistical Testing of Distance to selected Target by Interface

7.3.2 Interpolation Phases

As already identified, from this study there was once again a strong suggestion that the participants use the interfaces differently during three distinct phases of the interpolation. As the phases have different user characteristics, as identified from the pilot study, NHST was undertaken to establish if there was a significant difference in cursor speed or number of high-speed moves made between the different phases. To assess this the mouse data from all the tests (twenty participants, undertaking six test each, giving one hundred and twenty total) was divided into thirds of the total test time for the three interpolation phases. The average cursor speed and the number of high-speed moves were calculated, where a high-speed move was defined as greater than 0.5 units/100mS. This value was chosen as the moves showed as medium spikes on the mouse speed plots, such as in Figure 68 and Figure 69. The data was tested for normality (Skewness/Kurtosis, Shapiro-Wilk and visual inspection) and again found to lack a normal distribution so non-parametric statistical methods were used.

It was hypothesized that during the first phase (exploration) the participants would have a higher average cursor movement speed and make more high-speed moves. These would then both reduce during the second phase (localisation) and then again during the final (refinement) phase (H_A: $Median_1 > Median_2 > Median_3$). Thus, the null hypothesis was that the different phases had no effect on the cursor speed or number of high-speed moves (H_0: $Median_3 = Median_2 = Median_1$).

A Friedman test was used to establish that there was a statistically significant difference between the phases for the average cursor speed and the number of high-speed moves, $\chi^2(2) = 120.5167$, $p < 0.001$ and $\chi^2(2) = 82.5489$, $p < 0.001$ respectively. As a result, post-hoc testing was undertaken in the form of pairwise comparisons using Wilcoxon signed-rank test with a Bonferroni correction for multiple comparisons. The results from the pairwise comparisons are shown in Table 13

Variable	Test: Wilcoxon	Median (IQR)	Significance	*r*	PS_{Dep}
Speed	Z = -6.408	$Median_1$ = 0.130 units/sec (IQR = 0.170 units/sec – 0.095 units/sec)	p < 0.001	-0.414	0.742
	Z = -8.771	$Median_2$ = 0.093 units/sec (IQR = 0.123 units/sec – 0.072 units/sec)	p < 0.001	-0.566	0.917
	Z = -7.466	$Median_3$ = 0.048 units/sec (IQR = 0.077 units/sec – 0.028 units/sec)	p < 0.001	-0.482	0.833
High-Speed Moves	Z = -5.256	$Median_1$ = 26.50 mvs (IQR = 50.75 mvs – 14.50 mvs)	p < 0.001	-0.339	0.667
	Z = -7.772	$Median_2$ = 20.00 mvs (IQR = 34.00 mvs – 11.25 mvs)	p < 0.001	-0.502	0.85
	Z = -6.079	$Median_3$ = 9.50 mvs (IQR = 18.00 mvs – 3.00 mvs)	p < 0.001	-0.392	0.742

Table 13 Results of the Statistical Testing of Interpolation Phases for Mouse Speed and Number of High-Speed Moves

In all cases significance was shown with a medium to large effect size. This shows that there are differences in how the users interact with the interfaces during the different phases of the interpolation.

7.3.3 Usability Questionnaire

First the standard SUS scores were analysed for the six different interfaces so that comparisons could be made to the pilot study results and the published benchmarks, already detailed in Section 6.2.2. The descriptive statistics for the standard SUS scores by interface are presented in Table 14.

	Interface 1	Interface 2	Interface 3	Interface 4	Interface 5	Interface 6
Mean	82.00	70.75	73.00	71.50	84.50	78.00
Standard Error	3.666	4.176	3.824	4.858	3.268	4.070
Median	86.25	75.00	75.00	77.50	88.75	85.00
Standard Deviation	16.396	18.675	17.103	21.725	14.613	18.202
C.I. (95%)	7.673	8.740	8.004	10.1676	6.8393	8.519

Table 14 Usability Study Descriptive Statistics for Standard SUS Scores by Interface

As can be seen from these results, there is a much wider range of SUS scores here than was seen in the pilot study. This appears to show that an interpolator's visual model does have an impact on the perceived usability of the system. Based on this hypothesis, due to the data failing normality testing, a Friedman Test was used with post-hoc pairwise comparisons using Wilcoxon signed-rank test with a Bonferroni correction. The Friedman test showed a statistically significant difference between the interface's SUS scores in the experiment, $\chi^2(5) = 20.304$, $p = 0.001$. However, when the pairwise comparisons were undertaken, no significant results were obtained meaning that it was not possible to identify where the differences existed. This is likely to be a result of multiple factors such as the small sample size, the reduced sensitivity of non-statistical methods and the conservative nature of the Bonferroni correction. From these results it appears that the perceived usability of the interpolators is affected by the visualisation, but it has not been possible to identify any significant differences between individual interfaces. However, differences can be seen in the descriptive statistics (Table 14) between the interfaces and in all cases the SUS scores are higher than average of 68 from the database analysis [234]. To interpret these results the mean SUS scores of each interface have been mapped to the different metrics that have been used to categorize systems tested with SUS in the past [242]. These are summarised in Table 15.

	Interface 1	**Interface 2**	**Interface 3**	**Interface 4**	**Interface 5**	**Interface 6**
Mean SUS Score	82.00	70.75	73.00	71.50	84.50	78.00
Grade	A	C	B-	C+	A+	B+
Percentile range	90-95	41 – 59	65 – 69	60 – 64	96-100	80-84
Adjective	Excellent	OK	Good	Good	Best Imaginable	Good
Net Promoter Score	Promoter	Passive	Passive	Passive	Promoter	Passive

Table 15 Average SUS scores by Interface Associated with Percentiles, Grades, Adjectives, and NPS categories

From this it can be seen that Interface 5 and 1 perform the best and are comparable to the top 10% of all systems tested within the database. It is also worth noting that Interface 6 although being graded as B+, the means SUS score is only just below the next grade boundary (78.9) which would give it an A grade as well. This grade band also matches up with the Net Promoter Score (NPS) that suggests that users tend to be promoters and likely to recommend the system [242]. The results for these interfaces are comparable to those obtained in the pilot study. The final three interfaces (2, 3 & 4) all perform above the average but are not perceived as being as usable as interfaces 1, 5 & 6. They are also perceived as being worse than all the interfaces in the pilot study. Given that in the pilot study two of the interfaces had limited visual cues it may suggest that the visuals for these interfaces are hindering the sound design process.

As detailed in section 7.2, the standard SUS was extended with the addition of four items that were intended to provide extra detail about the graphical interpolators. These were firstly combined with the standard SUS scores to see if they provided a more detailed view of the different interpolator's usability. Table 16 shows the descriptive statistics for these extended SUS scores.

	Interface 1	**Interface 2**	**Interface 3**	**Interface 4**	**Interface 5**	**Interface 6**
Mean	76.87	65.54	67.86	69.11	79.11	73.93
Standard Error	3.414	3.953	3.847	4.588	3.351	3.913
Median	80.36	71.43	69.64	71.43	83.04	80.36
Standard Deviation	15.268	17.679	17.206	20.517	14.986	17.500
C.I. (95%)	7.146	8.274	8.053	9.602	7.014	8.190

Table 16 Usability Study Descriptive Statistics for Extended SUS Scores by Interface

As can be seen from these results, the average scores for all interfaces reduced in comparison to the standard SUS, but they appear to parallel the previous results. To confirm this the same testing methodology as applied to the standard SUS was also used for the extended SUS scores. The Friedman test showed a statistically significant difference between the interfaces during the experiment, $\chi^2(5) = 19.579$, $p = 0.001$, but the pairwise comparison with Wilcoxon tests showed no significant results. These results parallel those for the standard SUS. It was hoped that the extended questions would shed further detail on the usability of interpolators, specifically for sound design, however, it would appear that they were not successful in this respect. The standard SUS provides very similar results and has the added benefit of providing scales that aid comparison to sector wide norms. For completeness, Table 17 shows the descriptive statistics for just the additional SUS questions.

	Interface 1	Interface 2	Interface 3	Interface 4	Interface 5	Interface 6
Mean	64.06	49.37	55.00	63.125	65.62	63.7500
Standard Error	4.027	5.108	5.433	4.688	5.297	4.725
Median	71.87	50.00	56.25	62.50	68.75	68.75
Standard Deviation	18.009	22.843	24.299	20.97	23.69	21.132
C.I. (95%)	8.428	10.691	11.372	9.812	11.088	9.890

Table 17 Descriptive Statistics for Additional Question Scores by Interface

The second part of the questionnaire was analysed using the same methodology as was adopted in the pilot study. First the participants were asked which of the interfaces they preferred using and the results are shown in the frequency table in Table 18.

	Frequency	Percent	Cumulative Percent
Interface 1	6	30	30
Interface 3	1	5	34
Interface 4	1	5	40
Interface 5	8	40	80
Interface 6	4	20	100
Total	20	100	

Table 18 Usability Study Frequencies for User Preferred Interface Choice

Of the participants, 40% preferred using Interpolator 5 (triangulation), 30% chose Interpolator 1 (nodes), 30% selected Interpolator 6 (tessellation) and the remaining 10% were split between Interface 3 (radius-based) and Interface 4 (light). None of the participants selected Interface 2 (gravitational). These results appear to exactly mirror the results from the SUS scores providing greater confidence in the results. The mean ranks were also calculated for each interface, based on the preferred interface rank order selected by each participant. These are shown below in Table 19.

	Mean Rank
Interface 1	2.55
Interface 2	4.72
Interface 3	4.57
Interface 4	3.55
Interface 5	2.55
Interface 6	3.05

Table 19 Usability Study Mean Ranks for Preferred Interface Choices

From the results, it can be seen that Interface 1 & 5 are considered the preferred, with identical mean ranks. Next is Interface 6 followed by Interface 4 and then there is a large increase for Interfaces 3 & 2. This provides more detail to the participants selection of preferred interfaces and shows a clear predilection for two of the interfaces. A Friedman test was used to show that there is a statistically significant difference in the rank order between the interfaces, $\chi^2(5) = 26.856$, $p < 0.001$. Pairwise comparisons were then undertaken using Wilcoxon signed-rank tests with a Bonferroni correction, to identify where the significance was present. Significance was shown for the following cases with an accepted p-value of $p < 0.0033$:

Interface 2 – Interface 1 ($Z = -2.967$, $p = 0.003011$)
Interface 3 – Interface 1 ($Z = -2.986$, $p = 0.002825$)
Interface 2 – Interface 5 ($Z = -3.231$, $p = 0.001232$)

The Bonferroni correction has a reputation for being fairly conservative and other methods may have shown significance in more cases [256], but it was adhered to in alignment with all of the previous tests.

As with the pilot study, the participants were asked "How many years have you been using music technology" to gauge their experience levels. The data was evaluated with respect to their choice of preferred interface giving the results shown in Table 20.

	Interface 1	Interface 2	Interface 3	Interface 4	Interface 5	Interface 6
Mean	9.00	0.0	5.0	2.0	8.25	9.00
Standard Error	0.516	0.0	0.0	0.0	0.250	0.707
Median	9.50	0.0	5.0	2.0	8.00	9.50
Standard Deviation	1.265	0.0	0.0	0.0	0.707	1.414
C.I. (95%)	1.330	0.0	0.0	0.0	0.590	2.250

Table 20 Usability Study Descriptive Statistics for Music Tech. Experience by Interface

These results show that users who preferred the three most selected interfaces had very similar levels of experience using music technology with an average close to 9 years. As opposed to the other three interfaces with lower participant experience levels – 5 years for Interface 3 (radius-based), 2 years for Interface 4 (light) and no one selected Interface 2. As already identified, most participants selected Interface 5 as their preferred and it has an average participant experience level of 8.25 years with a Standard Deviation of 0.707. Interfaces 1 and 6 have larger Standard Deviations of 1.265 and 1.414 respectively, indicating a slightly wider spread of values. However, given the sample size between the groups, it is difficult to read too much into these results and as such they should be considered inconclusive. When all the participants responses were analysed together it did confirm that the participants had a range of different experience levels and there appears to be little bias and are like the results obtained from the pilot study. Nonetheless, the sample size (20) should be noted as a larger sample may show other trends. Table 21 shows the results of this analysis.

	Music Tech. Experience
Mean	9.20
Standard Error	2.108
Median	6.50
Standard Deviation	9.429
C.I. (95%)	4.410

Table 21 Usability Study Descriptive Statistics for Music Tech. Experience

To gauge the participants satisfaction with the sound they designed using their selected preferred interface, they were asked “how confident were you that you were able to design the best possible sound from the given starting points” and “to what extent did the preferred system allow you to create sounds that you would not ordinarily have discovered". This data was analysed for all the participants and the result are shown in the Table 22.

	Confidence in Sound Design	Unexpected Results
Mean	7.80	8.50
Standard Error	0.296	0.303
Median	8.00	9.00
Standard Deviation	1.322	1.357
C.I. (95%)	0.620	0.640

Table 22 Usability Study Descriptive Statistics for Confidence in Resulting Sound Design

From these results it can be seen that the participants appeared to be fairly confident with the sounds they were able to create with their preferred interface with a mean score of 7.80. Yet the participants gave higher scores (mean 8.50) to the question that the interfaces provided the ability to create sounds they would not ordinarily have discovered. These results were also evaluated with respect to their choice of preferred interface, but the results showed little variation was present. Again, this is probably a result of the small sample size, and a larger number of participants may have yielded more detail in the results.

Analysis of the critical incidents noted by participants, both positive and negative, allowed an insight into why the participants responded the way they did during this study. These will be presented in order, based on the participants preferred interface, from the least popular to the most. Interface 2 (Gravitational) was not selected by any of the participants as their preferred interface and by examining the comments it appears that the main reason for this was the lack of visual cues provided by the interface. Positive comments about the interface related to the free space providing "more freedom for searching and exploring to create sounds". However, other comments related to lack of visual feedback to indicate difficulty judging "where the sounds were coming from".

Interfaces 3 and 4 were each selected by a single participant as their favoured choice. Interface 3 (radius-based) was criticised for a lack of precision, resulting in sounds morphing "very quickly as [the] cursor moved…..making it harder to blend some of the sounds". This appears to have "made it hard to pinpoint a sound" with this interface. Another participant also commented that the interface's "visual feedback was focussed on the mouse location [rather] than the sound's origin". Interface 4 (light) was acclaimed to provide an interface where the "visual block colours made clear definitions where the sounds began to change and differ". However, it also received more comments that the interface was not clear and "the purpose [of the overlaps] was not immediately obvious". In addition, comments were made about the manner in which the sounds changed when controlled with this interface, with comments such as "[sound changes] between shapes felt very abrupt and creatively limiting".

Interface 6 (tessellation) was the third most popular interface with selection by four participants and received mainly positive feedback, even from participants that did not select it as their preferred. It was praised for ease of use and the intuitive interface that made it "obvious which portions of the different area were being used to generate the sounds". It was also commented that the interface was providing more visual cues with comments about it being more "informative regarding the sound I thought I was making".

Interface 1 (nodes), selected by six participants, received many comments about being very familiar and obvious to use as a result of the overlapping circles. This is not surprising as the nodes object is freely available in the Max programming environment and given the high experience level of the participants it is likely that a number may have come across this paradigm previously. Comments supported this notion with phrases such as "clear representation of each sounds location [within the space]" and "overlaps make sense so [the users] can focus on the qualities of the sound".

Interface 5 (triangulation) was the most popular interface chosen with eight of the twenty participants stating it as their preferred and participants that did not select it still made positive comments. They appear to have found the "colours and triangulation helpful in navigating the space" and intuitive to use. Observations were also made about the interface providing additional visual cues with comments like "the extra visual elements helped to guide me in where the sonic changes where [sic] being blended which allowed me easier movement towards the desired sounds". Additionally, some participants stated "certain sounds were unexpected", but yet "easy to predict where the sounds were going". This appears to fit with the underlying results from the questionnaire.

The general comments at the end of the study also back up a number of points already touched upon here and in the pilot study. The ease of use came up many times, with one participant stating that the interpolators "make [sound] design a much easier process, especially for prototyping sounds". Again, many of the participants found sound design with the interpolators "enjoyable" and stated the activity was "fun". A couple of participants also identified that the interfaces "helped [them] come across possibilities [they] would not normally" have found. This aspect was eloquently summarised by one participant as the interpolators provided a "balance between control and serendipity" when designing sounds. Several participants also commented that more visual cues may provide further enhancement to the systems and a number of suggestions were made. The most popular was the request for the inclusions of “markers” so that interesting locations could be tagged to aid being able to return to areas of sonic interest without having to rediscover them. Another participant commented that maybe the preset location could provide “more feedback on the sounds” to aid the sound design process.

7.4 Usability Study Evaluation

Although this study was limited to twenty participants it is still a reasonable number when compared to other studies in the music/sound technology area. As noted from the pilot study, formal usability testing of interpolators has not been undertaken and in the sector where usability testing has been performed, it is often only done with smaller numbers [244-247]. Although the number of participants was higher than the pilot study, this time it was not possible to show a significant difference between the interfaces for time, speed and distance. Even where significance was shown for accuracy it was only possible to show significance for one pairwise case. This also appears to be corroborated by examining the descriptive statistics where there does not appear to be any obvious trends. For completeness the descriptive statistics for all four parameters are presented in Table 23.

		Interface 1	Interface 2	Interface 3	Interface 4	Interface 5	Interface 6
Time (seconds)	**Mean**	89.83	67.57	91.5	84.7	79.64	91.05
	Median	100.7	49.45	80	81.55	68	63.9
	Range	171.2	168.4	170.1	164.9	139.2	243.1
	Min	7.8	17.3	27.4	9.7	21.7	23
	Max	179	185.7	197.5	174.6	160.9	266.1
	IQR	96.95	44.27	80.37	61.00	90.87	105.33
Speed (units/sec)	**Mean**	1.020	1.101	0.979	0.853	0.905	0.960
	Median	0.877	1.076	0.955	0.900	0.873	0.915
	Range	1.677	1.268	1.075	0.983	0.948	1.572
	Min	0.537	0.583	0.431	0.338	0.533	0.333
	Max	2.214	1.851	1.506	1.322	1.481	1.905
	IQR	0.531	0.456	0.615	0.408	0.300	0.498
Distance (units)	**Mean**	20.61	18.11	18.92	17.74	16.81	17.92
	Median	19.30	15.29	15.45	14.39	15.23	15.45
	Range	49.38	44.82	40.23	47.28	23.45	50.91
	Min	5.83	6.10	7.35	4.64	6.79	3.20
	Max	55.21	50.92	47.59	51.92	30.24	54.11
	IQR	19.39	9.16	15.05	10.04	12.06	11.88
Accuracy (units)	**Mean**	0.342	0.377	0.420	0.431	0.470	0.511
	Median	0.319	0.367	0.416	0.444	0.411	0.533
	Range	0.564	0.538	0.623	0.518	0.577	0.365
	Min	0.100	0.150	0.220	0.180	0.216	0.341
	Max	0.664	0.688	0.843	0.698	0.793	0.706
	IQR	0.169	0.182	0.121	0.165	0.171	0.198

Table 23 Usability Study Full Descriptive Statistics by Interfaces

These results appear to show that although the alternative interfaces present the users with different visual cues, these do not appear to affect the user performance when undertaking a sound design task with the interface. Even where it was possible to show significance it was only for one case and could just be from natural variation given that a confidence interval of 0.95 was used. However, when examining the mouse traces and calculated standard distance deviation (Table 9) for each interface, there is a correlation between those interfaces that the dimension space analysis showed as providing more visual cues and the those that had a larger deviation.

This appears to show that interfaces with more visual cues encourage users to explore more of the interpolation space. This fits with what was discovered in the pilot study, but it has not been possible to demonstrate this conclusively from the mouse data here. As with the pilot study, the testing showed the presence of three phases in the identification of a sound with a graphical interpolator system (exploration, localisation and refinement). From this study it has been possible to show there are significant differences in both the speed of cursor movements and the number of high-speed moves made, between each phase. This gives further confidence that the effect observed in the pilot study is real and present regardless of the visual cues presented to the user. Given that the visual cues for each interface were static and did not change during the experiment it seems that an Interactive Visualization (IV) paradigm could be of further benefit by allowing the user to change the level of the visual detail on the interface during the different phases. Moreover, given that as the user gets closer to their intended location, they tend to make smaller moves and travel less distance, some form of zoom function could be advantageous to provide a finer level of control for the user, allowing more detailed sound design. Such an interface has already been shown to offer benefit to a spatial exploration process [257].

Unlike the pilot study, the visual cues appear to have a considerable impact on the interfaces perceived usability. In this study, it was shown through both the standard SUS and the extended SUS that there was a significant difference between the perceived usability of the interfaces. Indeed, the mean usability scores match the user's preferences and there is a strong correlation to the dimension space analysis that was undertaken prior to the tests. Using these it is possible to interpret a perceived order of preference for the different interfaces based on the results. In all cases, Interface 5 (triangulation) was perceived as the best interface in comparison to the others and for the standard SUS received a score which against the standard benchmarks is within the top 4% of all SUS scores undertaken. Although, from the dimension space analysis Interface 6 (tessellation) got the next best results, in the user testing it did not do as well and was beaten in all areas of testing by Interface 2 (nodes) which was perceived as the second best in the questionnaires. It also got the same mean ranks for the preferred interface choice as Interface 5. As a result, Interface 6 was firmly pushed into third position in all areas of the usability questionnaire testing. Interfaces 3 (radius-based) and Interface 4 (light) from the dimension space analysis appear to be similar and in all areas of the testing they got very comparable results, swapping position in terms of their perceived usability. That said, in the mean rank for the preferred interface choice, Interface 4 got a much lower value of 3.55 compared to the value of 4.57 received by Interface 3. It is also worth pointing out that in terms of the participants selecting their preferred interface, although both Interface 3 and 4 were each selected by a single participant when examining the average music technology experience, it was less than the other selected interfaces. This may suggest that these interfaces are more appealing to users with lower experience levels or that users with higher experience levels are able to make more objective assessment of the virtues of a particular interface. From the dimension space analysis Interface 2 (gravitational) resulted in the plot that covered the smallest area illustrating that it provides the users with less visual cues than all of the other interpolators. This is borne out through the interface receiving the lowest scores for both the standard SUS and the extended SUS as well as receiving the highest value for mean rank for the preferred interface choice. Although this suggests that this interface is perceived as being the least desirable through the user testing, it is worth noting that in the standard SUS it still got a score (70.75) which is higher than the sector average score of 68 [234].

Like the pilot study these results appear to show that interpolators for sound design are considered to provide a high level of usability. This is supported by the values given by each participant for their confidence in the sound design created with their preferred interface, resulting in an average score of 7.8 and little variation between the different interfaces selected as the preferred choice. It is also interesting that the average score for the preferred interpolator allowing the participants to create sounds they would not have discovered using traditional methods was even higher at 8.5. Again, little variation between the different interfaces was observed indicating that the unexpected element was a result of using an interpolator rather than the specific interpolation model.

7.5 Usability Study Conclusions

Given the limited number of participants in this study care is required when interpreting the results and a cautious approach has been taken. In the future, when the Covid-19 pandemic passes it will perhaps be possible to continue this study with more participants. This may allow a clearer view of any trends and may provide a greater confidence in the results.

First it should be noted that this experiment only examined the process of interpolator navigation and so all other attributes were controlled and unchangeable by the participants (preset sounds, locations, field-of-influence, etc.). As a result, the experiment results only capture a part of the overall usability of these tools and if the users were given a greater range of controls, potentially it will further affect the usability. Future experiments will look to assess the impact that these additional controls will provide.

The observation of three distinct phases of use during both the pilot study test and this study adds further strength that the observed effect is real and that users interact with the interpolators differently at different stages during their journey through the space. It has now been shown that there is a significant difference in the user's interactions during the three phases and these are present for all the different interfaces. This suggests that the affect is from the interpolator navigation rather than the particular interface presented. Moreover, it may also be the case that this phenomenon is a result of the navigation/exploration process and not unique to interpolation. In which case, the results maybe applicable to many other areas where spatial searching is undertaken. In this study all of the visualisations were static, not changing during the experiment and were not selected to provide users with cues related to the navigation of the space. Future work should be undertaken based on these findings to design new interfaces that provide users with visuals that facilitate the different phases of a sound design task. In addition, it is suggested that the visuals should not remain static, but should be controllable based on the user's interactions, either through allowing the user to select different visualisations or based on their interactions with the space.

During this study all of the interfaces under test provided the users with a full visual interface, as opposed to the partial or blank interfaces in the pilot study. As a result, in this study there was no significant difference in the performance between the interfaces. This adds further strength to the results obtained in the pilot study and implies that the number of visual cues provided by a full interface does not affect the user's performance with the interface. However, the number of visual cues does appear to encourage the users to explore more of the space and also directly impacts on the perceived usability of the interface.

Overall, there appears to be a solid correlation between the dimension space analysis and the relative usability of the interpolators. Moreover, it appears to be evident that those interpolators that present the users with more visual cues are perceived as having better usability. The exception to this appears to be the nodes interpolator that did not present the participants with as many visual cues as the Voronoi tessellation interpolator, yet did better in the SUS questionnaire and the preference choice. This maybe because the nodes object is freely available within the Max environment, and it is therefore possible that previous familiarity with the model may have biased the results from some participants. Nonetheless, the dimension space analysis did provide a good pictographic guide for evaluating the visual cues that the different interfaces present the user and so could be used in subsequent interpolator developments to gauge the potential usability of different interfaces.

From the SUS questionnaires it appears that all of the interpolators are perceived as highly usable with the average scores all above the benchmarks. However, it is evident that the visual cues do affect the perceived usability of the interface, even though the user performance was not affected. This gives further strength to the idea of developing new visual interfaces that provide additional visual cues to the user. Moreover, given that it appears that some users preferred different interfaces the visual cues could be selected and adapted by the users to suit their personal needs.

It was hoped that the extended SUS questionnaire would allow additional detail to be gathered specific to the use of interpolators. However, it appears that the responses to the supplementary items mirrored the results from the standard SUS, just with lower scores. This being the case there does not appear to be much benefit for using the extended SUS over the standard version which provides the benefit of known norms. In future it would seem sensible to stick with the standard SUS.

Unlike the pilot study, the mission in this study was specifically developed to try and replicate a real sound design task. Given that sound design is a highly creative and individual practice it is no surprise that different participants made different choices when undertaking the task. However, having examined the sound selection choices made by the participants with each interface it is interesting to note there was often some commonality in the selected locations. This appears to be more pronounced with some interfaces than others and when all selected locations were examined it is apparent that some locations were never selected with any interface. If it were possible to identify areas that produce suitable outputs and those that do not when populating the space, it may be possible to improve the effectiveness of the interpolators. Moreover, as each interface generates a unique sonic palette there is a possibility that some sound design tasks are better suited to certain interpolation models then others. Allowing the user to select and change the interpolation model during the task might be of benefit, especially as users have different preferences of interpolation models.

Finally, graphical interpolators for sound design seem to be perceived as highly usable because they provide a predictable mechanism for modifying sounds. That is, moving towards a preset makes the sound generated more like the preset and moving away makes it less like the preset and more like other presets as the cursor approaches them. Nonetheless, the participants gave a very high score for the question that asked if the interpolator allowed the creation of sounds that they would not ordinarily have found. This likely comes from the complex *many-to-few* parameter mapping that was used in the experiment meaning that although the interpolators provide a predictable control mechanism, the mappings can result in unexpected parameter manipulation. Furthermore, as the interpolator is simultaneously changing the numerical values of many synthesis parameters it may not be obvious how these relate to the sonic outputs. This "predictable unpredictability" that the interpolators provide, appears to be highly beneficial to the creative task of sound design and may be less well suited to conventional music composition performance. When developing future interpolation systems, it would be prudent to consider both the predictable nature of the control mechanism, but also the unpredictable nature of the mappings. Moreover, there might be additional benefit to providing visual cues that relate to the parameter changes and so the sonic output.

Chapter 8 Star Interpolator

With the results from both of the studies undertaken suggesting the importance of visual cues when using an interpolation model, it was decided to try to design a new visualisation that provides all the visual cues identified in Chapter 7, but also new visual cues that so far have not been implemented. One thing that is common to each of the visualizations that have been implemented so far within the interpolation framework is they all relate directly to the interpolation model and do not relate to the parameter space or the eventual sonic output. From the wider literature review that was undertaken in Chapter 2, it was seen that this is the case for the vast majority of prior systems. There have been some exceptions and from these it is possible to categorise the visualisation of the space into three different forms:

1. Representation of the interpolation function (interpolation space)
2. Representation of the sonic output (timbre space)
3. Representation of the parameter values (parameter space)

Although most of the systems give a graphical representation of the interpolation function there are those that have provided a different model. The concept of defining a timbre space as a controllable mapping of synthesis parameters was suggested many years ago by Grey [135] and work still continues in the area. These systems do use interpolation to provide parameter mappings between the preset locations, however, the presets all have fixed locations within the space based on their sonic outputs. Although the interpolation allows new timbres to be created between the presets, from a sound design perspective it is restrictive as the locations of the presets cannot be modified as part of the process, given that the output timbre defines their location. From this perspective, the use of an interpolation space model that allows presets to be freely located anywhere in the space naturally suits the sound design goal. One previous solution, known as spike-guiding, does allow the free-form positioning of presets as well as providing a visual representation of the sound signals at preset locations within the space [186]. This was achieved by providing a time/frequency plot, known as a spike code [187], that gives a representation of a preset sound at its position in the space. Although this does provide a guide to sonic differences between locations, the spikes can be difficult to interpret and do not relate to the parameters and their adjustment within the space. Moreover, no indication is provided for the output between the preset locations and as large sonic changes can be created with an interpolator it could be difficult for the user to anticipate how the sound will change between locations.

A number of interpolators have offered shading and/or colour interpolation to imply the weightings of presets through the interpolation space [123], [162], [177] & [180]. However, this relates to the interpolation and not the effect on the parameters or the resulting sound. Although this idea could be modified so that colour shading is shown for the interpolation of individual parameters, it would still make it difficult to consider the simultaneous manipulation of multiple parameters. Either the parameter maps would have to be shown individually or some form of colour interpolation would have to be performed. From the exploration of shading that was undertaken in Chapter 5 it was found to be computationally intensive so unsuitable for real-time operation so would require pre-computation. Therefore, when it has been implemented by previous interpolators, they have two operating modes to set-up the space and then interpolate. This is restrictive as it means that the layout of the interpolation space cannot be changed as part of the sound design process. It was also found that interpreting colour interpolation can be difficult as there are many algorithms for performing colour interpolation, so it is not always obvious. One possible solution that was considered to reduce the computation overhead of the shading was to replace the shading with contour lines, similar to a geographical map, that would show lines of constant parameter values across the space. However, this solution does not really provide a mechanism for being able to simultaneously view multiple parameters.

From the review, beyond the basic *one-to-one* mappings offered by solutions such as the Korg Kaoss Pad [120], there have not been any examples where any representation of synthesis parameters has been provided. This seems counterintuitive given that the sonic output is a direct function of a set of parameters. Moreover, sound designers will likely have some knowledge and experience of synthesis principles and the impact that certain synthesis parameters will have on a sonic output.

8.1 Star Plots

Given the large number of parameters that most synthesis engines possess and their sizeable data range, to solve this problem a high-dimensional visualization is required that will work in the interpolation context. Radial based plots were chosen as the centre point provides a precise location for the represented preset's position within the interpolation space. Although area-based radar plots are a popular form of multi-dimensional visualization, they were not chosen here as area increases quadratically rather than linearly, which could result in users thinking small parameter changes are more significant than they actually are. In addition, the area views may interfere with the user's ability to interpret the other visual cues previously shown in Table 7. One possible solution is to use a glyph-based display, known as a star plot [258], which offers a representation of individual parameter values while also providing minimal interference with other visual aspects. With this method, each of the preset parameters is represented as a "beam" within the star and the beam's length is proportional to the parameter value within the preset. Each beam is centred at the preset location and angularly distributed within a unit circle to show the relative value of each parameter. Typically, with plug-in based synthesis engines, continuous parameters are normalized for external control to have a numerical range 0.0 – 1.0, but if this is not the case, scaling can be applied. At this stage, only plug-in based synthesis engines have been used so this has not been explored any further. The star plot can be used to give a pictorial representation for the parameter values of each preset in the interpolation space. With this the order of parameters in the star plot can have a large impact on its ability to effectively communicate the information. The order has been defined by the synthesis engine's parameter list, as this generally matches the signal-flow through the particular engine. Once the user selects the desired interpolation parameters, each star plot is constructed with these in the defined order. This provides a logical order for the parameters and allows parameter values to be directly compared between the chosen presets. All the non-selected parameters remain "locked" at their non-interpolated/last values and are not shown in the plots to aid the reading of the parameters that are being interpolated. As can be seen in the example shown in Figure 72, each preset has different parameter values giving each star a unique beam arrangement.

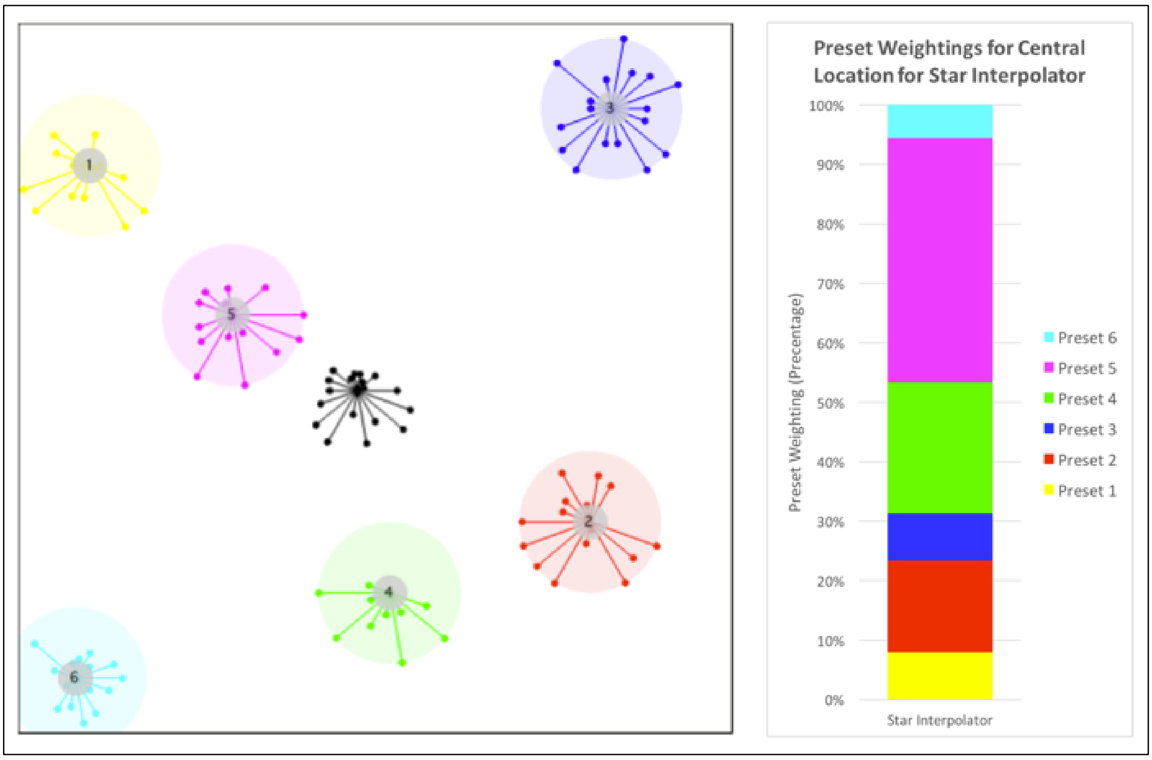


Figure 72 Star Interpolator Visualization with Eighteen Interpolation Parameters

In this example, eighteen parameters (of the 149 available in this synthesis engine) have been selected for interpolation[21]. In this case, the interpolation is realized as a generalized IDW model [166] to generate the preset weightings shown. If a different set of parameters is selected the new star plots are generated and displayed. In this way it is possible to represent different numbers of parameters using

[21] Example of this Star Interpolator Visualization: https://youtu.be/Eaf631rsvA8

the same mechanism. It can also be seen that the normal interpolation point's crosshair cursor, used in the other interpolators, has also been replaced with a star plot, the shape of which is updated in real-time as the interpolation is performed (shown in black in Figure 72). As the cursor is moved to coincide with one of the preset locations so the cursors star becomes the same as the preset's star. In this way, the interpolation point provides direct visual feedback on the parameter values that provide the biggest contribution to the current sonic output. Moreover, when moving the interpolation point within the parameter space this visualization also provides feedback on which parameter value changes are producing different sonic outputs. To provide the user with a more detailed picture of these parameter value changes, a separate larger viewer is provided that shows the parameter names, their position within the star plot and their current numerical values. An example of the cursor viewer is shown in Figure 73. This viewer is also updated in real-time to offer the user instant visual feedback that when combined with the interpolation space and the real-time sonic output, provides a powerful platform for sound design tasks.

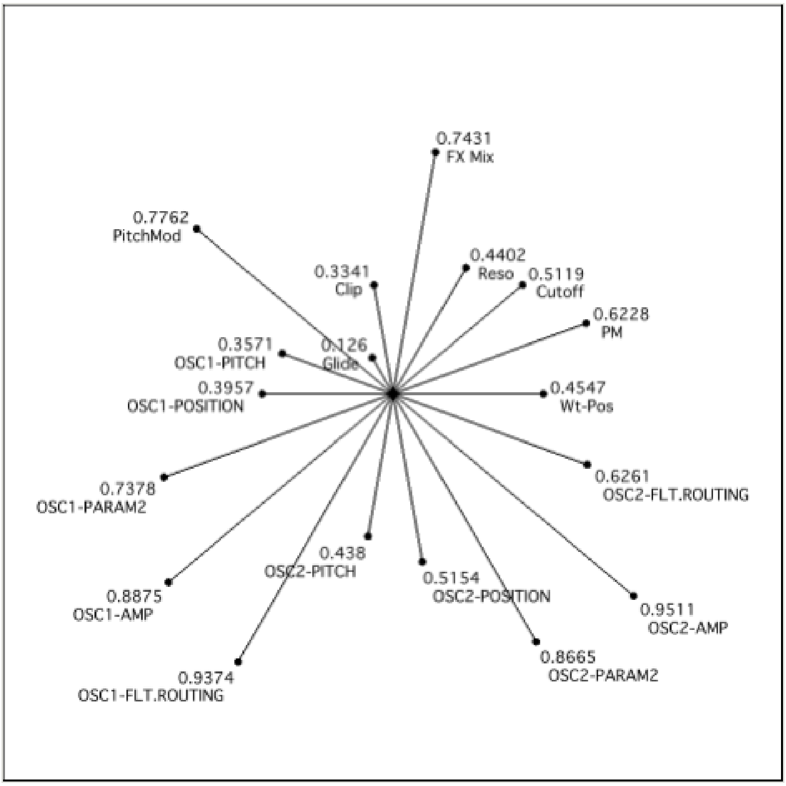


Figure 73 Star Interpolation Cursor Parameter Visualizer

An example of the full Star Interpolator interface is shown in Figure 74 incorporating all of the features covered[22]. As can be seen, the relative preset weightings of the interpolation are displayed via a bar-graph as was determined in Chapter 7.

[22] An example of the full Star Interpolator Visualization interface: https://youtu.be/BaFP_W5cnlI

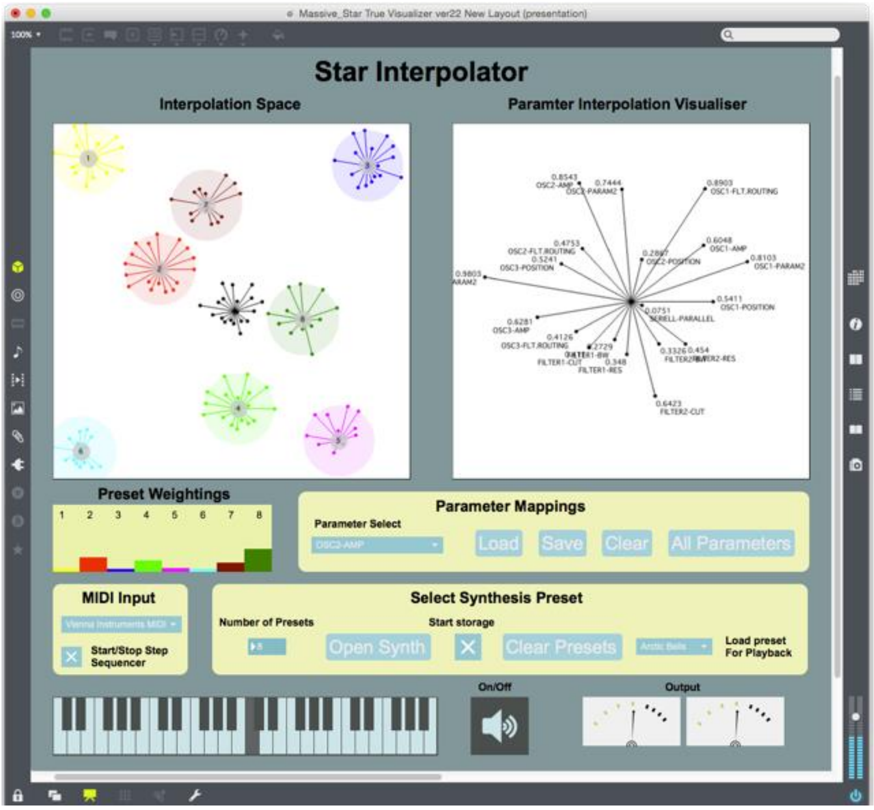


Figure 74 Star Interpolator Full Interface

8.1.1 3D Star Plots

A 3D version of the Star Interpolator was considered by expanding the model to three dimensions as others have previously done [167]. The use of a 3D model results in two different aspects that require consideration: having a representation of a 3D space allows the 3D arrangement of presets within the space and provides the possibility of having 3D glyphs representing the presets. The use of a 3D space provides users with additional degrees-of-freedom when using the interface. Additionally, the use of 3D glyphs provides a greater area for displaying a large number of parameters. These two things are actually independent of each other but have both been considered here. However, on implementation, challenges with this strategy became apparent. With a 3D version of a star plot it is harder to compare parameters and see the relative values directly, as the 3D projection makes it difficult to determine depths without additional visual cues, such as lighting and shading. Moreover, the view of the parameters depends on the viewer's perspective within the 3D space. This can be seen in Figure 75(a) which shows a 3D star plot, of 25 parameters, where all parameters have identical values, in this case 1.0. As can be seen, it is difficult to determine the parameter values due to their different positions within three dimensions. Figure 75(b), shows the same star and parameter values from a different orientation giving a different perspective on the parameters. This makes it hard to identify the position of individual parameters within the star and to be able to compare parameter values between presets. In light of these difficulties, a 3D display has not been pursued further at this time. Potentially allowing the user to rotate their viewing position within the space as well as providing additional visual cues (colour/shadowing) could aid the clarification of depth.

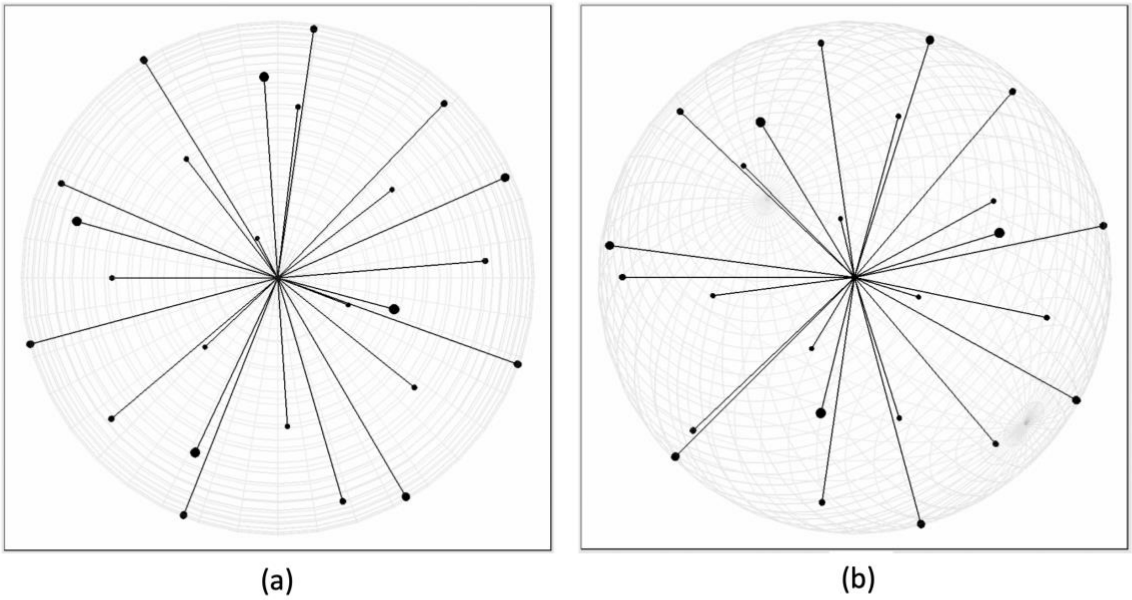


Figure 75 3D Star Plots with 25 Parameters all with Value of 1.0 (a) Positioned at *x*=0.0, *y*=0.0 & *z* =0.0, (b) Orientated +30 degrees rotation on *x*-axis and *y*-axis

8.2 Star Interpolator Exploration

The Star Interpolator was initially included in back-to-back sound design comparisons with the six different visualizations used in the usability tests (Chapter 7). This was done by populating it with the same presets and layout as the other interpolators. The key difference between the Star Interpolator and the comparative systems is its ability to not only provide guidance on parameter differences between the presets, which others do not, but to make these differences explicit. Moreover, since the cursor star shows the parameters for the current sonic output from the interpolator, it helps in gaining an understanding of the complex relationships between the parameter values and the audible output. Moving the cursor within the parameter space then allows the user to visually see which parameters are changing and by how much. This can be done to individual parameters or groups, thus providing detail of transitions in the mappings. When this is combined with the sonic output, it provides a powerful mechanism for understanding the sonic palette that the interpolation space is providing. Furthermore, the real-time update of the interpolation star plot allows the users to not only establish values for the parameters, but also gain a feel for the rate-of-change of the parameters when moving between locations within the space. This was found to be very useful when trying to establish desirable parameter mappings to use in the interpolator. During bench testing of the system, it was discovered that a good sound design strategy was to initially start with all the preset parameters mapped to the interpolator. An example is shown in Figure 76 of the Star Interpolator with all parameters mapped, in this case 149.

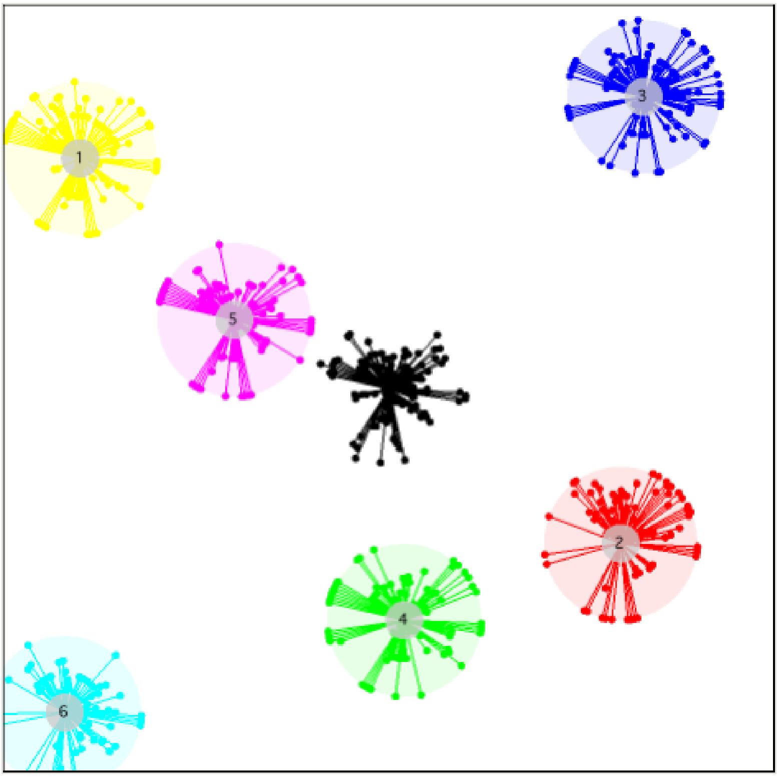


Figure 76 Star Interpolation Visualization with All Preset Parameters Mapped

Although the star plots become very crowded with this number of beams, the visuals still provide useful information as it is possible to instantly identify which parameters are not changing between the selected preset sounds and so are not affecting variations in the sonic output. The mappings can then be modified to remove these parameters and simplify the sound design process. With the refined star plots it is then possible to recognize which parameter values/changes are producing the most significant impact on the sonic output and an iterative refinement approach can be adopted in the sound design task. In this, parameters can be selectively removed to establish their sonic footprint and if the results are found to be unsatisfactory, they can always be reintroduced to recover the desirable sonic manipulation. Through this approach it is possible to identify regions within the interpolation space that generate specific audible characteristics. This was found to be particularly valuable when trying to design sonic expressions by moving the cursor between different locations within the parameter space. This process is not easy with any of the original interpolator visualizations and although their parameter mappings can also be changed, it is very difficult from the sonic output alone to identify which parameters are generating the differences between location changes within the space.

Another benefit of the Star Interpolator was found to be in the selection of presets and the setup of their layout within the interpolation space. This is often a process of trial-and-error in selecting presets that possess desirable sonic characteristics and then randomly exploring different arrangements of them within the space. However, with the star plots providing the identification of changing parameters it is possible to leave the interpolation point stationary and then individually move each preset towards and away from the interpolation point to hear the contribution that the mapped parameter changes provide. Repeating this successively provides a mechanism for constructing an interpolation space that provides the sonic changes which are desirable for the specific sound design task in-hand.

During the exploration of the Star Interpolator, it was tested with a number of different synthesis engines (additive, subtractive, FM, and granular). In each case, the star plots showed the values of the synthesis parameters that are mapped to the space – for the presets and the interpolation point. However, it was found that interpreting the parameter values and particular changes was easier to do with some forms of synthesis than others. For example, additive and subtractive produced fairly predictable results, whereas FM and granular were much harder to interpret. In addition, the same star plots and therefore parameter values, result in completely different sonic outputs if the synthesis engine is changed. However, it was found that the star plots did make it much easier to identify the parameter changes causing sonic textures. This in itself led to a much easier process for learning the subtleties of sound design with a particular synthesis engine.

8.3 Star Interpolation Visualization

Despite the advantages covered in the previous section, the Star Interpolator has not been designed as a new method of interpolation, but as a different form of visualization for existing methods. Especially as from the testing that was undertaken in Chapter 7, it is evident that users prefer different interpolation models. This being the case, the Star Interpolation visualization could be applied to all the different interpolation models already created with the interpolator framework. In this way, the generation of the preset weightings remains the same, but the visuals directly relate to the parameter changes and so the sonic changes being heard. However, as determined in the testing undertaken in Chapter 7, the original visualizations still provide useful cues, such as, regions-of-interest, that have been shown to be beneficial to users in sound design tasks. Therefore, an Interactive Visualization (IV) paradigm can be adopted, where the user can choose the visualization displayed between the original visualization, Star Interpolator or a hybrid visualization which combines both. Figure 77 shows an example of a hybrid visualization for the nodes interpolator that includes the star representation. Nodes was chosen initially for the hybrid visualisation as the dimension space analysis in Section 7.1.1 illustrates, it provides an intermediate number of visual cues and also provides familiarity due to its inclusion in the Max environment.

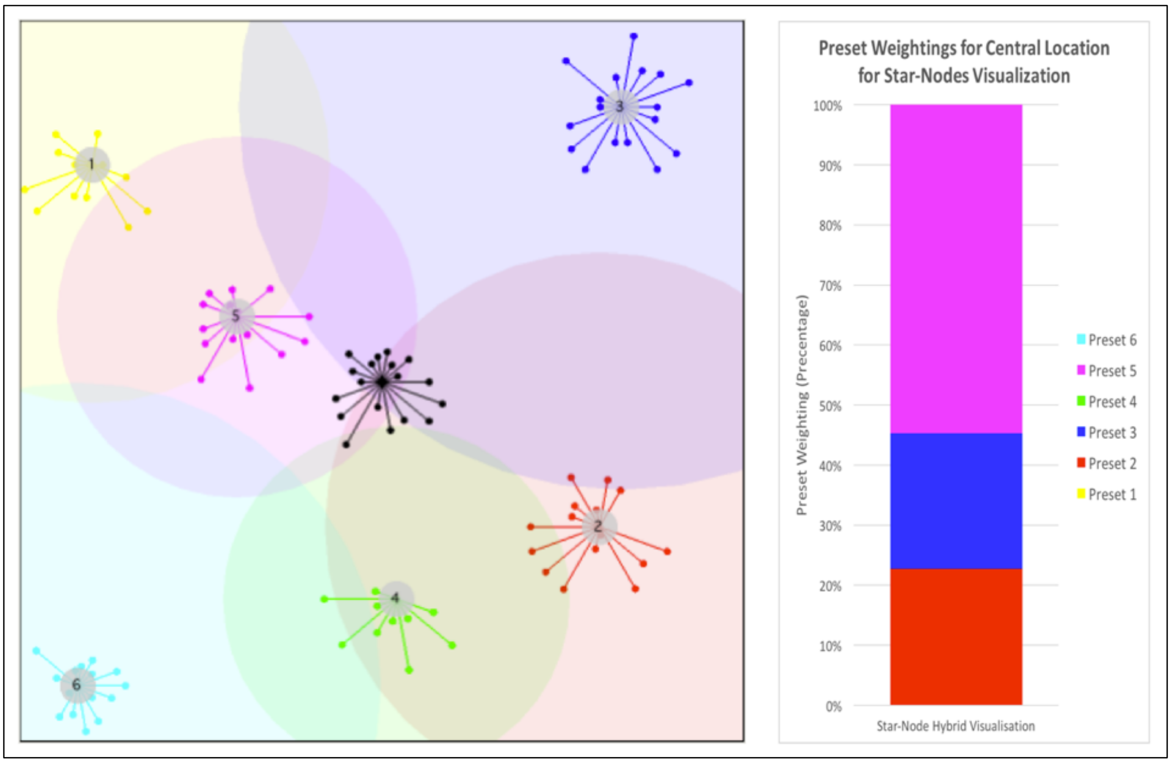


Figure 77 Hybrid Nodes-Star Visualization with Same Preset Layout and Interpolation Parameters

Here the preset layout and interpolation parameters are the same as was shown in Figure 72, but the interpolation model is the intersecting circles provided by nodes[23]. As a result, the weightings generated match those for the nodes model where only the presets included in the intersections are present.

In the current version, the star visualization can be accessed by the users with hotkeys on the computer keyboard. In this way, the user can select when and how the different visualizations are used. However, the hybrid visualization, shown in Figure 77, offers all of the visual cues provided by the original nodes model as well as the additional cues delivered by the star visualization. Through exploration in bench testing, it has been found that the hybrid visualization provides excellent detail of not only where within the space the sound will change (intersections in this case) and which parameter changes are providing sonic changes, but also when changes are not occurring and exactly when a preset is being recalled (i.e. has 100% weighting). For example, for the system shown in Figure 77, with nodes as the interpolation model, the preset sounds can be recalled by placing the cursor on a node area where no intersection exists, if available. With complex preset layouts, these areas may not be easily apparent, especially when the sonic output is long evolving sounds. However, with the hybrid visualization it is much easier to see, as when the cursor is moved within non-intersecting regions all the star beams remain static.

[23] An example of the hybrid Nodes-Star Interpolator Visualization interface: https://youtu.be/yjfXakh3LBU

As was found from the usability testing in Chapter 7, the most popular interpolator was the triangulation interpolator which, from the dimension space analysis, also provided the most visual cues. To ensure that the idea of a hybrid visualisation also worked for interpolators that provide more visual cues a triangulation-star visualization was also created. This is shown in Figure 78 with the same layout and eighteen parameters mapped to the space[24]. Although this interpolation model does provide the users with more visual cues, when combined with the star plots the visuals remain very usable.

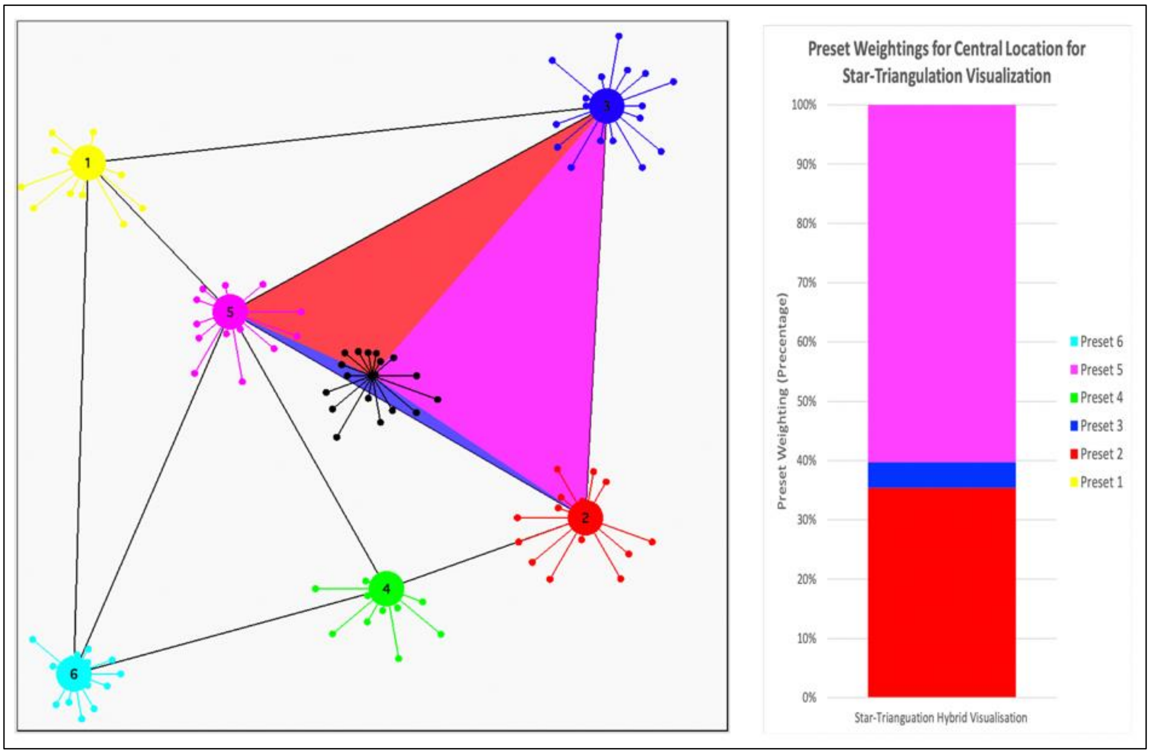
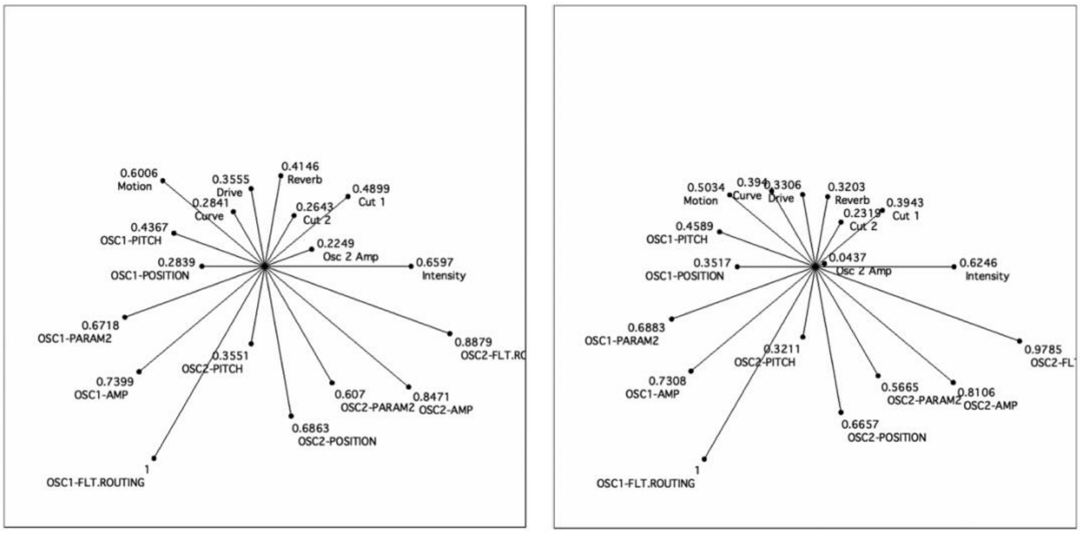


Figure 78 Hybrid Triangulation-Star Visualization with Same Preset Layout and Interpolation Parameters

The only modification that was made to the previous visuals of the triangulation model was the transparency of the triangles representing the preset weightings was increased to provide additional contrast for viewing the star plots against them. As can be seen from the cursor's star although all other aspects remain the same, the change of interpolator model causes a different output. Moreover, the star plots provide the ability to identify exactly which parameter changes are causing the sonic changes as a result of switching the interpolation model. This can be seen clearly when examining the cursor viewer as it allows the individual parameter values to be compared. For example, comparison of the cursor viewers between the hybrid nodes-star

[24] An example of the hybrid Triangulation-Star Interpolator Visualization interface: https://youtu.be/KbioRMkZWRk

visualization (shown in Figure 77) and triangulation-star visualization (shown in Figure 78) can be seen in Figure 79.



(a) (b)

Figure 79 Comparison Between the Cursor Viewer of two Hybrid Visualisations (a) Nodes-Star, (b) Triangulation-Star

From this it is possible to see the exact differences between the parameter values purely from changing the interpolator model from nodes to the triangulation. If the user is provided with a mechanism for switching between different interpolation models, then the star visualisation provides a mechanism for identifying exactly where the differences in their sonic pallets are being generated.

Through exploratory testing of this visualisation, it was found that as the triangulation method always uses only three presets to generate the output the star plots were very useful for identifying which parameters were causing sonic changes. Although this would be the case for all interpolators with a star visualisation, it was found to be easier to work out the linkage between the changing parameters and the sonic result, due to the constant nature of performing the interpolation between three presets. Whereas, with other interpolator models the number of presets is not constant and could be continually changing across the interpolation surface. Moreover, the star plots at the vertices of each triangle gave a clear indication of how individual parameters would change when moving within that triangle. This was equally true for identifying which parameters do not change within a triangle.

8.4 Star Interpolator Conclusions

As identified in the previous chapters, the original graphical interpolators provide a number of visual cues as well as a corresponding audio output. Although there is some linkage, albeit subtle, between the two, they do not directly correspond to each other. While one of the typical functions of a visual interpolator has been to conceal or abstract the user from the details of an underlying sound synthesis process, allowing them to focus purely on the sonic changes induced through navigation of the interpolation space, the Star Interpolator visualization provides a powerful mechanism for obtaining additional visual cues that do directly relate to the sonic output obtained from the interpolator, still without necessarily having to understand the technical details of the underlying synthesis algorithm. The real-time relationship between the visuals and the corresponding audible output offers a combined audio-visual cue that has the potential to provide more efficient navigation of the space, as has been found with other systems [260]. It also appears to offer further assistance in determining desirable parameter mappings when undertaking sound design tasks. Moreover, this visualization potentially offers a platform for beginners to become familiar with synthesizer programming techniques and understand the complex relationships between changes to multiple parameters and the resulting sound. Adopting an Interactive Visualization (IV) approach means that users that do not want the additional cues provided by this visualization can choose, through their interactions, what is displayed and when, providing increased user flexibility.

From the exploration that has been undertaken so far it appears that the Star Interpolator visualization offers many potential benefits, but to further understand these and to measure their effectiveness, formal usability tests should be undertaken using a similar methodology to that previously used for the evaluation of graphical interpolators in Chapter 6 & Chapter 7. This previous work has shown that the visual feedback provided by an interpolator interface changes how users interact with the interpolator and this has subsequently been shown to impact on the use of an interface during a sound design task, with interfaces that provide more visual cues being considered as more usable. Furthermore, these interfaces also result in a high perceived usability. Through applying a similar methodology to the Star Interpolator, it should be possible to empirically test the usability of this new visualization paradigm through a comparative approach.

Finally, as highlighted in the results from the usability questionnaires (Section 7.3.3) a number of the participants suggested that the ability to place markers within the interpolation space would be of benefit as it would provide an easy mechanism to return to the same point. However, as previously highlighted there are additional benefits to being able to indicate landmarks within the sound design space [188]. Therefore, an area that should be explored further is the provision of a mechanism that would allow users to place markers at points of interest within the space, but instead of just using a generic marker, a star plot could be used to provide a representation of the parameter values at that location.

Chapter 9 Conclusions and Further Work

Graphical interpolation systems provide a mechanism for the discovery of novel sounds through the navigation of a parameter interpolation space defined by the locations of existing sounds, the selection of interpolation parameters and the implementation of the interpolation model. The abstraction layer these systems embody provides a potentially highly usable tool for creative tasks, freeing the user from technical details of synthesis specifications and allowing them to focus on the sonic results. The work detailed in this thesis, particularly Chapter 5 to Chapter 8, has provided new insights into the use of synthesis parameter interpolation for sound design and helped develop a new understanding of the performance and usability of previous interpolator designs. This includes the identification of interpolation phases and the importance of visual cues, including their direct relationship to the systems performance and usability. Following on from these studies, a new interpolator visualisation has been devised that provides additional visual cues to enhance graphical interpolator usability. This chapter summarises the important results from the research, together with conclusions and recommendations for further work.

It is important to identify limitations of the studies completed so that the results can be evaluated in context and potentially addressed in subsequent research. A key constraint of the usability test was the number of participants that were recruited for the usability experiment. As a result of the Covid-19 situation, it was not possible to recruit more participants and so it was necessary to complete the data analysis phase with the limited number that had undertaken the experiment prior to lockdown. The significance values obtained in some areas of this study may be attributed to this issue and it is possible that access to more participants would have helped to provide more conclusive results.

Another potential limitation in the studies undertaken is that the users were constrained to a single layout for the presets and so configuration of the interfaces. The decision to control the user options was taken to limit the experimental variables so that like-for-like comparisons could be performed between the different implementations. However, it is possible that different layouts of the interpolation spaces would have led to different results. Moreover, if the users were allowed to modify the layout themselves during the experiment, control the number of presets and make their own preset choices, these could have further impacted on the results, at the cost of experimental complexity. An associated issue is the assumption that the different models all operate at the same efficiency when populated with the same number of presets. It might be that some models are more effective with a dense population of presets, while others benefit from a sparser arrangement.

A final area that may have influenced the results from the experiments is that the interpolation visuals were not explained to the users. Although this allowed the intuitive nature of each interface to be assessed, it may also have affected how the visuals were interpreted by some participants and this could have affected their performance, i.e. by misconstruing the meaning of the graphics, rather than the nature of the individual cues. Explaining the concept of each visual model to the participants before allowing them to use it might have eliminated any confusion, but would not have allowed the models level of intuitive usability to be established.

9.1 Interpolation Framework

To aid the development and testing of different graphical interpolation systems a framework was proposed that would facilitate comparisons between alternative implementations. This framework has been used exclusively through this body of work and has been the backbone of the usability testing that has been undertaken. The compartmentalisation of the framework made it possible to change a single aspect of an interpolator in isolation to the other components. In the usability testing this provided confidence that any effects observed were purely a result of the changes and not caused by secondary factors.

Moreover, the benefits of being able to rapidly prototype alternative interface designs have been recognised [260] and developed into user centred approaches [261]. Indirectly it has been shown that this framework offers a mechanism to allow the rapid prototyping of full interpolator systems, with a number of alternatives having been built. In addition, the framework has also been shown to offer the ability to rapidly develop and test alternative user interfaces for the same underlying interpolation systems. This has been shown in the development of the different interfaces used in the pilot study (Chapter 6) and the star interpolator (Chapter 8). From the testing that has been undertaken so far it appears to suggest that the graphical interpolator's visuals have a direct impact on the usability of the system for sound design. Although during this body of work the framework has been predominantly used to examine the impact that the visualisations provide, a similar strategy could be adopted for any of the other layers in the framework. This work would be beyond the scope of this study, but the framework offers a good foundation for initiating this work as it provides rapid prototyping and would afford a consistent architecture for the construction of interpolators to be used in further investigations, exploring other areas of the systems.

9.2 Interpolation Phases

From the experiments undertaken in Chapter 6 and Chapter 7 three distinct phases have been identified in a user's interactions with an interpolator. These are independent of the interpolators visual display which suggests that they are a function of the search process with a graphical interpolator. This is reinforced as interfaces with different visual cues have been tested and yet when examining the participants interactions, it has been shown that the same phases are present and there were no obvious differences between the phases generated with different visualisations. However, as identified in Chapter 8, given that all of the visual cues from the previous interpolators related to the interpolation model itself and not the synthesis parameters there is a possibility that different cues could enhance the process. Moreover, with all of the testing that has been undertaken so far, all the interfaces remained constant throughout. Given that the users clearly interact with the interfaces differently during each of the phases it would seem plausible that an adaptive interactive interface would be beneficial.

It should also be considered that in the analysis of the experiment results (presented in Chapter 6 & Chapter 7), the total test time was divided into three and each third labelled as the three phases. This appeared to work for many of the participants and provided a quick and easy way to identify differences in the user's interactions with the interfaces through the activity time. However, it was noted that not all participant's interactions split evenly into thirds of the testing time. Possibly using a more data driven approach to defining the phases, such as using the frequency of high-speed moves or distance moved against averages, would provide more accurate results. Given that the data-sets from both experiments are available this is further analysis that will be undertaken in the future. If a consistent method can be found to automatically detect the phases it could provide the possibility of being able to design interfaces that would automatically adapt the visualisation delivered to the users, depending on which phase they are in.

9.3 Interpolator Models and Visual Representations

This investigation has examined the relationship between the interpolation model and its visual representation in graphical interpolators. The results from the studies presented in Chapter 6 and Chapter 7 imply that there is a strong relationship between the number of visual cues that the visual representation provides and the perceived usability of the interface. Moreover, in the pilot study it was shown that a lack of visual cues can have an impact on the user's effectiveness with the interface. Although this was not corroborated in the usability study, it was shown that including a greater number of visual cues in an interface resulted in greater exploration of the available interpolation space by the participants. That is, the cues appear to encourage the users to visit more locations and as such, assess more of the available sounds that the interpolator can generate.

Given that in the usability study it was not possible to show significant differences in the performance metrics, it may suggest that as long as an interface offers certain visual cues, there is less impact on the user performance when additional cues are provided. This does offer a wider consideration that the visual cues may not be equal in their ability to communicate information to the user and that certain visual cues might be more important than others. The relative importance of each visual cue could be established through further testing using a similar methodology to the pilot study, except where only individual cues are provided, one at a time. The interpolation framework could again be used and through the comparison of individual cues their effectiveness could be evaluated. This would allow insight into the impact that individual cues have on the interface's performance and usability.

9.4 Interpolation Surfaces

In this body of work colour shading has been used to examine the different interpolation surfaces that the individual models create and the differences between them (Chapter 5). However, they have not been used as a visual cue to aid the interpolation process. The main reason for this is the computational load to calculate and render the surface in real-time. Although several other graphical interpolators [123], [177] have offered this feature the compromise is that the interpolator then requires two modes of operation: setup and interpolate. This also means that it is not possible to change the interpolation surface by moving preset locations or adding or removing presets as part of the sound design process. For this reason, it was decided that for this body of work the interpolators would not require two operating modes, in preference for allowing sound design through modifications to the surface. It is also worth noting that the shaded interpolation surfaces presented in Chapter 5 represent weightings of the presets across the surface but do not give much indication of how the individual parameter values will change within the space. For example, individual parameters may change over a large or small range between preset locations or the direction of parameter changes maybe different to the direction of cursor movement towards or away from preset locations.

Despite these issues there might be some usability benefits to showing this visual cue to users. As computer processing power and storage increases it is likely that it will become possible in the future to generate the interpolation surfaces real-time. Moreover, reimplementing with a programming architecture that is more efficient for graphics programming (such as C/C++ and/or using the Graphical Programming Unit (GPU)) may allow real-time generation. This would require a further in-depth investigation into this area. In the meantime, some non-real-time testing, where the surface is pre-computed for a preset layout, could be undertaken to see if there is any benefit and the level of usability that it provides. This would allow the level of return to be evaluated to establish if there is an overall benefit to further pursue in this area. This can be achieved by repeating the experiment already undertaken in the usability study, but this time providing a pre-computed surface for each interpolator based on the preset layout. The results could then be directly compared to those presented in Chapter 7 to see if differences can be identified and assess the potential impact the surface provides.

9.5 Star Visualization

The star plot visualization was introduced as a mechanism for providing users with additional cues that relate to the data being interpolated, namely the synthesis parameters. Although all of the other interpolators provide a sonic output that corresponds to the current interpolation, none provide any visual cues that relate to the output. The star visualisation could be considered as contradictory to the initial premise of using a graphical interpolator to mask the underlying synthesis parameters from the user. However, given the complex relationships that are created when even a few parameters are used, the overall star shapes are still fairly abstract, but they do appear to forge an audio-visual relationship which may enhance the interpolation process. As well as providing visual cues for the current interpolation output, cues are also provided for what output will be generated at the preset locations within the space. This provides the user with insight into how the parameters will change between the current cursor location and the preset locations. Because users may have different requirements during the different phases of the interpolation process the star visualisation has been made interactive through the use of hotkeys so that the user can control when and how the interpolator visuals are displayed.

Through bench-testing, anecdotal evidence has been gathered that appears to indicate that there is a benefit to using the star visualization. However, at this stage formal user testing, similar to that undertaken in both the pilot study (Chapter 6) and the interpolator usability study (Chapter 7), has not been undertaken. This will be completed in future work, using a similar methodology to the previous studies, but it will look to compare the usability between a standard interpolator interface, the star visualisation and the hybrid visualisation that combines both. This will be undertaken for two different interfaces: Nodes as it has been used as a benchmark throughout this body of work and the Delaunay Triangulation which was the preferred interface from the usability study. A further study can then be undertaken to establish if when using hotkeys to control the visualisation, there is any linkage to the interpolation phases.

It should also be highlighted that the star visualisation has a number of additional areas where they can be used in interpolation systems, and these will be explored in further work. Given the challenge of real-time shading to represent an interpolation surface (as highlighted in Section 9.4), the visualizer model could be adapted to give a star-based, interpolation space visualisation that would show the influence of the various presets in the interpolation. This would provide an alternative to the current bar-graph implementation (used in Chapter 8) and if a layered approach were taken, the visualiser could provide both the parameter values and the preset weightings simultaneously.

Another area that could be explored is the use of the star visualisation to display perceptual parameters of the sonic output (e.g. brightness, pitch, dynamics, harmonicity, etc.). As highlighted in Section 2.4.1.6 the use of perceptual parameters to map to sound generation has already been established and parameter classification undertaken [148]. The visualisation could be modified to display perceptual parameters instead of the synthesis parameters. The two alternative visualisations could then be comparatively tested to see if there is an impact. It may also be possible to allow the user to switch between the two alternative representations as it is conceivable that each visualisation would be beneficial at different points in the sound design process.

9.6 Interactive Interpolation Models

From the usability testing that was undertaken in Chapter 7 there was some consensus of the preferred interface, however, it is evident that some participants preferred different interpolation models. It is also worth considering that for the experiment conducted there was a single task being undertaken and other tasks might be better suited to different interfaces. Consequently, there might be value in making the selection of the interpolation models interactive so that they can be selected by the user. This could be done through a menu system selection or possibly through additional hotkeys and would give the users the ability to choose and change between the different interpolation models. In this way the users would have the ability to select the most suitable model for the task in hand. As was shown in Chapter 5, the different models produce different surfaces for identical layouts of presets. As a result, some models will produce outputs that are different to those that are achievable with alternative models. Allowing the user to be able to dynamically change between the models will produce discontinuities in the outputs when the selection is changed, but may allow unique sounds to be identified that would not have been possible without selecting different models. Moreover, given that as identified in Chapter 7 each model provides the user with different visual cues, the ability to be able to change the model may improve the overall usability of the interpolation system. It was also shown in Chapter 8 that the

Star Interpolator visualization offers a mechanism for clearly identifying the differences in parameter values caused by changing the model.

At this stage the ability to change the interpolation models has not been implemented, but it is planned for future development as well as further usability testing to establish if it aids the sound design process. This will also be undertaken with the Star Interpolator visualisation to explore its benefits in identifying the resulting parameter changes from the selection of a different model.

9.7 Interpolator Dimension Space Analysis

From the dimension space analysis and subsequent usability testing that was undertaken, detailed in Chapter 7, there appears to be a strong correlation between them and in most cases the dimension space analysis predicted the preferred usability results. As this is just one set of results it is difficult to be certain, but it does appear to show there is some further value in the use of this form of analysis. This could provide a benefit in being able to anticipate the usability of graphical interpolators that provide certain visual cues. Moreover, given further positive results between the analysis and user testing it might be possible to use the analysis as a means for designing new interfaces that maximise the potential of the visual cues. In this way, it should be possible to design interfaces that provide better usability.

To further evaluate the dimension space analysis, it will be applied to all the subsequent development and testing that is being proposed in this chapter. This should help to corroborate the results already obtained and provide greater confidence in subsequent results. It is also worth noting that the dimension space analysis was designed specifically for the interpolators that were tested in the usability study. As a result, it has not been designed as a definitive solution for all scenarios and it may require modification and/or redesign depending on the demands of subsequent interface testing. However, there is a degree of confidence in the basic premise and key principle of applying dimension space analysis to the design of visual interfaces for interpolators.

9.8 Interpolation Maps

Through this body of work, it has been shown that there appears to be a correlation between offering users' additional visual cues and increased perceived usability of the interpolators. Moreover, the interpolation process appears to offer a number of parallels to spatial exploration and navigation tasks. For example, from the interpolation traces obtained from testing (Chapter 6 and Chapter 7), the start of the exploration phase appears to be very similar to a random foraging spatial navigation pattern seen in animals, such as rats [262]. Given that humans are the only animals that possess the ability to recognise cartographic landmarks and to be able to perform visually guided navigation [263], there might be value in expanding the idea of constructing an interpolation map that could provide further visual cues. In fact, the Star Interpolation visualisation does provide visual markers for the preset locations within the space, similar to symbols on a cartographic map. Moreover, as identified in Chapter 8 there would possibly be a benefit in allowing the user to place down additional markers at areas of sonic interest. This would allow points of interest to be quickly returned to and if stars were used it would give an indication of parameter values at that point. This idea could be further expanded to perhaps provide additional cues. For example, as map symbols are used to represent different geographic characteristics so different symbols could be used to show locations of different sonic characteristics. In addition, contour lines or shading could be used to show values of individual parameters across the surface of the space and create an interpolation contour map. It has also been shown in GIS that multivariate maps can be used to estimate desired locations within the maps [264]. A similar approach could be taken to not only find locations where desired sounds exist, but also estimate locations with specific sonic characteristics. This area will require considerably more investigation, but could prove a fruitful area of research after the star interpolator usability study has been completed.

9.9 Creativity Support Index

From the testing that was undertaken in both the pilot study (Chapter 6) and the usability study (Chapter 7) the use of SUS has shown that participants perceive the graphical interpolators as highly usable against sector wide benchmarks. Moreover, in the usability study although there was little deviation in the standard SUS scores between the different interpolators, the differences between them do largely match the participants preferred interface choices. However, efforts to provide more detail on the usability of interpolators for sound design, with the extended SUS, merely replicated the standard SUS results and failed to provide any additional detail.

Recently another potential measure has been proposed in the form of the Creativity Support Index (CSI) [265]. This is based on the NASA Task Load Index Survey, but it has been specifically modified for the evaluation of tools that support creative tasks. This should allow greater detail to be gained into the user experience. It has been applied to a number of different music technology systems [266], [267], [268], and appears to provide an additional metric for the evaluation of these systems. It has also been used to successfully evaluate another audio visualisation system similar to this research [269]. Moreover, in a number of these examples it has also been used in conjunction with SUS to provide additional insight into the creativity that is possible with the system, as well as its usability. Given that sound design is considered to be a creative task it would be desirable to evaluate graphical interpolators in this respect. Therefore, for all future testing both SUS and CSI will be applied. Also given that usability and creativity are not necessarily dependent on each other there might be value in repeating the testing of the different interpolation models to evaluate their creativity with CSI to examine if there is any correlation to the SUS scores. This will be considered further after the testing of the star interpolator visualisation has been completed using both SUS and CSI.

9.10 Synthesis Engine Transfer Function Visualization

Although the star interpolator visualisation provides an indication of how the parameter values change for movements within the space, it does not offer much insight as to how the parameter changes are translated to sonic results by the synthesis engine. For example, many different types of synthesis use the same component such as, oscillators, amplifiers, mixers, envelopes, LFOs (Low Frequency Oscillators), etc. and as such offer the same control parameters. How these control inputs are then translated into sonic outputs, that is, the system's transfer function, is dependent on how the components are connected together. Even when the same synthesis paradigm is applied each implementation may have a slightly different configuration of components so that each engine will have a different transfer function resulting in a unique sonic pallet. For example, changing the parameters of an envelope generator will result in different sonic changes dependent on where and how it is connected to other synthesis components. In this body of work the synthesis engine has remained fairly constant to aid the evaluation of the impact that the visualization has on the interpolation process. However, in normal circumstances the synthesis engine could and would be changed regularly for sound design. This being the case, an area of further investigation that could prove beneficial would be providing the user with some form of visualization for not only the parameter values, but also the system's transfer function. Hybrid graph visualization [270] could be considered for this as it would allow node-link diagrams to be used for the system's transfer functions and then other visualisations, for the parameter values. A generic example of a NodeTrix diagram is shown in Figure 80, where the node-links show the network and then matrix representations display the denser relationships of the individual parameters.

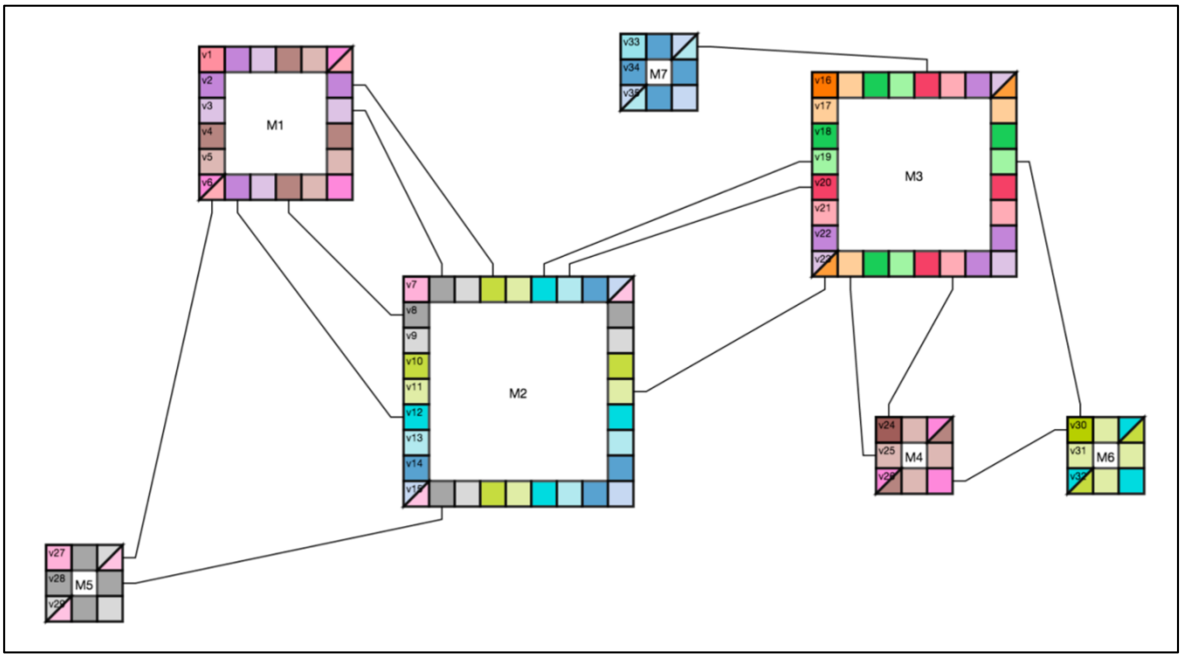


Figure 80 Example NodeTrix Diagram [271]

For example, in this context the node-links could be utilised to show the synthesis components and their interconnections and then maybe the star visualization could be used to show the values and changes of either perceptual or synthesis parameters. This area will require more in-depth study and investigation, but could provide substantial benefits in not only the visualisations for interpolation, but also the wider area of synthesis interface design.

9.11 Final Thoughts

Through this body of work, as well as all the specific points raised in this thesis it has been shown that interpolators offer an excellent platform for undertaking sound design applications. Sound design through interpolation has been shown to be easy and instinctive. It also appears that the exploratory nature of sound design bears a natural synergy to parameter searching offered through interpolation. The interpolators provide an intuitive interface for interacting with the often complex synthesis models that can be difficult to programme otherwise. This was backed-up by all the positive participant feedback gathered through the testing. Given the results from all the areas highlighted in this chapter, the use of interpolators in sound design shows future promise and has the potential to be developed further, either by this author or others.

9.11.1 Sound Design and Beyond

Although this body of work has focused on the application area of sound design, there are a number of other application areas where the new knowledge highlighted here maybe applicable.

Interpolation is used extensively in GIS to predict values in-between scattered data points. However, as the data points often relate to geographic features, the display normally relates to a cartographic map and not to the interpolation method or parameter values. However, given that it has been shown that the use of multivariate maps is possible [264], a display such as the Star Interpolator would allow users to see values when searching the space. Moreover, graphical interpolators could be used in any application area where users need to search for solutions that are constrained by known values. Examples such as, control of graphics, animation, texturing, image-processing, database transactions, avatar generation, game-level design, etc., may all benefit from the use of graphical interpolation. In all cases, a greater appreciation of the cues provided by alternative graphical models and their display will be invaluable for the selection of the interpolator. Finally, the knowledge contained in this thesis will provide those in other application areas with an insight into the importance of visual cues so that in future designs, due consideration can be given to the cues provided.

Appendix A Pilot Study Paperwork

Consent Form (04/01/17 – Version 1)

Participant Information Sheet (04/01/17 – Version 1)

Visual Interpolation Space for Sound Design - Questionnaire (04/01/17 – Version 1)

UNIVERSITY OF Southampton

CONSENT FORM *(04/01/17 – Version 1*)

Study title: Visual Interpolation Space for Sound Design

Researcher name: Darrell Gibson
Staff/Student number: 2 25808648
ERGO reference number: 25006

Please initial the box(es) if you agree with the statement(s):

I have read and understood the information sheet (insert date /version no. of participant information sheet) and have had the opportunity to ask questions about the study.

I agree to take part in this research project and agree for my (i) cursor movements, (ii) screencapture and (iii) questionnaire data to be used for the purpose of this study

I understand my participation is voluntary and I may withdraw at any time without my legal rights being affected

Data Protection
I understand that information collected about me during my participation in this study will be stored on a password protected computer and that this information will only be used for the purpose of this study. All files containing any personal data will be made anonymous.

Name of participant (print name)…………………………………………………………

Signature of participant…………………………………………………………………..

Date…………………………………………………………………………………………

UNIVERSITY OF Southampton

Participant Information Sheet (04/01/17 – Version 1)

Study Title: Visual Interpolation Space for Sound Design

Researcher: Darrell Gibson **Ethics number**: 25006

Please read this information carefully before deciding to take part in this research. If you are happy to participate you will be asked to sign a consent form.

What is the research about?
The research project is investigating suitable visual representations for Graphical User Interfaces (GUIs) that will be used for the design of sounds. The GUIs allow you to navigate a two dimensional pane with a mouse cursor that's position controls the sound generated by a synthesizer. Specific known sounds are placed at different locations within the pane and moving the cursor between these locations will create new sounds. To evaluate different visual representations for the pane, usability tests are being undertaken where you will be asked to design target sounds with different visualisations for the panes. Cursor movement data will then be collected from your interactions with the visual interface and this will be analysed to establish desirable properties. Your interactions with the pane will also be recorded (as a screen capture video) to try and identify trends in the usage of each interface that may not be evident from the captured interactions. Upon completion of each usability sound design task you will be asked to complete a questionnaire on your user experience.

As a participant in this study you will first be given an opportunity to audition the target sound three times before the test is started, by pressing the "Audition" button. When you begin the test with the "Start" button you will be presented with a two dimensional pane and will be given 10 minutes (maximum) to explore the sonic space defined by the pane, by moving the cursor around the pane area. When you believed that the target sound has been identified press the "Target" button. The test can be finished at anytime by pressing the "Stop" button or the test will automatically stop after 10 minutes. When the test has concluded you will then complete a questionnaire about the usability of the interface. Should you feel distressed by the sound output at anytime during the test it can be stopped by pressing the "Off" button. (The sound can then be reinstated with the corresponding "On" button).

This research is being undertaken by Darrell Gibson and is being completed through the Music Department at University of Southampton. When the participant testing has been completed it will be analysed to try and establish trends in the use of different visualisations for the interfaces. The results will then be published and used to inform the development of new interface visualisations.

Why have I been chosen?
You have been chosen for this study due to some familiarity with music technology and awareness of sound design principles. Only adults (18+ years) may participate in this study.

UNIVERSITY OF Southampton

What will happen to me if I take part?
If you agree to participate in this study you will undertake three sound design tasks, each using different visualisations for the control pane. Following each task a short questionnaire will be completed. The entire test should take approximately an hour to fully complete.

Are there any benefits in my taking part?
There is no individual benefit to the participating in this study, but there maybe a wider benefit to the music technology sector by adding to current knowledge.

Are there any risks involved?
The only minor risk from participating in the study is the potential of hearing damage. To minimise this risk the output of the audio system has been calibrated to a safe maximum level before the test is undertaken. You will then be able to adjust output sound a comfortable listening level, but will be restricted to the calibrated maximum. In addition, should you feel distressed to your hearing at anytime during the test the audio output can be halted by pressing the "Off" button. (The sound can then be reinstated with the corresponding "On" button).

Will my participation be confidential?
Participation in this study is completely anonymous and confidential. No personal data will be stored. Only numerical data from the mouse movement interaction with the interface will be stored and the results from the anonymous questionnaire. This will be stored on a password protected data server and will be stored in compliance with the Data Protection Act/University policy.

What happens if I change my mind?
If at any stage during the test you change your mind about participating then let the researcher (Darrell Gibson) know and you can stop immediately and any collected data will be deleted.

What happens if something goes wrong?
In the unlikely case of concern or complaint, you should provide a named independent contact with phone number and email address. This should normally be the Chair of the Faculty Ethics Committee Prof. Denis McManus (D.Mcmanus@soton.ac.uk).

Where can I get more information?
If you should have any further questions about this test or its context in the wider field ask the researcher (Darrell Gibson) in person or email (gibsond@bournemouth.ac.uk).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design

ERGO reference number: 25006

Test 1 Questionnaire: Participant number........... Interface number.............

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Test 1 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design

ERGO reference number: 25006

Test 2 Questionnaire: Participant number........... Interface number.............

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Test 2 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design

UNIVERSITY OF Southampton

ERGO reference number: 25006

Test 3 Questionnaire: Participant number........... Interface number.............

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Test 3 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

UNIVERSITY OF Southampton

Which of the three interfaces did you find best for the sound design task?

Test 1	Test 2	Test 3

Rate the preferred interface on the following scale:

Poor — Good

1	2	3	4	5	6	7	8	9	10

How many years have you used music technology:

Rate your competence in Sound Design:

Poor — Good

1	2	3	4	5	6	7	8	9	10

Final Comments

Please use the space below (and overleaf) to make any final comments:

Appendix B Usability Testing Paperwork

Consent Form (07/12/17 – Version 2)

Participant Information Sheet (07/12/17 – Version 3)

Usability Test - Sound Design Briefs - version 5

Visual Interpolation Space for Sound Design - Usability Testing - Questionnaire (07/12/17 – Version 1)

UNIVERSITY OF Southampton

CONSENT FORM *(07/12/17 – Version 2)*

Study title: Visual Interpolation Space for Sound Design - Usability Testing

Researcher name: Darrell Gibson
ERGO number: 31088

Please initial the box(es) if you agree with the statement(s):

I have read and understood the information sheet (07/12/17 – Version 2) and have had the opportunity to ask questions about the study.	
I agree to take part in this research project and agree for my data, (i) cursor movements, (ii) screencapture and (iii) questionnaire data, to be used for the purpose of this study.	
I understand my participation is voluntary and I may withdraw at any time for any reason without my rights being affected.	

Name of participant (print name)..

Signature of participant..

Date...

[07/12/17 Version 2] [31088]

UNIVERSITY OF Southampton

Participant Information Sheet (07/12/17 – Version 3)

Study Title: Visual Interpolation Space for Sound Design - Usability Testing

Researcher: Darrell Gibson
ERGO number: 31088

Please read this information carefully before deciding to take part in this research. It is up to you to decide whether or not to take part. If you are happy to participate you will be asked to sign a consent form.

What is the research about?
The research project is investigating suitable visual representations for Graphical User Interfaces (GUIs) that will be used for the design of sounds. The GUIs allow you to navigate a two dimensional pane with a mouse cursor that's position controls the sound generated by a synthesizer. Specific known sounds are placed at different locations within the pane and moving the cursor between these locations will create new sounds. To evaluate systems that use different visual representations for the pane, usability tests are being undertaken where you will be asked to design target sounds with different visualisations for the control panes. Cursor movement data will then be collected from your interactions with the visual interface and this will be analysed to establish desirable properties. Your interactions with the pane will also be recorded (as a screen capture video) to try and identify trends in the usage of each interface that may not be evident from the captured interactions. Upon completion of each usability sound design task you will be asked to complete a questionnaire on your user experience.

As a participant in this study you will first be given a "brief" detailing the type of sound that requires designing and a visual context where it is intended that the designed sound will be used. You can then begin the test by pressing the "Start" button you will be presented with a two dimensional graphical pane and will be given 10 minutes (maximum) to explore the sonic space defined by the pane, by moving the cursor around the pane area. When you believe that you have found a target sound that meets the specification in the brief you can register this by pressing the "Target" button. The test can be finished at anytime by pressing the "Stop" button or the test will automatically stop after 10 minutes. When the test has concluded you will then be asked to complete a questionnaire about the usability of the interface you have just used. Should you feel distressed by the sound output at anytime during the test it can be stopped by pressing the "Off" button. (The sound can then be reinstated with the corresponding "On" button).

As a participant in this study you will undertake six different sound design tasks, each using a different system with its own unique visualisations for the control pane. When each task is completed you will answer a short questionnaire to assess the usability of the system used. When all tasks are completed a final questionnaire will be undertaken to compare the merits of each interface. The entire test should take no longer than two hours to fully complete.

This research is being undertaken by Darrell Gibson and is being completed through the Music Department at University of Southampton. When the participant testing has been completed the data will be analysed to try and establish trends in the use of different systems and their visual interfaces. The results will then be published and used to inform the development of new interface visualisations.

Why have I been asked to participate?
You have been chosen for this study due to some familiarity with music technology and awareness of sound design principles. Only adults (18+ years) may participate in this study.

07/12/17 – Version 3 Ethics Number: 31088

UNIVERSITY OF Southampton

What will happen to me if I take part?
If you agree to participate in this study you will undertake six sound design tasks, each using a different system with unique visualisations for the control pane. Following each task a short questionnaire will be completed. The entire test should take approximately two hours to fully complete.

Are there any benefits in my taking part?
There is no individual benefit to the participating in this study, but there maybe a wider benefit to the music technology sector by adding to current knowledge.

Are there any risks involved?
The only minor risk from participating in the study is the potential of hearing damage. To minimise this risk the output of the audio system has been calibrated to a safe maximum level before the test is undertaken. You will then be able to adjust output sound a comfortable listening level, but will be restricted to the calibrated maximum. In addition, should you feel distressed to your hearing at anytime during the test the audio output can be halted by pressing the “Off” button. (The sound can then be reinstated with the corresponding “On” button).

Will my participation be confidential?
Participation in this study is completely anonymous and confidential. No personal data will be stored. Only numerical data from the mouse movement interaction with the interface will be stored and the results from the anonymous questionnaire. This will be stored on a password protected data server and will be stored in compliance with the Data Protection Act/University policy.

What should I do if I want to take part?
If you want to participate in this study you can inform the researcher (Darrell Gibson) in person or email (gibsond@bournemouth.ac.uk). You will then be provided with a Consent Form to read and complete, before the study and be started.

What happens if I change my mind?
If at any stage during the test you change your mind about participating then let the researcher (Darrell Gibson) know and you can stop immediately and any collected data will be deleted.

What will happen to the results of the research?
The resulting data will be evaluated, written-up and disseminated through the publication of the results. These will be made available through UoS ePrints server. Following publication the individual data collection will be securely destroyed. When the project ends all the analysed data will be securely destroyed.

Where can I get more information?
If you should have any further questions about this test or its context in the wider field ask the researcher (Darrell Gibson) in person or email (gibsond@bournemouth.ac.uk).

What happens if something goes wrong?
In the unlikely case of concern or complaint, please contact the Research Integrity and Governance Manager (023 8059 5058, rgoinfo@soton.ac.uk). The University has insurance in place to cover its legal liabilities in respect of this study.

Thank you.

07/12/17 – Version 3 Ethics Number: 31088

UNIVERSITY OF Southampton

Thank you for taking the time to read the information sheet and considering to take part in this research study.

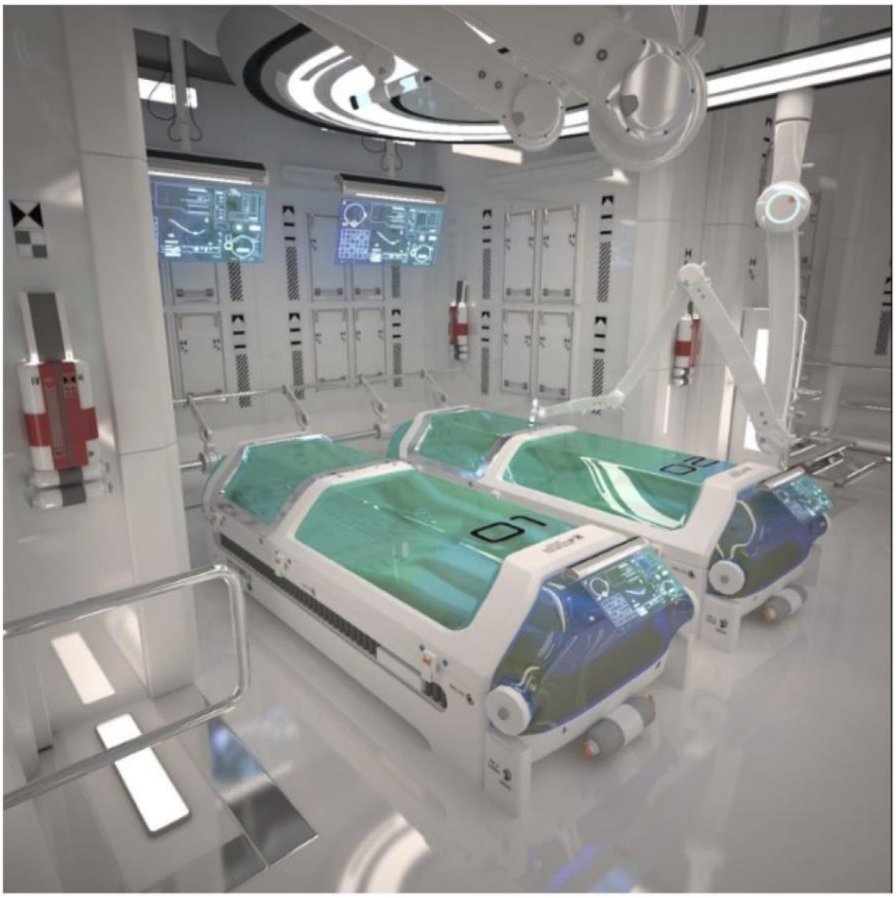
07/12/17 – Version 3

Ethics Number: 31088

Interpolator 1 – Sound Design Brief

You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a "soundbed", over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:

Soothing and healing sound for a medical hospital spaceship:



Interpolator 2 – Sound Design Brief

You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a "soundbed", over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:

Manic and chaotic sound for a spaceship owned by a psychopath:

Interpolator 3 – Sound Design Brief

You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a "soundbed", over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:

Calm and tranquil sound for a spaceship owned by a battle hero:

Interpolator 4 – Sound Design Brief

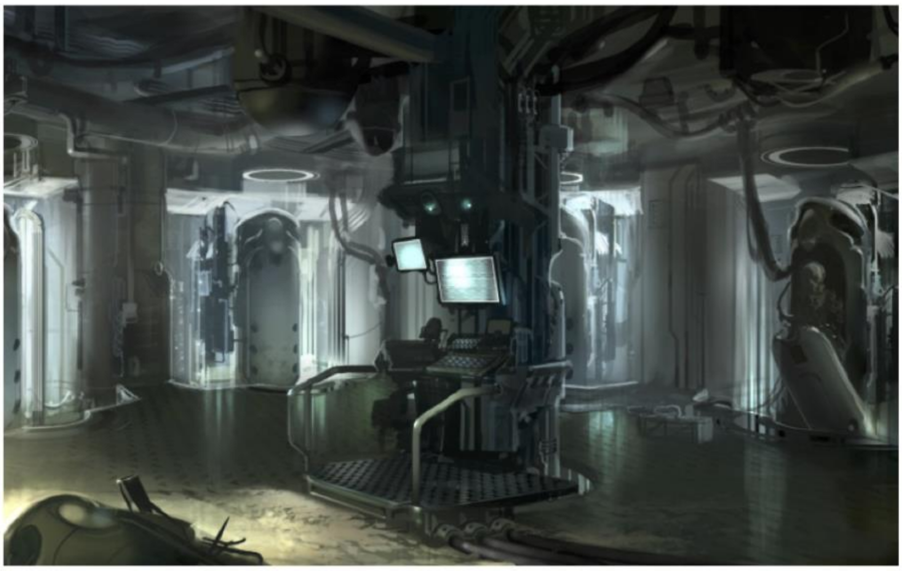
You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a “soundbed”, over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:

Threatening and scary sound for a spaceship where a killer is hunting the crew members:

Interpolator 5 – Sound Design Brief

You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a "soundbed", over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:

Despair and despondency for the sound spacecraft that is stranded in deep space with no engines and dwindling life-support systems:



Interpolator 6 – Sound Design Brief

You have been tasked with designing a background ambience sound for a spacecraft in a science fiction film. The sound will be used as the internal sound of the spacecraft and will be used as a "soundbed", over which other soundtrack elements (dialogue, music, effects) will be added. The level and pitch of the ambience will be adjusted during post-production so you should just concentrate on designing the tonal characteristic of the sound to best meet the following criteria:

Sombre and gloomy sound for a dying spaceship that is being eaten by parasitic space slime:

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design - Usability Testing ERGO reference number: 31088

Test 1 Questionnaire: Participant number………… Interface number…………..

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5
11. It was intuitive how the resulting sound was being generated.	1	2	3	4	5
12. The system's control mechanism was not obvious	1	2	3	4	5

UNIVERSITY OF Southampton

13. The visual display aided the exploration of the sound space

1	2	3	4	5

14. The system produced unexpected sounds

1	2	3	4	5

Test 1 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design - Usability Testing ERGO reference number: 31088

Test 2 Questionnaire: Participant number........... Interface number.............

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5
11. It was intuitive how the resulting sound was being generated.	1	2	3	4	5
12. The system's control mechanism was not obvious	1	2	3	4	5

UNIVERSITY OF Southampton

13. The visual display aided the exploration of the sound space

1	2	3	4	5

14. The system produced unexpected sounds

1	2	3	4	5

Test 2 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design - Usability Testing ERGO reference number: 31088

Test 3 Questionnaire: Participant number………… Interface number…………..

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5
11. It was intuitive how the resulting sound was being generated.	1	2	3	4	5
12. The system's control mechanism was not obvious	1	2	3	4	5

UNIVERSITY OF Southampton

13. The visual display aided the exploration of the sound space

1	2	3	4	5

14. The system produced unexpected sounds

1	2	3	4	5

Test 3 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design - Usability Testing ERGO reference number: 31088

Test 4 Questionnaire: Participant number………… Interface number…………..

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5
11. It was intuitive how the resulting sound was being generated.	1	2	3	4	5
12. The system's control mechanism was not obvious	1	2	3	4	5

UNIVERSITY OF Southampton

13. The visual display aided the exploration of the sound space

1	2	3	4	5

14. The system produced unexpected sounds

1	2	3	4	5

Test 4 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design - Usability Testing ERGO reference number: 31088

Test 5 Questionnaire: Participant number………… Interface number…………..

	Strongly disagree 1	2	3	4	Strongly agree 5
1. I think that I would like to use this system frequently					
2. I found the system unnecessarily complex					
3. I thought the system was easy to use					
4. I think that I would need the support of a technical person to be able to use this system					
5. I found the various functions in this system were well integrated					
6. I thought there was too much inconsistency in this system					
7. I would imagine that most people would learn to use this system very quickly					
8. I found the system very cumbersome to use					
9. I felt very confident using the system					
10. I needed to learn a lot of things before I could get going with this system					
11. It was intuitive how the resulting sound was being generated.					
12. The system's control mechanism was not obvious					

13. The visual display aided the exploration of the sound space

1	2	3	4	5

14. The system produced unexpected sounds

1	2	3	4	5

Test 5 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Visual Interpolation Space for Sound Design - Usability Testing ERGO reference number: 31088

Test 6 Questionnaire: Participant number………… Interface number…………..

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5
11. It was intuitive how the resulting sound was being generated.	1	2	3	4	5
12. The system's control mechanism was not obvious	1	2	3	4	5

UNIVERSITY OF Southampton

13. The visual display aided the exploration of the sound space

1	2	3	4	5

14. The system produced unexpected sounds

1	2	3	4	5

Test 6 - Critical Incidents

Please use the space below (and overleaf) to report any critical incidents that occur during the test, both positive (something worked particularly well or had a positive effect) and negative (system failed to respond as expected or produced poor results).

UNIVERSITY OF Southampton

Rank the six interfaces used in order of preference for the sound design tasks, with the best being 1st and the least favourite being 6th.

Test 1	
Test 2	
Test 3	
Test 4	
Test 5	
Test 6	

Rate the preferred interface on the following scale:

Poor — Good

1	2	3	4	5	6	7	8	9	10

With the preferred interface how confidents were you that you were able to design the best possible sound from the given starting points:

Poor — Good

1	2	3	4	5	6	7	8	9	10

To what extent did the preferred system allow you to create sounds that you not ordinarily have discovered:

Poor — Good

1	2	3	4	5	6	7	8	9	10

How many years have you used music technology:

Rate your competence in Sound Design:

Poor — Good

1	2	3	4	5	6	7	8	9	10

UNIVERSITY OF Southampton

Final Comments

Please use the space below (and overleaf) to make any final comments:

Glossary of Terms

Framework	A software framework is an abstraction and template that provides generic functionality for developing specific software applications. This is achieved through code reuse using libraries.

List of References

[1] Mancini, M. The Sound Designer. Film Sound: Theory and Practice. Weiss. and Belton, J.(eds) op. cit. 1985.

[2] Gibbs, T. The Fundamentals of Sonic Art and Sound Design. Ava Publishing; 2007 Jul 10.

[3] Ament, V. T. The Foley Grail: The Art of Performing Sound for Film, Games, And Animation. CRC Press; 2014 Apr 3.

[4] Altman, R. The Material Heterogeneity of Recorded Sound. Sound Theory, Sound Practice. 1992:15-31.

[5] Pauletto, S. Editorial. The New Soundtrack - Perspectives on Sound Design. 2014;4(2):v-vi.

[6] Holland S, Mudd T, Wilkie-McKenna K, McPherson A, Wanderley MM. Understanding music interaction, and why it matters. In New directions in music and human-computer interaction 2019 (pp. 1-20). Springer, Cham.

[7] Bødker S. When second wave HCI meets third wave challenges. In Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles 2006 Oct 14 (pp. 1-8).

[8] Bødker S. Third-wave HCI, 10 years later---participation and sharing. Interactions, August 2015 28; 22(5):24-31.

[9] Tanaka A. Embodied musical interaction. In New Directions in Music and Human-Computer Interaction 2019 (pp. 135-154). Springer, Cham.

[10] Brown D, Nash C, Mitchell T. A User Experience Review of Music Interaction Evaluations. In Proceedings New Interfaces for Musical Expression, May 2017.

[11] Crafton, D. The talkies: American cinema's transition to sound, 1926-1931. University of California Pr, 1999.

[12] Sound Of Cinema: The Music That Made The Movies - The Big Score. City, 2013.

[13] Holman, T. Sound for film and television. Taylor & Francis, 2010.

[14] Chion, M., Gorbman, C. and Murch, W. Audio-vision: sound on screen. Columbia University Press, 1994.

[15] Steiff, J. The Complete Idiot's Guide to Independent Filmmaking. Penguin, 2005.

[16] Sonnenschein, D. Sound design: The expressive power of music, voice and sound effects in cinema. Michael Wiese Productions, 2001.

[17] Yewdall, D. L. Practical art of motion picture sound. Taylor & Francis US, 2007.

[18] Sergi, G. In defence of Vulgarity: the place of sound effects in the cinema. Scope: An online journal of film and television studies: ISSN 1465-9166, 27, June (2006), 2008.

[19] Wright, B. What do we hear? The pluralism of sound design in Hollywood sound production. The New Soundtrack, 3, 2 (2013), 137-157.

[20] Burtt, B. Ben Burtt-Sound Designer of Star Wars. Filmsound. Ed. Sven E. Carlsson, 15 (1999).

[21] Sergi, G. Tales of the Silent Blast: Star Wars and Sound. Journal of Popular Film and Television, 26, 1 (1998), 12-22.

[22] Jarrett, M. Sound Doctrine: An Interview with Walter Murch. FILM QUARTERLY-BERKELEY-, 53, 3 (2000), 2-11.

[23] Ben Burtt - Awards - IMDb. The Internet Movie Database. IMDb.com, Inc, n.d. Web., City, 2018.

[24] Walter Murch - Awards - IMDb. The Internet Movie Database. IMDb.com, Inc, n.d. Web., City, 2018.

[25] Ben Burtt Special: Star Wars – The Sounds [Part 1]. Designing Sound, Art and techniques of sound design, designingsound.org, Web, 10 September 2009, City, 2018.

[26] Sragow, M. The sound of Vietnam: How wizard Walter Murch created a soundtrack of horror for Francis Ford Coppola's "Apocalypse Now.". Salon Media Group, Salon.com, Web, APR 27, 2000, City, 2000.

[27] Laudadio N. The Synthesizer. Electronic Musical Sound in Film Song and Sound. Harper G, Doughty R, Eisentraut J, editors: Bloomsbury Publishing USA; 2014.

[28] Niebur, L. Special Sound: the creation and legacy of the BBC Radiophonic Workshop. Oxford University Press, 2010.

[29] Kaye, D. and LeBrecht, J. Sound and Music for the Theatre: The Art & Technique of Design. Taylor & Francis, 2013.

[30] Kulezic-Wilson, D. Sound design is the new score. Music, Sound, and the Moving Image, 2, 2 (2008), 127-131.

[31] Puronas, V. Sonic hyperrealism: illusions of a non-existent aural reality. The New Soundtrack, 4, 2 (2014), 181-194.

[32] Pauletto, S. The sound design of cinematic voices. The New Soundtrack, 2, 2 (2012), 127-142.

[33] Yang, W. and Kang, J. Soundscape and sound preferences in urban squares: a case study in Sheffield. Journal of Urban Design, 10, 1 (2005), 61-80.

[34] Théberge, P. Sound maps: music and sound in cybercartography. Modern Cartography Series, 4 (2005), 389-410.

[35] Serafin, S. and Serafin, G. Sound Design to Enhance Presence in Photorealistic Virtual Reality. City, 2004.

[36] Alves, V. and Roque, L. A deck for sound design in games: enhancements based on a design exercise. ACM, City, 2011.

[37] Gal, V., Le Prado, C., Merland, J., Natkin, S. and Vega, L. Processes and tools for sound design in computer games. City, 2002.

[38] Tsang, L. 8. SOUND AND MUSIC IN WEBSITE DESIGN. Music, Sound and Multimedia: From the Live to the Virtual (2007), 145.

[39] Nykänen, A. Methods for product sound design. Department of Human Work Sciences, Luleå University of Technology, 2008.

[40] Rocchesso, D., Serafin, S., Behrendt, F., Bernardini, N., Bresin, R., Eckel, G., Franinovic, K., Hermann, T., Pauletto, S. and Susini, P. Sonic interaction design: sound, information and experience. ACM, City, 2008.

[41] Jee, E.-S., Jeong, Y.-J., Kim, C. H. and Kobayashi, H. Sound design for emotion and intention expression of socially interactive robots. Intelligent Service Robotics, 3, 3 (2010), 199-206.

[42] Suied, C., Susini, P., Misdariis, N., Langlois, S., Smith, B. K. and McAdams, S. Toward a sound design methodology: Application to electronic automotive sound. In Proceedings of the Proceedings of ICAD 05-Eleventh Meeting of the International Conference on Auditory Display (Limerick, Ireland, July 6-9,, 2005).

[43] Pirhonen, A., Tuuri, K., Mustonen, M.-S. and Murphy, E. Beyond clicks and beeps: In pursuit of an effective sound design methodology. Springer, City, 2007.

[44] Visell, Y., Fontana, F., Giordano, B. L., Nordahl, R., Serafin, S. and Bresin, R. Sound design and perception in walking interactions. International Journal of Human-Computer Studies, 67, 11 (2009), 947-959.

[45] Hug, D. Investigating narrative and performative sound design strategies for interactive commodities. Springer, City, 2010.

[46] Schaffert, N., Mattes, K. and Effenberg, A. O. A sound design for the purposes of movement optimisation in elite sport (using the example of rowing) (2009).

[47] d'Escriván, J. Sound Art (?) on/in Film. Organised Sound, 14, 01 (2009), 65-73.

[48] Dalton, D. Synthesized Sound Effects For Star Trek The Motion Picture. City, 1980.

[49] Kaminskij, K. Electronic Synthesis of Special Sound Effects in Soviet vs. Electrified Voices: Medial, Socio-Historical and Cultural Aspects of Voice Transfer (2012), 273.

[50] Serafine, F. Sound Effects Design and Synthesis for TRON. City, 1982.

[51] Koldau, L. M. Sound effects as a genre-defining factor in submarine films. MedieKultur. Journal of media and communication research, 26, 48 (2009), 13.

[52] Burtt B. Sound of Indiana Jones. 2003.

[53] Blocker, J. Synthesized sound for Brainstorm. City, 1984.

[54] Candusso, D. Aural Landscapes: Designing a sound environment for screen. Screen Sound Journal, 3 (2012).

[55] Farnell, A. An introduction to procedural audio and its application in computer games. City, 2007.

[56] Böttcher, N. Current problems and future possibilities of procedural audio in computer games. Journal of Gaming & Virtual Worlds, 5, 3 (2013).

[57] Thom R. Designing a movie for sound. IRIS-PARIS-. 1999;27:9-20.

[58] Interview of Ben Burtt in the Star Wars Trilogy: *The Definitive Collection*, Laserdisc Box Set, 1993.

[59] Avid Everywhere Momentum Continues With New Pro Tools - S6 Version 2.1 Software and Innovative Mixing Modules. 2016. [Available from: https://globenewswire.com/news-release/2016/01/21/803757/0/en/Avid-Everywhere-Momentum-Continues-With-New-Pro-Tools-S6-Version-2-1-Software-and-Innovative-Mixing-Modules.html].

[60] Young J. Sound morphology and the articulation of structure in electroacoustic music. Organised sound. 2004 Apr 1;9(01):7-14.

[61] Teruggi D. Technology and musique concrète: the technical developments of the Groupe de Recherches Musicales and their implication in musical composition. Organised Sound. 2007 Dec 1;12(3):213.

[62] Carlsson S. E. The Sound Design of Star Wars. 2016 [Available from: http://filmsound.org/starwars/].

[63] Butler, M. J. Unlocking the groove: Rhythm, meter, and musical design in electronic dance music. Indiana University Press, 2006.

[64] Brown, B. The noise instruments of Luigi Russolo, Perspectives of New Music, vol. 20, no. 1/2, pp. 31-48, 1981.

[65] Vail, M. Keyboard Magazine presents Vintage synthesizers: pioneering designers, groundbreaking instruments, collecting tips, mutants of technology: Backbeat Books, 2000.

[66] Curtis Roads. The computer music tutorial. MIT press; 1996.

[67] Russ M. Sound synthesis and sampling: Third Edition, Routledge; 2012.

[68] Farnell A. Designing Sound. Cambridge: Mit Press; 2010 Oct 29.

[69] F. Avanzini FF, L. Ottaviani, M. Rath, D. Rocchesso. Models and Algorithms for Sounding Objects. Project funded by the European Community under the 'Information Society Technologies' Programme (1998-2002). Report No.: Contract No.: SOb - the Sounding Object IST Project no. IST-2000-25287.

[70] Moss W, Yeh H, Hong JM, Lin MC, Manocha D. Sounding liquids: Automatic sound synthesis from fluid simulation. ACM Transactions on Graphics (TOG). 2010 Jun 1;29(3):21.

[71] Miklavcic SJ, Zita A, Arvidsson P. Computational real-time sound synthesis of rain. Department of Science and Technology (ITN), Campus Norrköping, Linköping University [Institutionen för teknik och naturvetenskap (ITN), Campus Norrköping, Linköpings universitet]; 2004.

[72] Chadwick JN, James DL. Animating fire with sound. In ACM Transactions on Graphics (TOG) 2011 Aug 7 (Vol. 30, No. 4, p. 84). ACM.

[73] An SS, James DL, Marschner S. Motion-driven concatenative synthesis of cloth sounds. ACM Transactions on Graphics (TOG). 2012 Aug 5;31(4):102.

[74] Kahrs M, Avanzini F. Computer synthesis of bird songs and calls. In Proc. COST G6 Conf. on Digital Audio Effects (Limerick, Ireland, December 2001) 2001 Dec (pp. 23-7).

[75] Tschuch G, Brothers DJ. Modeling vibration and sound production in insects with nonresonant stridulatory organs. The Journal of the Acoustical Society of America. 1999 Dec 1;106(6):3706-10.

[76] Martino R. Synthasaurus: An Animal Vocalization Synthesizer (Masters dissertation, NorthWestern University), 2000.

[77] Cook PR. Modeling Bill's gait: Analysis and parametric synthesis of walking sounds. In Audio Engineering Society Conference: 22nd International Conference: Virtual, Synthetic, and Entertainment Audio 2002 Jun 1. Audio Engineering Society.

[78] Reiss J, Hendry S. Physical Modeling and Synthesis of Motor Noise for Replication of a Sound Effects Library. In Audio Engineering Society Convention 129 2010 Nov 4. Audio Engineering Society.

[79] Baldan S, Lachambre H, Monache SD, Boussard P. Physically informed car engine sound synthesis for virtual and augmented environments. In Sonic Interactions for Virtual Environments (SIVE), 2015 IEEE 2nd VR Workshop on 2015 Mar 24 (pp. 1-6). IEEE.

[80] LE Sound by AudioGaming - Testimonials, 2019 [Available from: http://lesound.io/testimonials/].

[81] iTsugi’s GameSynth - The Procedural Audio Middleware, 2019 [Available from: http://www.procedural-audio.com].

[82] LE Sound by AudioGaming - Products, 2019 [Available from: http://lesound.io/product-category/plugins/].

[83] Krotos - Professional Audio Software for Creative Minds, 2019 [Available from: https://www.krotosaudio.com/]

[84] Horner A. Evolution in digital audio technology. In Evolutionary Computer Music 2007 (pp. 52-78). Springer London.

[85] Dahlstedt P. Evolution in creative sound design. In Evolutionary computer music 2007 (pp. 79-99). Springer London.

[86] Grey, J., and J. Moorer, *Perceptual Evaluations Of Synthesized Musical Instrument Tones*, Journal of Acoustical Society of America, Volume 62, Issue 2, pages 454-462, 1977.

[87] Moorer, J., The Use of the Phase Vocoder in Computer Music Applications, Audio Engineering Journal, Volume 26, Issue 1, 1978.

[88] Kreutzer, C., J. Walker and M. O'Neill, *A Parametric Model for Spectral Sound Synthesis of Musical Sounds*, Proceedings of the IEEE International Conference on Audio, Language and Image Processing (ICALIP) 2008, pages 633-637.

[89] Le Groux, S., and P. Verschure, *Perceptsynth: Mapping Perceptual Musical Features to Sound Synthesis Parameters*, Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), March 30 – April 4, 2008, Las Vegas, USA.

[90] Sethares, W.A, A.J. Milne, S. Tiedje, A. Prechtl, and J. Plamondon, *Spectral Tools for Dynamic Tonality and Audio Morphing*. Computer Music Journal, Summer 2009, Volume 33, Issue 2, pages 71-84.

[91] Misra, A., P. R. Cook and G. Wang, A New Paradigm For Sound Design, Proceedings of 9th International Conference on Digital Audio Effects (DAFx-06), Montreal, Canada, September 18-20, 2006.

[92] Misra A. Tapestrea: techniques and paradigms for expressive synthesis, transformation, and re-composition of environmental audio. Princeton University; 2009 Jan 1.

[93] Misra A, Wang G, Cook P. Musical Tapestry: Re-composing Natural Sounds†. Journal of New Music Research. 2007 Dec 1;36(4):241-50.

[94] Klingbeil M. Software for spectral analysis, editing, and synthesis. In Proceedings of the International Computer Music Conference 2005, Sep (pp. 107-110).

[95] Klingbeil, M., Spectral Analysis, Editing, and Resynthesis: Methods and Applications, Doctor of Musical Arts in the Graduate School of Arts and Sciences, Columbia University, 2009.

[96] Quackenbush, S., and A. Lindsay, *Overview of MPEG-7 Audio*, IEEE Transactions On Circuits And Systems For Video Technology, Volume 11, Number 6, June 2001.

[97] Casey, M., General Sound Classification and Similarity in MPEG-7, Organised Sound, Volume 6, Issue 2, Cambridge University Press, 2002.

[98] Hoffman, M., and P. Cook, Feature-Based Synthesis: Mapping from Acoustic and Perceptual Features to Synthesis Parameters, Proceedings of the International Computer Music Conference, New Orleans, 2006.

[99] Schwarz, D., R. Cahen and S. Britton, Principles And Applications Of Interactive Corpus-Based Concatenative Synthesis, Journées d'Informatique Musicale (JIM). Albi : Mars 2008.

[100] Schwarz D, Britton S, Cahen R, Goepfer T. Musical applications of real-time corpus-based concatenative synthesis. In International Computer Music Conference (ICMC) 2007 (pp. 47-50).

[101] Miranda, E. R., An Artificial Intelligence Approach to Sound Design, Computer Music Journal, Volume 19, Issue 2, pages 59-75, 1995.

[102] Miranda, E. R., Machine Learning and Sound Design, Leonardo Music Journal, Volume 7, pages 49-55, 1998.

[103] Garcia, R. A., Growing Sound Synthesizers Using Evolutionary Methods, Proceedings of ALMMA 2001: Artificial Life Models for Musical Applications Workshop (ECAL).

[104] Garcia, R. A., Automating The Design Of Sound Synthesis Techniques Using Evolutionary Methods, Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-01), Limerick, Ireland, December 6-8, 2001.

[105] Dahlstedt, P., Creating and Exploring Huge Parameter Spaces: Interactive Evolution as a Tool for Sound Generation, Proceedings of 2001 International Computer Music Conference, Havana, Cuba, ICMA.

[106] Johnson, C. G., Exploring Sound-Space With Interactive Genetic Algorithms. Leonardo Music Journal, Volume 36, Issue 1, pages 51-54, 2003.

[107] Schatter, G., E. Zuger and C. Nitschke, A Synaesthetic Approach For A Synthesizer Interface Based On Genetic Algorithms And Fuzzy Sets. Proceeding of International Computer Music Conference 2005.

[108] McDermott, J., N. J. L. Griffith and M. O'Neill, Toward User-Directed Evolution of Sound Synthesis Parameters, Applications on Evolutionary Computing, Springer, pages 517– 526, 2005.

[109] McDermott, J., N. J. L. Griffith and M. O'Neill, Evolutionary GUIs for Sound Synthesis. Proceedings of Fifth European Workshop on Evolutionary Music and Art (EvoMUSART), 2007.

[110] McDermott, J., N. J. L. Griffith and M. O'Neill, Interactive EC Control of Synthesized Timbre, Evolutionary Computation, Volume 18, Issue 2, pages 277–303, 2010.

[111] Yee-King, M., and M. Roth, SynthBot: An Unsupervised Software Synthesizer Programmer, International Computer Music Conference 2008.

[112] Dykiert, M., and N.E. Gold, Support for Learning Synthesiser Programming, Proceedings of 8th Sound and Music Computing Conference 2011, 6 – 9 July 2011, Padova, Italy.

[113] Miranda, E. R. At the Crossroads of Evolutionary Computation and Music: Self- Programming Synthesizers, Swarm Orchestras and the Origins of Melody. Evolutionary Computation, Volume 12, Issue 2, pages 137-158, 2004.

[114] Jenkins, M. Analog Synthesizers: Understanding, Performing, Buying. Focal Press, 2007.

[115] Hunt A, Wanderley M, Kirk R. Towards a model for instrumental mapping in expert musical interaction. In Proceedings of the 2000 International Computer Music Conference 2000 Sep (pp. 209-212).

[116] Hunt, A. & Wanderley, M.M., Mapping Performer Parameters to Synthesis Engines. Organised Sound, Volume 7, Issue 2, pages 97–108, 2002.

[117] Hunt, A., M. Wanderley, and M. Paradis, The Importance Of Parameter Mapping In Electronic Instrument Design. Journal of New Music Research, Volume 32, Issue 4, page 429–440, 2003.

[118] Goudeseune, C., Interpolated Mappings for Musical Instruments. Organised Sound, 7(2):85–96, 2002.

[119] Momeni A, Wessel D. Characterizing and controlling musical material intuitively with geometric models. In Proceedings of the 2003 conference on New interfaces for musical expression 2003 May 22 (pp. 54-62). National University of Singapore.

[120] KORG KAOSS PAD KP-3 Dynamic Effect/Sampler, Owner's Manual. 2006.

[121] Rovan JB, Wanderley MM, Dubnov S, Depalle P, editors. Instrumental gestural mapping strategies as expressivity determinants in computer music performance. Proceedings of Kansei-The Technology of Emotion Workshop; 1997.

[122] Wanderley MM, Depalle P. Gestural control of sound synthesis. Proceedings of the IEEE. 2004; 92(4): 632-44.

[123] Bencina R. The metasurface: applying natural neighbour interpolation to two-to-many mapping. In Proceedings of the 2005 conference on New interfaces for musical expression 2005 May 1 (pp. 101-104). National University of Singapore.

[124] Paine G. Towards unified design guidelines for new interfaces for musical expression. Organised Sound. 2009 Aug 1;14(2):142-55.

[125] Bowler I, Purvis A, Manning P, Bailey NJ. On mapping N articulation onto M synthesiser-control parameters. In Proceedings of the International Computer Music Conference 1990 (pp. 181-184).

[126] Van Nort, D., M. Wanderley and Philippe Depalle. On the Choice of Mappings based on Geometric Properties. Proceedings of the 2004 International Conference on New Interfaces for Musical Expression (NIME 04), Hamamatsu, Japan, June 3-5, 2004.

[127] Wanderley MM, Schnell N, Rovan J. Escher-modeling and performing composed instruments in real-time. In Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on 1998 Oct 11 (Vol. 2, pp. 1080-1084).

[128] Lam NS. Spatial interpolation methods: a review. The American Cartographer. 1983 Jan 1;10(2):129-50.

[129] Van Nort D, Wanderley M. The LoM Mapping Toolbox for Max/MSP/Jitter. In Proc. of the 2006 International Computer Music Conference (ICMC) 2006 (pp. 397-400).

[130] Choi I, Bargar R, Goudeseune C. A manifold interface for a high dimensional control space. In Proceedings of the 1995 International Computer Music Conference 1995 (pp. 385-392).

[131] Bevilacqua F, Müller R, Schnell N. MnM: a Max/MSP mapping toolbox. In Proceedings of the 2005 conference on New interfaces for musical expression 2005 May 1 (pp. 85-88). National University of Singapore.

[132] Van Nort D, Wanderley MM. Exploring the effect of mapping trajectories on musical performance. In Proc. 2006 Sound and Music Computing Conference (SMC) 2006 Jun 4 (pp. 19-24).

[133] Van Nort D, Wanderley M. Control strategies for navigation of complex sonic spaces. In Proceedings of the 7th international conference on New interfaces for musical expression 2007 Jun 6 (pp. 379-382). ACM.

[134] Van Nort D, Wanderley MM, Depalle P. Mapping control structures for sound synthesis: functional and topological perspectives. Computer Music Journal. 2014 Sep;38(3):6-22.

[135] Grey J. M. Multidimensional perceptual scaling of musical timbres. The Journal of the Acoustical Society of America. 1977 May 1;61(5):1270-7.

[136] Wessel D. L. Timbre space as a musical control structure. Computer music journal. 1979 Jun 1:45-52.

[137] Vertegaal, R., and E. Bonis, *ISEE: An Intuitive Sound Editing Environment*. Computer Music Journal, Volume 18, Issue 2, pages 21-29, 1994.

[138] Vertegaal, R., and B. Eaglestone, Comparison of input devices in an ISEE direct timbre manipulation task. Interacting with Computers, Butterworth- Heinemann, 1996.

[139] Rolland PY, Pachet F. A framework for representing knowledge about synthesizer programming. Computer Music Journal. 1996 Oct 1;20(3):47-58.

[140] Ethington, R. and B. Punch, SeaWave: A System for Musical Timbre Description. Computer Music Journal, Volume 18, Issue 1, pages 30-39, 1994.

[141] Gounaropoulos, A., and C. Johnson, *Synthesising Timbres and Timbre- Changes from Adjectives/Adverbs*. Applications of Evolutionary Computing, Volume, 3907 of Lecture Notes in Computer Science, pages 664-675, 2006.

[142] Aramaki M., Kronland-Martinet R., Voinier Th. & S. Ystad, *Timbre Control Of Real-Time Percussive Synthesizer*. Proceedings of 19th International Congress On Acoustics Madrid, 2-7 September 2007, Madrid, Spain.

[143] Nicol, C. A., *Development and Exploration of a Timbre Space Representation of Audio*. Ph.D from Department of Computing Science. Glasgow, University of Glasgow, 2005.

[144] Seago, A., A new user interface for musical timbre design. Ph.D in Music Computing, Open University, 2009.

[145] Seago, A., S. Holland, S., and P. Mulholland, *A Novel User Interface for Musical Timbre Design*. 128th Convention of the Audio Engineering Society, London, May 2010.

[146] Loureiro, M. A., H. B. D. Paula and H. C. Yehia, *Timbre Classification Of A Single Musical Instrument*. In 5th International Conference on Music Information Retrieval (ISMIR 04), 2004.

[147] Burred, J.J., A. Röbel and X. Rodet, *An Accurate Timbre Model for Musical Instruments and its Application to Classification*. Proceedings of International Workshop on Learning the Semantics of Audio Signals (LSAS), Athens, Greece, December 2006.

[148] Zacharakis, A., K. Pastiadis, G. Papadelis and J. Reiss, *An Investigation of Musical Timbre: Uncovering Salient Semantic Descriptors and Perceptual Dimensions*. 12th International Society for Music Information Retrieval Conference (ISMIR 2011), pages 807-812.

[149] Winkler, T., Making Motion Musical: *Gestural Mapping Strategies For Interactive Computer Music*. In Proceedings of 1995 International Computer Music Conference, pages 261–264.

[150] Camurri C., De Poli G., Leman M., Volpe G., *A Multi-layered Conceptual Framework for Expressive Gesture Applications*. Workshop on Current Research Directions in Computer Music, Barcelona, Spain, pp. 29-34, 2001.

[151] Arfib, D. and Kessous, L., *Gestural Control of Sound Synthesis and Processing Algorithms*. Lecture Notes in Computer Science 2002, Volume 2298, pages 55-85.

[152] Arfib, D., Couturier, J.M., Kessous, L. & Verfaille, V., *Strategies of Mapping Between Gesture Data and Synthesis Model Parameters Using Perceptual Spaces*. Organised Sound Volume 7, Issue 2, pages 127-144, 2002.

[153] Arfib, D., Courturier, J.M. and L. Kessous, *Expressiveness and Digital Musical instrument Design*. Journal of New Music Research, Volume 34, Issue 1, pages 125-136, 2005.

[154] Verfaille V, Wanderley MM, Depalle P. Mapping strategies for gestural and adaptive control of digital audio effects. Journal of New Music Research. 2006 Mar 1;35(1):71-93.

[155] Caramiaux, B., F. Bevilacqua and N. Schnell, *Study on Gesture-Sound Similarity*. 3rd Music and Gesture Conference, McGill University, Montreal, 2010.

[156] Caramiaux, B., F. Bevilacqua, N. Schnell. "Sound Selection by Gestures" New Interfaces for Musical Expression (NIME 2011), Oslo, Norway, 2011.

[157] Verfaille, V., J. Boissinot, P. Depalle, and M. M. Wanderley, *Ssynth: A Real Time Additive Synthesizer With Flexible Control*. Proceedings of the International Computer Music Conference (ICMC'06), New Orleans, 2006.

[158] Guillemain P, Verfaille V. Combining physical modeling and additive synthesis as a mapping strategy for realtime control. In Proceedings of the International Computer Music Conference, Copenhagen, Denmark 2007 (Vol. 1, pp. 442-9).

[159] Pendharkar, C., Gurevich, M., and Wyse, L. *Parameterized Morphing As A Mapping Technique For Sound Synthesis*. Proceedings of the 9th International Conference on Digital Audio Effect. Montreal, Canada. 2006, 1-6.

[160] Wyse, L., and N. Dinh Duy *Instrumentalizing Synthesis Models*. Proceedings of the International Conference on New Interfaces for Musical Expression, June 15-18, 2010, Sydney, Australia.

[161] Brandtsegg, Ø, S. Saue, & T. Johansen, *A Modulation Matrix for Complex Parameter Sets*. Proceedings of the 11th International Conference on New Interfaces for Musical Expression, 30 May - 1 June 2011, Oslo, Norway.

[162] van Wijk JJ, van Overveld CW. Preset based interaction with high dimensional parameter spaces. In Data Visualization 2003 (pp. 391-406). Springer US.

[163] Allouis JF. The SYTER project: Sound processor design and software overview. In Proceedings of the 1982 International Computer Music Conference (ICMC). Ann Arbor, MI: Michigan Publishing, University of Michigan Library; 1982.

[164] Geslin Y. Digital Sound and Music Transformation Environments: a Twenty-year Experiment at the “Groupe de Recherches Musicales”. Journal of New Music Research. 2002 Jun 1;31(2):99-107.

[165] Todoroff T. Control of digital audio effects. DAFX: Digital Audio Effects. 2002:465-97.

[166] Shepard D. A two-dimensional interpolation function for irregularly-spaced data. In Proceedings of the 1968 23rd ACM national conference. 1968 Jan 1 (pp. 517-524). ACM.

[167] Todoroff T, Reboursière L. 1-d, 2-d and 3-d interpolation tools for max/msp/jitter. In Proc. ICMC’09, 2009.

[168] Pottier L, Stalla O. Interpretation and Space. Trends in Gestural Control of Music. 2000.

[169] Li J., Heap A. D. Spatial interpolation methods applied in the environmental sciences: A review. Environmental Modelling & Software, 53, 173-189 2014.

[170] Favreau E. Les outils de traitement GRM Tools. Publications du LMA. 1998.

[171] Bencina R. Oasis Rose the composition-real-time DSP with AudioMulch. In Proceedings of the Australasian Computer Music Conference 1998 (pp. 85-92).

[172] KORG KAOSS PAD QUAD Dynamic Effects Processor, Owner's Manual. 2010.

[173] Native Instruments FM8, Operation Manual. 2006.

[174] Griffiths D. Logic 10.2, including Alchemy, revealed! : musicradar; 2015 [Available from: http://www.musicradar.com/news/tech/logic-10-2-including-alchemy-revealed-626880.

[175] Todoroff T., Traube C. Graphical NeXTSTEP objects as FTS clients to control instruments in the new FTS client/server architecture. In Proceedings of the 1996 International Computer Music Conference 1996 (pp. 28-31). The International Computer Music Association.

[176] Todoroff T., Traube C, Ledent JM. NeXTStep graphical interfaces to control sound processing and spatialization instruments. In Proceedings: International Computer Music Conference 1997, Thessaloniki, Hellas, 25-30 september 1997, (pp. 325-328). The International Computer Music Association.

[177] Spain M., Polfreman R. Interpolator: a two-dimensional graphical interpolation system for the simultaneous control of digital signal processing parameters. Organised Sound. 2001 Aug 1;6(02):147-51.

[178] Freed A, MacCallum J, Schmeder A, Wessel D. Visualizations and Interaction Strategies for Hybridization Interfaces. In NIME 2010 (Vol. 10, pp. 343-347).

[179] Harman C, Johns M. Voronoi Natural Neighbors Interpolation. In Class of 2008 Senior Conference on Computational Geometry (p. 49).

[180] Larkin O, editor INT.LIB–A Graphical Preset Interpolator For Max MSP. ICMC'07: Proc of the 2007 International Computer Music Conference, 2007.

[181] Grosse D. Artist Focus: Andrew Benson, Cycling 74 Community Interview, 2015, [Available from: https://cycling74.com/2015/11/25/artist-focus-andrew-benson/#.VsnfhYQ0g9H].

[182] Benson A. nodes, Cycling 74 Community Forums, 2009 [Available from: https://cycling74.com/forums/topic/nodes/].

[183] nodes. Max Reference. Cycling 74, 2016.

[184] Stuart J. Sound Shapes and Spatial Texture: Frequency-Space Morphology. Ann Arbor, MI: Michigan Publishing, University of Michigan Library; 2014.

[185] O'Sullivan L. MorphOSC-A Toolkit for Building Sound Control GUIs with Preset Interpolation in the Processing Development Environment. Linux Audio Conference, Austria, 2013.

[186] Adiloglu K, Drioli C, Polotti P, Rocchesso D, Delle Monache S. Physics-based spike-guided tools for sound design. In Conference on Digital Audio Effects 2010 (DAFx-10), Sep 6.

[187] Smith E, Lewicki MS. Efficient coding of time-relative structure using spikes. Neural Computation. 2005 Jan;17(1):19-45.

[188] Drioli C, Polotti P, Rocchesso D, Delle Monache S, Adiloglu K, Annies R, Obermayer K. Auditory representations as landmarks in the sound design space. In Proceedings of Sound and Music Computing Conference 2009.

[189] Farin GE. Curves and surfaces for CAGD: a practical guide. Morgan Kaufmann; 2002.

[190] Marier M. Designing Mappings for Musical Interfaces Using Preset Interpolation. In Conference on New Interfaces for Musical Expression (NIME 12), 2012.

[191] Marier M. The Sponge - A Flexible Interface. In Conference on New Interfaces for Musical Expression (NIME 10), 2010 (Vol. 10, pp. 356-359).

[192] Bell R, Bartel A, Barrass S. A Systematic Method for Describing and Cataloguing Percussive Sounds, Based on Spectral, Temporal and Perceptual Criteria. International Journal of Arts and Sciences, 3(9):240-251 (2010).

[193] Fasciani S. Voice-Controlled Interface For Digital Musical Instruments. Doctoral Dissertation, National University Of Singapore, 2014.

[194] Arturia Prophet-V User Manual, 2014.

[195] Davidson PL, Han JY. Synthesis and control on large scale multi-touch sensing displays. In Proceedings of the 2006 conference on New interfaces for musical expression 2006 Jun 4 (pp. 216-219). IRCAM—Centre Pompidou.

[196] Schlei K. Relationship-Based Instrument Mapping of Multi-Point Data Streams Using a Trackpad Interface. In NIME 2010 Jun (pp. 136-139).

[197] Jordà S, Geiger G, Alonso M, Kaltenbrunner M. The reacTable: exploring the synergy between live music performance and tabletop tangible interfaces. In Proceedings of the 1st international conference on Tangible and embedded interaction, 2007 Feb 15 (pp. 139-146). ACM.

[198] Schlei K. TC-11: A Programmable Multi-Touch Synthesizer for the iPad. In NIME 2012.

[199] Le Vaillant, G., T. Dutoit, and R. Giot, Analytic vs. holistic approaches for the live search of sound presets using graphical interpolation. In International Conference on New Interfaces for Musical Expression (NIME 2020), Birmingham, UK, 2020.

[200] Le Vaillant, G., and R. Giot, "Multi-touch Interface for Acousmatic Music Spatialization," in 40th International Computer Music Conference (ICMC 2014), Athens, Greece, 2014.

[201] Misra A, Cook PR. Toward synthesized environments: A survey of analysis and synthesis methods for sound designers and composers. Ann Arbor, MI: Michigan Publishing, University of Michigan Library; 2009.

[202] Verron C, Pallone G, Aramaki M, Kronland-Martinet R. Controlling a spatialized environmental sound synthesizer. In Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA'09. IEEE Workshop on 2009 Oct 18 (pp. 321-324). IEEE.

[203] Jönsson A, Breslin R, Ma M. The Ambience Table: A Serious Gaming Interface for Aiding Sound Design. InSerious Games Development and Applications 2013 Sep 25 (pp. 151-164). Springer Berlin Heidelberg.

[204] Cook PR. Toward physically-informed parametric synthesis of sound effects. In Applications of Signal Processing to Audio and Acoustics, 1999 IEEE Workshop on 1999 (pp. 1-5). IEEE.

[205] Monache SD, Polotti P, Rocchesso D. A toolkit for explorations in sonic interaction design. In Proceedings of the 5th Audio Mostly Conference: A Conference on Interaction with Sound 2010 Sep 15 (p. 1). ACM.

[206] Van Den Doel K, Kry PG, Pai DK. FoleyAutomatic: physically-based sound effects for interactive simulation and animation. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques 2001 Aug 1 (pp. 537-544). ACM.

[207] Menzies D. Physical audio for virtual environments, phya in review. In Proceedings of the 13 International Conference on Auditory Display, Montréal, Canada, June 26 - 29, 2007.

[208] Menzies D. Physically motivated environmental sound synthesis for virtual worlds. EURASIP Journal on Audio, Speech, and Music Processing. 2010 Jan 1;2010:6.

[209] Ren Z, Yeh H, Lin MC. Example-guided physically based modal sound synthesis. ACM Transactions on Graphics (TOG). 2013 Jan 1;32(1):1.

[210] Mullan E, Physical Modelling Sound Synthesis by Digital Waveguide Extraction with Application to Computer Games and Virtual Environments (Doctoral dissertation, Queen's University Belfast), 2013.

[211] Gohlke K, Black D, Loviscach J. Leveraging behavioral models of sounding objects forgesture-controlled sound design. In Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction 2011 Jan 22 (pp. 245-248). ACM.

[212] Lubart TI. Models of the Creative Process: Past, Present and Future. Creativity Research Journal. 2001 Oct 1;13(3-4):295-308.

[213] Messitte N. The Changing Times Of Hollywood Sound Design: Forbes, Media & Entertainment; 2015 [updated 30th September 2015. Available from: http://www.forbes.com/sites/nickmessitte/2015/09/30/the-changing-times-of-hollywood-sound-design-part-1/?ss=lists#4fc29a2059f3].

[214] Audio plug-in [Available from: https://en.wikipedia.org/wiki/Audio_plug-in].

[215] Orio N, Schnell N, Wanderley MM. Input devices for musical expression: borrowing tools from HCI. In Proceedings of the 2001 conference on New interfaces for musical expression 2001 Apr 1 (pp. 1-4). National University of Singapore.

[216] Fernandes G, Holmes C. Applying HCI to music-related hardware. In CHI'02 Extended Abstracts on Human Factors in Computing Systems 2002 Apr 20 (pp. 870-871). ACM.

[217] Kiefer C, Collins N, Fitzpatrick G. HCI Methodology For Evaluating Musical Controllers: A Case Study. In NIME 2008 Jun 4 (pp. 87-90).

[218] Hsu WT, Sosnick MH. Evaluating Interactive Music Systems: An HCI Approach. In NIME 2009 Jun (pp. 25-28).

[219] Xambó A, Laney R, Dobbyn C, Jorda S. Collaborative music interaction on tabletops: an HCI approach. In: BCS HCI 2011 Workshop on When Words Fail: What can Music Interaction tell us about HCI?, 4 July 2011, Newcastle Upon Tyne.

[220] Malloch J, Birnbaum D, Sinyor E, Wanderley MM. Towards A New Conceptual Framework For Digital Musical Instruments. In Proceedings of the 9th International Conference On Digital Audio Effects, 2006 Sep 18 (pp. 49-52).

[221] O'modhrain S. A Framework For The Evaluation Of Digital Musical Instruments. Computer Music Journal. 2011 Mar 9;35(1):28-42.

[222] Flueckiger B. Sound Effect: Strategies For Sound Effects In Films. Sound And Music In Film And Visual Media. 2009:151-79.

[223] Birnbaum D, Fiebrink R, Malloch J, Wanderley MM. Towards a dimension space for musical devices. In Proceedings of the 2005 conference on New Interfaces for Musical Expression, 2005 May 1 (pp. 192-195). National University of Singapore.

[224] Dewey C, Wakefield J. A Guide To The Design And Evaluation of New User Interfaces For The Audio Industry. In Audio Engineering Society Convention 136 2014 Apr 25. Audio Engineering Society.

[225] Marquez-borbon A, Gurevich M, Fyans AC, Stapleton P. Designing Digital Musical Interactions in Experimental Contexts. In NIME 2011 International Conference on New Interfaces for Musical Expression, 2011.

[226] De Vreede G. J, Fruhling A, Chakrapani A. A repeatable collaboration process for usability testing. In System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on 2005 Jan 3 (pp. 46-46). IEEE.

[227] Barbosa J, Malloch J, Wanderley MM, Huot S. What does “Evaluation” mean for the NIME community? In NIME 2015-15th International Conference on New Interfaces for Musical Expression 2015 May 31 (p. 6). Louisiana State University.

[228] Gama S., Gonçalves D., Studying Color Blending Perception for Data Visualization. Presented at the Eurographics Conference on Visualization (EuroVis), 2014.

[229] Lee D. T., Schachter B. J., Two algorithms for constructing a Delaunay triangulation. International Journal of Computer & Information Sciences, 9(3), 219-242, 1980.

[230] Sarter N. B., Multimodal Information Presentation: Design Guidance and Research Challenges. International Journal of Industrial Ergonomics, 36(5), 439-445, 2006.

[231] International Organization for Standardization, Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts (ISO 9241-11:2018). https://www.iso.org/standard/63500.html, 2021.

[232] Brooke, J. SUS: A "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & A. L. McClelland (Eds.), Usability Evaluation in Industry. London: Taylor and Francis, 1996.

[233] Bangor A., Kortum P., Miller J.A., The system usability scale (SUS): An empirical evaluation. International Journal of Human-Computer Interaction, 24(6), pp.574-594, 2008. https://doi.org/10.1080/10447310802205776

[234] Brooke J., SUS: a retrospective. Journal of usability studies, 8(2), pp.29-40, 2013.

[235] Lewis J.R., Sauro J., Item benchmarks for the system usability scale. Journal of Usability Studies, 13(3), pp.158-167, 2018.

[236] Cohen J., Statistical Power Analysis for the Behavioral Sciences. New York, NY: Routledge Academic, 1988. https://doi.org/10.4324/9780203771587

[237] Mitchell A., The ESRI Guide to GIS Analysis, Volume 2: Spatial Measurements & Statistics. ESRI Press, 2005.

[238] Pallant J., SPSS Survival Manual: A Step by Step Guide to Data Analysis Using SPSS for Windows. Routledge, 2020.

[239] Leech N. L, Onwuegbuzie A. J., A Call for Greater Use of Nonparametric Statistics. Presented in Mid-South Educational Research Association Annual Meeting, 2002.

[240] Rosenthal R., Parametric measures of effect size. In H. Cooper & L. V. Hedges (Eds.), The handbook of research synthesis. (pp. 231-244). New York: Russell Sage Foundation, 1994.

[241] Grissom R. J., Kim J. J., Effect sizes for research: Univariate and multivariate applications. (2nd ed.). New York, NY: Taylor & Francis, 2012.

[242] Sauro J., A practical guide to the System Usability Scale: Background, benchmarks, & best practices. Denver, CO: Measuring Usability LLC, 2011.

[243] Kortum P. T., Bangor A., Usability ratings for everyday products measured with the System Usability Scale. International Journal of Human-Computer Interaction, 29(2), 67-76, 2013.

[244] Yang Y. C., Wang S. T., Tseng Y. J., Lin H. C. K., Light up! Creating an interactive digital artwork based on Arduino and Max/MSP design. In 2010 International Computer Symposium (ICS2010) (pp. 258-263), IEEE, 2010. https://doi.org/10.1109/COMPSYM.2010.5685506

[245] Order S., 'ICreate': Preliminary usability testing of apps for the music technology classroom. Journal of University Teaching & Learning Practice, 12(4), 8, 2015.

[246] Spiliotopoulos K., Rigou M., Sirmakessis S., A Comparative Study of Skeuomorphic and Flat Design from a UX Perspective. Multimodal Technologies and Interaction, 2(2), 31, 2018. https://doi.org/10.3390/mti2020031

[247] Stolfi A. S., Milo A., Barthet M., Playsound. space: Improvising in the browser with semantic sound objects. Journal of New Music Research, 48(4), 366-384, 2019. https://doi.org/10.1080/09298215.2019.1649433

[248] Keefe D. F., Isenberg T., Reimagining the scientific visualization interaction paradigm. Computer, 46(5), 51-57, 2013. https://doi.org/10.1109/MC.2013.178

[249] Wilson A., Fazenda B. M., User-guided rendering of audio objects using an interactive genetic algorithm. Journal of the Audio Engineering Society, 67(7/8), 522-530, 2019.

[250] Magnusson T., An Epistemic Dimension Space for Musical Devices. In International Conference on New Interfaces for Musical Expression (NIME), 2010 Jan 1 (pp. 43-46).

[251] Wójcik-Augustyniak M., How to measure and compare the value of organizations. The case study of HEIs. Entrepreneurship and Sustainability Issues. 2020;7(3):2144-69.

[252] Whittington, W. B., Sound Design and Science Fiction. University of Texas Press, 2007.

[253] Konzett M., Sci-Fi Film and Sounds of the Future. Teoksessa Bartkowiak, Mathew J.: Sounds of the Future. Essays on Music in Science Fiction Film. McFarland & Company, Jefferson. 2010 Mar 10:100-15.

[254] Grøn N., The Sound of Horror-Silence & Sound Contrasts in Sci-Fi Horror Movies. Journal of Media, Cognition and Communication. 2013 Jun 10;1(1).

[255] Lakens D., Calculating and reporting effect sizes to facilitate cumulative science: a practical primer for t-tests and ANOVAs. Frontiers in psychology. 2013 Nov 26;4:863.

[256] Bender R., Lange S., Multiple test procedures other than Bonferroni's deserve wider use. The British Medical Journal (The BMJ), 318(7183), 1999 p.600.

[257] Tubb R., Dixon S., The Divergent Interface: Supporting Creative Exploration of Parameter Spaces. In International Conference on New Interfaces for Musical Expression (NIME), 2014 (pp. 227-232).

[258] Ward M. O. Multivariate data glyphs: Principles and practice. In Handbook of Data Visualization (pp. 179-198). Springer, Berlin, Heidelberg, 2008.

[259] Gröhn M., T. Lokki, and T. Takala. Comparison Of Auditory, Visual, and Audiovisual Navigation In a 3D Space. ACM Transactions on Applied Perception (TAP), 2(4), 564-570, 2005.

[260] Wilson J, Rosenberg D. Rapid prototyping for user interface design. In Handbook of human-computer interaction 1988 Jan 1 (pp. 859-875). North-Holland.

[261] Meister D, Enderwick TP. Human factors in system design, development, and testing. CRC Press, 2001 Jun 1.

[262] Aronov D, Tank DW. Engagement of Neural Circuits Underlying 2D Spatial Navigation In a Rodent Virtual Reality System. Neuron. 2014 Oct 22;84(2):442-56.

[263] Ekstrom AD, Spiers HJ, Bohbot VD, Rosenbaum RS. Human Spatial Navigation. Princeton University Press; 2018 Aug 7.

[264] Krivoruchko K., Wood N., Using Multivariate Interpolation for Estimating Well Performance, ESRI Technology, Summer 2014.

[265] Carroll E.A., Latulipe C., The creativity support index. In Conference on Human Factors in Computing Systems (CHI'09) Extended Abstracts on Human Factors in Computing Systems (pp. 4009-4014), 2009.

[266] Stolfi A., Milo A., Ceriani M., Barthet M., Participatory musical improvisations with playsound. space. In Proceedings of the Web Audio Conference, 2018.

[267] Turchet L, Barthet M. An ubiquitous smart guitar system for collaborative musical practice. Journal of New Music Research. 2019 Aug 8;48(4):352-65.

[268] Frich J, MacDonald Vermeulen L, Remy C, Biskjaer MM, Dalsgaard P. Mapping the landscape of creativity support tools in HCI. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems 2019 May 2 (pp. 1-18).

[269] Graf M, Chijioke Opara H, Barthet M. An Audio-Driven System for Real-Time Music Visualisation. In Audio Engineering Society Convention 150. Audio Engineering Society, 24 May 2021.

[270] Di Giacomo E, Didimo W, Montecchiani F, Tappini A. A User Study on Hybrid Graph Visualizations. In Proceedings of the 29th International Symposium on Graph Drawing and Network Visualization (GD 2021), 2021.

[271] Da Lozzo G, Di Battista G, Frati F, Patrignani M. Computing NodeTrix representations of clustered graphs. In International Symposium on Graph Drawing and Network Visualization 2016 Sep 19 (pp. 107-120). Springer, Cham.