

Technische Univeristät München
Forschungs- und Lehreinheit XI
Angewandte Informatik / Kooperative Systeme

Collaborative Music-Making with Interactive Tabletops

Niklas Klügel



Technische Univeristät München
Forschungs- und Lehreinheit XI
Angewandte Informatik / Kooperative Systeme

Collaborative Music-Making with Interactive Tabletops

Niklas Klügel

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr. Thomas Huckle
Prüfer der Disseration:	1.	Univ.-Prof. Dr. Johann Schlichter
	2.	Univ.-Prof. Bernd Brügge, Ph.D.
	3.	Priv.-Doz. Dr. Georg Groh

Die Dissertation wurde am 08.04.2015 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 27.07.2015 angenommen.

Copyright © 2015 Niklas Klügel

Licensed under Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)

November 2015

Abstract

Electronic music and musical creativity pervades contemporary culture. Music technology thus acts as a mediator, providing creativity support for musical engagement. However, few commercial products and research projects focus on musical collaboration, despite evidence that music, musical engagement and -interaction are deeply embedded into socio-cultural and collaborative constructs. Hence, certain socio-technological processes are not leveraged for this creative task. The goal of this thesis is to support the collaborative creation of electronic music in the form of co-located creativity support systems that utilize multi-touch tabletops. For this, musical practises are analysed and requirements to accomplish this goal are deduced. Based on several viewpoints, it is argued that long-term musical engagement requires both low-thresholds and heigh-ceilings with respect to difficulty. The contributions are subjected to this requirement. Music related topics, such as the control of audio synthesis, groove production and melody creation are approached with respect to structural, perceptional and emotional qualities. Several of the devised methods have been employed in prototypical applications. The appropriateness of each of these approaches has been evaluated in user-studies. The results indicate that these applications can be used to foster creative collaboration and support the understanding of musical domains for novices.

Zusammenfassung

Elektronische Musik und musikalische, kreative Betätigung sind Teil der Gegenwartskultur. In diesem Zusammenhang agiert Musiktechnologie als Vermittler, der in der musikalischen Kollaboration kreativitätsunterstützend wirkt. Obwohl es Belege gibt, die zeigen, dass Musik, musikalische Betätigung und musikalische Interaktion tief in soziokulturelle und kollaborative Konstrukte eingebettet sind, setzen dennoch wenige kommerzielle Produkte und forschungsnahe Projekte einen Fokus auf Kollaboration in der Elektronischen Musik. Im Zuge dessen werden kreativitätsfördernde Abläufe, die aus einer sozialen Situation heraus entstehen, nicht vollständig instrumentalisiert. Ziel dieser Dissertation ist es, Kollaboration in verschiedenen musikalischen Domänen der Elektronischen Musik zu unterstützen. Hierfür werden zunächst Praktiken in der Elektronischen Musik untersucht und entsprechende Anforderungen für Systeme zur Kreativitätsunterstützung am gleichen Ort durch Multi-Touch Tisch-Bildschirme ermittelt. Diese Anforderungen beinhalten unter anderem, dass musikalische Betätigung im kollaborativen Umfeld sowohl niedrige Einstiegshürden als auch weiterführende Ausdrucksmöglichkeiten bieten muss, um über längere Zeiträume attraktiv zu bleiben. Die Beiträge in dieser Dissertation sind darauf ausgerichtet, diesen Anforderungen zu genügen. Hierbei wird musikalische Struktur, Wahrnehmung, sowie Emotion auf einzelne Domänen in der Elektronischen Musik übertragen, modelliert und in prototypischen Anwendungen evaluiert. Die Domänen beinhalten Audio-Synthese, Groove Produktion und Melodie-Erzeugung. Die Ergebnisse von Nutzerstudien zeigen, dass die präsentierten Ansätze musikalische Kollaboration animieren und fördern können, sowie das Verständnis von musikalischen Zusammenhängen unterstützen können.

Acknowledgements

Over the past years I have received valuable support and encouragement without which this dissertation would not be made possible. First and foremost, I owe my advisor Dr. Johann Schlichter a debt of gratitude for giving me the opportunity to join his research group and to continue with the quest that started with the Diploma Thesis at his chair and to set about a project that I am so wholeheartedly devoted to. I would also like to thank him for thoroughly reviewing this thesis and providing me with invaluable feedback. I am grateful to Prof. Bernd Brügge, Ph.D. for being my second advisor, and giving me feedback, especially concerning software architectural aspects of the involved frameworks. Furthermore, I would like to thank my advisor P.D. Dr. Georg Groh for his constructive criticism in the countless discussions to focus and elaborate my research - these have been highly appreciated. I must also express the appreciation of my dear colleagues - present and former ones - for their constant encouragements and support. The welcoming and amicable work climate they have created made my stay especially enjoyable. I would like to thank Dr. René Frieß for introducing me to tabletop computing. His work on Creativity Support Systems contributed significant insights for this thesis. Credits also go out to the students whose theses I had the pleasure to advise and whose enthusiasm and dedication to the topic made the related practical work for this dissertation possible. Last and definitely not least, I owe my girlfriend Angelina the sincere appreciation for her constant support and patience.

Contents

1 Introduction 1
1.1 Motivation & Research Question . . . 1
1.2 Musical Collaboration in this Thesis . . . 2
1.3 Paradoxes and Wickedness . . . 4
1.4 Methodology . . . 6
1.5 Outline . . . 8
1.6 Related Work . . . 9

I The Rich Picture 13

2 From Musique Concrète to the Bedroom Producer 15
2.1 Breaking with History . . . 15
2.2 Studio Works in the Mainstream . . . 16
2.3 The Studio as an Instrument . . . 17
2.4 The Bedroom Producer . . . 18
2.5 Contemporary Techniques for Collaboration in the Studio . . . 19

3 Social Aspects: Creativity, Music, Interaction and Flow 21
3.1 Creativity . . . 21
3.2 Music and Social Being . . . 26
3.3 Motivation in Music . . . 27
3.4 Embodied Practice & Communication . . . 29
3.5 Flow . . . 31

4 Collaboration: Music and Human Computer Interaction 35
4.1 Collaboration and the Tabletop Environment . . . 35
4.2 Common Design Guidelines for Creativity Support Systems . . . 39
4.3 Musical collaboration . . . 39
4.4 Music and Human Computer Interaction . . . 42
4.5 Feedback . . . 44
4.6 Evaluation Methodologies . . . 44

5 Résumé: Framing the Rich Picture, Goals and Outlook 47
5.1 Human-Computer Interaction . . . 47
5.2 Protocols in Music Collaboration . . . 50
5.3 Electronic Dance Music . . . 52
5.4 User-group . . . 54
5.5 Metonymy . . . 55

6 Framework Architecture 57
6.1 Event Processing . . . 60
6.2 User Interface . . . 70
6.3 Audio Synthesis . . . 72
6.4 I/O Abstractions . . . 74

II Timbre 75

7 Introduction to Timbre 77
7.1 On the Importance of Timbre . . . 77
7.2 Musical organisation . . . 78
7.3 Semiotics, Metaphor & Imagination . . . 81
7.4 Conclusion . . . 83

8 Digital Audio Effects 85
8.1 Unit Generators & Modifiers . . . 86
8.2 Additive Synthesis . . . 87
8.3 Subtractive Synthesis . . . 89
8.4 Special Filters and Basic Effects . . . 91
8.5 Simple Reverberation . . . 92
8.6 Abstract Audio Synthesis . . . 94
8.7 Perceptually Salient Audio Features to Describe Timbre . . . 102

9 Controlling Audio Synthesis by a Perceptually Motivated Mapping 107
9.1 Perceptually Motivated Synthesis . . . 107
9.2 Related work . . . 109
9.3 Concept and Preliminary Tests . . . 110
9.4 Synthesis & Corpus . . . 113
9.5 Features . . . 116
9.6 Mapping & Visualization . . . 117
9.7 Prototype . . . 119
9.8 Evaluation . . . 121
9.9 Discussion & Conclusion . . . 123

III Emotion 127

10 Introduction to Emotion 129
10.1 Emotion and Music . . . 129
10.2 Which Emotions Can Music Express? . . . 132
10.3 Emotion and Timbre . . . 132
10.4 Affective Algorithmic Composition . . . 133
10.5 Presented Research . . . 134

11 Emotional Affect of Sounds 135
11.1 Introduction . . . 135
11.2 Potential Applications . . . 137
11.3 Related Work . . . 137
11.4 Data set for Training . . . 139
11.5 Machine Learning . . . 142
11.6 Conclusion . . . 152

12 Controlling the Emotional Affect of Simple Melodies 155
12.1 Introduction . . . 156
12.2 Related Work . . . 156
12.3 Conceptual Framing . . . 157
12.4 Implementation . . . 158
12.5 Evaluation . . . 162
12.6 Conclusions . . . 168

IV Structure 169

13 Groove Production 171
13.1 Introduction . . . 171
13.2 Concept . . . 173
13.3 Prototypes . . . 174
13.4 Evaluation . . . 178
13.5 Results & Conclusion . . . 179

14 The Shared Composition as Optimization Problem 185
14.1 Introduction . . . 185
14.2 Genetic Algorithms . . . 187
14.3 Concept . . . 189
14.4 Interface & Prototype . . . 192
14.5 Evaluation . . . 194
14.6 Results . . . 194
14.7 Discussion & Conclusion . . . 195

V Conclusion 197

15 Summary and Future Work 199
15.1 Summary . . . 199
15.2 Approaches . . . 201
15.3 Prospects for Future Work . . . 202

Bibliography 209

List of Figures

1.1 The *reacTable* application . . . 9
1.2 Picture of a previous prototype . . . 10
1.3 *mixiTUI* . . . 10
1.4 *Xenakis* . . . 10
1.5 *ANTracks* . . . 11
1.6 *Composition on the Table* . . . 12
1.7 Scenario of Iltur in use . . . 12

3.1 Compositional process of music as proposed by Gelineck *et al.* . . . 23
3.2 Systems model of creativity . . . 24

5.1 Three approaches to rhythm programming with hardware . . . 53
5.2 The Roland TB-303 . . . 55

6.1 Framework as high-level component diagram . . . 59
6.2 Exemplary patches (programs) in Max 6 and OpenMusic . . . 64
6.3 Class diagram of the Reakt framework . . . 65
6.4 Exemplary graph structure implicitly generated in the framework . . . 69
6.5 Performance comparison . . . 70
6.6 MT4j architecture overview . . . 71
6.7 MT4j component structure . . . 73

7.1 Screen-shot of an arrangement session in a DAW . . . 79

8.1 Amplitude modulation of a signal by an ADSR . . . 86
8.2 Simple waveforms using integer-relationships for partials in their spectra . . . 87
8.3 Spectrum of a bell sound . . . 88
8.4 Amplitude responses of common filter types . . . 89
8.5 Common structure of a subtractive synthesizer. . . . 91
8.6 The frequency response of a comb filter . . . 91
8.7 The phases of reverberation . . . 93
8.8 Common two operator FM synthesis topologies . . . 94
8.9 Spectrum of simple FM exhibiting sidebands . . . 95
8.10 First kind Bessel functions . . . 96
8.11 Example of constructive interference . . . 97
8.12 Ring modulation as multiplication of two signals . . . 98
8.13 Example of a spectrum generated by Ring Modulation . . . 99
8.14 Phase function in Vector Phaseshaping Synthesis . . . 100
8.15 Topology of digital oscillators . . . 101
8.16 Examples of waveforms in Vector Phase Shaping . . . 102
8.17 Waveforms and their respective spectra in Vector Phase Shaping . . . 103
8.18 VPS waveform morphologies . . . 103
8.19 Timbral audio features and their extraction techniques . . . 104

9.1 Visualization of the proposed process . . . 111

9.2 Example plots of preliminary data sets . . . 112
9.3 Final design of the synthesis topology . . . 113
9.4 Example plot of the search method for three dimensions (three parameters) . . . 115
9.5 Example of the non-linear mapping in GTM . . . 118
9.6 Screen-shot of our prototype . . . 119
9.7 Deployment diagram for SoundDesigner . . . 120
9.8 Screen-shot of the alternative prototype . . . 122
9.9 Participants of the user study evaluating our prototype . . . 123

10.1 Russel's circumplex model for Valence / Arousal . . . 131
10.2 Highly rated emotions from listener studies . . . 132
10.3 Overview of an affective algorithmic composition system . . . 133

11.1 The processing data-flow of the proposed method . . . 136
11.2 V/A values of all samples in the data set . . . 139
11.3 Histogram of values of top six features according to clusters . . . 140
11.4 Histogram of V/A values (left to right) in our data set . . . 141
11.5 A graphical representation of a simple Restricted Boltzmann Machine . . . 144
11.6 Layer-wise training of a DBN . . . 149
11.7 The effect of layer size and pre-training . . . 149
11.8 Schematic of the applied model (number of layers/nodes is exemplary) . . . 150
11.9 An excerpt of a 25 tile grid with the V/A values . . . 151
11.10 Performance evaluation . . . 151
11.11 Extension of the mapping proposed in chapter 9 with affect features . . . 152

12.1 Exemplary user interface in [WIC08] . . . 157
12.2 Architecture of the underlying synthesizer . . . 159
12.3 Screen-shot of the prototypical implementation . . . 160
12.4 Illustration of drawing a melody contour . . . 161
12.5 Deployment diagram for FugueGenerator . . . 163
12.6 Median values of user generated valence / arousal positions from the evaluation . . . 163
12.7 The median values of the pitch heights for each emotion (sorted by valence) . . . 164
12.8 The median values of the pitch heights for each emotion (sorted by arousal) . . . 164
12.9 Plot of created melody contours in relationship to targeted emotions . . . 165
12.10 User interaction: plot of the average melody contour over time . . . 166
12.11 User interaction: plot of the V/A values for each user over time . . . 166
12.12 Answers in the questionnaire that relate to appreciation . . . 167
12.13 Answers in the questionnaire that relate to collaboration . . . 168

13.1 Screen-shot of the first prototype (p1) . . . 173
13.2 Screen-shot of the second prototype (p2) . . . 174
13.3 Topology of the implemented subtractive synthesizer . . . 175
13.4 Topology of the a synthesizer inspired by physical modelling . . . 176
13.5 Deployment diagram for TreeQuencer . . . 177
13.6 Excerpt from the transformed log-output . . . 178
13.7 Exemplary plots of the created rhythmic complexity in the user studies . . . 179
13.8 Exemplary plots of the logged session data . . . 181
13.9 A group experiencing user behaviour as captured on video . . . 182

14.1 Schematic of a GA algorithm . . . 188
14.2 Screen-shot of the prototype . . . 193
14.3 Deployment diagram for Mutator . . . 194

List of Tables

11.1 Excerpt from the 25 most frequently used sentiment related tags with V/A value 142
11.2 Preliminary performance (classification error) of common ML-methods 142

12.1 Influence of arousal on properties of music 161
12.2 Influence of valence on properties of music 162
12.3 Comparison of definitions as they appear in [Liv+10] and in our contribution 164

14.1 Number of patterns used in the application w.r.t. instrument class 191

15.1 Musical domains, underlying models and methods of implementation 202
15.2 Studies and methodologies 203
15.3 Abstractions in interaction, music / artistry and signal processing 205

List of Contributions

The work on this thesis has led to the following, relevant publications (in chronological order):

- N. Klügel, M. R. Friess, G. Groh and F. Echtler. "An Approach to Collaborative Music Composition". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by A. R. Jensenius, A. Tveit, R. I. Godoy and D. Overholt. 2011, pp. 32–35
- N. Klügel and G. Groh. "Towards Mapping Timbre to Emotional Affect". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by W. Yeo, K. Lee, A. Sigman, J. H. and G. Wakefield. Graduate School of Culture Technology, KAIST, May 2013, pp. 525–530
- N. Klügel, G. Hagerer and G. Groh. "TreeQuencer: Collaborative Rhythm Sequencing A Comparative Study". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by B. Caramiaux, K. Tahiroglu, R. Fiebrink and A. Tanaka. Goldsmiths, University of London, June 2014, pp. 50–53
- N. Klügel, T. Becker and G. Groh. "Designing Sound Collaboratively Perceptually Motivated Audio Synthesis". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by B. Caramiaux, K. Tahiroglu, R. Fiebrink and A. Tanaka. Goldsmiths, University of London, June 2014, pp. 327–330
- N. Klügel, G. Hagerer and G. Groh. "FugueGenerator - Collaborative Melody Composition Based on a Generative Approach for Conveying Emotion in Music". In: *Proceedings of the Joint International Conference Music Computing (ICMC) and Sound and Music Computing (SMC)*. ed. by A. Andreopoulou, G. Kosteletos, N. Georgiou, A. Kolkou and A. Charoumenidou. National and Kapodistrian University of Athens, Sept. 2014, pp. 1043–1050
- N. Klügel, A. Lindström and G. Groh. "A Genetic Algorithm Approach to Collaborative Music Creation on a Multi-touch Table". In: *Proceedings of the Joint International Conference Music Computing (ICMC) and Sound and Music Computing (SMC)*. ed. by A. Andreopoulou, G. Kosteletos, N. Georgiou, A. Kolkou and A. Charoumenidou. National and Kapodistrian University of Athens, Sept. 2014, pp. 286–293

Publications which are tangentially relevant to this thesis are:

- M. R. Frieß, N. Klügel and G. Groh. "Lzrdm: Collaborative Multi-touch Sequencer". In: *ACM International Conference on Interactive Tabletops and Surfaces*. ITS '10. (Demonstration). ACM, 2010, pp. 303–303
- M. R. Frieß, G. Groh, N. Klügel and M. Kleinhans. "A tabletop application environment for generic creativity techniques". In: *International*

Journal of Computer Information Systems and Industrial Management Applications 3.4 (2011)

- M. R. Frieß, G. Groh, N. Klügel and M. Kleinhans. "An Analysis Framework for Quantitatively Characterizing Collaboration in Tabletop Based CSCW Environments". In: *International Journal of Computer Information Systems and Industrial Management Applications* 4.2 (2012)
- N. Klügel. "Practical Empirical Mode Decomposition for Audio Synthesis". In: *Proceedings of the International Conference of Digital Audio Effects*. University of York, Sept. 2012, pp. 15–18

Furthermore, some of the source codes for the implemented frameworks and prototypical applications have been made available:

- N. Klügel, C. Ruff and U. Laufs. *UltraCom - UI-Framework*. Framework / Library, Distributed under GPL version 3. 2010-2014. URL: https://github.com/lodsb/UltraCom, chapter 6
- N. Klügel. *Reakt - Reactive Event System*. Framework / Library, Distributed under GPL version 3. 2010-2014. URL: https://github.com/lodsb/reakt, chapter 6
- N. Klügel. *Library for scales and tuning*. Framework / Library, Distributed under GPL version 3. 2013. URL: https://github.com/lodsb/scales, chapter 12
- N. Klügel. *Genetic Algorithm Toolbox for Scala*. Framework / Library, Distributed under GPL version 3. 2013. URL: https://github.com/lodsb/mutant5000, chapter 14
- T. Becker and N. Klügel. *SoundDesigner*. Application, includes data sets, Distributed under GPL version 3. 2013. URL: https://github.com/lodsb/SoundDesigner, chapter 9
- G. Hagerer and N. Klügel. *TreeQuencer*. Application, Distributed under GPL version 3. 2013. URL: https://github.com/lodsb/TreeQuencer, chapter 13
- G. Hagerer and N. Klügel. *FugueGenerator*. Application, Distributed under GPL version 3. 2013. URL: https://github.com/lodsb/FugueGenerator, chapter 14
- A. Lindström and N. Klügel. *SoundDesigner*. Application, Distributed under GPL version 3. 2013. URL: https://github.com/lodsb/Mutator, chapter 12

List of Abbreviations

CSCM	Computer Supported Collaborative / Cooperative Music(-making)
CSCW	Computer Supported Cooperative Work
DAW	Digital Audio Workstation
HCI	Human-Computer Interaction
MIDI	Musical Instrument Digital Interface
NIME	New Interfaces (for) Musical Expression
OSC	Open Sound Control
TUI	Tangible User Interface
TUIO	Tangible User Interface Object(-Protocol)
WIMP	Windows Icons Menus Pointer

Chapter 1

Introduction

" In a world of more than 4,500 singing species, only one species - Homo sapiens - lives on the ground, and only one species - Homo sapiens - has the ability to follow precise rhythmic patterns so as to permit group singing, drumming, and dancing. What explains the unique place of humans among singing species?"

Brown and Jordania, “Universals in the world’s musics”, [BJ13]

Music performance is typically a group activity, in which the common case is for musicians to use separate instruments. Indeed, regarding the historical development of music performance, traditional instruments are usually designed for individual use and do not favour performers taking an active role in controlling not only their own musical output but also that of their peers [Jor05a]. However specially designed electronic communication channels can allow this. As such, the idea of a multi-user music instrument is not exactly new. In the context of electro-acoustic music it dates back at least to the works of John Cage *Imaginary Landscapes No. 4* (1951) and Karlheinz Stockhausen *Mikrophonie I* (1964) and grew into the various instruments and IT environments developed nowadays by the computer music community.

1.1 Motivation & Research Question

Yet, the context in which such multi-user music instruments were applied changed from the necessity to translate a composition into a performance [Mac08] to the creativity support in a social musical *experience* [BF03]. Most of these new systems are designed to be used in a public setting (*e. g.* exhibitions) and therefore focus on the social event of making music together. Hence the social experience takes precedence over the generation of music itself (ibid). As a result, they are limited in musical expressiveness to provide an instant access to making music for novices [BF03]. The problem with this is, as elaborated in the following chapters, that expressive musical use of the system is denied. But expression is important for fostering creativity, further engagement with the system, feeling of Flow in the group and expressing, developing and learning musical meaning. In this manner Wessel et al. [WW02, p. 1] point out: *"many of the simple-to-use computer interfaces proposed for musical control seem, after even a brief period of use, to have a toy-like character and do not invite continued musical evolution".*

· ◇ ·

However, ideally, such a system can be adaptive and allow for novices' and experts' participation alike, thus, truly *supporting* the creative process. Approvingly, Chadabe [Cha02] comes to the conclusion that it allows *"for anyone, from the most skilled and musically talented performers to the most unskilled members of the large public, to participate in a musical process"*, while Ferreira-Lopes *et al.* further admit such systems the role of *"a musical instrument, an interface, an instrument of composition or an almost autonomous performer"* [FC05]. In this regard, such a system is as well the mediator between the collaborating participants and between human and sonic output as an interactive partner.

· ◇ ·

Single Display Groupware for co-located collaborative creativity support has been recommended [SBD99] and applied in various works in this field [Jor+06; PH09; XLD11; PRI02] as well as elaborated on in my Diploma Thesis [Klü09]. The utilization of multi-touch for interaction further offer means to facilitate social communication protocols as summarized by [WW11, p. 39]: *"The advent of touch computing and relatively large sharable screens combined to produce systems that are not limited to relatively specific and routinized transactions. [...] This enables a new kind of computing: the social NUI*[1]*, where people can share experiences in the here and now and use the computer simply as a support mechanism"*. Therefore the use of a multi-touch tabletop environment is argued to be the preferred way to enable group interaction. The merits of multi-touch and tangible interfaces are further discussed in section 4.1. Additionally, the pervasiveness of electronic music establishes shared knowledge which is beneficial for group collaboration. It is furthermore argued that synthesizing music or sounds with electronic means has advantageous properties for creativity support and aspects of Human-Computer Interaction (cp. section 5.3).

[1] NUI - Natural User Interface

· ◇ ·

Concluding, this thesis contributes to the following research question:

> *How can the performative collaborative creation of electronic music in a Single Display Groupware environment be supported for novices and experts alike?*

1.2 Musical Collaboration in this Thesis

Because the range of possible applications and implied music related context is vast, it is first necessary to narrow the domain specific context in which the contributions of this thesis are to be placed. Some points of the general argumentation are briefly anticipated here and will be explicated and corroborated in the next chapters which highlight continuative viewpoints.

· ◇ ·

It is difficult to create a consistent taxonomy for systems for Computer Supported Collaborative Music-making (CSCM) as many prototypical implementations differ in musical and social context, such as installations, co-located and networked music ensembles, new instrument designs or collaborations with robotic performers [Dru09]. However, it is possible to regard the context of such applications (in accordance with [BF03]): *Focus, Location, Scalability and Player Interaction*; additionally of importance

is the *Musical Range, the Physical Interface and the Pathway From Novices to Expert Performance*. The *Focus* distinguishes for whom the sonic output of most musical interactions is intended for: this can be the audience (audience transparency, [Vin+04; CBC11]), or a group of players. Thus, it refers to the relationship between the players' control and the music produced in order to communicate meaning of compositional structure and performance, and the related feedback-cycles (cp. section 3.5). If the Focus is on the group of players, then the implicit goal is to enhance the social interaction between players, where the interactions of players with the instrument serve as means for communicating intent and meaning in the creative process [Dru09; SR04]. *Location* is important for the aspect of verbal, and more importantly, non-verbal communication. Depending on the location, players can either directly see each others' actions or have to rely on additional communication channels (*e.g.* for remote collaboration). Much of the communicative functions in a cooperative process are based on embodied actions (cp. section 3.4) and their public and simultaneous availability to the actor and the perceptions of the participants. With this mutual perception much of the intent in the ongoing creation of shared meaning is communicated [Rob97]. Especially the non-verbal communication is important as speech and music cognition can demonstrably interfere with each other, since the brain resources used in speech and music processing partially overlap [Sto+09]. Another property that is dependent on the location is the specific kind of audience that is intended to interact with the environment and with this the expected players' aptitude (the skill of the target demographic) and willingness to spend time to explore the environment. For example, if such an environment is placed as an installation in an exhibition, the general assumption is that a flow-through audience engages with the environment spontaneously [BF03; Jac+10]. Thus, it is used mainly by novices for a short period of time, therefore the system is designed for a social event and in result to be as simple as possible and self-explanatory. In contrast, a system placed in a location where players are less constrained in regards to time may lend more affordances towards creative and artistic expressiveness (long-term engagement, cp. section 4.4). Regarding *Scalability*, the number of players greatly influences the types of interfaces that can be used or music that can be produced with the system. Factors such as turn-taking protocols and gesture-sound correspondences shift as the numbers of players increase. In case of a shared tabletop interface, the limited number of concurrent users makes turn-taking patterns to coordinate collaboration feasible. Within the musical collaboration one can differentiate between the interfaces used for interaction. *Grosso modo*, the *Player Interaction* can be divided into either a single interface for the whole group or separate interfaces for each player. In both cases the players' activities contribute to the global mix for the sonic output, but the methods and opportunities for expert play, respectively novice play, can vary accordingly. For example, experts usually have the ability to separate the sound sources in the global sonic mix and therefore can more easily distinguish between their contribution to the musical whole and the one of the others [Nij09]. For novice players this may not be obvious if the action of others cannot be otherwise perceived, especially if each player has a separate interface [Swi11]. The Player Interaction is also dependent on the *Physical Interface* used. On one hand it is of primary importance as it provides the affordances that make action visible and on the other it provides the affordances what the player should do to interact [Nor88]. The most common technique to provide for an easily learned interface for novices is to limit the *Musical Range* by predetermining most

musical events. Thereby the potential for musical interaction is limited (cp. section 3.1). However, from the standpoint of musical creativity, the exploration of musical expression is vital [Nij09; Hol11]. This issue is also linked to *The Pathway from Novices to Expert Performance* that would allow novices to continue to refine their range of musical expression (see sections 3.1, 3.5 and 4.4) in a similar way that traditional acoustic music instruments have different entry levels or afford the development of a high degree of musicality [WW02]. Additional viewpoints on taxonomies are given in [Mag10] and [HW12].

· ◇ ·

For this thesis it is assumed that the collaborative creative process is not taking place in a public setting nor in front of an audience. It is therefore further assumed that the Location affords collaborators to get acquainted with the system and to assist in musical intent communication. In this way the Focus is the support of social interaction using mainly multi-touch for Player Interaction for 4-6 people as afforded by the multi-touch tabletop interface (see section 4.1). As will be argued the aim is to offer a broad Musical Range to allow The Pathway from Novices to Expert Performance (section 3.1 and 4.4) and to further foster the musical intent communication.

1.3 Paradoxes and Wickedness

Designing CSCM applications is a non-trivial task because it is difficult to frame design traits and to understand their impact, and because viewpoints on the context of (collaborative) music making can be conflicting or incomplete. As expressed by Dahl, there is inherent difficulty in locating research that is related to New Interfaces For Musical Expression (NIME), even at the most fundamental level:

> " *Attempts to precisely locate our research projects within various classifications of research are complicated by the difficulty of defining what knowledge we are generating, or whether knowledge is even our goal.* " [Dah14]

The following aims to outline broadly where these difficulties originate from. Furthermore the applied research methodology is motivated.

1.3.1 The Music Paradox

Computational processes to create music or support music-making face the problem that music composition cannot be considered properly as a modelling activity [Vag01]. The reason for this is that music is not dependent on logical constructs and, consequently, a formalization as a foundational role - although operational formalization, tools such as algorithms, or knowledge can be applied for composing music:

> " *nobody can say what music is, other than by means of a normative proposition, because "music itself" is in fact a non-demonstrable thing, and its practice is neither arbitrary nor based on physical or metaphysical foundations [...]* " [Vag01]

Thus, regarding established musical concepts, this means that "consistency implies existence" (ibid), such that similarly, even established musical primitives can be modified by means of consistent practice (ibid). An example for this is the introduction of twelve-tone music (cp. [Sch75]) that challenged the classical notion of consonance and dissonance. Musical "rules" are therefore a codification of evolving practices. Because of

the lack of formalization and the inherent interdependency between actions and perceptions - between what is planned and executed, and what is heard, since a score cannot reflect music formally - it should be clear that music lacks objective functions to allow evaluating a CSCM formally or optimizing it towards these objective functions. Another point is that musical processes involve a plurality of representational systems (ibid), which abstract musical primitives in the representational domains. These representational systems can encapsulate the production of musical processes (*e.g.* formal tools that generate or transform) or strategies to qualify results and choices (*e.g.* interactive methods). In consequence, the choice of musical primitives and representational systems manifests varied musical processes which in itself result in differing musical outcomes. Hence, designing a CSCM is a form of meta-creation that determines the wealth of musical processes and outcomes [MM07]. The plasticity of the digital realm (software) affords *digital lutherie* (cp. [Jor05b]) which describes the activity of altering or designing new instruments and musical environments according to an artistic context. However, at the same time, the user of such systems is *"the receiver and interpreter of information from the designers of the software and the sender of information in the form of the music being composed using the tool"* [Mag06]. Thus, the user creates meaning but the software is a system of representational meanings that influence the user (ibid). This duality deviates from the conception that the user is a receiver of information in traditional task-based Human Computer Interaction (HCI). In a collaborative environment, these relationships between meaning-making and signifying are disproportionally more complex, because the creative process is not embedded in a dialogue (human and machine) but in a social situation. There are more instances where music related HCI breaks with common paradigms in traditional HCI. For example, usability in HCI generally requires that a system provides utility (accomplish tasks that are meaningful to the target group); is effective (functions and information are organized in an appropriate manner); is efficient (appropriate effort to perform a task, *e.g.* "simple, intuitive use"); is easy to learn; is safe (operations have low risk) [BTT05, pp. 84]. However, as argued throughout chapter 4, musical engagement requires that a certain degree of difficulty in accomplishing tasks exists (long-term engagement vs. efficiency) that there are "unknowns" in the system (exploration vs. effectivity), that mistakes happen (serendipity / inspiration vs. safety). Therefore, usability is not an appropriate benchmark to evaluate a system or to motivate its design. Other related insights are that *"inspiration does not always precede execution"*, and that music making is a problem-*seeking* activity [FB13] which is only peripherally concerned with creating a product (artefact) [McD+13]. This is analyzed in more detail in chapter 4 and section 5.1.

1.3.2 Wicked Problems

In [Dah14] Dahl draws parallels between designing new methods for music composition and wicked problems as they were elaborated upon by Buchanan. Wicked problems are a *"class of social system problems which are ill-formulated, where the information is confusing, where there are many clients and decision makers with conflicting values, and where the ramifications in the whole system are thoroughly confusing"* [Buc92] (cited from ibid). They have a critical dependence upon human cognitive and social abilities [Hev+04]. The consequences are (cited from [Dah12]):

- They *"have no definitive formulation, but every formulation of a wicked problem*

corresponds to the formulation of a solution".

- There is no exhaustive list of admissible operations for solving wicked problems.
- *"No formulation and solution of a wicked problem has a definitive test".*

In a similar way, the design of new musical instruments faces the issues that few clear guidelines exist, that many components interact and that there is a lack of explicit criteria for success. It follows that designing has to be pragmatic and iterative, where *"problem setting and problem solving co-occur"* [Dah14]. This pragmatic account is a *"hermeneutic process of interpretation and creation of meaning"* in which prototyping is integral. It contrasts with the conservative account where design can begin once the problem is defined leading to a synthesis of solutions for the deconstructed sub-problems. Another point that is raised by Dahl is that this process itself is dichotomous: it engages in design-oriented research as prototypes are used to understand fundamental aspects of music making and at the same time it engages in research-oriented design as prototypes are applied in musical performance[2].

[2] Fallman distinguished between design-oriented research where research is the focussed area and design is the means, and research-oriented design where design is the focussed area and research is the means [Fal03].

1.4 Methodology

The previous section pointed towards a research methodology that emphasizes an iterative and exploratory design practice which is grounded in a pragmatic discourse. More specifically, this resembles a design science approach in which user experience is the criteria for assessment.

1.4.1 Experience Design

It is apparent that musical instruments (in the broadest sense) are simply not utilitarian [McD+13]. In fact, musical engagement is to a large degree hedonistic [Che14], thus, the tool oriented methodology in traditional HCI of designing such a system is inept [BBP07]. It follows that the design for musical engagement is a design for music experience with a focus on ludic interaction [3]. The approach of "experience design" fits into this conceptualization since it *"is concerned with all the issues that go into providing an engaging and enjoyable experience for people in both the short and longer term. This includes aesthetics, pleasure and emotional engagement in terms of both the product and the service provided."* [BTT05, p. 112]. McDermott *et al.* make similar observations and point out the relation of experience design in long-term engagement: *"Music interaction, it seems, must be studied in its native environment. More broadly, the language of experience design is perhaps more appropriate than that of usability for discussing music interaction. Experience design privileges consideration of the holistic experience of the interaction, which by nature is longitudinal, and must incorporate temporal changes in the human due to the interaction [...]"* [McD+13]. In this way, experience design is human centric which requires an understanding of the activities, aspirations and contexts involved in human-computer as well as human-human interaction [BTT05, ch. 1]. Thus, Benyon alludes to the necessity to frame the conceptual design which precedes physical design[4]. The conceptual design is abstract, the aim is to avoid assumptions about the specific distribution of information or functions (focus on "what" instead of "how") and rather create a rich picture of the main conceptual relationships between the main entities in a (socio-technical) system (ibid).

[3] Of course, usability is an essential prerequistite in order for a pleasurable system [Bec13]. The point is that design for usability alone is not sufficient.

[4] There are three components to physical design: operational design, representational design, and design of interactions (cited directly from [BTT05, p. 52])

This implies understanding the structure of the situation in which this system is *embedded*. Embedding is a crucial point in this thesis: music making and collaboration have a rich vocabulary of interaction and meaning making which is either established (culturally and socially) or natural (cognitive, embodied and social). To disregard the involved factors can ultimately lead to the disruption of socio-technical processes.

1.4.2 Design Science

Design science *"seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts"* [Hev+04] (cited from [Fri13, p. 4]). As characterized in [Hev+04], it focusses on the design of information systems that address the aforementioned class of wicked problems. Hevner *et al.* propose guidelines for this design process which are also useful for the presented research area although some of these are motivated by business related considerations. Based on the summary in [Fri13, p. 4], these are:

Design as an Artefact: *"Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation."*

Several concepts and prototypical implementations of socio-technical and socio-cultural environments for supporting the collaborative creation of music have been proposed and developed. These address the music related subjects of structure (rhythm and harmony), emotion and timbre.

Problem Relevance: *"The objective of design-science research is to develop technology-based solutions to important and relevant business problems."*

The market for music technology has been constantly expanding in the last decades. However there are few technologies that allow for collaboration in electronic music making. Creative endeavours, especially musical ones have been linked to mental health and well-being in numerous sources (cp. [RD00a; MKM12]). Positive effects of collaboration have also been shown in music therapy (cp. [Ans+13; Mag+11]).

Design Evaluation: *"The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well executed evaluation methods."*

Instead of the HCI related notion of utility and efficacy, evaluations focus on user experience. All prototypes have been evaluated in user studies including quantitative and qualitative approaches.

Research Contributions: *"Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies."*

Contributions have been made to the fields of NIME, HCI and CSCW / CSCM. These can be found in chapter 5 and the respective chapters describing the prototypical implementations.

Research Rigor: *"Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact."*

The design requirements have been deduced by theoretical work and ana-

lysis of related work as well as iterative prototype development. User studies have been performed to validate or dismiss the raised points.

Design as a Search Process: *"The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment."*

Multiple paths for approaching the related subjects have been proposed and tested using the prototypes. The iteration is based on a previous contribution which has been developed in the scope of my Diploma Thesis [Klü09] and subsequently published [Klü+11]. This approach has been criticised in [Xam+13]: *"...the authors adopt a traditional approach of supporting music composition and notation with less consideration to contemporary music practices. A contemporary music approach tends to use alternative musical instructions more focused on the music process (Cox and Warner 2004; Gresham-Lancaster 1998), a practice close to the notion of unpredictability and uncertainty, which arguably tends to be present in music performance with novel interfaces for music."* This valid point is addressed in chapter 3. My own points of criticism are that this prototype depends on specific technical knowledge which conflicts with supporting heterogeneous groups in the musical collaboration (discussed in the same chapter). It covers many aspects of music creation (rhythm, harmony and sound-design) and provides equally broad controls. In consequence, it gives importance to peripheral tasks which is detrimental to creative engagement with the domain (chapter 4). The breadth of possible operations and the implementation thereof makes evaluation and deduction of insights challenging. These are the reasons why the musical domains of rhythm, harmony and timbre have been approached with separate prototypes.

Communication of Research: *"Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences."* The prototypes and results of the evaluations have been presented and published in international, specifically music technology related conferences.

1.5 Outline

The aim of the following chapters is to create the aforementioned ”rich picture” in order to frame the scope of the socio-technical environment and its relationships to traits of musical engagement. Inter alia, this involves creating conceptual scenarios, characterizing conceptual personae and their mental models with respect to musical and social engagement, as well as considering the role of the physical environment and social context in which collaboration takes place. Chapter 3 is concerned with exploring these contexts, it also shows that music is in many parts rooted deeply in social, cultural and embodied practices. Chapter 4 argues from the perspective of Human-Computer Interaction and practices in collaborative music improvisation to point out several desirable design goals for prototypical implementations. Furthermore, it is argued that the tabletop as shared interface can be embedded fittingly (section 4.1) into these contexts and that electronic dance music is a suitable scope of the focussed musical activity from the viewpoint of shared knowledge and collective creativity (section 5.3). Chapter 2 gives a short overview of the history of electronic music production and practices. More importantly, it is revealed that most tools are focussing on solo use and require specialized know-



Figure 1.1: The *reacTable* application makes use of a modular synthesizer metaphor and employ multi-touch and tangible interaction for controlling sound generating and modifying building blocks. Taken from [Wik07]

ledge. This conflicts with prerequisites for collaborative creativity support and thus motivates the focus on alternative interaction and organization methods in applications. The main insights of this "rich picture" are summarized in chapter 5. This is followed by an overview of the software architecture of the generic framework that has been used to build the prototypes including additional software packages.

· ◇ ·

After this first foundational part, prototypical applications and models are presented. These are subsumed into three categorical aspects of music: timbre, emotion and structure. This partition is mainly motivated by the focus of the prototypes. However, an alternative scheme may be equally valid. All prototypes have been evaluated in user studies with focus on user experience. Furthermore, certain studies are tangential to musicology, such that additional insights that are not mainly related with the core research question have been gained.

· ◇ ·

The thesis concludes with a summary and outlook, including starting points for future works.

1.6 Related Work

While the most relevant related work for the approached topics is presented in the respective chapters, the following gives an overview of applications which can be considered as directly related, preceding contributions to the presented field.

· ◇ ·

In the last decade several music applications have been developed that are based on a single display with multi-touch or tangible input modalities. Most notable is the seminal work of the *reacTable* team [Jor+06], which has created a system that is intended for collaborative use and has been widely used in exhibitions, concerts and educational institutions for children (the system is shown in figure 1.1). It is roughly based on the concept of a modular synthesizer in which sound generating and modifying blocks can be interconnected to generate audio. The interfaces uses a mixture of tangible and multi-touch interaction such that the audio related blocks

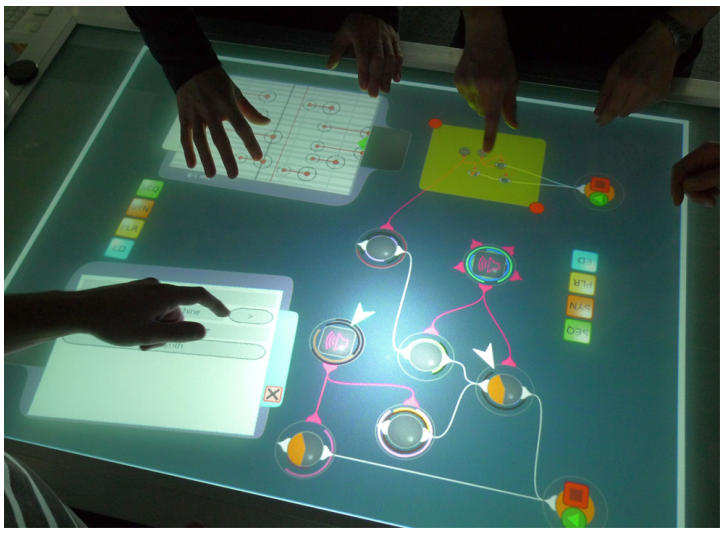


Figure 1.2: The first iteration of a collaborative music-making environment *lzrdm* by the present author, [Klü+11]

are represented physically as *tangibles*. Physical interaction (*e.g.* rotation) with these alters synthesis related parameters while additional functionality is accessed via touch interaction. It supports the explorative creation of music without preceding limitations on the musical expressiveness for the setting of co-located collaboration as a design principle in a playful way. Regarding the expressiveness, it allows both, experts and novices, to take part in the creative process of creating music. However, one main point of criticism is that it does so in a quite technical way, which makes the targeted exploration of the sonic possibilities offered by the system difficult for novices. This viewpoint will be elaborated in chapter 2. Even nowadays the *reacTable* remains the most influential collaborative music making system such that later contributions to this field use similar interaction paradigms.

· ◇ ·

Our previous contribution [Klü+11] was similarly technically inclined as the *reacTable* since the main focus was to translate common studio practices in a Digital Audio Workstation (DAW) to a collaborative use-case. It allows the composition of harmonic structures, song arrangements, limited synthesis control and real-time playback of song fragments for improvisation. Compared to the presented related work and the contributions in this thesis, the application follows a more comprehensive approach to music composition. However, unlike the present thesis, the design considerations for this prototype (see fig. 1.2) were principally informed by design traits from HCI and CSCW. The critique in the previous section 1.4.2 already expressed the issues with the design and this comprehensive approach.

· ◇ ·

SoundXY4 [Xam+14] and *Touchtr4ck* [XLD11] are recent contributions by the research group that co-created the original *reacTable*. These applications have been developed as part of Anna Xambó's PhD thesis. *SoundXY4* allows interlocutors to compose musique concrète in a style inspired by Russolo's *The Art of Noise*. Musically, the scope of the application is limited as samples of recorded sounds can be only played back and filtered. Similar to the *reacTable*, tangibles were used to represent these sounds and filters. However, what is interesting is that the prototypical implementation was concerned with extending awareness considerations that

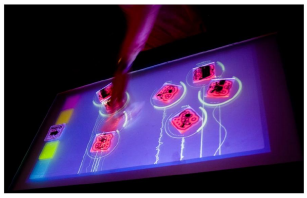


Figure 1.3: Picture of *mixiTUI* application for mixing audio samples and applying effects. Focus is the performer-audience relationship (taken from [PH09])



Figure 1.4: *Xenakis* is a generative music application for single users (taken from [Bis+08a])

have thus far been mainly applied to tabletop interaction to the audio delivery method. An ambisonic speaker arrangement was used to give each sound a location in the room depending on the position of the respective tangible on the tabletop. *Touchtr4ck*, focuses on the design principles that allow collaboration in a democratic way, thus implementing many of the design principles proposed in [Sco+04; Kru+05; Hor05].
The application design is inspired by the approach to collaborative electronic music performance Stockhausen chose for his aforementioned piece *Mikrophonie I* (1963) in which there is a real-time division of labour between producing, recording and transforming sounds. For this, shared multi-touch controls are employed to record, play-back and process four audio loops.



Figure 1.5: Similar to *Xenakis*, *ANTracks* allows to generatively create music on a tabletop (taken from [Wol+10])

· ◇ ·

Similar to the last two prototypes, *mixiTUI* [PH09] (figure 1.3) and Audiopad [PRI02] allow users to mix pre-recorded audio loops and to apply further effects. However, the tabletop is seen here as a mediator in the performer-audience relationship[5] rather than the musical collaboration between musicians. Other single user tabletop applications include *Xenakis* [Bis+08a] (figure 1.4) or *ANTracks* (figure 1.5) [Wol+10]. These applications use a generative approach to making music by steering stochastic processes for music generation. *Scrapple* [Lev06], for example, allows to design and resynthesize spectrograms in real-time. These single user interfaces show interesting alternative approaches to create music which are independent of the classic loop playback or modular synthesis approaches and still allow a fair degree of expression. Of course these systems circumvent the (design) issues that arise for a collaborative environment.

[5] This relationship can be seen as a form of collaboration as well.

· ◇ ·

Of course, with the widespread adoption of multi-touch technology in consumer devices, supporting co-located collaboration in a similar fashion to tabletop applications became feasible for tablet computers. *Cotracks* is an application designed for the *Apple iPad* [App10] that provides comparatively sophisticated tools to create loop-based music collaboratively [Fut13]. It offers several interface layouts to concurrently use virtual instruments in groups of two to four people. Collaboration is roughly based on a division of labour similar to a band setting where each user has control over a dedicated (set of) instrument(s). Because of the small size of the shared display certain beneficial design traits such as supporting territoriality (cp. chapter 4) are not leveraged. Further information about the user appreciation is not available and a more general evaluation of the interface design has not been published, possibly because the application has been released just recently.

· ◇ ·

Another interesting approach to collaborative music composition is *Composition on the Table* [Iwa99], which allows users to construct rhythmic and harmonic loops by creating paths along a grid structure on which user generated objects can move around (as can be seen in figure 1.6). Albeit the system severely limits the expression since the harmonic pitches are fixed and the created rhythmic loops have a short length, it provides an interface which is easy to grasp for novices and still allows some musical complexity to emerge [BF03].

· ◇ ·

An example for non-democratic use of a collaborative musical system that does especially support expert and novices alike is *Iltur* [WD05]. The system uses special controllers to interact with the application while experts interact with their traditional instruments only as exemplified in figure 1.7.

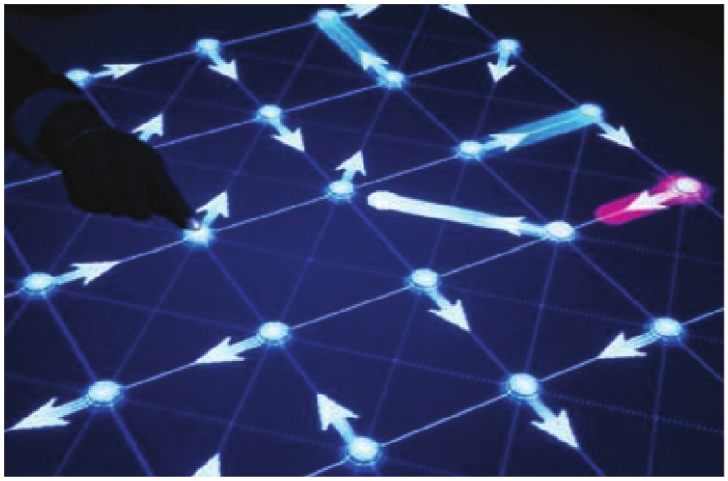


Figure 1.6: *Composition on the Table* [Iwa99]

The support is achieved by a hierarchical process in which the sound played by experts is recorded and analysed in real-time upon request by novices using their special controllers. Novices can then play the analysed recordings back with the further possibility to generatively permute them e.g. transposing or rearranging the melody line played by a trumpet. The system keeps track of the overall musical and adapts the control input of the novices to fit to e.g. to the harmonic content that is played by the experts. Here, different interfaces and processes are used for experts and novices while the system has the role of a mediator between those two. This approach of employing a hierarchical music-making process and giving the computer the role of a mediator seems very promising and may make sense in the setting of using a Single Display Groupware environment and creating electronic music as well.



Figure 1.7: *Iltur* [WD05] used in a studio recording; the trumpet playing of the person to the left is recorded so it can be altered in realtime by the person to the right interfacing the system using a specialized controller [WD05]




Part

I

The Rich Picture




Chapter 2

From Musique Concrète to the Bedroom Producer

"This revolution of music is paralleled by the increasing proliferation of machinery sharing in human labor. In the pounding atmosphere of great cities as well as in the formerly silent countryside, machines create today such a large number of varied noises that pure sound, with its littleness and its monotony, now fails to arouse any emotion."

Luigi Russolo, 1913

Chapter Abstract

The following outlines the history of electronic music in the context of studio works. Emphasis is given to the set of skills acquired for modern (electronic) music production, the transition from shared group processes to one that centres around highly qualified individuals. It is argued that this technology driven change is also reflected in the involved tool chains that make integration of collaborative practices difficult.

As argued by Simon Emmerson in [Emm86, ch. 1], Electroacoustic Music is the single novel music form that developed within the 20th century, because it signified a point of departure from core conceptions of (classical) music. Within this development, the creation and understanding of macro-, micro structures and timbre of music pieces was transformed. Electroacoustic Music is a meta-genre which is nowadays a catch-all term for music that is produced with electronic means in an institutional context or within classical tradition [Dem10, ch. 1]. The approximate pendant in the popular culture is called *Electronica* and includes various electronic music forms such as Electronic Dance Music, Ambient, Industrial, Indietronica, Krautrock, Synthpop, Breakbeat and Glitch, to name a few (ibid).

2.1 Breaking with History

The technological development that served as a basis for this change came with the establishment of public radio, the entertainment industry [Wor09],

and the invention of music related devices between 1920 - 1945 [Hol12, p. 19]. Especially the backlash against Old World arts and values [Hol12, p. 42] and the post-war availability of tools for editing and recording tapes as well as shellac discs [Man03] made the artistic examination of these new technologies feasible and its conditions favourable. The direct result of this is *musique concrète*, pioneered by the French (broadcast technician and then composer) Pierre Scheaffer and Pierre Henry. Cox summarizes the history and impact of Pierre Schaeffer' work, or *musique concrète* in general, on music in the popular culture as:

> *"[...] Schaeffer's method of composition bore a closer resemblance to cinematic montage than it did to traditional music composition. The major European avant-garde composers (Stockhausen, Boulez, etc.) flocked to his Paris studio; but, ultimately,the impact of Schaeffer's work was felt most strongly outside classical music, for example, in the early tape experiments of Les Paul, the studio manipulations of Beatles producer George Martin, [...], the sampling and turntablism of HipHop DJs from Grandmaster Flash to Q-Bert "* [CW04, p. 5]

Shortly after he founded the *Groupe de Recherches Musicales* (GRM) with a dedicated studio with support of his former employer *Radio-diffusion Television Françaises* (RTF) in 1951 to nourish research in *musique concrète*, the Nordwestdeutscher Rundfunk established a similar facility under supervision of Herbert Eimert [Hol12, p. 51]. However, the technical and aesthetic direction was different: the studio for *elektronische Musik* followed the footsteps of serialist composers and focussed predominantly on the creation of purely synthesized sound objects as opposed to sampled and manipulated ones at GRM. The animosity between these two studios for the next two decades has been documented in various sources (ibid). Although the *techné* employed in both studios differed, the artistic products were oftentimes indistinguishable [Hol12, p. 76]. As will become apparent in chapter 8, even vastly different technological approaches in signal processing can lead to similar acoustic results.

2.2 Studio Works in the Mainstream

Following the international recognition of works created in these studios, in a second wave, new studios opened throughout Europe and America [Man03]. Perhaps the most well known is the BBC Radiophonic Workshop (cp. [Nie10]) where sound effects for the BBC's radio and television programmes were devised. In this second wave the acoustic results of the newly introduced technologies and techniques were publicised through mass media, not only in the form of sound effects for movies, radio or television but also in the form of new compositions, *e.g.* the Doctor Who Theme by Ron Gainer (1963), which was realised by Delia Derbyshire at the Radiophonic Workshop as a mixture of *musique concrète* tape modifications and synthesized sounds. With this, these processes became gradually adapted and established as production methods in music studios. For example, The Beatles experimented with tape loops (and tape based effects) in their late period [Mac07, ch. 1] for the albums *Revolver* (1966) or *Sgt. Pepper's Lonely Hearts Club Band* [1] and The Beach Boys used tapes to layer the voice harmonies (and re-recording the results) in *Pet Sounds* (1966), mixed synthesized bass sounds with the playing of a real Fender bass or created song arrangements by re-assembling recordings taken from different versions of a song (*Good Vibrations*, 1966) (ibid).

[1] Interestingly the cover art for this album also shows a picture of Karlheinz Stockhausen

· ◇ ·

The raise of the recording studio not only as a creativity enabling institution for artists but also as an economic keystone of the music industry can be seen in the shift from live performances to recorded works as the *"dominant industrialized commercial medium of popular music"* [Har14]. The affordability of studio technology at the end of the 1970s finally led to the adoption of these production tools in the underground music culture, for example in Punk, Disco and the later early Electronica with its emerging Techno scene, or Hip Hop [Mil13]. Thus, the egalitarian effects of mass-producing means for electronic music creation (starting in the 1980s with the introduction of affordable digital synthesizers), resulted in that the *technical* acquisition methods of material (sound objects, composition structure, ...) within the two spheres of "high arts" and the mass-appeal electronic music merged [Dem10, ch. 1]. However, the distinction between these two is still apparent in the scientific discourse (ibid); a more in-depth essay about this relationship can be found in [Emm07, ch. 3].

2.3 The Studio as an Instrument

It was in the same period when the music studio became an instrument of its own, a new compositional device, that altered the role and habitus of the classic record producer and composer - apart from offering the means to create new sonic forms [Mac07, ch. 1], [PB04]. The producer-composer Brian Eno (Roxy Music), who has been vocal about this change since his solo-works in the late 1970s summarizes this in the following:

> *" In a compositional sense this takes the making of music away from any traditional way that composers worked [...] and one becomes empirical in a way that the classical composer never was. You're working directly with sound, and there's no transmission loss between you and the sound -- you handle it, it puts the composer in the identical position of a painter -- he's working [...] directly with the substance [...] "* [Eno04, p. 129]

The following explosion of popular electronic music can therefore not just be reduced to the affordability of technology or the larger sonic palette but also involves a socio-cultural change: single persons can create and shape music and sound, omitting the necessity to perform with or compose for a group of musicians in order to manifest a sonic or compositional vision [Hom09]. Furthermore, this development was accelerated with the advent of MIDI, a protocol that allows recording and editing of performance-data (using dedicated hardware or a computer) and controlling electronic instruments in detail (ibid) and, later, with the affordability of music studio solutions that rely entirely on software. Nowadays, this is apparent in the vast majority of new electronic music being produced by "bedroom producers", people who have set up a studio in their own home, not relying on commercial studios in order to record their music [Ram06; Car12]. Thus, this third transition also signals a departure from studio work as a collaborative process to one that is centred around an individual (ibid) in its entirety. The implications will be discussed in the next section.

· ◇ ·

Another departure in this regard is that music production work is shifting towards interactive performance, where music is created in a feedback cycle between the creative individual and the recording or composition tools [Vag01]. This "interactive play" is therefore a mixture between real-time music composition and musical improvisation [Ram06]. Thus, apart from sound material, representational and social implications on music creation, with the establishment of recording technologies also music composition processes have changed. Central to this aspect is the sequencer,

a hardware device or software that has the purpose of linearisation of musical material [DNB05] and as such create a linear, temporal ordering of performance data for (sound producing) studio equipment which can be in general repeatedly edited, played back and also recorded in real-time[2]. The resulting impact on the music creation process is that composition and performance both become non-linear and are free to be structured into a musical form at any time since performance data can be rearranged and every improvisation can be submitted to a reelaboration [Tar95][p. 7-9]. Furthermore, the intermediate level of note or harmonic structures can be omitted, it allows to construct sonorities and their temporal linearisation exactly. As performance data and performance are decoupled, so are the roles of the composer and performance skills of instrument playing. Especially this gave raise to new musical forms that are detached from the physical abilities of any human performer (cp. [GP05][p. 165-177]). Thus, one can see that the sequencer itself is another technological development based on serialist music principles and concepts from electro-acoustic music.

[2] A taxonomy of the various modern sequencing concepts is given in [DNB05]

· ◇ ·

The influence of (technological) developments in Electroacoustic music on Electronica is still apparent, for example, in the numerous tools originally developed at academic institutes such as IRCAM (Institut de Recherche et Coordination Acoustique/Musique) or GRAME (Centre National de Création Musicale) that have been opened to the music producing public as commercial products[3].

· ◇ ·

Given the commonalities between the academic and popular electronic music that have been presented thus far, we can subsume that - excluding antipodal cultural phenomena - phenomenological and theoretical argumentations and insights that regard Electroacoustic Music also hold in the context of Electronica [4].

[3] *e.g.* the IRCAM Studio Tools that have been developed in collaboration with Flux, a professional studio software company, [Flu13], or the Kyma hardware and software environment that is frequently used in commercial sound-design which was originally developed at the CERL Sound Group at the University of Illinois [Sym90]

2.4 The Bedroom Producer

De Carvalho *et al.* discuss the various roles of the "bedroom producer" or "home recordist" in [Car12], stating that home recording is a one-person job involving a large array of roles and associated expertises because of the aforementioned technological and inherent social changes in studio work. Depending on the literacy in the activity of home recording, the expertises overlap with classic professions in the recording studio, such as the mixing engineer, sound engineer, studio designer, and the producer and (session) musician [Ram06; Car12]. The acquisition of those required skills and know-how is generally seen as an ongoing process that aims to accomplish the proficiency on par of professionals. Furthermore, the latter requires, to some degree, the accumulation of professional tools and therefore presupposes both, sufficient dedication to activity and the necessary financial assets. Apart from the accumulation of such hard skills, other necessities are the ability for critical self-evaluation (feasibility of musical ideas) and, of course, the musical artistry. Moreover, in the professional realm, freelance work is predominant while "real" work experience and expertise is preferred over certificates or qualifications from institutions [Car12; HB10]. Hence, with project related work that is carried out in home studios and the necessity of freelancers to invest into their carriers [HB10], affordable musical tools for home recording are bound to become more professional. To

[4] I am pointing this out, since the related work presented in this thesis stems mainly from research that is concerned with Electroacoustic music due to the scarcity of musicological publications that are related to popular electronic music in the discussed fields.

summarize, the home recordist as a prototypical, modern music creator is a person that represents very high musical and technological literacy conjoined with high motivation and dedication to acquire capital and skills in a time-consuming process.

2.5 Contemporary Techniques for Collaboration in the Studio

As outlined, the music production process changed from a shared group process to an individual one throughout the last decades. Expectedly, the production tools reflect this [Ram06]. To gain a broader view and understand the consequential implications, it is worth exemplifying how current studio technology integrates into collaborative practices. Generally, the distinction between composing and performing music is blurred in (real-time) electronic music production [XLD11] since the interpretation which transforms the symbolic representation of music to sound is largely omitted. From a superficial and purely mechanics informed viewpoint, collaboration in electronic music making at least requires the following:

Composition: Sharing of artefacts which represent musical products such that these can be re-integrated into another (alternative, iterative) musical context.

Performance: Shared control over artefacts and generating processes. This means that the process of creating of artefacts facilitates multiple points of access for interaction.

There are few established techniques to exchange artefacts due to the scarcity of standardized document formats or protocols. Apart from standards for video and sound authoring such as *Open Media Framework* (OMF, [EDL97]) or *Advanced Authoring Format* (AAF, [Adv06]) which do not have reasonably widespread support outside of the broadcasting industry, the most common method is to either exchange MIDI file data or rendered audio material (called stems). Stems lack plasticity since they represent the recorded result of the generating processes (symbolic data, automation, synthesis), transformations can only be applied as a post-processing method within a comparatively limited scope. MIDI files on the other hand represent just symbolic data (note events) and automation data. However, apart from a comparatively simple mapping of symbolic data to instrument control in the General MIDI standard, the MIDI file format does not contain any additional ways of saving mappings of control data to the sound generating processes. To summarize, exchanging data between platforms (software or hardware) is either not technically possible or results in a loss of information. Exchange using the same software platform instead is more common practice. Here, usually project files are exchanged. There are several (online) services that facilitate such a project oriented file exchange and remote collaboration. Examples of these are *Ohmstudio* [Ohm11] (as an integrated Digital Audio Workstation and remote collaboration environment), or *blend.io* [Ble14] and *Splice* [spl14]. which are more similar to a cloud storage system with additional project management facilities. However, to my knowledge no common software platforms currently allow co-located collaboration for sharing documents and concurrently working on these.

· ◇ ·

Another practice that is supported in most modern DAWs is to create mappings from hardware control surfaces to software parameters for tactile real-time control. In this way control over the generating processes can be shared. Since most DAWs are still exclusively single-user environments, certain functions generally cannot be performed concurrently, for example editing or recording performance data which means that iterative processes in creative practice are not easily facilitated (cp. chapter 3). In this scenario a single computer acts as a hub for the collaborative music performance.

· ◇ ·

In studio setups where devices are distributed (*e.g.* hardware, several software systems running on the same machine or on different ones), there are additional infrastructural problems. These include synchronization of all involved devices, and sharing or routing control and audio signals. Common methods employ MIDI for clock synchronization of the devices and sharing MIDI or OSC control, so one device is able to control another. MIDI clock synchronization is done with one device acting as master clock, setting the tempo for all other (MIDI) connected devices and setting start and stop signals. The sharing of MIDI or OSC control can be achieved in a similar way. However, this adoption of MIDI can already be seen as misuse, as this protocol and its hardware is devised to be unidirectional, connecting devices either end-to-end or by a bus. One central problem is that there is no standardized mapping in MIDI or OSC, therefore setting up such control networks involves technical skills, knowledge about the device specific MIDI or OSC implementation since this tends to differ significantly from device to device. In a hardware centric setup accessories such as MIDI expanders, repeaters and additional cables are needed. Similar configurations are necessary for routing audio. There are additional issues involved such as latency compensation for digital processors, format conversion of digital control protocols and analogue signals (voltage, balanced / unbalanced connections). With such an infrastructure it is possible to shape the audio signals collaboratively or distribute control over the involved devices. However, sharing artefacts is not facilitated.

· ◇ ·

To conclude, it is apparent that even simplifying the means for collaboration to a purely technical and functional environment that completely disregards prerequisites for supporting human interaction, social context and collaborative musical practices, supporting collaboration is a difficult undertaking because devices integrate improperly into a shared environment.

Chapter 3

Social Aspects: Creativity, Music, Interaction and Flow

"If all meanings could be adequately expressed by words, the arts of painting and music would not exist. There are values and meanings that can be expressed only by immediately visible [...] and audible qualities, and to ask what they mean in the sense of something can be put into words is to deny their distinctive existence"

Dewey, *Art as experience*, [Dew05, p. 77]

Chapter Abstract

This chapter summarizes the various contexts in which creativity, embodiment and musical engagement and its proximate functions exist. It is shown that in each of these, social practices are embedded deeply such that focussing on the support for collaborative creation of music in this thesis is valid.

While the previous argumentation has already implied that collaboration and music-making are part of a complex interplay between technology, social factors of engagement and creativity, it is nonetheless necessary to further analyse these matters. This establishes fundamentals that in the following chapters serve to frame the requirements for successful CSCM applications.

3.1 Creativity

Music-making is a creative endeavour. This raises the question what creativity is and how it relates to the socio-musical context. As with any matter that is multifaceted, there exist multiple definitions for it, with the most commonly held that creativity involves the creation of a novel idea that serves some purpose [Ste99, p. 137-144]. Thus creativity represents a balance between the originality of an idea and its usefulness, the quality to solve a problem, resolve a difficulty or fulfil some desire. There are two viewpoints in which creativity produces such ideas, the personal view and

the social view ([Ste99, p. 450-452] and [Csi97]). In the former viewpoint it is argued that human cognition offers creative capacity as an essential property, in the latter creativity stems from the assessment of value of an original idea for a certain domain by means of social and cultural processes as explained by Csikszentmihalyi in [Ste99, p. 313-335]. The relationship between these creative identities need not to be an antipodal one [Fis05b] but up to now a unifying theory is still missing. However, the general consensus in recent research is that a confluence of factors are responsible for creativity to emerge [TM13], thus admitting the socio-cultural interactions between the individual and their environment. Therefore, from the position of the Systems View of Creativity (e.g. [Csi88]), it is the system itself that produces creativity.

3.1.1 The Individualist Perspective

Individual creativity assumes that the individual brings in situated actions (actions that cannot be planned as a whole) to cope with the current problem or situation [Fis05b]. Although there are investigations that found that creative individuals exhibit special personal traits, there is no conclusive empirical evidence that special personalities are a precondition for creative behaviour [TM13]. In this way, Csikszentmihalyi concludes (cited from ibid):

> " *Creative individuals are remarkable for their ability to adapt to almost any situation and to make do with whatever is at hand to reach their goals. If nothing else, this distinguishes them from the rest of us. But there does not seem to be a particular set of traits that a person must have in order to come up with a valuable novelty [...]* " [Csi97, p. 51]

With the initial definition given above, the most generic attribute of creative individuals is the *"capacity to identify a pertinent problem in their chosen field and the development of related questions to solve it"* [TM13], which is the result of experience and knowledge rather than a biological predisposition (ibid). Apart from this, creative individuals most likely exhibit several personality traits. Csikszentmihalyi proposes a four stage model that consists of four phases leading to the creative or innovative product: preparation, incubation, insight and elaboration. These stages can overlap and involve various iterations depending on the creative task. This model can be seen as an elaboration of the more general phases of divergent thinking, the generation of many ideas that are possibly adequate as a solution and convergent thinking, where the generated ideas are evaluated and selected based on appropriateness [Saw11, p. 88].

· ◇ ·

A similar model is proposed by Gelineck *et al.* in [GS09] for the case of music composition of electronic music by individuals. It is based on a study that involved the analysis of the work-flows for music composition from 18 expert music producers. As shown in figure 3.1, the model consists of three phases, exploration, editing and working pragmatically. Similar to Csikszentmihalyi's work, the model is non-linear concerning the sequence of stages. Exploration involved trail and error to express an idea but also getting inspiration from the music tools itself, as *"[m]ost prefer tools that they don't understand fully, tools that are unpredictable in some way, or tools that they can use in unintended ways"* (ibid). Intuitive and predictable tools that are, however, preferred for the editing and pragmatic mode as they lend itself to accurate control over the composition. In this way it becomes apparent that these musicians have a dichotomous behaviour, the one of

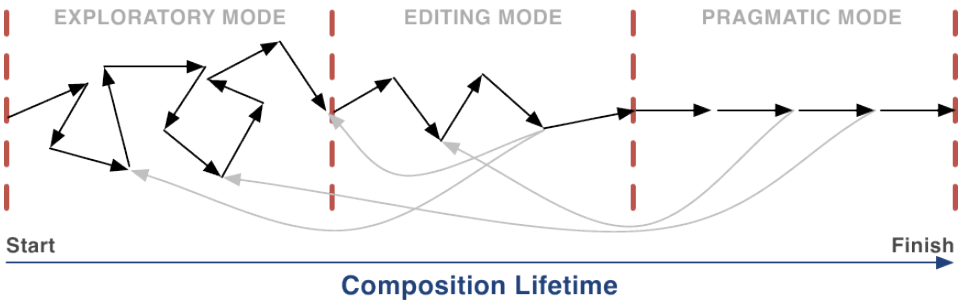


Figure 3.1: Compositional process of music as proposed by Gelineck *et al.* that consists of three modes. An exploratory mode, an editing mode and a pragmatic mode.

the "explorer" and the one of the "worker", and similarly, tools or a set thereof have to reflect this. Furthermore, it was revealed that most subjects impose artificial boundaries, rules or dogmas to their options in order to either guide or challenge their creative processes. Hence, freedom, with respect to tools, is seen as the ability to subvert such boundaries or initial directions when seen fit without externally imposed restrictions. Therefore this form of freedom is seen as equally important to the creative process. Concluding to this study, Gelineck *et al.* propose several design goals for related electronic music instruments. Relevant for this thesis are the proposals *Design for unintended use* and *Design for a balance between an intuitive tool and an unpredictable tool*. Throughout this thesis, these are one of the main points for argumentation and evaluation of prototypes, namely the targeted creation of musical artefacts, which is analogous to a convergent phase, and the exploration of the respective musical domain (cp. chapter 5), which is identical to a divergent phase. Of course, there is a large body of work conceptualizing and analysing the creative processes in music composition that are largely in agreement with the presented concepts and emphasize the associated qualities of linear / step-wise and recursive / iterative practices and repeated aggregation of compositional elements ("chunking", cp. [Bam77]) in this task [Col05]. However, Collins criticised previous related contributions for being too contrived to allow for a naturalistic creative setting to take place (ibid). Hence, he analysed the composition processes of a professional composer in a three year study that on the one hand similarly supported previous work but also revealed that solution spaces are highly context driven, that the processes are concurrent and cumulative rather than distinct. While these results are specified within the domain of music composition, on a more general level, the highly parallel and iterative nature of music composition (and simultaneous handling of multiple operations) and the support thereof overlaps with elemental requirements for collaborative Creativity Support Systems (cp. section 4.1). Admittedly, Collins' study does not focus on the composition of electronic music (as the other contributions) but rather "traditional" computer aided music composition. However, apart from implications based on the medium, the main issue with this study is that it focusses on a professional musician because the motivation for musical engagement differs from that of other groups (*e.g.* novices, amateurs; cp. 3.3) which may result in altered exploration strategies. There are also differences in individual composing processes [Bia12].

3.1.2 The Group Perspective

For the case of social creativity, the differences in knowledge, expertise, and perspectives that exist among individuals provide the ground to col-

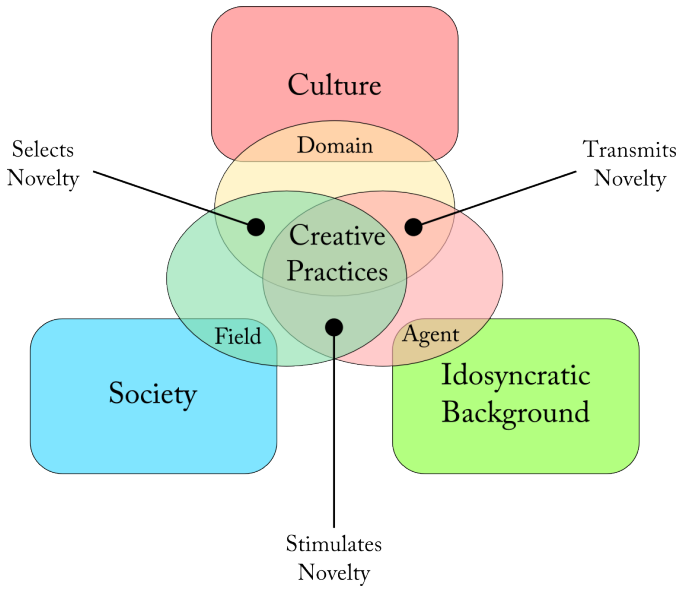


Figure 3.2: Systems model of creativity (adapted from [TM13])

laborate toward more creative and sustainable solutions (ibid). Similar to Fischer, Uzzi [US05] shows within the notion of "small world networks", that many key innovations in art, science, and politics have been created within a network of artists or scientists that shared and judged their ideas together with results that supersede what a single person could have been done in isolation (more on this will be given in the context of Flow, section 3.5) [Fis05a]. In the matter of Collaborative Computer Supported Work, Fischer points out [Fis05a]: "*Much human creativity arises from activities that take place in a social context in which interactions with other people and the shared artefacts are important contributors to the process. Social creativity comes alive in socio-technical environments in which communities collaborate.*". Both Csikszentmihalyi [Ste99, p. 450-452], [Mor80] and Fischer [Fis05b] argue that for social creativity the heterogeneity of individuals, the overall diversities in knowledge, experience and expertise are the key elements to foster creativity: "*The challenge is often not to reduce heterogeneity and specialization, but to support it, manage it, and integrate it by finding ways to build bridges between local knowledge sources and by exploiting conceptual collisions and breakdowns as sources for innovation.*". Sawyer further points out that, generally, cognitive diversity rather than ethnic, national or gender diversity contributes to group creativity and that heterogeneous teams outperform homogeneous teams in complex tasks [Saw11, p. 234]. However, he also makes clear that there is an optimal degree of diversity, such that extremes may impede creativity and that a shared sense of purpose and commitment to the group's goals are a necessity. In such a setting, the computer can be seen as a mediator for the collaboration, helping to communicate musical intent or to facilitate the realization of musical ideas in this heterogeneous group.

3.1.3 Creative Practice: Confluence System

One of the aforementioned confluence systems is presented in [TM13] which itself is based on the work of Kerrigan (cp. [Ker13]) but adapted to the creative practice in record production and studio recording (education). The creative practice is described as the result of the interaction between three

overlapping environments (see fig. 3.2). The domain is part of culture and contains the body of knowledge and a set of symbol systems in which an individual is knowledgeable. It is *"informally and formally dependent upon the category or style of music with which [the involved] musicians engage"* (ibid). Essential to the individual is the domain acquisition which is generally an ad-hoc process of mentoring and auto-didacticism. Depending on the field of music making, this can be entirely informal as has been shown for Hip-Hop and dance music producers. This is the process where musicians get musically encultured (cp. chapters 2 and 7). Being knowledgeable in the rules and conventions that govern a domain is therefore a necessity to validate a product within that domain but also to be able to deconstruct these in order to achieve novelty. The field is a part of a community that evaluates and validates the creative product (assess, reject or accept ideas) of an individual, thus it is able to utilize and understand the knowledge of the domain. Therefore this also means that the field validates the domain implicitly. As a social organization it mediates the production of music by filtering novelty, being proactive or reactive in its stimulation of novelty or supporting creativity through association with the broader social system. In the case of music improvisation within a collective, such as the intended use case, the evaluation of new musical ideas is a collaborative process. Here, the ideas introduced by a performer are immediately evaluated by the other performers, *"determining whether or not the performance will shift to incorporate the proposed new idea"* [Saw11, p. 346]. Sawyer further summarizes group creativity within the field (as defined in this section) in the following:

> " *Once we realize that performance creativity is like a collaborative conversation, we see that it's not created by any one of the performers. The individualist approach [which has been presented above] can partially explain what's going on, but it can't provide the complete explanation of group creativity. To explain group creativity, we also need to incorporate the sociocultural approach, bringing all approaches together in an interdisciplinary approach that analyzes collaboration, interaction, and group dynamics [...]* " [Saw11, p. 347]

The third component is the individual, whose creation joins the domain if it is validated by the field and if it survives subsequent re-evaluation.

· ◇ ·

Another insight is that the relationship between (electronic) music tools and the domain is bidirectional, as pointed out by Bennet in [Ben10]: *"[...] this is where the studio or music computer plays a significant part in the creative process - it becomes arguably the "invisible writer" in that it generates or enables stimulus material that would not be part of the creative process in a non-studio environment"*. A similar stance is taken in [MM07], albeit from the viewpoint of digital lutherie (cp. [Jor05b]) which highlights one of the seminal differences between electronic and acoustic music creation: *"The process of designing the instrument becomes a process of composing at the same time. The fact that people talk about "composing instruments" [8] yields a clear distinction from the acoustic world [...]"*. Further support for this viewpoint can be found in [Mag06] and [DNB10].

· ◇ ·

The two design principles, *Design for unintended use* and *Design for a balance between an intuitive tool and an unpredictable tool*, are also echoed by Sawyer with respect to group creativity:

> " *Teams generate innovation through interaction among members of the team. If the interactions are predictable [...] or if they're too strongly controlled [...] then no unexpected innovations will emerge. Teams are more likely to generate innovation when the group task*

> *design allows for a group process that is emergent, unpredictable, and improvisational (Sawyer, 2007). Yet research also shows that completely unstructured team interactions aren't ideal; the introduction of an appropriate task design to guide the team's creative process can enhance the innovation potential of"* [Saw11, p. 347]

It becomes apparent, that the dichotomy that was identified by Gelineck (freedom versus boundaries) also holds for the collaborative case of musical creativity. In this regard the question arises if rules should be imposed on the creative process or whether the functional focus of the prototypical implementations is sufficient enough as a restriction.

3.2 Music and Social Being

In [Cro05], Cross argues that music is a fundamental part of the human communicative toolkit and that it is equally symbolic and a natural domain of human thought and behaviour. The following section will summarize several points raised in this contribution and in [Bis09] because these are valuable for the continuing argumentation in this thesis and most importantly, it underpins the impact of social contexts on music and vice-versa. Furthermore, there is strong evidence that, from a biological standpoint, language and music share many similarities and that humans are primed for music. The capacity to engage with or in music is not only a cultural construct.

3.2.1 Proximate functions

When we look at the collaborative creation of music, we also have to frame the underlying social processes and foundational motivators. Joint intentionality is a specific feature of human psychology which involves the conception and maintenance of shared intentions [Bis09] it is *"manifested in the multifarious and dynamic systems of shared meanings through which we may understand each other as ourselves and act together on the world"* [Cro05]. The resulting joint actions therefore describe *"shared intentional actions as shared goals and coordinated action plans [...] in terms of principles of [...] regulatory systems [...] which adjust actions in order to minimize the discrepancy between the actual states and goal states of the interactants"* [Bis09]. Accordingly, this furthermore implies comprehensions and enactment of roles. These processes are fundamental to musical engagement and have been shown empirically to take place [Cro05]. Thus, also taking the involved cultural processes into account, one can state that the two generic functionalities of music are *"in underwriting social flexibility and in facilitating intellectual flexibility"* (ibid). Music is frequently held to be locatable in the affective domain. In essence, it has been shown to be used to modulate emotions for and between individuals (e.g. lullabies) and group contexts across cultures. This regulating function of music, can be seen as related to joint intentionally in interacting within a mutual and synchronized framework as the proximal goal of musical interaction is also sharing convergent psychological states. Therefore, even with respect to simple musical engagement (proto-musical or ”musicking”), positive affect is promoted and *"the co-regulation of affect and motivational states, and the maturation, development, and repair of intersubjectivity are seen to be inextricably linked"* [Bis09]. Music has been shown to be meaning bearing, *"exhibiting degrees of intentionality or aboutness"* (ibid). Conceiving music because of this quality has been shown to take place across cultures. However, the interpretation varies between individuals based on

own experience and conditioning, cultural learning and interaction as well as cultural contexts. Although it follows that in a collective musical act, the individual musical experience differs between participants, *"[m]usic can be construed as guaranteeing the success of social interaction by conditions for the minimisation of conflict through its semantic openness"* [Cro05]. Exactly this openness and the subjectivity in that meanings are individually associable are therefore fundamental to joint intentionality. Encouraging the general topic of this thesis, to support collaborative music creation, Cross concludes:

> *" Music's capacity to guarantee the success of social interaction makes it an excellently adapted framework for interaction in situations that are on the edge, situations where outcomes are neither clear beforehand nor retrospectively (easily) definable [...] . But it also means that music is able to act as a medium within which a capacity for flexible social interaction may be rehearsed and perhaps formed "* [Cro05]

3.2.2 Structure

It has been shown empirically, that *"only global features such as intensity -- volume -- tempo, tessitura and timbre reliably elicit similar types of emotion states"* (ibid). Not all of these parameters are culturally stable, for example timbre (cp. section 7.3). Inferring rhythmic structure and thus *"organizing our experience of the ongoing glow of music around that pulse"* (ibid) is one of the few musical features that are stable across cultures. In this regard it is frequently seen as enabling a joint sense of shared action because of these commonly experienced temporal regularities. For example, the participatory context of music across cultures *"involving overt and interactive engagement in musical activities"* can be related to this quality. In particular, rhythm facilitates entrainment in music participation which *"involves the coordination in time of one participant's musical behaviours with those of another"* (ibid).

3.3 Motivation in Music

"[M]otivation is highly valued because of its consequences: Motivation produces" [RD00b]. Motivation energizes and directs behaviors, thus it equally represents the basis to explain the behaviour in musical collaboration. One valid viewpoint is, of course, that the social situation alone motivates collaborators to engage musically. In this way, the activity has become primarily a social event in which the technology serves as catalyst for social interaction, rather than as the focus of the musical experience [BF03] [BB07, p10-25]. However, the focus are personal motivators and the social context which acts as an enabler or catalyst. As will be shown, the motivation to engage with music is highly subjective but can still lead to a classification of user groups that are representative within the scope of this thesis.

3.3.1 Music Participation: Types

To obtain an overview of typical users and their characteristics it is helpful to analyse the types of human activity and motivations that precedes music participation. In [Gat91], Gates characterizes six types of music participants and their corresponding motivational benefits that fall roughly in the behavioural classes play, serious leisure and work/survival:
The dabbler is curious enough about an activity so that he gets involved as a participant. Once this curiosity has been satisfied he will become a non-participant. Recreationists look for entertainment benefits and will sustain

costs over benefits as long as the music activity is entertaining or provides a source of curiosity. Amateurs and hobbyists see music participation as serious leisure. They identify with the activity and will overcome difficulties even for a long time, thus, sustain costs over psychological benefits. Although the benefits are not easily catalogued, the result is that this group is serious about developing music related skills and knowledge. Apprentices and professionals will engage in music participation for as long as it is economically feasible.

· ◇ ·

While feedback in social contexts is an important motivational factor for all groups, it is useful in this regard to further differentiate between various motivational conditions. Stebbins' P-A-P system [Ste82; Gat91] describes a social system that exists between professional, amateur and public, where members of the system share a set of values that are agreed-upon *"that is used to regulate behavior and decide issues of quality, including settling questions of stratification"* [Gat91]. Furthermore, certain individuals are *"either or both the "definers" and "modelers" of high quality for other"* (ibid). In this system, economic values motivate some of the important behaviours and as such, professionals, apprentices and amateurs are sustained and reinforced in the P-A-P system. As stated above, others are motivated idiosyncratically or have more personal values. Therefore, the difference between the two groups, amateurs and hobbyists, is that *"hobbyists do not adopt the values of professionals that could result in economic benefits"* [Gat91]. However, regarding the social aspects of music participation, the setting and social dynamics determine the cost benefit category rather than the need to maintain a particular persona, role or social position to music participation. There are additional musical and non-musical factors and values that are attached to the experience in music participation, and therefore are influential to the type of music participation, such as genre (content), social and political values (context). Gates subsumes the potencies of these musical and non-musical value hierarchies that affect musical participation as *affective potencies* and further characterizes the aforementioned groups accordingly. He also points out that affective potencies may change with the focus of the subject's attention. The group of professionals, apprentices and amateurs exhibits high affective potency for both musical and non-musical values. For hobbyists, on the other hand, non-musical values have low or neutral potency levels as opposed to the musical ones. Compared to those of the hobbyist, the affective potencies for the recreationist are reciprocal. Finally, for the dabbler the potencies for both value classes are low.

· ◇ ·

Of course, the given characterizations reflect only one viewpoint, there are also alternatives, such as the various cognitive styles of composers presented in [Eag+08]. However, I will focus on the presented characterizations since there is more overlap with existing literature.

3.3.2 Intrinsic and Extrinsic Motivation

Apart from concrete manifestations of affective potencies, motivation can be also characterized by intrinsicality or extrinsicality. In [RD00b] intrinsic motivation is defined as *"the prototypic manifestation of the human tendency toward learning and creativity [...] seek out novelty and challenges [...] extend and exercise ones capacities [...] to learn [...] of intrinsic motivation describes this natural inclination toward assimilation, mastery, spontaneous interest, and exploration that is so essential to cognitive and social development and that represents a principal source of*

enjoyment and vitality throughout life". Extrinsic motivation, on the other hand, refers to attaining a separable outcome from performing an activity, such as a reward or avoiding punishment (ibid). In the context of music making, extrinsic motivation plays only a minor role (e.g. seeking of fame and survival) [Bis09] and thus will not be considered in further discussions. Music making is an activity of intrinsic interest as it has appeal for novelty, challenge and aesthetic value [RD00b]. Only for this class of activities intrinsic motivation can take place. Intrinsic motivation depends on supportive conditions to flourish, otherwise it is disrupted. These can be distinguished as those that enhance feelings of competence and those that foster social relatedness. The latter involves a relational social base even when behaviours are performed in isolation. The former relies on positive social-contextural events that act as promoters for feedback (feedback and communication) and freedom from demeaning evaluations. Another important factor is the feeling of autonomy such that behaviour is experienced as self-determined. This requires immediate contextural support for autonomy or competence or the *perception* thereof. In the scope of this thesis, it means that the result of actions is visible or suggestive in music performance although these may be (partially) controlled by others or a mediator. Additionally, choice and opportunities for self direction enhance this feeling of autonomy and is also related to "The Pathway to Expert Performance" in section 1.2 for motivation in approaching a musical instrument. Thus, environments that allow autonomy foster intrinsic motivation. Regarding collaboration, social contexts that are positive and without conflict can *"[...] catalyze both within- and between-person differences in motivation and personal growth, resulting in people being more self-motivated [...]"* [RD00b]. The three psychological needs - competence, autonomy and relatedness - are thus beneficial to (intrinsic) motivation, however, if thwarted, this can lead to a diminishment, both in motivation and well-being (cp. section 3.5). Regarding the types in music participation, one can see that the group of hobbyists and recreationists or dabblers is mainly intrinsically motivated whereas in the group of professionals extrinsic motivation may be dominant.

3.4 Embodied Practice & Communication

Sensori-motor experience and cognition are inextricably linked to both, Human Computer Interaction (HCI) and tangible interfaces [Hor05], and music participation [Lem10; Lem+07]. This is addressed in the following topic areas (paraphrased from the summary in [Pri+09]):

The relationship between action and perception: This is based on the theory of affordance in which the perceived properties of objects or the environment afford or guide action. Furthermore, engagement with the world forms the basis for our understanding of it.

Meaning-making: Conceptual systems are grounded in bodily experience and as such meaning-making is in many cases body-based.

Situated cognition: This considers the physical and social context of a task which shapes action and cognition. It includes the familiarity with the task *"where cognition is conscious (and therefore time intensive) in unfamiliar situations or tasks, but becomes 'unconscious' in familiar situations"*.

External support: The environment acts as an external support because

space and gesture support memory and communication.

Since these forms of embodiment are highly relevant to co-located collaboration facilitated by tangible or multi-touch interfaces, especially in the context of music participation, I will expand upon the themes of embodied cognition and interaction to further frame the implicated benefits. However, with respect to music participation only instrument interaction will be regarded for the sake of brevity.

· ◇ ·

So far, the discussion of creativity and engagement has focussed implicitly on distributed approaches to cognition, where a socio-cultural process is part of a socio-technical environment [Lin15]. As pointed out, the physicality of musical interaction also demands the discussion of an additional viewpoint, which is embodied cognition. From the stance of a radical embodiment position, observing bodily actions and movements offers an approach for studying cognitive processes without the *"need to infer the content of human's perceptions, intentions, [...]"* [Lin15]. It thus challenges the concept of Cartesian cognition. Lindblom highlights several insights into human meaning-making based on embodied practice in recent research, especially neuroscience (ibid). Since bodily states mirror cognitive states, meaning making emerges at an unconscious and perceptual level. Even more, it is suggested that sensorimotor processes are cognitive in themselves, because performing and perceiving actions are inextricably linked to each other (outwardly and inwardly). Thus, the purpose and intention behind movement is informed. Furthermore, facial communication creates a shared space in which *"cognitive and bodily states of the interacting partners are reflected in both in themselves and in-between them"* (ibid). Gestures by themselves allow the expression of concepts that are hard to articulate in speech and are *"able to present different pieces of information simultaneously, which in speech would need to be expressed sequentially"* (ibid).

· ◇ ·

In consequence, these forms of embodiment facilitate several highly desirable practices, namely the communication of intent, the extended expression which is particularly useful for music, where intentions can be difficult to articulate with speech, and, importantly, knowledge construction and learning strategies based on the mirroring property. Co-located collaboration allows to exploit these mechanisms naturally.

· ◇ ·

Besides embodied cognition to frame interaction, another approach is to examine the phenomenological aspects of embodiment, the embodiment of interaction [Dou04]. With embodied cognition, these are the two main strands that are concerned with the role of embodiment in ("third wave") HCI [XJP14]. Similar observations to the last paragraph have been made in the music related sciences in search of answers how essential aspects of music communication, the sharing of intentions and expressions are mediated. The physical layer of this mediation consists of the human body, the musical instrument as a form of artificial prosthesis and different mediation channels which can be sonic, visual and haptic [Lem+07]. Ideally, it would disappear from the cognition since the physicality of interaction helps to reduce cognitive load [DMS07] because an already established set of the player's skills is leveraged for interaction [Ovi06; Dou04]. As above, this admits that the mind is not the sole entity that springs human action [Ing01] and thus, such interaction is commonly called "intuitive". Therefore, the integration of the instrumental movements into the bodily coordinate

system of a player leads to functional and relational transparency of the musical instrument so that it disappears from the consciousness [Nij09; Lem10]. This embodiment of interaction is therefore an embodiment of musical knowledge, skills and experience.

· ◇ ·

From the social stance, embodied interaction can be seen as embodiment of social interaction [BF03] as "it engages people in "muscular bonding" at the physical level, which, at the intentional level, results in experiences of collaboration, bonding and coalition signaling" [Lem+09] and therefore it implies how performance cues are perceived, selected and processed in order to guide executive, interpretative or expressive actions (ibid).

· ◇ ·

It should be stressed that recent research in organizational behaviour suggests a linkage between creativity and embodiment exactly because of the characteristic that creativity results from interaction with the environment and other agents. While their findings are not exactly related to computer interaction, Leung *et al.* reveal in a series of studies that *"physically and psychologically embodying creative metaphors promote fluency, flexibility, and/or originality in problem-solving [...] we provide the first evidence that embodiment can also activate cognitive processes conducive for generating previously unknown ideas and connections"* [Leu+12]. To summarize, certain actions initiate or promote creative (mental) processes. More related to the field of HCI is the contribution by Klemmer *et al.* [KHT06], suggesting that less constrained interaction facilitates thinking and communication, thus fostering creativity. Another important aspect is that embodiment significantly contributes to Flow (see section 3.5) since it facilitates low-level motor learning [Mag06; NB11] and because the multisensory characteristics of music interaction permit immediate feedback cycles - sensory feedback is simultaneous with the music experience - which are necessary for Flow to take place [Cus12]. To summarize, embodiment in a musical activity is not only a means for communication (the more common HCI perspective) but also an requirement for musical creativity and collaboration.

3.5 Flow

The strong affective involvement with an activity is at the heart of Csikszentmihalyi's theory of Flow [Csi90]. The experience of Flow can be defined as "a holistic sensation that people feel when they act with total involvement" (ibid). The subjective experience of Flow is the merging of activity and awareness, complete concentration on the task at hand, sense of potential control, loss of self-consciousness and an altered sense of time. To reiterate points raised regarding engagement and motivation, creativity necessitates a strong affective involvement such as Flow [Hew05]. Certain conditions are responsible for the occurrence of Flow: a challenge between the musician's skills and the challenges posed by the performance, clear goals every step of the way and immediate and unambiguous feedback [Nij09]. These requirements are repeatedly echoed from the various viewpoints elaborated in this chapter (e.g. sections 4.4, 3.3, 3.4). Flow is important as it stimulates an implicit learning process, this creates a bidirectional link to the embodiment of music apart from the previously mentioned feedback in embodied interaction. Furthermore it stimulates enjoyment of the activity and contributes to the feeling of personal engagement leading to the empathical involvement with the music (ibid). The latter is a result of

social presence, which in itself is part of the Flow process, it is "the feeling that one has some level of access or insight into the other's intentional, cognitive, or affective states" [BN01] (cited from [Nij09]). If this affective and embodied state is experienced collectively it is commonly called as being "in the groove", referring to peak moments in a performance that are collectively *felt* to work [Swi11].

· ◇ ·

The collective experience of Flow is also referred to as Group Flow [Saw06], which has been shown to happen in CSCM environments [SR04; Swi13], Jazz improvisation groups (ibid), string ensembles as well as improvisation theatre groups [Saw06]. In the empirical study by MacDonald [Mac06] which focused of collaborative music composition, it was furthermore revealed that groups that reported the experience of Group Flow correlated with more valuable compositions in comparison to groups that did not experience it. This insight is supported in other sources, such as:

> " *Group Flow can inspire musicians to play things that they would not have been able to play alone, or that they would not have thought of without the inspiration of the group [...]* " [Saw06, p. 158]

In this way Group Flow is a motivator and means for the group to innovate in a creative task. It is not primarily a function of the skills and inclinations of the individual members, but the particular way contributions to the musical whole are (empathically) interpreted and, in the course of performance/improvisation, are built upon. Similar to the argumentation in the last section, Group Flow requires a fertile environment that allows freedom to experiment:

> " *People love to be in group Flow (Eisenberg, 1990). As a result, the companies that create an improvisational work environment will be better able to attract and retain the most creative professionals. [...] In business, the bottom line is what the whole organization creates collectively. There might be a lot of creative employees, but if they work in a stifling organizational structure they won't innovate.* " [Saw11, p. 248]

The support of the implicit learning process furthermore helps mediating the interaction with the shared CSCM environment and finally the social interaction with peer members. Thus, Group Flow creates a tri-directional link between individual, group and the music created.

· ◇ ·

Since the embodiment of action (see 3.4) is strongly linked to Flow, great care has to be taken that the interaction with the CSCM environment is designed to allow for the inherent detachment of consciousness from the physicality of interaction, i.e. the reduction of cognitive load for interaction. Gelineck's design proposals in section 3.1 can also be approached from the argumentation of experiencing Flow when approaching a musical instrument. This argumentation is similarly two-fold [FC05]: on one side is the endogenous motivation, for which the initiator is the will to realize an artistic idea. On the other is the exogenous motivation, the "play instinct" to explore the music instrument. While the first one is a directed creative undertaking which requires knowledge and competence to achieve the goal in a constructing manner, the second one is undirected and does not principally presume any musical skills. As can be seen from the study by Gelineck *et al.*, both approaches to music and its experience are valid in that they lead to musically "meaningful" results. Yet, they also presume different skill levels from the engaging person. This reflects on how experts and novices can differ in approaching a CSCM environment and thus, it has

to facilitate both approaches in order to support heterogeneous groups. Naturally there is a spectrum with respect to skills between novices and experts, therefore this also leads to the argumentation that such an environment has to allow ”mastery”, the incremental learning of the tool in order to become an expert (cp. “The Pathway to Expert Performance”, section 1.2).

Chapter 4

Collaboration: Music and Human Computer Interaction

"As complicated as computers and other products of our technology may be, it is easier to understand the machine side of the human-machine interface than to come to grips with the far more complex and variable human side."

Raskin, *The Humane Interface*, [Ras00, p. 9]

Chapter Abstract

The following analyses practices for Creativity Support Systems that are embedded into a social context and practices in collaborative music making. It is argued that Shared Display Environments, specifically a tabletop interface, allows embedding into this social context. The main perspective is on Human-Computer Interaction (HCI) and the implications on it from musical practice and collaborative music improvisation.

Creativity Support Systems aim to enhance creative outcomes for individual and group-level problem solving [Fri13]. Within the field of Computer Supported Cooperative Work, a CSCM is then a Creativity Support System focussed on group support with the essential constraint, that the collaborative process does not benefit a problem solving task but creates additional value by itself as hedonistic environment [Che14]. Because of this, traditionally artefact oriented process models in Creativity Support Systems are difficult to integrate. This problem is aggravated because the motivations for engagement with the collaborative process are personal.

4.1 Collaboration and the Tabletop Environment

Frieß refers to Creativity Support Systems that are *"tailored to explicitly support the properties of co-located creative collaboration situations"* [Fri13, p. 66] as "Situative Creativity Support". He emphasizes that such systems exploit the social situation in which the task takes place and thus these systems

have to be embedded into it. In an extensive literature review, he points out several factors that are mandatory or beneficial for creativity and collaboration. Furthermore, he shows and devises corresponding design traits and concludes that these can be utilized effectively for applications that make use of a tabletop in shape of a Single Display Groupware that falls into the category of same place / same time collaboration (cp. [Joh+91]). Two insights are central for this argumentation: first, "information sharing, knowledge of group and individual activity, and coordination are central to successful collaboration" [DB92, p. 3] and second, studies *"confirm that the nonverbal behavior plays a major role in shaping the perception of social situations"* [VPB09][p. 3] (both references are cited from [Fri+12]). In tabletop environments, cues about non-verbal behaviour manifest in the interaction between people, with the application and within the physical space around the tabletop display [Fri+12]. In his dissertation [Fri13], Frieß illustrates how the tabletop environment facilitates situative behaviour and how it in relates to facets of teamwork (-quality, cp. [HG01]):

Positioning: Collaboration in a tabletop environment also includes the physical space around it. Compared to Single Display Groupwares that use wall-projection or similar presentation based interfaces, the tabletop affords a more fine grained differentiation of user positions and thus imposes less limitation on the group interaction and creative productivity [Fri13, p. 68-72]. This includes the position people naturally take depending on the social interactions necessary to perform a specific task (e.g. conversation, cooperation, co-action, competition; cp. [Rus67]) and hierarchy / dominance (e.g. dominant people tend to seat themselves on the shorter sides of a rectangular table [LS67]). Another facet of situative behaviour that can be respected is proxemics, which correlates physical distance to social distance (cp. [Hal63]). The implications are summarized in [Fri+12] as *"[the] distances are mainly dependent on the social relationships people have with respect to each other, psychological characteristics also play an important role: extrovert people, for example, tend to prefer an arrangement that minimizes interpersonal distances, while introvert ones do the opposite"*.

Body orientation: Another aspect of non-verbal communication is the orientation (e.g. shoulder line) that people have towards each other. "For example, facing in the opposite direction with respect to others is a clear sign of non-inclusion. [...] Face-to-face interactions are in general more active and engaging [...], while people sitting parallel to each other tend to be either buddies or less mutually interested." [VPB09][p. 24] (cited from [Fri+12]). Thus body orientation is also part of a signaling mechanism that informs cooperativeness.

Visibility: Wall-projection based Groupwares face the issues that, on the one hand, visibility and physical access for individuals may be blocked due to physical constraints of the interface by their peers and, on the other hand, that looking at the projection instead of peers hinders face-to-face communication [Fri13, p. 68-72]. Thus, production blocking impedes true parallel work and the physical position may be responsible for a fragmented visibility between participants that interferes with communication and visibility of action. In contrast, since the tabletop is at the centre of the (social) interaction, it can help avoiding these issues.

Coupling and territoriality:[1] With respect to the different forms or schemas of collaboration mediated by the computer on the tabletop interfaces, vari-

[1] The following paragraph is taken from a section in [Fri+11] which I authored

ous forms of coupling can be identified [Tan+06]. Coupling is a dynamic process within collaboration on such interfaces resulting in a steady flux of the group configuration. Supporting such situative group transitions is the key to permitting the dynamics of collaboration to happen, since they reflect a natural style of communication and interaction between collaborators. In a similar way, the spatial usage of the tabletop environment for personal and group tasks is comparable to the human division of space in the physical reality, helping to coordinate actions with artefacts on the interactive table or collaborators [SC06; SGM03]. Such territories can be either personal, group related or shared among all collaborators. Coupling and territoriality complement each other: coupling demands that the application can be interacted with concurrently in a flexible way by several persons from arbitrary positions around the table, thereby realizing dynamic group structure transitions. Territoriality requires that the interaction in one territory does not disrupt or interfere with interaction in others.

Orientation and Sharing of Artefacts: Collaborative activities can also be related to adjusting to the orientation of artefacts on the tabletop interface [SGM03]. Such actions are performed for comprehension (e.g. to improve readability for oneself or others) and communication (e.g. to signal the direction of communication) [Fri+12] (cp. [Kru+03]). Furthermore, they mediate coordination by communicating ownership (and the transfer thereof, when sharing artefacts) and the aforementioned facets of territoriality.

· ◇ ·

So far, we have seen what forms of non-verbal behaviour the tabletop environment facilitates and how this directly reflects on situative behavior. However, there are additional relationships that are important for collaboration. In the classical HCI sense, group awareness is dependent on the visibility of action [DB92]. It *"describes the ability of individuals in the group to share and gain knowledge about the process of activities performed by other participants [...] a good sense of group awareness allows collaborators to coordinate activities and to access shared resources by simplifying communication"* [Fri+11]. Improving group awareness (visibility of interaction) can therefore result in *"free[ing] up mental resources for the creative task"* [Fri13, p. 64].

· ◇ ·

However, there is also another viewpoint to (group) awareness that emphasizes the social dimension: awareness of another person includes social presence. The definition for social presence was given in section 3.5 and related it to the process and experience of Flow. From the perspective of situative behavior, it *"enhance[s] closeness to and nonverbal interaction with another"* [AB99] (cited from [Ret03]) and reduces misunderstandings and conflicts [Fri13]. Social bonding and group cohesion (members of a group participate readily and stay together because of mutual interest) increase prosocial in-group behaviour, cooperation [KT10] and can lead to building mutual trust and thus increase the well-being of the group. Apparently, these socio-emotional processes depend on the support of social interaction which the co-located collaboration on a tabletop can provide [Fri13, p. 63]. However, they also depend on and influence communication, coordination and group awareness (ibid), which are also constrained or assisted by the interaction design of an application. With respect to teamwork quality [HG01], socio-emotional processes are reflected in mutual support (participants work together instead of competing) and effort (the group task's workload is shared and prioritized over other obligations) [Fri13, p. 63].

· ◇ ·

Another important aspect that guides the practical facilitation of collaboration in CSCW is to support group articulation [CJ95; DB92]. Group articulation refers to the ability to partition the group task into units that can be combined, substituted and reintegrated into the group task. In this way the partitioning of the group task allows to work in parallel on subtasks and to democratize the collaboration with respect to workload and involvement. Therefore, a prerequisite to meaningfully support group articulation is to facilitate the various aspects of territoriality. Similarly, free positioning around the shared interface is integral for coupling [Fri13, p. 73].

· ◇ ·

From a more general point of view, multi-touch displays provide the means for parallel input, such that in principle several users can control an application concurrently using the same input method [2]. The one-to-one mapping of digital and physical objects, spaces and interaction fosters immediate joint use of interaction. By that, *direct manipulation* alleviates the need for social protocols to coordinate the actions within the group [Mor04]. Moreover, tangible and touch interaction support embodied practices (cp. section 3.4) and thus further reduce barriers to computing [WW11, ch. 1]. Morris et al. explored the impact of shared, centralized controls versus individual controls with the tabletop application "Teamtag" [Mor+06]. Regarding the group performance and the users' personal preference, it was found out that the latter was clearly the preferred method of control. This insight is supported by the research of Scott et al. [SGM03], emphasizing the importance for parallel input and control of the application to support group awareness and group articulation. Centralized controls enforce active group awareness which is desired in certain group tasks [Hor05] at the expense of not allowing for simultaneous action and creating disruptions in the workflow of the collaborators: *"Having to alternate and sequentialize actions caused multiple breakdowns, even though participants were highly aware of each other. Alternating actions was felt to be demanding"* [Hor05]. Parallel interaction also implies a more democratic group interaction: non-verbal contribution is possible for all group members as the right to control artifacts is distributed. In this way, individual users are not prioritized and thresholds for collaborating are lowered as particularly self-assured individuals can not gain full control over the application or group task in favor of "shy" ones [ESH02]. Thus, this public interaction triggers only desired communication and negotiation[3].

[2] This paragraph is largely taken from a section in [Fri+11], a contribution which I co-authored.

· ◇ ·

To conclude, the tabletop interface can be seen favourable for supporting (creative) collaboration since it can be embedded into situative behaviours. Although the full spectrum of tangible interaction is not represented in multi-touch interfaces [TKI10], interaction with and around the tabletop with peers can still enable rich embodied practices compared to prevalent methods of Human Computer Interaction (e.g. Window Icon Mouse Pointer). Ma *et al.* advocate the use of tangible interfaces over purely multi-touch based ones for collaboration based on the results of a user study (data exploration at an exhibition) because they better afforded touching and (direct) manipulation [Ma+15] and thus resulted in a broader exploration of the data. However, no significant trends with respect to social interaction or improved understanding of the domain have been shown. The widespread adoption of affordable multi-touch technology rather than highly customized tangible interfaces [Xam+11] may also make the discussions in this chapter more applicable to other scenarios.

[3] There are also socio-psychological factors that revolve around the physical interaction such as accidentally touching the hands of other users while performing an action, which is generally regarded as unpleasant and frequently occurring with centralized controls [Mor+06]

4.2 Common Design Guidelines for Creativity Support Systems

Michael Resnick et al. [Res+05] proposed a set of design principles for an environment that is suited to foster creative behaviour. As we will see, many of the design themes that are posited here overlap with those of collaborative tabletop applications and music HCI. These design principles respect the viewpoints regarding prerequisites for the Flow experience and approaches to the CSCM laid out thus far (taken from [Klü+11]): It should support exploration and therefore allow for experimenting with many different alternatives (→ group articulation, coupling), as *"creative work means, that the final design is not necessarily known at the outset [...]"*. The necessity to support such parallel paths that are implicitly based on iterative processes is also expressed in other literature such as [Hew05] and [TM02]. Additionally, the inclusion of direct feedback mechanisms is emphasized to shorten iteration cycles [Lit+10]. In order to support explorative behaviour, tools in the environment should be intuitive and ideally be *"self-revealing, so that it is clear to users what can be done"*. It should provide a simple user experience without loosing the possibility to do complex work and offer *"a low threshold, high ceiling, and wide walls"*: a low threshold allows novices to get started by providing the means that gives users an *"immediate confidence that they can succeed"*. On the other hand it should also be able for experts to create *"sophisticated, complete solutions"* (→ Pathway to Expert Performance). This also reflects on the "primitive elements" that users are interacting with. The granularity and choice of those elements determines what can be explored and what remains hidden from the users' view. The term "wide walls" refers to the wideness of the exploration space, which is provided by the environment. There should be *"very general primitives that users must learn how to combine"*. This also relates to the guideline to *"support many paths and many styles"* which mirrors the various human styles of exploration and construction, the different ways of approaching and mastering a problem solving process in consonance with social creativity that thrives on cognitive diversity [Mor80]. Furthermore, embodied practices for collaboration in creative tasks are important as highlighted in section 3.4.

4.3 Musical collaboration

After having framed creative collaboration on a tabletop interface, we now turn to implications and patterns of musical collaboration. There is a significant overlap of the generic requirements for tabletop applications that have been presented thus far with those that are specific for group musical engagement. Xambó *et al.* [Xam+11], for example, propose candidate design patterns based on the viewpoint that musical interfaces have intrinsic constraints and affordances that determine the (collaborative) instrument's interaction possibilities (cp. [Mag06]): There is necessity for both, shared and personal spaces, since personal spaces allow participants independent control of their contributions which then can be incorporated into the shared composition (→ territoriality) and, additionally, for a *"division of tasks which can be executed in parallel towards a single shared goal"* (→ coupling, group articulation). In the studies by Fencott *et al.*, strong territorial behaviour was frequently identified [FB12; FB10] and it was shown that having both kinds of workspaces is favoured by participants. Similar observations have been made in [Lan+10]. Finally, Xambó *et al.* affirm that

multiple pathways for novices and experts should be afforded (→ Pathway to Expert Performance). Other design considerations include the gestural input such as discrete versus continuous controls, the multiplicity of gestures with respect to control granularity (e.g. multi- versus single-touch gestures) or the fact that the missing sense of touch requires more audiovisual feedback (cp. section 4.5) [Lan+10]. Laney *et al.* also point out that the responsiveness of the application determines the perceived emotiveness (ibid) and that audiovisual feedback can enhance the perceived response of the system.

· ◇ ·

As stated before successful collaboration depends on group articulation and group awareness. In [BHL07] mutual engagement is seen as indicator and result of such successful collaboration. Mutual engagement can be seen in proximal interaction (collaborators will work close to each other), participants mirroring (re-producing) and transforming contributions (reproducing), and joint contributions. Here additional design traits that foster mutual engagement are elaborated such as mutual modifiability (being able to edit each other's contributions) because it *"implies an egalitarian approach to role assignment within the tool rather than explicitly enforcing role mechanisms - in such an approach participants co-ordinate their activity in a subtle and dynamic manner"*. Other design traits encourage the use of shared display environments, because they allow the use of shared and consistent representations, and facilitate to communicate in and around a shared product (annotation). Even more, in a study that measured affective and psychophysiological features (e.g. ECG, EEG) of collaborating drummers with respect to visibility, it was found out that the visibility of collaborators has effects on their physiology and performance [MGB14]. In this way it can be posited that certain factors of group awareness or mutual engagement are not limited to altering solely cognitive and affective states. With respect to modes of interaction, talking, gestural, auditory (listening carefully) and visual (looking at each other) modes have been shown to take place [Lan+10]. Occasionally leaders emerged, even for quasi-democratic environments.

· ◇ ·

Apart from the presented observations that have been made within the context of computer mediated collaboration, insights about the processes involved in musical collaboration (instrumental) can also be gained from analysing music improvisation using traditional instruments. Collective Free Improvisation (CFI) involves a mix of top-down and bottom-up processes because the music evolves through ad hoc cues and larger contexts of possible responses to these cues based on a shared understanding [CA14]. In its most "pure" form it means that participants have neither played together before nor prepared any material for the collaboration [Can13]. It is a good example of emergent structures, because the self-organization of collective signals is possible despite the lack of a priori structures. Thus, similar to the presented use-case, the performance is the product of the musical engagement of multiple agents. The success of a performance is therefore a coordination problem which tries to establish a shared representation that is constrained by satisfying individual aesthetic preferences and achieving and maintaining coherence for the piece (evaluating the overall musical result as satisfying) [CG12]. This dichotomy of maintaining personal excitement while ensuring continual development of the piece manifests itself in two entities that shape the performance: attractors which are collective sequences or fixed points of the musical stream

and transitions which are cues to detect a change of the sequence. The collective performance converges towards different type of attractors: erratic (unstable and uninteresting, phase of dis-coordination) as result of a conflict between different ideas; unstable but interesting, in which the sequence is a long transition either as aesthetic quality to create suspension or fragility or as accepted instability; stable but uninteresting as low-complexity region (”problem-finding” phase); stable and interesting which is a ”true” collective sequence and the paradigmatic case of coordination and convergence which results in maintaining excitement. Furthermore, with respect to transitions it has been stated that accidents in performance can trigger new variations of a sequence or act as a signal for change [Can13].

· ◇ ·

As the previous discussions implicated, communication involves on one hand the acoustic signal, musical or gestural content [CG12], the physical orientation [HLB05; Lan+10] or vocalizations and verbal communication [FB10]. On the other hand, it involves the intentional content. With respect to verbal communication, it has been shown that for computer aided group composition, dialogue is frequently spontaneous and transactive [HLB05]. Apart from the previously mentioned signals, physical orientation not only offers the opportunity to signal (dis)engagement but also mediates anticipatory information in order to manage the temporal structure of the piece (ibid). A great deal of cognitive resources is devoted to the encoding and decoding of such activities because the competency of musical agents in an improvised interaction depends on their ability to convey semantic content of their own musical ideas and to make accurate judgements about the semantic content of their peers' signals [CG12]. As such, the semantic content deduced from signals can be: formal, in order to communicate the objective or musical evolution; interactive to signal a musician to enter a certain form of interaction; evaluative. In this way this process also involves strategic reasoning as improvisors make musical decisions since they anticipate some kind of response or reaction from their peers.

· ◇ ·

Shared representations involve one's own representations of a given situation, the own representation of another musician (as argued above) and his/hers intention and objectives as well as the more global team preferences. Such representations can be contrastive, transparent or formal. While the former two are situative, the latter is also dependent on the context of the piece. I argue that the focus on electronic dance music in this thesis is an appropriate formal choice because it is pervasive (as argued before) and thus the formal, or formalistic aspects of such music are implicitly understood or ”felt”. This framing can be seen as a strategy to create a shared representation based on the ”musical logic” (of that style). In general, apart from a formal musical logic, there are strategies to establish a shared representation. These are based on cognitive factors or the discourse between the individual and the collective result according to the individual's own musical logic. The strategies involve:

Stabilization by repeating the pattern in order to propose an attractor for others to join in or by operating as a mediator between two heterogeneous ideas presented by other improvisors in order to synthesize an attractor or converge towards a new one.

”Wait and see” strategy in transitional phases maintaining a low information signal while the group finds a new attractor.

”Play along” strategy in a way to acknowledge the state of affairs and do what is implicitly expected to play, thus cooperate.

”Densification” strategy in which complexity is deliberately created to provoke a transition.

· ◇ ·

With respect to mental models, in [CA14] it was shown empirically that the *"similarity in participants' mental models predicted the degree of musical familiarity with better-than-random accuracy"*. Thus sympathy is an important factor of a successful musical collaboration. Similar results have been shown in [MMM02] where the musical collaboration between children differed qualitatively (more transactive communication) depending on ties of friendship. However, other observations also showed that participants did successfully collaborate while at times holding entirely different representations of the music they were creating [FB10] - as such this issue is contrasted in literature [CA14].

4.4 Music and Human Computer Interaction

Much of the argumentation presented analyses the reasons, principles and mechanics of musical and creative engagement. It is here, were musical HCI and mainstream HCI, which tends to focus on usability, divert [Hol+13]. As has been illustrated the patterns of musical activity and motivation are manifold. They may seem unremarkable from the viewpoint of musical interaction but these have not been a central concern in mainstream HCI, especially since they encompass distinctive goals of interaction (e.g. fun, utility, positive user experience). Music systems occupy a spectrum of autonomy, the player paradigm (the system is an agent) and the instrument paradigm [McD+13]. Even for the latter, instruments are not utilitarian as they are not played to accomplish a goal but, as pointed out before, playing itself is often the primary goal (ibid). Thus, when regarding HCI, efficiency is less important than engagement. Especially long-term engagement is of interest because it reflects common practice: instrument performance, mixing and production amongst others are all activities that take years to master and in which sessions can last several hours [Hol+13]. Therefore, the benefit of investigating interfaces that foster long-term engagement is that these are more likely to become popular interfaces and thus have special value for research.

· ◇ ·

In [Wal+13], Wallis *et al.* use the previously mentioned self-determination theory (SDT) in amateur engagement and its relatedness to the three intrinsic motivators - mastery, autonomy and purpose - to examine several properties that are transferable to the field of HCI. The properties outlined are insofar interesting as they argue for heuristics for the design of user interfaces that increase long-term engagement with musical applications. Several streams of argumentation that are presented in this chapter ultimately lead to these heuristics. The following are the most relevant ones for the scope of this thesis:

Incrementality describes the progression in difficulty from beginner to expert - it is the gradualness of the learning curve in ”The Pathway to Expert Performance”. It impacts persistence within an activity and can help

maximize Flow state in users if the challenges in performing the activity grow appropriately with the acquired skills. This follows from the previously presented Flow theory; similarly, in the extremes it either results in frustration if challenges increase too quickly for the user or in boredom, if these increase too slowly. Video games, are for example, applications that frequently manifest incrementality with levels and scoring. With respect to music, such challenges are in many cases self-managed (e.g. classical instrumentalists). Approaches to manifest the gradual increase of difficulty are, for example, layered affordances, where a hypothetical instrument has sensible restrictions on the degrees of freedoms (to make errors) which are gradually lifted by the player or, adaptive affordances, where the machine itself controls such mechanisms based on heuristic learning methods [McD+13].

Complexity or open-endedness is related to the former as it comprises the potency of an instrument to offer limitless incrementality or to define a ceiling of expertise. For example, a musician regardless of skill will always find some facet of challenge (improvement) in playing a non-trivial instrument. To pick up a former argument: it can be inferred that *"[i]nstruments that do not afford much complexity are [...] often perceived as toys and are usually not the source of long-term engagement"* [Wal+13]. It follows that interactive systems should offer the potential for complex interactions but on the other hand not force it upon users (especially beginners) - thus a design should be *appropriately complex* for the musical task. In this vein, Linson urges that *"instrument designers should not take a limited view of human cognitive capabilities with respect to coping with difficulty and complexity"* since acclimation to new modes of interaction is a feature of human cognitive flexibility [Lin11].

Immediacy describes whether there are obstacles in participating in the activity, it impacts the number of people initiating and persisting in it. This includes many factors such as ubiquitousness, portability and, most importantly for the presented use-case, latency.

Whether interaction seems to be driven by the user interface or the user is denoted as operational freedom. The lack of free operation can result in boredom as this prohibits innovative interactions to produce complex outcomes.

As presented throughout chapter 3, cooperation fosters motivation and a community of sharing among other things. This is important to long-term engagement because of the community building factors laid out previously and, since teaching and inspiring one another speeds up the attainment of skill and attracts new users.

The aspect of incrementality leads to a counter-intuitive observation that contrasts with a common idiom in HCI, namely that ease of use in musical interaction is not always beneficial. Apart from the implications of maximizing Flow states based on appropriate levels of difficulty in interaction, there is also the component of spontaneous creativity that arises from e.g. errors in the performance, leading to new interpretations or goals [McD+13]. This is related to the characterization of the "explorer" in section 3.1.1 where inspiration is drawn from unpredictable behaviour of tools. Here, a possibly pragmatic stage may be disrupted in order to explore new creative outcomes. There are various dimensions of difficulty that are applicable to music interaction. Some of them are avoidable while others

are not, they are presented in [McD+13]. Difficulties such as physical difficulty or difficulty of dexterity and coordination are non-existent or seldom for interactive music systems (cp. chapter 2). However issues related to non-linearities, discontinuities and interaction in control can be problematic, they are especially common in synthesizer parameters. Difficulties between imagination and realization, polyphony and multiple streams (related to cognitive load) and conceptual difficulties can be mediated by the interface. On a more abstract level, especially peripheral and technical tasks benefit from simplification as these are not essential to the musical undertaking. What is exactly peripheral depends on the individual [4]: while some may see burden in "tinkering" with synthesis parameters in order to set up timbres that fit to a musical theme, others may be more interested in the sound design process itself rather than a harmonic structure. Contrary to this, difficulty is expedient in the context of open-endedness because it allows to accomplish goals of appropriate difficulty (w.r.t. incrementality), and experience unlimited growth in skills and thus it is fundamental part of the musician's self-narrative and self-image.

[4] Of course, there are tasks that are agreeably overtly technical, even from the standpoint of the bedroom producer, for example, mixing and mastering a song. This is apparent in the demand for external services performing these tasks. For example, services that provide hired mixing and mastering engineers or recent online services that offer automatic / algorithmic mixing [Mix14b] and mastering [Mix14a].

4.5 Feedback

We have seen multiple pointers where feedback from the system is integral for interaction. Feedback is first and foremost necessary for the continuous evaluation of the iterative creation processes [TM02]. As mentioned before, a tight loop of action and evaluation is also necessary for the feeling of control, embodied practices and intrinsic reward, thus the Flow experience. This implies that feedback is rapid, constant and consistent [BTT05, p. 90]. Electronic music allows for short delays in this loop, since in most cases the sonic output can be rendered in real-time. However, for a collaborative use-case this feedback loop is not enough since sound is pervasive with the implication that *"participants may be unable to gesture or orient around a sound, or use spatial ordering strategies to manage tasks and conversation"* [FB10]. Therefore feedback has to be audiovisual, because an artefact representation can facilitate communication around and interaction towards the sound object [FB10; CJ06], and because it allows the machine to communicate the states and behaviours of musical processes that are otherwise hidden due to the internal complexity of relations that map user input to audio output [Jor03]. The latter is also related to physical instruments or tangible interfaces since they offer haptic feedback which multi-touch interfaces do not (ibid), therefore mechanisms need to be provided that inform about the state of the interaction [BTT05, p. 43]. One problem, however, is that visualizations can distort acoustic stimuli (personal communication with Kevin Schlei & Gerriet K. Sharma, Sep. 2014) or distract from their perception [Cro08]. In this regard, visual feedback that does not inform about the state of artefacts and musical processes in the software, or which does not foster a sense of control may be regarded as superfluous.

4.6 Evaluation Methodologies

In section 1.3.1, it was argued that music itself lacks formalization as a foundational role. Even for music genres that are seen as more "formulaic" (colloquially), measuring the quality of works is somewhat an ill-fated undertaking because it depends effectively on subjective traits (coined through dynamic socio-cultural contexts). As laid out before, experience design is

human centric, we are interested in fostering experiential use and ludic engagement. The success of an experience is determined by the process of playing and not the outcome of it [MAI06]. This includes that such evaluations are not task based. Stowell *et al.* furthermore point out that task-based evaluations tend to be highly reduced experimental setups which are too contrived to provide for creative interactions between human and computer, this leads to concerns about the validity of these tests [SM13]. In this way, controlled experimental comparisons, may have risk of compromising the musical situation. Open-task evaluations also contrast with practices in CSCW where studies *"[...] typically investigate specific aspect of group activity by presenting work teams with simplified tasks [...]"* [FB13]. As such, an evaluation most likely has to focus on collaborative instrument driven interaction [Xam+13]. Given the argumentation in the previous chapter, the ultimate goal is therefore experiencing group Flow [Swi13]. Swift proposes four approaches to evaluate whether participants are in a Flow state: expert judgements (e.g. based on Interaction Criticism [Bar09]), qualitative data analysis (e.g. analysis of interviews using Discourse Analysis [SPB08]), unsupervised learning and rich data collection (e.g. based on biometrics). With respect to expert judgements, Xambó *et al.* see video analysis as an appropriate tool because it allows to create a rich picture of the user interaction [Xam+13]. However, the process of video annotation for expert evaluation is lengthy and must be rigorous in order to avoid biasing the results [MAI06]. Typically, the analysis time to data sequence time ratio ranges from $5:1$ to $100:1$ (ibid), which makes this method difficult to implement, given that user studies require evaluation by several groups in an appropriate time frame of 20 minutes to one hour. Alternatively, observation and note-taking during the collaboration is also a valid tool to create a representation of the sequence of activities. Observational analysis including concurrent self-reporting (e.g. think-aloud evaluation, cp. [WM91]) allow qualitative and objective assessment [MAI06] while rich data collection based on biometrics can be used for quantitative and objective assessment. However, gathering user-generated data concurrently with the activities is generally impractical because they may distract from the music-making process in the case of self-report techniques or distort measurements of biometric sensors [SM13]. Hence, in most cases retrospective protocols such as interviews, for qualitative and subjective assessment, or questionnaires, for quantitative and subjective assessment, are applied. The added delay in retrospective statements may be problematic since access to former affective and cognitive states may not be available anymore. Subjective reporting is generalizable and amendable to rapid statistical analysis, and furthermore allows to develop an understanding of the attitudes of users [MAI06]. The drawback is that they *"are not conducive to finding complex patterns, and subject responses may not correspond to the actual experiences"*, thus they may not be applicable to understand behaviours (ibid). Other issues are that subjective techniques have a low evaluative bandwidth (data is only generated if a question is asked) and that the novelty of the environment may skew results. Fencott and Bryan-Kinns argue on the other hand that novelty also introduces control over the influence of previous experience, since it is ensured that participants use the software for the first time [FB13].

· ◇ ·

Another approach to gather quantitative and qualitative data is to perform logging of the interaction within the application itself. For example, in [Fri+12], we tracked the modification of artefacts specific to users in order to gain insight into how co-editing and contributions are reflected on

aspects of territoriality. Similar to this, Fencott and Bryan-Kinns highlight that logging of e.g. the amount of musical contributions and co-editing can reveal behavioural characteristics in the collaboration [FB13]. In more diversified environments, this may allow to distinguish between the modes of spontaneous interaction [Joh11], instrumental (detailed control), ornamental (instrument is expected to augment interaction) or conversational (the instrument is seen as a partner who influences the development of play). More generally, the divergent approaches of explorative and goal-oriented interaction may be traceable and thus can be embedded into the analysis of self-reports.

· ◇ ·

Dahl [Dah14] makes two interesting remarks about evaluation of musical interfaces, apart from the difficulty of evaluating experience. First, as music is embedded into culture which constantly changes and so does the implicit valuation for experience: *"what was once interesting or provocative may become commonplace or passé"*. In this regard, one could conclude that the evaluation and its results are constrained by a cultural context. Second, *"[p]rototyping in NIME*[5] *requires higher fidelity than in most other HCI research"* because for music performance they are subject to latency requirements (e.g. $\leq$ 100 ms for visual feedback [BTT05], $\leq$ 20 ms for sonic feedback [Cha+04]) and have to be sufficiently complete. Therefore "Wizard of Oz" techniques (cp. [DJA93]) are only useful in evaluating early stages of prototyping.

[5] NIME: New Interfaces for Musical Expression as a research community

· ◇ ·

Apart from the explicit relation to a feeling and evaluation whether the representational system is accepted, experience design also implies a study of utility. For example, this can include or relate to, discoverability, consistency, feedback, reliability and creative exploration [Mai14] and expression. Some of these aspects have more explicit guidelines from experience design literature, such as consistency and feedback, or are embedded into design traits like those concerned with supporting creative collaboration (e.g. group awareness, visibility, territoriality). However, the real impact on the experience can only be guessed, because *"[w]hen research involves incorporating novel technologies into a play experience, there are no "experts""* [MAI06]. Evaluation is therefore still bound to take subjective user reports into account or perform comparative analyses. The latter was applied in chapters 13 and 9.

· ◇ ·

With respect to the composition of the group, Fencott and Bryan-Kinns make some compelling arguments why having groups of people who have not previously worked together is preferable, despite the fact that social and musical rapport are an important aspect of group interaction [FB13]. Since there is an established history of experience to draw upon, communication may be obtuse and difficult to interpret or study. Participants that have previously worked together may introduce a bias between groups, since not all groups have an equal level of experience of working together. This may also be true for intra-group relationships with respect to coupling and territoriality. Another issue is that shared *"musical knowledge and established working strategies may be stronger than or resilient to effects brought about by the experimental conditions under investigation"*. Because of this, preference is given to heterogeneous, non-familiar participants in study groups.

Chapter 5

Résumé: Framing the Rich Picture, Goals and Outlook

Chapter Abstract

The following will summarize the observations from the previous chapters and highlight key insights and design goals. This leads to framing the musical domain, target user group and further methodologies.

The literature review in the previous chapters has argued that music, musical engagement and -interaction is deeply embedded into socio-cultural and collaborative constructs. Additionally, it was demonstrated that practices in music making and collaboration are embodied - this includes meaning-making, coordination and communication. Social protocols have been shown which form the basis for coordination in generic and music related collaboration. These rely on verbal and non-verbal communication. In the context of Situative Creativity, it was argued that the tabletop environment as shared interface and physical space can facilitate these social protocols and certain aspects of embodied practices. This resulted in the conclusion, that the technical environment can be embedded into the social context and support creative collaboration. With respect to practices in recording and electronic music creation, we have seen that there is lack of tools that wholly embed collaborative practices (*e.g.* exchange of musical artefacts, iteration) which could otherwise exploit the beneficial effects of creativity including group Flow, social presence and shared meaning making.

5.1 Human-Computer Interaction

Interfaces that support the task of music creation deviate from conceptions in mainstream or ”first / second wave” HCI because the mediation of such software does not fall into a single *"user-tool-object triangle of mediation"* [BBP07; Mag06]. Instead, it oscillates between being a tool to change a domain object and equally being the domain object itself (domain obect duality). Furthermore, the interaction itself oscillates between action, which is consciously directed towards the object and therefore implies control (and relates to the experience of Flow), and operation, which is triggered in the

context of the action. In this regard, the more classic notion of direct manipulation [Shn83] which mostly focusses on transparent interaction alone is not suitable [BBP07].

· ◇ ·

Another viewpoint is that traditional HCI tends towards characterizing the relationship between a user and a tool as static. In this way it does not reflect the dynamism of cultural or aesthetic evaluations in human computer interaction [Bød06; Ber04] with possible implications on the results of an evaluation as pointed out before. Bertelsen *et al.* therefore introduce three alternative concepts to describe this form of mediation more appropriately: instrumentness, materiality and metonymy. Instrumentness subsumes the domain object duality and complex *chains* of mediation that result in clusters of primary, secondary and tertiary artefacts. The chain of mediation refers to entities that are *"chained together and attain blurred roles as mediators and objects of interest; artefacts modifying other artefacts, artefacts turning into objects, etc. in elaborate structures"* (ibid). Artefacts can be used directly in productive acts (primary), transmit information about the productive *practice* (secondary) or represent an aesthetic (cultural) concept about the tool or practice (third). In this way, artefacts in the software live in several domains (*e.g.* technological, tool-related, socio-cultural). Materiality encompasses the user's viewpoint in that the software has *"material aspects such as resistance"*, and that an (software) object *"reifies and mediates functionality and at the same time be an object for reflection at several levels"*, it *"is exactly what constitutes the software as an instrument, as something to play on and with"* (ibid), thus it also implies a socio-cultural context (*e.g.* genre, form, instrumentation). The resistance to change or mould objects and thus creatively engage with the software is related to the aforementioned aspects of difficulty and exploration. This is also related to another deviation from classic HCI: the ambiguous role of efficiency and ergonomics where peripheral tasks should be efficient or easy to master, while the musical task can suffer from it, especially in the context of long-term engagement. As pointed out earlier, however, which task is peripheral and which one is not depends on the context of the CSCM environment (*e.g.* task domain, subculture).

· ◇ ·

Following the argumentation presented in section 3.5, giving users a palette of choices or varying possible paths within certain bounds to accomplish a musical task may, on one hand, foster the experience of self-determination and perception of control and, on the other, invite users to explore the medium and assimilate or master the tool. The exploration of the medium and compositional space is central to all of the proposed prototypes. Chapters 9 and 13 specifically address the degrees of freedom given to a musical task, and the motivation and patterns of engagement in music participation. Especially the latter presents a study, whether for the task of rhythmic music creation, the offered degrees of freedom in control are also reflected in users' frustration or feeling of being overwhelmed.

5.1.1 Building blocks

Open-endedness, exploration or the more differentiated perspective in CSCW, high-ceilings and wide walls, suggest a high granularity of building blocks. Additionally, they indicate that the malleability of these tools point towards a generic use (tailorability and Gelineck's *Design for unintended use*) in their respective musical domain rather than a specific one which is high-level [Hew05; Cro08]. The presented prototypes in chapters 9 and 13 are

designed towards the traits of high granularity and high tailorability. This approach has also been presented in a previous contribution (cp. [Klü+11]). Gelineck [GS09] and Vaggione [Vag01] see that individuals seek to create constrains on their creative processes in order to reduce the cognitive load that the breadth of musical possibilities imposes. The concept of materiality can be therefore extended to also reflect subjective traits. Thus they should not be part of the interaction design, since these constraints are self-imposed and are related to incrementality. Exploration also implies another deviation from common HCI, namely that tools can benefit from a certain degree of unpredictability and errors in human performance should be embraced (vs. safety in HCI). All prototypes allow human error without side-effects that render the application unusable. However, approaches that *actively* promote or embrace human error have not been pursued. Unpredictability is especially a theme for the study in the chapter 14 and a major part of the design for the accompanying prototype. It is also a trait of the application in chapter 13 with respect to control of the audio synthesis. Another point is that expressiveness also includes goal oriented use. This has been part of the design goals and user studies for all prototypes. To summarize, the dichotomy of the targeted creation of musical artefacts and the exploration of the respective musical domain has been part of the argumentation and motivation in design for all prototypes in this thesis.

· ◇ ·

As presented earlier, support of heterogeneous groups implies inclusion of novices while at the same time coping with difficulty and complexity has beneficial effects. Therefore the prototypes in chapter 9, 13 and 14 argue for the design paradigm to lower thresholds while complex interactions with artefacts (expressiveness) are retained. In chapter 14, these presumptions are reversed by giving users an extraneous or complex interface and evaluating whether an actively mediating application (see next section) can help manifest the shared musical intent. Although it does not evaluate the composition on aesthetic merits *per se*, its decisions are guided by the actions resulting from collaboration.

5.1.2 Mediation

More generally, with respect creativity support, the computer itself can be seen as taking part in the field of creative practices: as a mediator it can either actively or passively evaluate actions and therefore guide the interactions and shape ideas similarly explicitly or implicitly. As such, it acts or reacts in the spectrum from reactive, mediating user interface to active collaborator. In my view reactive mediation may not only involve providing a shared environment for a musical domain alone but also involve diminishing differences in users' skills and knowledge with alternative or orthogonal representations and interfaces. For example, one approach is to model certain musical domains computationally and thus give high level control over this domain. This is the case for the prototypes in chapters 9 and 12 or 11. Active mediation then either involves participation in the collaborative task or adaptivity to the user's skills, and the social or creative situation. The application could, for example, adapt the interface or the generative processes based on a measure of the musical output (is the composition too erratic?) or customized to a user's skill-level (provide means for machine-guided incrementality). Such decision making and the evaluation of the benefits thereof for system design requires knowledge of music perception and musical practices. However, in the related fields of musicology or

music psychology there is lack of applicable insights with respect to the context of this thesis, *e.g.* studio practices in electronic music making. In chapter 13 a comparative study is performed to find out whether there is discrepancy in creating and perceiving rhythmic complexity with respect to user skill for the task of groove production. Other questions were whether different modes in constructing rhythm can be identified in relationship to user skill and rhythmic complexity or whether certain presumed practices in groove production can be identified (they may follow naturally in the musical domain or may be influenced culturally). Since controlled experimental comparisons may have risk of compromising the musical situation (*e.g.* with respect to the prototype fidelity), this study evaluated prototypes that were self-contained and comprehensive in sufficiently long sessions to facilitate engagement. Another study (cp. chapter 12) evaluated the validity of a model for rule based transformations of harmonic structure in a creative context. Here, theorized outcomes and results from empirical listening tests in literature were compared with observations from creative use of a prototype built around this model for task based music composition.

5.1.3 Studies

The problem in such studies is that qualitative-subjective evidence is rarely sufficient to reason about application use and the related behavioural or music-psychological phenomena. One option is to use logging mechanisms that are embedded into the application that allow to generate quantitative-objective data. I argue that logging and the analysis of higher order information about the interaction such as changes to the musical structure and can also reveal discrepancies in self-reports leading to a more objective characterization of prevalently subjective attitudes and insights. These techniques have been employed specifically in chapters 13, 12 and 14. In this regard, the rich and open task of musical collaboration [SM13] can also benefit from rich evaluation involving several methodologies (observational analysis, self-reports and logging).

5.2 Protocols in Music Collaboration

In [BF03] Blaine and Fels argue that the distribution of roles (*e.g.* leader, director, soloist, ...) is a valid approach to facilitate group coordination. Such roles may be responsible for signaling "clear goals" which are needed for the experience of Flow. There is also the possibility that the CSCM environment itself imposes roles or turn-taking patterns such as in Beatbugs [WAJ02] or the field of music entertainment games (*e.g. RockBand* [Har07], *SingStar* [Son04]) or computer games in general, since they carry features such as goal-oriented and affective involvement and a degree of learning. However, in an improvisational group performance "clear goals" cannot be chosen *a-priori* as they are the result of a collaborative negotiation process within the improvisation itself and also involve personal viewpoints. Therefore, the distribution of roles in musical improvisation is a dynamic process that relies on social protocols. This presupposes that the CSCM affords democratic interaction, which is a design goal that resonates with other contributions regarding CSCM and HCI (cp. [Hor05]). The contributions in this thesis largely follow this conception - every participant can equally contribute to the shared composition / performance without distinguished roles imposed by the application. The exception is

the prototype in chapter 14 which makes a distinction between instrument roles akin to a traditional band.

· ◇ ·

To reiterate the main point of Situative Creativity Support, embedding the application in a social environment means that social protocols can happen around the system and with the system (*e.g.* shared spaces). It therefore implies that the system does not block communication channels that are used naturally in human-human communication and in meaning-making. This then also has to apply for protocols in music improvisation. In contribution [BBP08] several insights ("lessons") in complex mediation within collaboration are given (paraphrased):

L1 Use means more than one use situation. Situations and context are non-hierarchical, overlapping, transient and heterogeneous.

L2 Many actors and artefacts mediate the same activity/focus on the same object.

L3 The outcome of an activity becomes the artefact for others.

L4 The overall activity may be carried out through chains of objects turn artefacts.

L5 Substitution of artefacts is common.

L6 There is more than one artefact at a time. It almost necessarily functions in concert with other, complementary artefacts.

L7 One artefact mediates many purposes at once.

L8 Artefacts are clusters of primary, secondary and tertiary artefacts

This extends the previous communication and coordination centric characterisations of musical collaboration with an understanding of interaction on the shared interface that is focussed on the construction process of music rather than simply interfacing with one. From the HCI centric viewpoint of chapter 4, it is apparent that several design traits act as facilitators for these "lessons":

- L1 implies that interaction allows multiple paths in a egalitarian way (*e.g.* using shared controls).
- L2 requires visibility of (inter-) action with respect to group awareness as well as parallel input.
- L3 implies iteration and progressive evaluation. This depends on feedback for the interaction (sonic & visual).
- L1, L4, L5 depend on non-linear construction methods.
- L5-L8 are coupled to aspects of granularity which can be seen as a facet of operational freedom.

These relationships can be seen as heuristic guidelines to design the interaction for musical collaboration.

· ◇ ·

As argued before, visibility and physical interaction are part of understanding music, communication and coordination in a social situation (*e.g.* with respect to group awareness). They also facilitate learning from interlocutors

on how to interact with the application. Apart from fostering shared representations, visibility is also related to affective and cognitive states, and behavioural and psychophysiological responses. Visual and, naturally, sonic feedback has been employed in all applications. Especially the prototypes in chapters 13 and 9 make use of visual feedback to inform about the state of the shared composition and the result of changes such that the system behaviour can be anticipated. Additionally, feedforward mechanisms were applied to guide complex user interaction. Feedforward foreshadows the results of an action while it is transient, thus users get a response from the application that indicates how their action will affect the system [Bec13]. Users get a chance to cancel the action midway to avoid undesired outcomes which is beneficial for users unfamiliar with the prototype (novices).

· ◇ ·

The prototypes do not support a confluence system of different composition states (branching) or a more traditional undo / redo system because in a collaborative environment these are difficult to conceptualize (*e.g.* ambiguous states, global undo may disrupt other contributors, local branches need tracking of users and their contributions). However, the support for iteration is also the basis for additional design traits. The prototypes make use of flat interaction hierarchies and high plasticity such that changes are immediate and (manually) revertible. Shared compositions can be altered concurrently and non-linearly which is also the result of the generic support for concurrent modification in collaboration. There is preference for local controls and transformations of artefacts to reduce interdependency of concurrent interaction. The use of modes is either omitted or kept to a minimum in order to reduce user errors (a more in-depth discussion on this topic can be found in [Klü+11]).

· ◇ ·

In section 4.3 practices in music collaboration were presented, including the presence of attractors and transitions that are part of the coordination processes. Only a few attractors are musically relevant and only one is the result of the union of successful coordination and personal goals. While one may view the latter as the significant goal of a computer mediation, I argue that all attractors are natural and necessary to create a shared representation. This is because disagreement also implies that participants are able to map the relationships of their mutual internal representations. Also, the experience of Flow depends on the appropriateness of the task and does not "come for free", thus it means that conflicts or discrepancies are possible necessary outcomes of collective musical engagement. Hence, it is difficult to determine if and how a creativity support system should proactively engage in the music composition process itself. In this regard, the presented prototypes generally take a neutral stance towards the musical outcome.

5.3 Electronic Dance Music

Regarding creative practice (section 3.1.3), a suitable domain should relate to common knowledge and symbols, given that the group of collaborators is heterogeneous. Electronic dance music has nowadays a pervading ubiquity due to its global raise to the mainstream [Fra12]. With its preference for repetitive structures (cp. [But06]) and sound qualities (cp. chapter 2 and 7), it can be seen as a suitable demarcation of a music related domain. I argue that the focus on electronic dance music in this thesis is an appro-

priate formal choice because it is pervasive and thus the formal, or "formalistic" aspects of such music are implicitly understood or "felt". This framing can be seen as a strategy to create a shared representation based on "musical logic". A further differentiation of electronic dance music into genres or sub-genres may not only have limiting effects on the creative uses of the prototypical applications and thus the scope of insights, but may also complicate the line of argumentation as such factors as group identity, political-economic matters and socio-historical context would have to be regarded as well (cp. [McL01] and 7.3). Therefore, the vague and generic term of electronic dance music is here also seen as guideline to frame the general paradigms of electronic music production, the related processes and tools.

· ◇ ·

Another viewpoint is that the purely electronic generation of music can act as an enabler for interactive systems and creative practices, because musical engagement is not necessarily linked to factual *instrument playing* skills as the technical environment mediates the playing. From the perspective of a composer, the traditional process of interpretation for performing a piece is omitted, thus the composer has more control over the acoustic outcome. Importantly, audio synthesis by the computer enables immediate feedback cycles (see chapter 2) which are relevant for motivation (Flow) and also for the creative process itself, as put by Sawyer: *"Creativity takes place over time, and most of the creativity occurs while doing the work. The medium is an essential part of the creative process, and creators often get ideas while working with their materials"* [Saw11, p. 88]. To foreclose the conclusions from chapter 7, for electronic (dance) music it holds that "the medium equals sound". As a consequence, the largest part of this thesis is dedicated to this subject. One central issue is that controlling the sound qualities of the audio generation processes traditionally requires technical knowledge (cp. chapter 2, 7 and 9). However, the issue with shaping sounds generated electronically or computationally by an audio synthesizer is that its sound shaping parameters are in most cases not *grounded* in the domain of perception (timbral qualities) and affect but in the generating structure and its technical functioning. This is problematic with respect to group heterogeneity as it excludes non-knowledgeable interlocutors if the task is focussed on synthesis control, it adds difficulty to peripheral tasks if the focus is on other musical domains. The aim is therefore to reduce or eliminate the technical aspects of synthesizer control. This is approached either directly (chapter 7, with an extension in chapter 11) for the use case of sound design or indirectly where synthesis control is peripheral (chapter 12). More generally, peripheral tasks in the context of electronic music are technical ones. Within the framing of the bedroom producer, these rely to a large on specialized knowledge. Therefore, apart from being a distraction or "nuisance", it can be reasonably assumed that, in heterogeneous groups, participants do not have enough background to see value in performing these tasks. In this regard, we aim to minimize or omit these in our prototypical implementations.

· ◇ ·

Section 3.2 emphasized the participatory context of music and highlighted that rhythm facilitates entrainment (*e.g.* in drum circles). Electronic dance music itself revolves around rhythmic structures[1] therefore it allows to exploit these effects in a music creation context naturally (as opposed to the participatory context in dance or club culture). These are the reasons why the two prototypical implementations shown in chapters 13 and 14 allow creating comparatively extensive rhythmic structures. Specifically chapter



Figure 5.1: Three approaches to rhythm programming with hardware. Top to bottom: Roland TR-808 (1980) using "xOx-style" step sequencing, Akai MPC-60 (1987) based on real-time performance recording, a modular synthesizer using clock dividers, logic gates, etc. for pattern generation. There are few other devices that had such an influence on electronic music and mainstream pop as the TR and MPC series. (Pictures taken from [Wik11; Aud15; Res15])

[1] Throughout the thesis I will also use the term "beat" or "groove" for musical structures that are repetitive and focus on rhythm.

13 is concerned with rhythmic complexity and its implications on collaboration and the musical product.

· ◇ ·

Although this reverses the historic causality, one could argue that the repetitiveness of electronic music makes instrument play in the traditional sense more of a technicality than central to the musical performance. Sequencer based control of certain performance parameters in music is therefore, on one hand, more natural for the creation of electronic dance music and, on the other hand, an approach to simplify instrument performance. Furthermore this also simplifies performance strategies for coordination: stabilization, ”Play along” and ”Wait and see” (section 4.3) are wholly or partially based on pattern repetition. Another aspect related to interaction and feedback is that a sequencing based interface can circumvent the perceived whole system latency as touches do not have to lead to audible results directly or immediately. For example, this allows to pre-schedule events in the application to compensate for the processing latency in audio synthesis and it decouples from the latency introduced by the input device processing[2]. Chapters 13, 14 and 12 cover (step-) sequencing musical events with respect to specific musical domains or compositional intent. A previous contribution to this topic is available in [Klü+11].

[2] The multi-touch tabletop (Citron dreaMtouch) used in the studies is based on an infra-red multi-touch frame, these have typical latencies between 7 and 30 ms.

· ◇ ·

Section 3.2.1 showed that the affective domain is an important part of the proximate functions of music. Research suggests that both, structural parameters of a piece (*e.g.* harmony, rhythm and tempo) and of the sound itself (timbre) are related to the perception of emotion (cp. chapters 7 and 10). Two approaches are presented: the identification of emotions in tone colour using machine learning (ch. 11) and the generation of simple music pieces with respect to emotional intent (ch. 12).

5.4 User-group

In the previous chapters several characterizations of music participation were presented: on one hand, the bedroom producer as archetype of the semi-professional modern music creator and on the other hand, the more generic typing of music participants (*e.g.* dabbler, hobbyist and professional). From the viewpoint of motivation, one can give preference for the group of hobbyists and recreationists or dabblers since they are primarily intrinsically motivated. The latter group may be more suited to scenarios where short-term musical engagement is prevalent, for example in installations or exhibitions. However, given the environment of collaboration in creative contexts and the reasoning to emphasize long-term engagement, the hobbyist can be deemed the best suited target group of users. This is because he is genuinely interested in musical engagement itself and continued adoption of musical skills and knowledge. The generally low affective potency of non-musical values may furthermore lower thresholds for engaging in the collaboration as differences in values between collaborators are valued less important. The difference between the bedroom producer and the other types is that the former has technical background knowledge while the others are first of all distinguished by their motivational stance alone - independent of specific skills or knowledge. As such, the targeted user-group is antipodal to that of the bedroom producer: the related metaphors, techniques and tools are not necessarily applicable to the presented use-case, since, by extension, they rely on specialized knowledge derived

from the recording industry. It is therefore necessary to abstract from this *lingua franca* for the prototypes presented in the following chapters.

5.5 Metonymy

Stowell *et al.* argue that there is inherent difficulty in finding metaphors for interaction because neither the music domain nor the breadth of possibilities provided by the computer can be coherently mapped to metaphors that hold cognitive advantage [SM13]. Even more, given the points raised in the previous sections, the multitude of different individual viewpoints and therefore concurrent metaphors make the design of collaborative musical interfaces more complex [FB13; Xam+13]. However, as argued by Bertelsen *et al.*, metaphor (is-a-kind-of relationship) is a trope based on similarity, which may be restrictive, because it makes use of analogy and substitution and thus may *"create constricting user models and narrow conceptions of the domain object"* [BBP07]. Furthermore, *"metaphors have been successful but also problematic in the sense that a good metaphor closes the software and locks users unless they are creative people who feel stimulated and provoked to try to break a too perfect metaphor"*. Metonymy (stands-for relationship) is another viewpoint as it is based on contiguity which more appropriately models musical activities that are not grounded in metaphor and encourages users to develop their own ways of using software. While both can be combined and seen as valid approaches to design a user interface, Bertelsen *et al.* conclude that *"metaphor may be the master trope for creating user-friendly and transparent software, while metonymy works better when designing for new and unforeseen uses like creativity and play"*.



Figure 5.2: The Roland TB-303 (1982) is one of the most influential bass synthesizers and has shaped Acid and Techno-House music. It is notorious for its characteristic sound and for its arcane sequencer interface. (Picture is taken from [Mat14b])

· ◇ ·

The prototypes presented in this thesis have a tendency towards metonymy this is also partially because there are few clear and established analogies that fit into a multi-user instrument paradigm. One could think of the ”virtual studio” metaphor but this is not sufficient because a ”hardware” metaphor does not represent an *established* domain object by itself. It encompasses multiple viewpoints on achieving certain musical tasks: there are different devices that do a similar task but are highly specific in their own musical logic (see figure 5.1 for an example). If at all, they have been popularized in niche genres. The ”virtual studio” metaphor is therefore flawed since virtual hardware itself does not represent a culturally and technically stable or strong analogy and therefore may not be plain to generic users without the necessary background. Another issue is that hardware by itself does not necessarily represent a ”good” user interface although a certain device may be popular and even known to novices (see figure 5.2 for an example). The compartmentalization inherent in different virtual hardware devices that are task specific further prevents sharing, and chaining and substitution of artefacts - the ”virtual studio” is therefore a metaphor of shared control over objects but not necessarily over artefacts. It has low granularity and low tailorability. The last point that I’d like to raise is that the skeumorph of hardware-like interfaces in a multi-touch environment reduces the interaction [NB11] to being low-bandwidth, because it only allows a one-to-one mapping in the parameter space, and because it inherits the mechanical constraints of the hardware interface (knobs, sliders, spacing) with limited tangibility compared to the original device. As pointed out before, this may also reduce Flow (cp. [NB11] for a related comparative study).

Chapter 6

Framework Architecture

Chapter Abstract

The applications presented in this thesis have been prototyped with a software framework, which offers reusable abstractions and components for rapid development. In the following, its software architecture is motivated and implementation details are elaborated.

The source codes for the presented frameworks have been made available:
N. Klügel, C. Ruff and U. Laufs. *UltraCom - UI-Framework*. Framework / Library, Distributed under GPL version 3. 2010-2014. url: https://github.com/lodsb/UltraCom
N. Klügel. *Reakt - Reactive Event System*. Framework / Library, Distributed under GPL version 3. 2010-2014. url: https://github.com/lodsb/reakt

---

Production of a viable artefact that can be evaluated is central to the design science approach. Furthermore, it was argued that prototypes in NIME require high fidelity in order to permit sensible evaluation (section 4.6). To facilitate this, a framework was developed to provide reusable components and abstractions. As stated before in section 3.1 and 1.3.1, development of music tools involves functional *and* aesthetic evaluations. It is to a part in itself a creative activity that relies on rapid feedback cycles for iteration. To borrow some terms from the field of creative computing and interactive programming, creative programming benefits from low diffuseness (verbosity of the language), low viscosity (resistance to change) and progressive evaluation among other cognitive dimensions that characterize the notation of programming languages or Domain Specific Languages [McL11, ch. 5]. With respect to a framework, prototypical applications make use of components in the following domains: the graphical user interface including gesture recognition, audio synthesis (signal generation and modification) and music related processes (high level musical events). Because the latter is specific to the prototypical application, the framework provides mainly subcomponents that enable the construction of graphical user interfaces and the control of the audio synthesis. However, the *focus* is the interoperability between these three domains.

· ◇ ·

From the software engineering perspective, we are interested in developing reusable parts or components for a family of applications and establish a system framework. Domain Engineering is one approach to achieve this goal [Cza98, ch. 3]. This includes:

- Domain Analysis: defining a set of reusable requirements
- Domain Design: establishing a common architecture for systems in the domain
- Domain Implementation: implementing reusable assets

The previous chapters have discussed the various facets of collaborative music making as part of the Domain Analysis (context analysis). Based on the argumentation in chapter 1, formally deriving a holistic Domain Model for a framework that facilitates the construction of applications for all use-cases within musical collaboration on a tabletop is difficult. However, it is possible to formulate high-level features that such a framework should provide to arrive at subsystems and components that can be reused. This loosely follows the Feature Oriented Reuse Method [Kan+98]. Based on previous applications in this field (*e.g.* [Jor+06; Xam+11; Xam+14; Bis+08b; Klü+11]), we can identify the following features, including capabilities which are functional and non-functional, the operating environment and domain technologies:

- Functional features:

F1 Perform audio synthesis	F2 Accept multi-touch input
F3 Perform gesture recognition	F4 Provide a graphical user interface
F5 Support multiple users working concurrently	F6 Provide a shared display environment (tabletop)
F7 Control audio synthesis	F8 Control music processes

- Non-functional features:

  NF1 Low-latency, suitable for interactive use. Audio synthesis therefore has to be performed in real-time.

  NF2 Plasticity of the framework for diverse prototypes including the design traits pointed out above for rapid development.

  NF3 Interoperability with other systems using standard (music related) protocols to enhance the functionality of the framework and leverage functionality provided by other environments.

  NF4 Framework functionality that provides tight interaction between musical processes, graphical user interface and audio synthesis for feedback and control.

  NF5 Reliable framework components that allow the creation of high-quality prototypes.

  NF6 Cross-platform since students may use the framework for (future) theses, thus specific platform requirements should be avoided.

Figure 6.1 outlines the component-wise framework architecture that will be discussed in more detail in the following sections. This includes Domain Design and Domain Implementation.

· ◇ ·

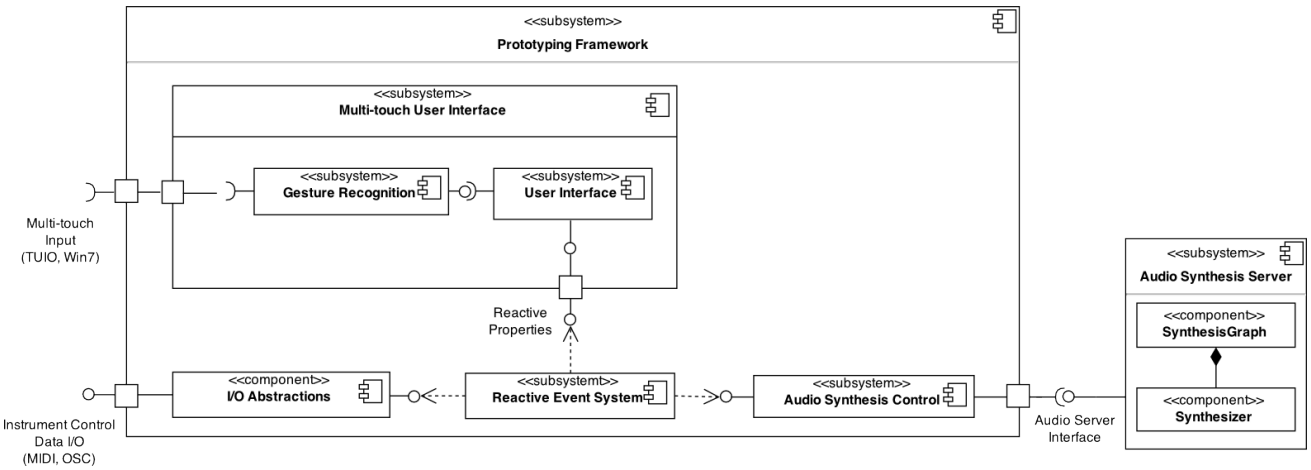


Figure 6.1: Framework as high-level component diagram

Several technology related choices were specifically made to facilitate the non-functional requirements. First, the framework relies on Scala [Ode+04] as programming language and runtime environment on top of the Java Virtual Machine. Thus, it provides a tool-chain for cross-platform development and code execution (→NF6). Scala allows seamless interoperability with Java programs. It is a multi-paradigm, general purpose programming language that supports object-oriented, functional, imperative and concurrent programming constructs. Furthermore, it features several design traits from functional programming languages such as lazy evaluation, algebraic data types, covariance and contravariance. It's type system relies on strong static typing and offers type inference (→NF5). In this way it allows to combine concise programming constructs and practices from both, object oriented and purely functional programming. Additionally, for progressive evaluation, Scala offers a shell for developing, modifying and testing code or applications interactively (→NF2). Second, a framework for developing rich multi-touch applications [LRZ10] has been adapted which already features graphical user interface elements and gesture recognition (→F2, F3, F4). It has been applied successfully to create several prototypical implementations for other application domains in the NUI community (→NF5). Third, a widely adopted and proven framework for real-time audio synthesis (SuperCollider, [McC02]) was used (→NF1, NF5).

· ◇ ·

Technically, there is the difficulty that the domains, audio synthesis, graphical interface and musical processes, result in heterogeneous subsystems which, in the case of musical applications, are furthermore inherently concurrent [Mag06]. On one hand, to recur to requirements in creative programming, one aim of the framework is to the reduction of ripple effects in prototypical development, which implies decentralized states, low coupling and high cohesion of processing elements or components (→NF2). On the other hand, tight interaction between these domains (→NF4) may lead to a different (*e.g.* monolithic) framework architecture. As an alternative, the presented framework provides a data flow inspired architecture to loosely couple components (see fig. 6.1). This is discussed in more detail in the next section. Finally extensions to this are provided to enable communication with external devices or applications using MIDI or OSC.

6.1 Event Processing

Interactive applications have to react and coordinate multiple events. There is difficulty in programming such applications because events are to a large degree generated externally, such that prediction or controlling the order of events is impossible. The effect of this is that control over the program is inverted (the control flow of the program is defined by the external events). This complicates the management of states in the computation and thus is error prone using programming solutions that rely on traditional event handlers (callbacks) and related design patterns [Bai+13]. Coordinating such asynchronous callbacks is daunting, because the respective code is isolated yet changes to the business logic, or data in general, occur on shared states as side effects of the event handling functions (callbacks don't have return values). For example, a recent analysis of Adobe desktop applications revealed that event handling comprises a third of the code and contained half of the bugs reported [Jär+08] (cited from [Bai+13]).

· ◇ ·

In case of the prototypes, events can happen internally and between several system components in an asynchronous way, yet there are also processes that demand synchronous updates (technically or domain specific). Furthermore, mappings between input and output events are complex. The following analyses the related technical issues:

Input modalities (mixed interaction)

- Tabletop and multi-touch:
  - Multi-touch positions are commonly sent via the TUIO protocol [Kal+05] for cross-platform applications using network transport protocols. Thus, input events are not part of interface event serialization mechanisms offered by the UI toolkit or operating system.
  - Multi-touch itself has intrinsic concurrent behaviour [Sch+11b], especially in a collaborative context as there are separate user-input streams. It follows that actions and reactions in the application are itself concurrent.
- External controllers: MIDI data input can be sent from external applications or external controller devices to a dedicated MIDI input device. Again there is generally no UI related support for integrating such event streams; the same applies for input methods that make use of the OSC protocol (network).
- Feedback from the audio-backend: Audio rendering is performed in another process and FIFO pipes or local network sockets are used to inform the application about state changes or real-time audio feedback such as amplitude levels of the rendered audio.

Internal processing (mixed temporal dependencies)

- Gesture recognition: Gesture events happen only if a specific gesture is (partially) recognized. This depends on the type of gesture and the stream of input data. For example, while a drag gesture may immediately result in events of the type 'drag', the recognition of handwriting relies on more input data and may delay generating the respective output events (online versus offline gestures).

- Graphics rendering: Rendering takes place in another context such that updates to the stateful graphics pipeline (*e.g.* hardware accelerated rendering with OpenGL) can only happen synchronously.
- Musical timing: Music itself relies on multiple, concurrent time-lines in which (musical) events take place. These processes run asynchronously and independently (to a degree) to the application specific event processing but they change application states.
- Complex mappings: In the NIME related context, instrument design requires complex mappings from multiple input event sources to multiple output destinations that also account for inter-dependencies of event data [DK06].
- High frequency updates and low latency: Musical interfaces demand intimate control, which implies that controller input and state updates should happen with low latency, high (sampling) rate and bandwidth [WW02]. State updates therefore happen frequently and concurrently.
- Expensive processing: Due to the real-time context and interactivity of the application expensive processing must not block use of the system. Processing may therefore happen asynchronously and create asynchronous events.

Output modalities (mixed environment interactions)

- Sound synthesis: As stated above, audio rendering is performed in an external process. It is controlled via a local network socket.
- MIDI and OSC output: Similar to the above synthesis may be controlled in other applications (*e.g.* a DAW) via MIDI / OSC control.

To summarize, event processing happens across heterogeneous components in the prototypical applications. It involves asynchronous and synchronous operations across thread and process boundaries with complex inter-dependencies and state behaviour.

· ◇ ·

In the following, (functional) reactive programming as recent approach to abstract from the ”mechanics” of event based programming is introduced. I will then broaden the context of reactive programming, because, to my personal frustration, the related literature and software products tend to ignore the rich history and practices in data flow oriented programming business appliances and NIME.

6.1.1 Functional Reactive Programming

Reactive programming has been proposed as a well-suited solution for event-driven programming [Bai+13]. It is originally based on the (synchronous) data-flow programming paradigm and thus provides abstractions to express programs as reactions to external events and time while managing the flow of data and computation dependencies. The main idea is to use an underlying execution model to automatically and efficiently propagate state changes across a network of dependent computations. In this way the aim is to facilitate a declarative programming paradigm for event processing in which computations are composable first-class types where stateful operations have been reduced or removed. Mutable values are

considered as time-varying values to better capture the temporal aspect of mutability [Cza12]. In case of *functional* reactive programming (FRP), a (higher-order) general-purpose functional language is extended with these abstractions [ICK06]. Generally, the time-varying values are called signals.

· ◇ ·

FRP was first formulated in [EH97] (cited from [Cza12]) as part of an implementation that is embedded into Haskell. Here, signals are expressed as *continuous*, time-varying values (Behaviors) that *constantly* change as a function from time to a value:

$$\text{Behavior } \alpha = \text{Time} \to \alpha$$

Another type that was introduced are Events as a sequence of discrete events that are a time-stamped list of values.

$$\text{Event } \alpha = [(\text{Time}, \alpha)]$$

Time values must increase monotonically. The definition of Behaviors allows expressing dependencies on past, present and future values. The causality of time-dependencies must be met in order for programs to be implementable. Programs that rely on past values may introduce additional memory usage that grows proportionally to the time the program has been running. The evaluation model is demand-driven, or *pull-based* [Bai+13]. It relies on *sampling* signals. While such a pull model offers the liberty to only request new values when they are needed, a significant latency between event occurrence and reaction may be introduced as all delayed computation have to be performed at once. This may result in space (memory for storing events between computation) and time (computation necessary to "catch up") leaks that arise over time, especially for implementations with lazy languages (*e.g.* Haskell). Subsequent implementations such as Yampa [Hud+03] or NewFran [Ell09] have remedied the issue of time and space leaks while sometimes limiting expressiveness [Bai+13]. Additional criticism is expressed by Czaplicki [Cza12]: these frameworks generally enforce a strict ordering of events while assuming that updates are instantaneous. Since, practically this may not be true, long updates may block all pending events. Furthermore, continuous signals are assumed to be constantly changing such that sampling is performed as frequently as possible. However, many practical events such as mouse clicks happen sporadically compared to the update frequency since these are discrete events (as all user input) and therefore unnecessary computations are performed. This is especially problematic, given that updates may not be instantaneous.

· ◇ ·

As an alternative to being demand-driven, the evaluation model can also be data-driven (push-based) [Bai+13]. This model is common in implementations using eager languages and involves some form of observer pattern. Such implementations allow latency between events and reactions. However, similarly to the above, computation can be wasteful if a change of input source leads every time to re-computations. Currently there are numerous push-based implementations that add abstractions over the observer pattern in form of a functional or functional-inspired framework for reactive programming. Examples are *ReactiveX*, with multiple language bindings [Var14c], *Scala.React* [MRO10], *Bacon.js* [Var14a] for reactive Web-Applications and reactive data structures for Scala [Var14b].

· ◇ ·

There have also been contributions that combine both models [Spe01; Ell09]. To recur to the Haskell based implementations, extensions have been proposed for first-order FRP. For example higher-order FRP which allows the computation graph to change dynamically [KBH12] or Arrowized FRP that relies on signal functions instead of signals to dynamically change the process order using dynamic collections of signal functions [CC13].

6.1.2 Data flow Programming

The cited related work fails to clearly point out the relationship between reactive programming and data flow programming. Reactive programming can be seen as a subset of data flow programming which is inherently based on reactive and composable behaviour, and transformational programming aspects [Car14]. The history of data flow programming dates back to at least BLODI [KLV61] (this and the following references are cited from [Car14, ch. 1]), which was a system for simulating signal processing devices using processing blocks that communicate over links. Sutherland's work in [Sut66] shows one of the first systems for visual data flow programming in real-time. An early definition of how data flow programs should operate is given in [Den74]. Data flow programming paradigms are also used widely in hardware description languages such as Verilog or VHDL.

· ◇ ·

As with reactive programming, data flow programming is declarative. It makes use of nodes that are processing elements that define input and output ports to connect to other nodes in an acyclic directed graph in which edges model the flow of data through the system. In most cases nodes are purely functional but some implementations allow stateful behaviour. The execution semantics per node involve consuming data from the input ports (if available), processing it and publishing the results of the computation to the output ports (if necessary). The graph structure allows reasoning about parallel, asynchronous processes. In many cases data flow programming is used to model or execute concurrent behaviour. There are additional properties that characterize a data flow implementation, such as number of ports allowed per node, edge capacity to hold data and graph structure, *e.g.* splits and joins are forbidden to simplify concurrent execution with mutable data, and whether the graph is dynamic, asynchronous or synchronous (fixed update rate). Some prevalent types of data flow programming are:

- Pipeline data flow: Nodes may only have one input and output and the graph structure is comparably simple, *e.g.* Unix-Pipes, stream processors [Car14, p. 60].
- Actor Model: The graph structure can change dynamically while nodes are executed concurrently. Data is exchanged via message passing between actors (nodes). Instead of input ports actors have an ordered list where messages are enqueued upon receipt. Messages can usually only be received by actors that are known to the sending actors, implying some form of addressing scheme. Actors are referentially transparent per invocation and messages are generally immutable. Since actors are independent of others, cycles and recursion are allowed. Some actor implementations are location transparent. One of the most influential actor systems is part of the Erlang programming language [Arm10].
- Flow-Based Programming: The model is static and asynchronous [Car14, p. 50]. It allows multi-port and compound nodes. Immutable and higher

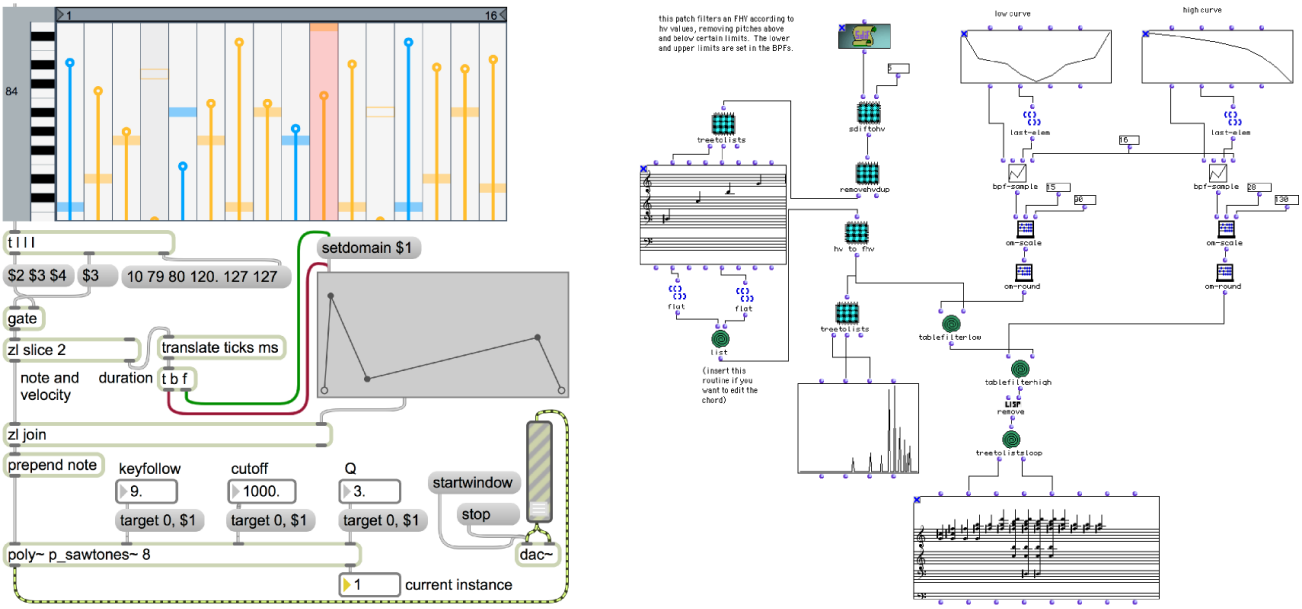


Figure 6.2: Exemplary patches (programs) in Max 6 and OpenMusic

order data-types such as lists and trees by means of grouping data fragments can be used. Nodes are stateful (*e.g.* uninitialized, terminated, active) and cycles in the graph are permissible.

6.1.3 *Data flow programming for NIME*

In the context of NIME, data flow based programming is by far the most common method to prototype new musical instruments and synthesis methods. Prevalent environments are Max [PZC90] and PureData [Puc+96] (Max family of languages), Reaktor [Nat96], OpenMusic [Ass+99] and Kyma [Sym90] (see figure 6.2). All of these languages use a visual programming paradigm (called patching) where programs can be edited while running. Furthermore graphical control elements and external controllers can be similarly added "just-in-time" to alter and experiment with the running program. This allows for extremely rapid feedback cycles for iterative development. In fact, this form of interactive programming[1] emphasizes the activity of writing a program as a creative endeavour, thus blurring the lines of what is the program (product) and programming (activity). As such, interactive programming also has a culture of programming as a performance dating back to early (collaborative programming) experiments by *The League of Automatic Composers / The Hub* (late 1970s) [Gre98]. It recently gained more interest in the form of international collectives (*e.g. TOPLAP* [McL+]), festivals (*e.g. Algorave*, [Alg]) and international conferences (*e.g. International Conference on Live Coding* [ICL]). It should be pointed out that in these contexts, textural languages are equally represented (*e.g.* SuperCollider [McC02] or Gibber [RK12]) and they make, at least partially, use of similar data flow related constructs.

[1] Synonyms include live-coding, on-the-fly-programming, learneable programming and conversational programming.

· ◇ ·

Apart from the performative aspect of programming, recent contributions to software development practices include mobile application [Bri+11], game development [Pho14] and a framework for designing musical tabletop applications [JGJ11] using PureData. With respect to research practices, frameworks have been developed for mapping gesture to sound [BMS05], ma-

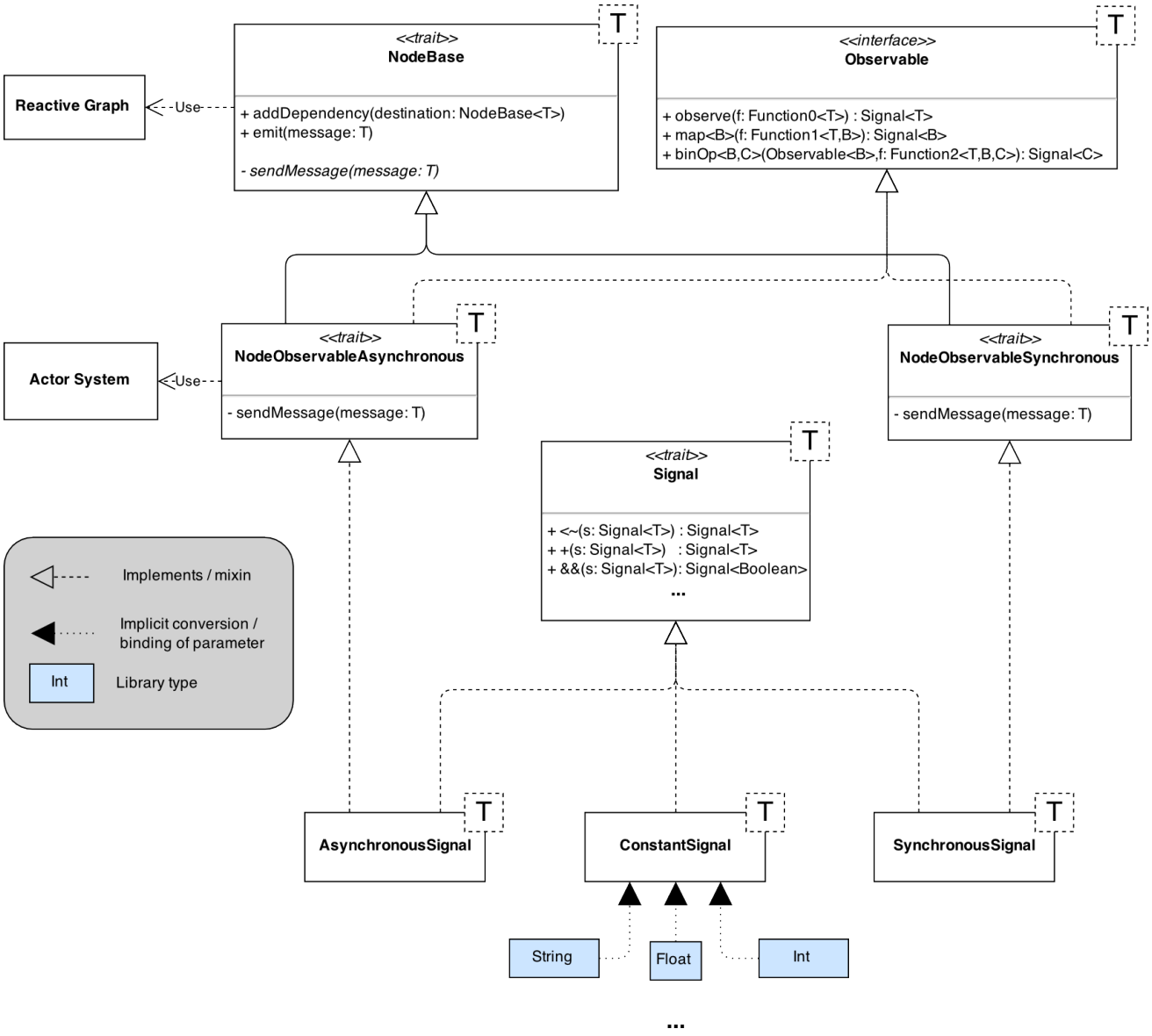


Figure 6.3: UML-class diagram of the Reakt framework (omitting some implementation details). As discussed, the framework consists of two parts, one that encapsulates the DSL for FRP and one that implements the synchronous and asynchronous data flow. Implicit type conversions are indicated that bind the template parameter of a `ConstantSignal[T]`. Functions are represented as types `Function*`, similar to Scala function objects with templated type parameters.

chine learning for musician computer interaction [GKO11] and studying gestural interaction [JGW05] using Max. Regarding studio practices, patches in Reaktor, Max (Max for Ableton Live) and PureData [Mat14a] can also be loaded into certain DAWs to extend the provided functionality. Languages like Max and Reaktor use a push-based model, thus apart from the audio synthesis they are fully event driven. OpenMusic is prevalently used in algorithmic composition and makes use of a pull-based approach since it largely relies on evaluation of functional expressions (in Common Lisp) although extensions for truly reactive behaviour have been added recently [BG14].

6.1.4 The Reakt Framework

When development on the framework started (in late 2010), FRP was not such a publicized topic as it is today. Consequently the aforementioned implementations were either not yet developed or unavailable for the public,

especially for Scala (or Java). The main inspiration came from the contribution *Deprecating the observer pattern* [MRO10] and data flow languages such as PureData. This is why the nomenclature may differ from the presented FRP systems. An extension to PureData or Max has not been considered since concurrency on a thread / process level is prohibited or difficult to achieve, also because high level data-structures cannot be added using an embedding language and type-polymorphisms are restricted to basic types, and because changing the processing graph programmatically is a non-standard operation (lacking documentation and testing). Similar to those languages, Reakt uses nodes as processing elements in a graph structure to express the data flow.

Implementation Details

Besides the generic functional and object oriented language features of Scala, the implementation makes use of mixin class composition, implicit type conversions and operator overloading. Additionally, the framework relies on type polymorphism in order to embed execution on any type of data in a type-safe way. It is push-based, uses a dynamic graph and supports concurrency on top of the *Akka* toolkit (cp. [Gup12]). Akka is based on the actor model, providing means to create distributed, concurrent and event driven applications. Communication is carried out via message-passing. Akka actors are location transparent, local instances are run on an additional thread pool, called "micro-kernel", that also provides highly efficient message-passing and low memory footprint. Besides this, the Scala libraries provide a Domain Specific Language (DSL) that abstracts from the low-level functionality and offers high-level abstractions such as Futures, Promises and Software Transactional Memory. The synchronous data flow is put into effect with delegation (of method calls). In the asynchronous case, the processing graph is traversed by actors that live within a worker-pool and messages that are passed contain the execution context of a node. These asynchronous and synchronous parts of the data flow can be mixed at will. A strict ordering of tokens is not guaranteed for asynchronous data flows. This relaxation of constraints has the benefit that execution can be done efficiently, circumventing the aforementioned issues of time and space leaks. While the ordering is not always necessary for all event types, it can be enforced by expressing the parts of the data flow in focus using synchronous types. To diminish wasteful evaluation of the whole graph, the implementation makes use of memoization at the node-level. This way only the child nodes of an event-source node have to be evaluated, even if sub-graphs that are independent with respect to the event source are joined at some point of the evaluation chain. Additionally, splits in the graph structure are allowed. To have some liberty in triggering the evaluation (similar to pull-based data flow), nodes can be forced to send messages based on the currently memoized value, this is similar to the *bang* message in PureData or Max.

· ◇ ·

The implementation relies on modelling two domains, the data flow behaviour for synchronous and asynchronous messaging and the functionality of computational abstractions and data transformations in FRP. For now, we call the nodes in the FRP domain simply a signal of type `a` (the data type), or simply `Signal[a]` to be close to the Scala notation. In the following, the implementation will be described in a simplified and more abstract manner to omit unnecessary details. The data flow behaviour relies on a global directed acyclic graph (one per runtime) and two node types, for

asynchronous and synchronous behaviour. Nodes are abstract and implemented as mixins with concrete implementation details about the message passing mechanism. This includes methods for connecting and disconnecting child nodes. More importantly, they offer methods for instantiation of observing or intermediate child nodes, such that the mixins abstractly expressing the FRP functionality can rely on the behaviour that the transformations produced stay within the same node-type class (synchronous or asynchronous). To be more explicit, child node instantiation follows the design pattern of *lifting* in functional programming. For example, given `Signal[a]` that outputs data of type `a` and a function $a \rightarrow b$, the lift operation returns a `Signal[b]` which outputs data of type `b` every time it receives input of type `a`, thus "lifting" the function to a higher-order type. Nodes are immutable and encapsulate exactly one function. Depending on the function, new output values are generated based on received (pushed) data. In case of purely observational nodes, the function returns no value and input data is simply emitted unmodified. This guarantees that nodes can always be chained and observation never results implicitly in a *functional* modification of the data flow (*e.g.* for debugging prototypes and rapid development). The following outlines the mixin[2] for an asynchronous node which is derived from the base class `NodeBase[A]` that encapsulates the core functionality for both node types (asynchronous and synchronous), the return type is made explicit:

[2] Mixins are confusingly called traits in Scala.

```
trait TNodeAsynchronous[A] extends NodeBase[A] with Observable[A] {
        def observe   (f: A => Unit)
                    : SignalAsynchronous[A]     = {...}

        def map[B]    (f: A => B)
                    : SignalAsynchronous[B]     = {...}

        def binOp[B,C](that: NodeBase[B], f: (A, B) => C)
                    : SignalAsynchronous[C]  = {...}

    @Override
        protected def sendMessage(message: A) = {...}
}
```

The asynchronous message passing is implemented in `sendMessage(message: A)`. `Observable[A]` is the interface definition for nodes that provides "observable" behaviour. Connections in the graph can either express unary or binary operations[3]. Therefore, apart from observing (method `observe`) and mapping or transforming (method `map`) a type, also two parametric functions can be lifted (method `binop`) and the observation depends on two nodes. These basic functions accommodate the higher level constructs used in FRP. As mentioned earlier, the FRP related mixins make use of operator overloading and implicit type conversions. This technique is common in Scala frameworks to construct DSLs (cp. [Ode06; Gho10]), especially since the Scala compiler allows omitting language symbols (*e.g.* dots, braces) if the expression is unambiguous. For example, arithmetic operators are provided as syntactic sugar in the form of object methods which are then evaluated to an element-wise arithmetic combination if the specific data types of the signals support an implicit conversion to a numeric base-type.

[3] Binary operations do not differentiate between hot or cold inputs, thus no preference is given on the input source to trigger re-computation However, for any node, the order of messaging output nodes depends on the order of connecting these destinations.

· ◇ ·

Similarly, implicit conversion rules have been supplied to convert any variable type in a program to a constant signal (of that type) if an expression

in combination with a signal demands it: a library type is simply lifted to a constant signal of that type by binding the type parameter of a special signal object and setting its output value. It should be noted that these type conversions do not obey Liskov's substitution principle. Memoization is implemented on the FRP side. Here, operator overloading allows simple means to push data into a node or to read a single value from it. Furthermore, there are operators for modifying the graph structure. Figure 6.3 outlines the described class hierarchy.

· ◇ ·

For convenient interoperability with Java frameworks, additional signal types have been defined that wrap intermediate objects produced by the Scala compiler (function objects), type conversation of primitive types and additional type information to circumvent errors produced by type erasure of the Java compiler for generics. These objects also perform delegation across thread boundaries (bidirectional) to support reactive updates *e.g.* from and to the graphics rendering thread. Unlike their Scala counterparts, they are only used as sources and sinks in the graph.

Example

In the following an example is given how the FRP framework can be used in the prototypical implementations. This example supposes that there are UI components, a slider, a button and a text label, with reactive properties. This API is equivalent to the one developed for the UI part of the system architecture.

```
1  val sliderValue <~ slider.value
2  sliderValue.observe { x =>
3          println("My value is "+x)
4  }
5
6  val aValue = new SignalAsynchronous(3.0f)
7  textLabel.text <~ aValue.map(x => "My value is "+x)
8  val mulitpliedValue <~ slider.value * aValue
9  val someBooleanStream <~ (multipliedValue < 200.0f) && button.pressed
10
11 aValue() = 4
```

Lines 1–4 show how a node can be explicitly observed using an anonymous function. Here, also an intermediate node is created (`sliderValue`). In line 6 a node is instantiated with an initial value which is updated in line 11. This triggers an re-evaluation of the graph starting from the (child) node that encapsulates the binary multiply function as part of the graph expressed in line 8. Similarly, this will also update the content of the text label which is dependent on the type conversion `Float` → `String` in line 7. Lines 8 & 9 exemplify how operator overloading is used to perform arithmetic computations on signals. These lead to an implicit type conversion in line 9. Furthermore, the red number indicates that an implicit type conversion from `Float` to the constant signal `ConstantSignal[Float]` has happened. Figure 6.4 shows the explicitly instantiated and implicit intermediate signals created and the resulting graph structure.

Performance estimation

To get a rough estimate how the overhead introduced by the decoupling (message passing) affects overall performance, a trivializing experiment

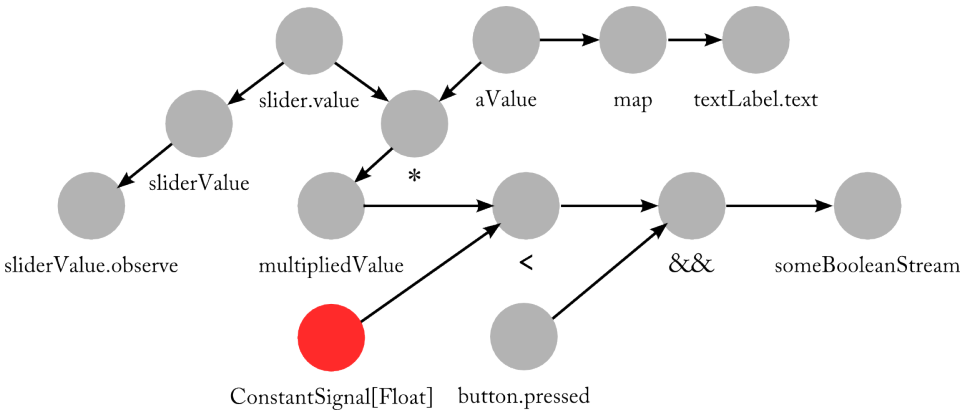


Figure 6.4: The generated graph structure from the code example above, including intermediate signals. Circles denote nodes and arrows, respectively, the edges in the graph.

was performed. This is by no means intended as a scientific examination. Figure 6.5 depicts the throughput of a network in various sizes either using direct function calls, synchronous graph elements only or asynchronous ones. Apart from the throughput also the relative overhead is shown. The graph is simply a pipeline structure (3 to 500 nodes) such that for the asynchronous case concurrency cannot be exploited and the overhead is accumulated in all three scenarios. The measurements were made by timing how long it takes for one million messages (average of ten rounds) to be pushed through the network from source to sink on a mid-range system (Intel i7 4930K CPU with 32GIB main memory). Otherwise, no additional operations are performed on a node-level. The throughput measurements using the direct function call are interpreted as base-line measurements. For this, a sequence of the length of the pipeline was simply *folded* (functional programming) to emulate the pipeline. It is apparent that the asynchronous message passing has at least a ten-fold execution overhead compared to direct function calls and, similarly an overhead of the factor $2-8$. This can be traced back to the mechanisms in the actor implementation (*e.g.* thread scheduling, resulting delays in message receipt). Conversely, the overhead of the synchronous graph execution compared to direct calls (roughly a factor 5 in the experiments) can be accounted to the delegation method. Here, the delegation is performed recursively down the graph, with the effect that the overhead introduced by the management of the stack increases with the length of the pipeline. This could be improved in future work.

· ◇ ·

While the asynchronous and synchronous message passing methods have a significantly lower throughput compared to the direct function call method, this does not diminish the applicability for the use case. Practically, longer pipelines than ten steps have been used rarely in the prototypical implementations and the event system never caused performance concerns. On the same machine where the throughput experiments were performed, a ten stage pipeline can still be processed with around 170000 (asynchronous) and one million (synchronous) messages per second. This is enough for interactive event processing, even if broadband event streams are used. Since synchronous and asynchronous parts of the graph can be mixed arbitrarily, the higher performing synchronous implementations can be used even in concurrent parts of the graph.

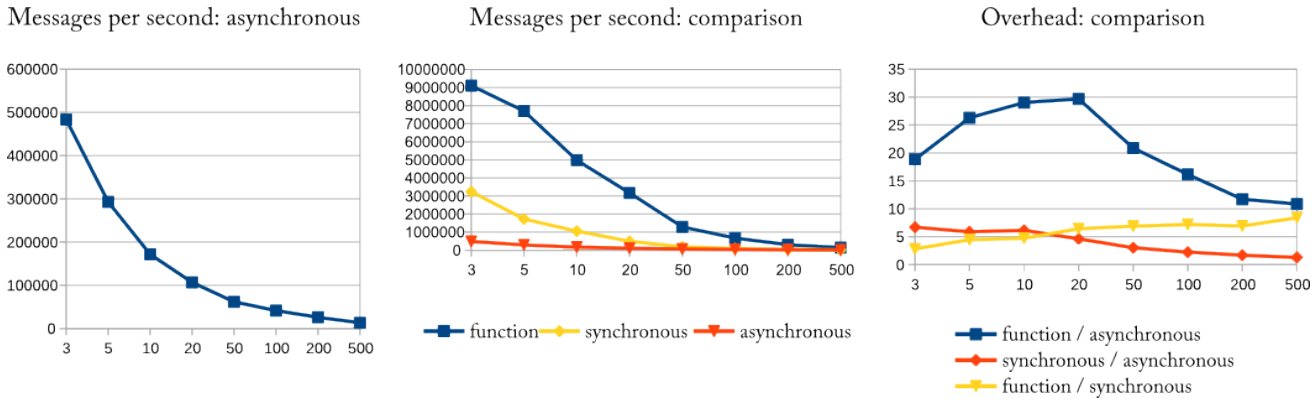


Figure 6.5: Plots of the numbers of messages processed for a pipeline graph configuration. The x-axis shows the length of the pipeline. The plot in the middle side shows the throughput based on direct function calls (function) and synchronous or asynchronous nodes. The plot on the left hand side shows the measurements for the asynchronous graph only. A comparison of the relative overhead is depicted on the right hand side.

Conclusion

The presented framework abstracts over event streams, their transformation and combination. It allows to design the event processing in a loosely coupled, state decentralizing fashion which emphasizes separation of concerns. It lies at the heart of all prototypical implementations to coherently control the heterogeneous parts of the system architecture. This is because they are cheap with respect to overhead in practical, interactive scenarios and because the abstractions result in an API that is mainly concerned with element-wise type transformations. Furthermore, because of type polymorphism, any data type can become an even type (*e.g.* lists, trees, graphs, UI elements) such that also processing on high-level data structures can be performed in an reactive fashion.

6.2 User Interface

The framework for rapid development of tabletop user interfaces is based on MT4j (Multi-Touch for Java), which was developed by the *Fraunhofer Institute for Industrial Engineering* [LRZ10]. At the time, frameworks for writing the prototypical applications were evaluated, few alternatives offered a cross-platform and open-source environment for application development. Most notable are Squidy [KRR09] and pyMT [Han+09]. Active development for all of the mentioned frameworks ceased while writing this thesis (MT4j is not maintained since late 2012). Frameworks that are proprietary or rely on proprietary technology at that time included Microsoft Surface SDK and Diamond Touch, Miria SDK and Breeze (based on Silverlight and WPF), and Adobe AIR [CN14; LRZ10][4]. Aside from frameworks that offer visualisation support, some focus on gesture recognition only, by providing a dedicated gesture server ([EK08], [Ram+09]) or establishing means to declaratively define new gestures [Kin+12; Sch+11b]. While this gives liberty to chose an arbitrary UI framework, the collaborative use-case requires that the UI toolkits implement some form of a scene graph to allow arbitrary rotation and scaling of UI components which was not the case for all toolkits at that time. Current frameworks such as Qt [Qt 14] and JavaFX 8 [Ora14] support true multi-touch gesture interaction and rely on scene graphs to display highly adaptive user interfaces. Alternatively, modern web-technologies can also be used to develop tabletop applications [Moz15].

[4] Apart from Microsoft Surface and Diamond Touch these frameworks are *de facto* deprecated.

· ◇ ·

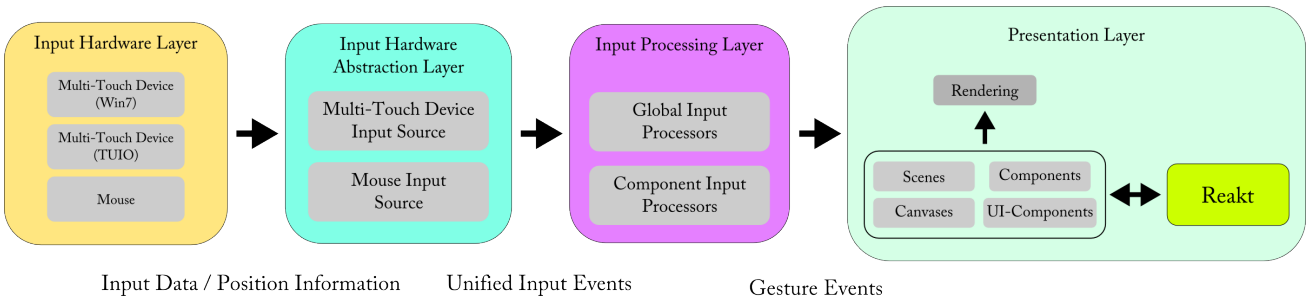


Figure 6.6: Overview of the MT4j architecture including modification in the current implementation (adapted from [LRZ10])

MT4j offered the most complete development stack, including input hardware abstraction, gesture recognition, and high performance rendering. It furthermore facilitates the creation of true three dimensional user interfaces. The framework allows online (events while the gesture is performed) and offline gesture (events at the end of the gesture) recognition. Gestures are expressed procedurally which makes the implementation of complex multi-touch gestures non-trivial [Sch+11b]. Nevertheless, such gestures are problematic if they are not part of common practice (how is their use communicated by the application and learnt by the user?) or if the input device can not differentiate between different user input streams, which is the case for most tabletop devices. MT4j supplies standard gestures known from mobile computing interfaces (*e.g.* pinch, rotate, drag) which proved to be sufficient for the implemented prototypes. Gesture processing relies on a layer-wise approach in order to abstract from the concrete input methods (see fig. 6.6) [LRZ10]. The input hardware layer supports several data sources, such as TUIO and multi-mouse (for debugging), these event streams are then combined based on input class in the hardware abstraction layer. Finally, in the input processing layer, gesture processors perform gesture recognition and transform these into logical gesture event stream data. Processors can run globally (per scene) or per UI component (*e.g.* for a button or slider), while multiple processors can be active and prioritised for a UI component. Propagation of gesture events through these layers makes use of the observer pattern. Another abstraction bundles common actions for UI components based on gesture streams. This makes component actions reusable. For example, the action triggered by a drag gesture leads to the translation of a UI component. The package `Event` in figure 6.7 outlines the class diagram of event related classes for component manipulation. Input events (class `InputEvent`) have a target UI component associated by the underlying event system. Depending on the processor (class `AbstractInputProcessor`) and action (class `AbstractAction`) that a UI component registered (method `Component::registerGestureProcessor`), these input events are converted in a high level gesture event (*e.g.* DragEvent) which are then put into effect by the action processor (*e.g.* translating the targeted UI component).

· ◇ ·

Component-wise gesture recognition relies on the internal scene graph in the presentation layer in order to project touch coordinates onto graphical elements for picking (hit detection and setting the target component). Rendering the scene graph and, accordingly, UI components is facilitated with Processing [F+07], a framework for creative computing. To maintain compatibility with recent versions of Processing, make use of new rendering features and facilitate deployment of applications, large parts of the rendering backend (based on OpenGL) have been subsequently rewritten

by the present author [5].

[5] Shortly before publication of this thesis a new revision of MT4j has been published. It incorporated similar updates to the core infrastructure based on the presented implementation [Mat15].

· ◇ ·

The component structure is hierarchic and recursive, it directly represents the tree-structure of the scene graph. In this way, every component can act as a container for child components which live in a coordinate system relative to their parent. Thus, transformations (translation, rotation) of parent components are re-applied to their children. Figure 6.7, package `SceneGraph`, shows the related class structure. The class hierarchy for the scene graph is divided into visible components (inherited from Shape) which can be based on three dimensional mesh data and 2.5 dimensional polygons for generic UI components found in common windowing toolkits. A canvas object is the root node of a scene graph and therefore part of every scene. Multiple canvases can be active in a root scene and therefore act similar to a window in WIMP toolkits.

· ◇ ·

For convenient expression of component relationships and styling of components, a DSL has been developed. Based on common practice in Scala tool-kits, companion objects (equivalent to the factory pattern) have been provided for most UI components.

· ◇ ·

The integration with the FRP framework is provided in the presentation layer as adapters for component properties (fig. 6.7, package `SceneGraph`). These are represented as bidirectional nodes in Reakt such that internal updates from MT4j or Java trigger Reakt events and vice-versa, messages lead to the modification of the Java property. In this way, transformations in the scene, interactions and styling of a component can also be expressed in a reactive way.

6.3 Audio Synthesis

There are several environments to perform and control real-time audio synthesis. Apart from the data flow languages presented earlier, other well known systems are Csound (a Music-N language, cp. [Bou+00]), ChucK [W+03] and SuperCollider (cp. [WCC11]). These rely on textural representations of musical and synthesis related processes. In case of the latter two systems, an own general purpose programming language is provided that embed these processes. They are directed towards interactive programming and feature domain specific language constructs such as strong timing among the more conventional paradigms of object oriented and functional programming. With respect to audio synthesis, especially Csound and SuperCollider provide an extensive list of signal generators and modifiers as well as constructs to abstract from the audio synthesis and control of synthesis parameters. Both environments provide various bindings to other programming languages and thus facilitate integration in existing applications. For the application framework, SuperCollider was employed since a language binding for Scala exists. It provides an extensive DSL to mimic idioms known from the SuperCollider programming language *sclang* and is also feature complete with respect to synthesis constructs that can be expressed in comparison to sclang. The language binding *ScalaCollider* was developed by Hanns Rutz as part of his PhD thesis [Rut10]. Unlike the other environments mentioned, SuperCollider itself is divided into the programming language part (sclang) and a synthesis part in form of an

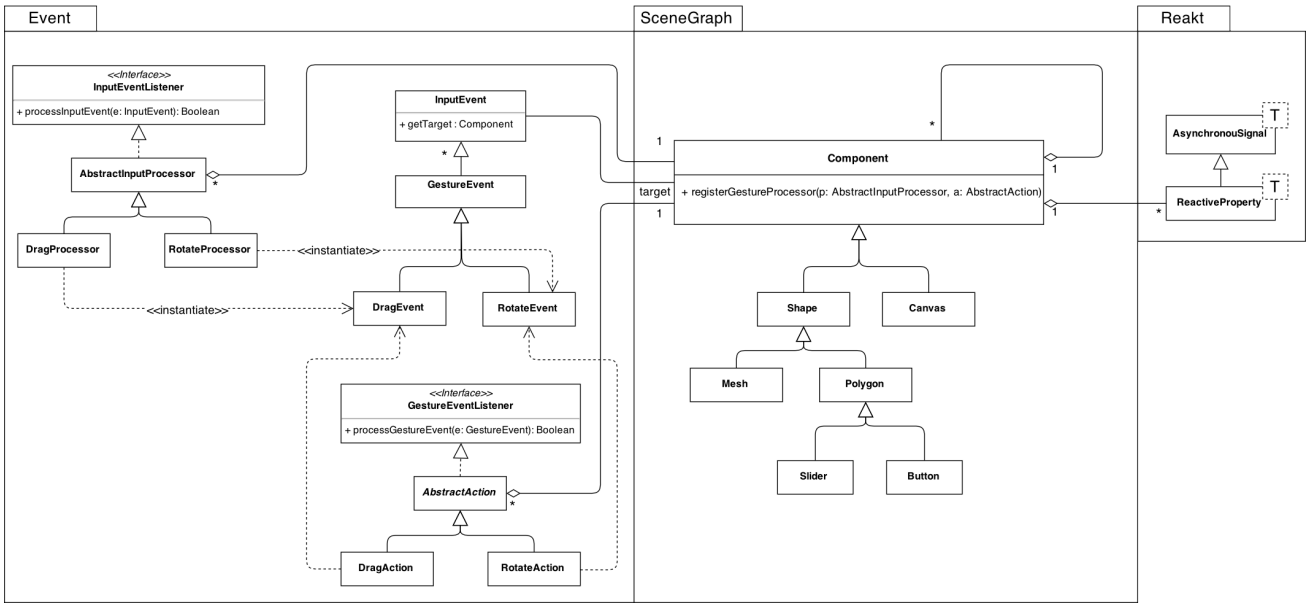


Figure 6.7: Simplified UML diagram of the component structure in MT4j, including event abstractions and processors (package `Event`), components in the scene graph (package `SceneGraph`) as well as adapter classes for Reakt (package `Reakt`)

external audio server (scsynth). Communication between the two is performed using the OSC protocol. Language bindings therefore wrap this communication and provide object mirroring of the remote objects in the scsynth process. Thus, integrating scsynth into an application can be done natively without foreign language interfaces. Current bindings include Clojure, Haskell, Ruby and JavaScript [Rut10].

· ◇ ·

Common with Music-N languages, SuperCollider abstracts the building blocks of a synthesis algorithm as Unit Generator (Ugen, more details are available in section 8.1) which can be combined to form a signal flow. The set of abstractions is therefore very similar to data flow programming. A signal flow can be grouped into a synthesizer (or effect) and Ugens can be defined for external control. Synthesizers can be instantiated and destroyed in real time. Furthermore, scsynth provides a bus system for audio and control data with various methods to route singals to and from it. Similar to Reakt, ScalaCollider uses implicit type conversions to transform expressions into Ugens or synthesis graph structures. In combination with companion objects and operator overloading this allows to express synthesis algorithms similar to sclang without added verbosity.

· ◇ ·

For integration with Reakt, a thin wrapper was added that attaches to each synthesizer instance in Scala an input node for parameter control and an output node that emits the current loudness of the synthesized signal sampled at low frequency. This involves bidirectional communication with scsynth. The following illustrates how UI components can be used to control a synthesizer.

```
1 val mySynthDefinition = SynthDef("Amplitude Modulated Sine"){
2         val frequency     = "freq".kr(440)
3         val outputSignal  = SinOsc.ar(frequency)*Saw.ar(0.01)
4
5         AudioServer attach outputSignal
```

```
6  }
7
8  val mySynth = mySynthDefinition.play
9
10 aRectangle = Rectangle(100,100)
11 aSlider    = Slider(0,10000)
12
13 aRectangle.fillColor <~ mySynth.amp.map( x => Color(x))
14 mySynth.parameters <~ aSlider.value.map(x => ("freq" -> x))
```

In the synthesizer definition (lines $1-6$) a simple sine wave is amplitude modulated (cp. section 8.6.2) with a sawtooth wave. The output signal is registered to the audio backend in line 5 which routes the audio output of the synthesizer to a standard audio output bus and sets up the mechanism for feedback of signal loudness from scsynth. Line 8 instantiates the synthesizer on the audio server and creates a mirrored object in the Scala runtime. The amplitude output of the synthesizer instance is then mapped to the colour of a rectangle UI component. Line 2 created a named control to bind a name for a control Ugen on the scsynth side so it can be controlled remotely. Tuples that contain the name of a control along with value can be sent to the server to update the respective control Ugen. In this example the slider value is simply mapped to such a tuple which is then sent to the generic synthesizer parameter control node in line 14.

6.4 I/O Abstractions

For interoperability with other applications outside of the framework, thin adaptor classes have been written that encapsulate MIDI and OSC communication in Reakt nodes based on implementations in the Java standard library and, respectively, in the ScalaOSC library [Rut15]. Furthermore, the protocol events themselves have been mapped to Scala case-classes such that the language provided pattern matching mechanisms can be used to conveniently convert event streams.




Part

II

Timbre




Chapter 7

Introduction to Timbre

"One of the most striking paradoxes concerning timbre is that when we knew less about it, it didn't pose much of a problem ..."

Philippe Manoury, 1991

Chapter Abstract

In this chapter I will discuss the role of timbre (also known as *sound colour*) in music composition and perception and further show its ramifications on collaboration and identity finding. I will highlight its importance as a compositional building block, especially for the context of electronically generated music. The presented rationale also leads to the conclusion that timbre has a significant role in collaborative music making.

The meaning of timbre is widely given by a non-definition similar to the one created by the American Standards Association [Ass60]: *"[Timbre is] that attribute of sensation in terms of which a listener can judge that two sounds having the same loudness and pitch are dissimilar".*

· ◇ ·

In this regard timbre is used to differentiate perceptual characteristics of tone quality as pointed out by [McA99, p. 85] as *"encompassing a complex set of auditory attributes as well as a plethora of auditory and musical issues"*. It is therefore the synthesis of multiple perceptual dimensions, *"the absence and presence of different properties of the sound, the perceptual weight of which is still in many ways unclear"* [DP97]. It should be highlighted that timbre is by no means independent of pitch and loudness but, on the other hand, it can exist for non-pitched sounds.

7.1 On the Importance of Timbre

Recent findings conclude that timbre is *perceptually more salient* than pitch [Hol11; Hur01; Pat08] and thus has a major impact on the creation, performance and reception of music. In the context of electronic music, timbre has a unique position since the breadth of sounds that can be electronically generated ranges from the *"musical and nonmusical, representational and*

abstract, familiar and arcane" [Dem10, p. 1]. Joanna Demers summarizes the implications of this with respect to the aforementioned development of electronic music in the following:

> " *Until electricity became available, music was generally produced with a relatively small choice of instruments and vocal types. Timbre, [...], has thus been one of the most stable elements of musical production; in other words, non-electronic music is identifiable as music (and identifiable in a very short period of time) on the basis of its timbres.* " [Dem10, p. 2]

Apart from the subsequent historical discourse about the identity of electronically created music (c.p. musique concrète), the manipulation of timbre and its role as musical object is now a central building block of this kind of music. Thus, it defies several organizational principles of classic Western music [Wha99].

7.2 Musical organisation

In many ways, timbral qualities constitute a central building block of music. From the perspective of a composer, timbre is integral to making voice leading effective [Hur01]. From a social perspective, in the setting of musicians playing in a band, timbre forms a significant part of the musician's assessment of *musical identity* since sound is the primary medium of interaction with the other musicians as pointed out by [Cot04] and especially [Hol11] as *"[v]arying timbre is one of the principal ways through which performers communicate musical structure, ideas, emotions and musical personality"*. With respect to music composition and the musical organization, in [Hur01] it was shown that *"established perceptual principles are able to account for the majority of traditional rules in Western voice-leading [...] the power of these perceptual principles was evident in the fact that non-traditional rules derived from the principles were also found to be consistent with actual musical practice"*. This means that methods such as drawing the listener's attention to a particular sound against a background of other sounds, setting voices in contrast or conveying polyphonic music are to a large part motivated and facilitated by the (differentiating) timbral characteristics of the used instruments. Thus, for listeners, it induces expectations towards the flow of a piece [Pat08]. The major obstacle is that for a thorough analysis of tone color, a conclusive foundation is still lacking from the viewpoint of music theory or musicology. This perplexing nature of timbre is expressed by Robert Cogan: *"tone color is perhaps the most paradoxical of music's parameters. The paradox lies in the contrast its direct communicative power and the historical inability to grasp it critically or analytically"* (cited from [Bla12]). The following issues are related to this.

7.2.1 Notation

The Western notational system does not specifically refer to timbral characteristics of of musical building blocks (except approximate articulations for certain instruments). Thus, it is simply not part of the classic Western viewpoint on musical structure. This is also reason among others why Electro-Acoustic composers (*e.g.* Xenakis, Varèse, Stockhausen) develop(ed) their own, personalized, graphical score systems. However, there are cases in other cultures where Timbral Listening (pitch is subordinate to timbre) has a tradition for certain instruments. For example, the Shakuhachi music in Japan also has a notational system to indicate timbral variations [Ber69].

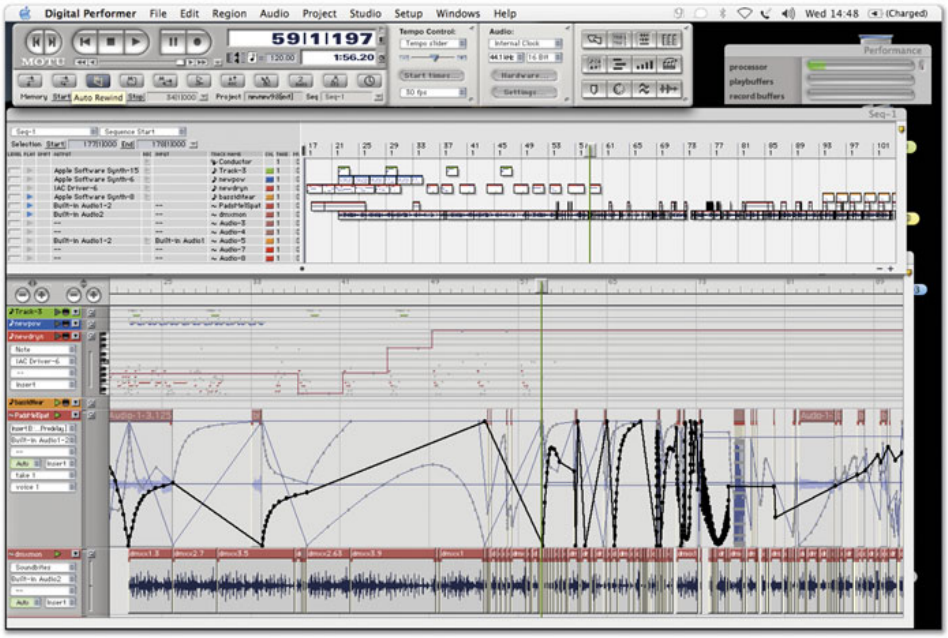


Figure 7.1: Screen-shot of an arrangement session in the DAW *Digital Performer*. The arrangement is a song of the electronic music duo Autechre (circa 2004, taken from [Tin04]). The timeline at the bottom shows the automation of various parameters of some synthesis method (undocumented).

In the practical reality of modern studio music productions, timbre modulations are represented in various, abundant, ways in audio processing environments. The most common technique is automation of parameters using a timeline and breakpoint operations (cp. figure 7.1). However, this representation is a purely technical one, describing solely the parameter development over time, there is no information about perceptual qualities of the sound produced. Thus, this form of composing simply does not fit into a classic notational system that may make use of extended articulations as (if we stick to the example figure 7.1) there is no agreeable set of defined articulations since it is highly dependent on the synthesis method and the number of sensible parameter configurations is generally vast. Even more, since music production results in an sound artefact(a recording) rather than performance instructions (a score), musicologists have little access to deconstruct such works for analysis. Interestingly, in 1936 Edgar Varèse made similar conclusions, which have close resemblance to the line of argumentation regarding the music studio as an instrument (cp. section 2.3) and which predict the use of automation in DAWs:

> " *I am sure that the time will come when the composer, after he has graphically realized his score, will see this score automatically put on a machine which will faithfully transmit the musical content to the listener. As frequencies and new rhythms will have to be indicated on the score, our actual notation will be inadequate. The new notation will probably be seismographic. And here it is curious to note that at the beginning of two eras, the Mediaeval primitive and our own primitive era (for we are at a new primitive stage in music today) we are faced with an identical problem: the problem of finding graphic symbols for the transposition of the composer's thought into sound. At a distance of more than a thousand years we have this analogy: our still primitive electrical instruments find it necessary to abandon staff notation and to use a kind of seismographic writing much like the early ideographic writing originally used for the voice before the development of staff notation. Formerly the curves of the musical line indicated the melodic fluctuations of the voice, today the machine-instrument requires precise design indications* " [VW66]

7.2.2 Analysis

There are no objective analysis tools to describe timbre. Although spectrogram analysis is able to explain the structure of sounds (e.g. overtone series), it does not necessarily account for their perception or represent isolated timbral parameters (see [Zat05] for a discussion of recent research). Another issue is that they merge several musical domains into one representation (microstructure, macrostructure, timbre) and that there are technical or mathematical limitations to the time and frequency resolution of the representations (i.e. Short-Time Fourier Transform) that make the analysis of certain perceptual effects infeasible. Because of the representational deficiencies of the spectrogram, usually higher level feature analyses are preferred (see section 8.7). However, comprehensive analysis relies on a multitude of (itself potentially high dimensional) feature sets. Thus, the high dimensionality of the resulting data makes further interpretation necessary as the various timbres themselves are in many ways represented as combination of features. Chapter 9 discusses this issue in more detail and shows an approach how machine learning can be used to generate such an ”interpretative layer”.

7.2.3 Language

The lack of common, agreeable language to describe timbral aspects impedes, even obfuscates, the scientific discourse. Because of this, timbre is described phenomenologically by analogy, metaphor [Bla12] or reference as discussed in section 2.4 and 7.3. In the latter, it is furthermore argued that metaphors and semiotics of timbres are indeed a product of communities and their meaning building.

· ◇ ·

However, a counter argument is that there exist agreeable *semantic* timbral dimensions. In a study conducted by Disley *et al.* [DHH06], musicians were asked to describe the timbre of samples from twelve classical western instruments by evaluating the appropriateness of adjectives from a predefined vocabulary. This resulted in a reduced set of descriptive adjectives along four perceptual dimensions since several words did not explain the variation of the data set well or had limited agreement. In a similar study involving musicians, non-linear correlations between a predefined vocabulary and a set of audio samples that included 8 synthesized tones among acoustic ones were examined [Zac+12]. The explaining perceptual dimensions are related to the description of *luminance, texture* and *mass* of a sound object. A subsequent study also indicates that these are language-independent [ZPR14]. In this way, one could argue that the existence of such semantic timbral dimensions indicates that linguistic concepts can be used to aid the specification of timbres for the creative use-case. However, Zacharakis *et al.* point out that:

> “ *The mapping between a semantic and a perceptual timbre space must be complex and partial since not all perceivable attributes of sound can be adequately verbalized, and also because verbalization might be a product of conditioning.* ” [ZPR14]

This is one of the reasons why an approach of a perceptually informed audio synthesis that omits linguistic processing and references has been devised in chapter 9.

· ◇ ·

To summarize, because of these issues, coherently analysing modern electronic music, thus music that is intrinsically composed of timbral manipulations of sound objects, is hard to realize. Regarding this thesis, certain assumptions are being made that are therefore hard to fully corroborate scientifically.

7.3 Semiotics, Metaphor & Imagination

The semiotics of music have always been a topic of hot debate in which timbre plays an important role for the performative interpretation of a musical score. Since differences in timbre can more easily distinguished than differences in pitch [Hur01] (e.g. humans can effortlessly identify instruments within a mixed auditory scene), timbre as a form-bearing dimension for a composer *"should also be central to interpretative decision-making and projection in performance"* and further as *"as a vehicle for communicating structural concepts and interpretative imagination"* as argued in [Hol11]. The qualities of timbre that transport metaphors and structural concepts underlie, of course, cultural and social principles wherein this implicates that metaphors and concepts of certain musics may be concealed for the unrelated listener. If we define timbre as *sound as it is perceived* and admit that pitch and loudness are qualities of timbre, then the listeners of western classical music would have strong grounds to argue that pitch is the principal dimension of the timbral quality. For listeners of electronic music, this principal dimension may be a certain parameter of the synthesizer or a timbral effect. Taking the example from McLean which tries to exemplify the vocabulary used for metaphors in the specific genre of Acid House which became popular in the late 1980s and centers around the timbre manipulation of electronic instruments:

> *"An exemplar is Josh Wink's 1995 track Higher state of consciousness, making heavy use of envelope modulation [of the filter cut-off frequency] and resonance parameters to slowly bring the music to an euphoric peak. The metaphors here are ENVELOPE MODULATION IS UP and RESONANCE IS UP, related to the metaphor of CONSCIOUSNESS IS UP in the title. [...] Metaphors such as RESONANCE IS UP, ENVELOPE MODULATION IS UP and metonyms such as DISTORTION IS LOUDNESS have particular meaning within acid house music culture"* [McL11]

In a similar way the quality of the singing voice in popular and experimental music is used to project different gender locations, images or states of emotion e.g. by rasping tones, head tones, growling or screaming [Cot04]. Related to this is also the standardization of certain sonorities in and around recording studios, e.g. the Nashville sound, Phil Spector's 'wall of sound' [PB04]. Thus, one can imagine that these contexts itself can be referenced by the respective sonorities.

· ◇ ·

In [Por04] Thomas Porcello conducted a study that compared the verbal communication in a recording studio between an intermediate and a professional (drum tuning student and a producer) with the interaction between two professionals (recording engineer and the same producer). The matter of both conversations is largely the same, the approach to manifest a certain sound that the producer envisions. These interactions contrast rather strongly. The conversation between the student and producer is dominated by a highly interpretive and ultimately, inexpedient discourse that makes use of lexical onomatopoesis (words that bear partial acoustic resemblance to sounds), pure metaphor (words that only describe timbral

characteristics), association (e.g. similar recordings) and evaluation. On the other hand, the conversation between the professionals is characterized by an extreme brevity, *"complete avoidance of hesitations or overlapping speech"*, with *"the conversational economy that accrues to linking sounds to pieces of recording technology"* (ibid) without further references. It becomes clear that a lexical overlap between the involved parties is not sufficient since shared knowledge is a must to use similar linguistic resources and therefore to communicate intended sonorities verbally. As pointed out in chapter 5, acquiring this knowledge is not feasible, therefore, organizing sonorities by language (e.g. tags, keywords) may not be sufficient.

· ◇ ·

One viewpoint on musical semantics is the relation of timbre to the physical causality [Bea07, p. 299-313]. We experience the importance of this relation every day as we bind certain sound objects to physical processes and their visual cues. In reverse, this experience further allows us to derive physical characteristics from a sound object and consequently its meaning in the form of *causal semantics* [Lem10]. The sound of footsteps in a hallway for example can give cues about the person causing the sounds, such as his/her figure, the kind of shoes and therefore gender, the material of the hallway's floor or the size of the hallway. Regarding causual semantics in a musical context Leman ([Lem10]) argues that *"it could be stated that causal semantics in music implies the activity of agents that might have caused the sonic (musical) patterns, or that the sonic patterns are themselves perceived as agents involved in causation. Rather than being imitations of real sound-sources [...] the focus is here on agency, that is, on the action or the intervention that produces a particular effect, however, concrete or abstract it may be"*. In this sense, the meaning formation of causality involves the imagination and representation of the sound source in the action-oriented ontology of the listener, the subjective viewpoint towards a biased physical environment [Lem10; Lem+07]. However, as indicated in chapter 2, electronic music again takes on a special position since such causality can be invalidated or imitated leading to a paradox outcome:

> " *One of the most profound differences between acoustic and electronically produced sound is the capability of the latter to invalidate the causal logic that has bound together productive, acoustic and perceptual domains since the beginning of auditory time. Electronic sound requires a redefinition of "source" and "production" [...] it deprives the perceptual domain of at least a part of its traditional role in processing musical timbre. And yet [...] modern listeners seem to roll along as usual, fully unaware of an occasional incongruity.* " [GP05, p. 164]

The reversal of the roles laid out thus far regarding musical semantics is on the other hand observable in the context of listener and performer interaction, as imagery is communicated or implied. In [Vin+04], it is highlighted that visual information communicating gestures, and therefore giving causal cues, of a performer playing greatly improves the music performance perception of structural and emotional information. Similar mechanisms can be found in the collaborative case for ensembles [SB09]. To summarize, timbre, communicating musical intent and embodiment are tightly linked, they form a symbioses of meaning transportation mediated via sound & visual cues by human action and interaction with musical instrument or additional means offered by the CSCM environment.

· ◇ ·

From the viewpoint of music composition, the referential nature of musical semantics (the foundation in musical metaphor) with the projection into the music performance forms an imaginary space in which *"the physical*

production of tone is guided by imagery [...] the desire and ability to create a particular sound at any one moment is at the heart of expert performance" [Hol11]. This is further summarized by Cottrell:

> " *[T]he ability to internalise sound and to develop a rich imaginative and comprehensive internal representation, not only of what has been previously experienced, but also for imaging new, innovative compositions and interpretations, would appear to be an essential feature of a musician's thinking* " [Cot04]

7.4 Conclusion

However, as argued before, to realize the endogenously motivated act of creativity requires expert-level abilities to shape timbre for a performer. For novices this barrier should be alleviated. Extending this imagery space is often admitted to be an exogenous act of creativity [Hol11] for composers and performers in which e.g. playing techniques are extended evoke new musical "vocabulary" without initial presumptions on a certain compositional form. This "interest in sound for its own sake" has also been mentioned previously - in the context of studio/home recording in section 2.4. In this way this experimenting approach is to a large degree the source for creating and coining new musical vocabulary, metaphors and their semantics and therefore extending and transforming the actors imagery space [Bai07]. To support such experimentation in an unbiased environment is a key to transport new musical language in a collaborative scenario and to allow novices to learn by observation of action versus by tutorial (from one another or more experienced collaborators) similar to *Knowledge Building Communities* [BJT05].

· ◇ ·

Having established timbre as medium for communicating musical ideas and concepts, it can be derived that an application in the field of CSCW for music composition or performance should afford the *unbiased* means of exploring (the dynamic evolution) of timbre. Any form of bias would otherwise presume a commonly located and shared imagery space of the collaborators, therefore shape expectations on the creative output (e.g. style/genres, modes of expression). This may counter timbre as a vehicle for meaningful creative discurse. Of course, practically some kind of bias will always exist in the form of user interface assumptions or chosen methods of audio synthesis.

Chapter 8

Digital Audio Effects

Chapter Abstract

This chapter is intended to give a review of common methods that are used to synthesize audio signals digitally. Most of these have been applied in the prototypical implementations in some form or another. The information provided here is intended to corroborate the application of the presented methods.

In general, a synthesis describes the combination of entities to form something new and, fundamentally in this context, it describes the creation of something by artificial means. While in the context of this thesis, one could speak of the synthesis of music at large, the general consensus of the term synthesis *in music* solely describes the (artificial) creation of perceptual *attributes of sound*. Thus, from a purely technical viewpoint, such an *audio* synthesis involves processes that control the spectral qualities of a sound and in this way, some form of signal processing in some domain (*e.g.* digital, analogue, electro-mechanical).

· ◇ ·

However, this technical process oriented viewpoint on the creation of perceptual attributes of sounds does evidently not admit other viewpoints that are of relevance for the musical context as a whole. Indeed, as pointed out in [ZA02, ch. 1] the communities involved in creating music are manifold: Digital Signal Processing (DSP) programmers, sound engineers, sound designers, electro-acoustic music composers, performers using various digital or augmented or extended acoustic instruments and musicologists. And naturally, within such communities different classifications of audio synthesis methods occur that are logically relevant for the given community but meaningless to another one:

> "*For instance, signal-processing techniques are rarely presented according to the perceptual features that are modified, but rather according to acoustical dimensions. Conversely, composers usually rely on perceptual or cognitive features rather than acoustical dimensions, and even less on signal-processing aspects.*" [ZA02, p. 3]

For this chapter, I will base the discussion on all presented synthesis methods in the signal processing or engineering domain. For the reasoning why these have been applied in certain prototypical implementations, I will, however, show their referential qualities in terms of compositional intent

(how and where they are used). It should be mentioned that the perceptual attributes that are associated with a processing method are not necessarily fixed. For example, extreme dynamic processing can also change the timbre of a sound and, similarly, so does extreme pitch processing.

8.1 Unit Generators & Modifiers

One of the most influential developments in computer music systems was the MUSIC-I system written by Max Mathews at Bell Labs in 1957 [Cd07, p. 58]. It introduced as a core concept the abstraction of audio synthesis processes into a graph of generic building blocks, called Unit Generators (Ugen). A Ugen abstracts the low level computation of a DSP algorithm as an atomic black box, which can be interfaced with input and output connectors. Input connectors thus provide a method to change internal parameters of the computation while output connectors emit the results. For the instrument designer and musician this meant that a synthesis method can be represented as signal flow between ready made modules where the specifics of signal generation have been abstracted away. The connections between Ugens can carry static or time-varying signals and, generally, they are processed at different sampling rates for reasons of efficiency. Audio-rate is therefore a processing that usually happens at the sampling rate of the audio interface while control-rate has a sampling rate one or two orders lower for processes that do not take place in the audible spectrum. Since this concept supports encapsulation and polymorphism it works well with object-oriented and flow-based programming paradigms [Kle+13, p. 20], and thus it still forms the basis of many more modern computer music systems such as *Csound* [Bou+00], *SuperCollider* [WCC11], *Reaktor* [Nat96] or *PureData* [Puc+96]. Throughout this thesis, most synthesis methods will be represented as a network of unit generators.

· ◇ ·

An oscillator is, for example, a Ugen which runs at audio-rate as it outputs a periodic waveform in the audible spectrum. It usually has input ports for frequency, initial phase and amplitude (cp. fig. 8.15). As most sounds we hear do not consist of a static spectrum of equal amplitude but rather develop over time [DJ97, p. 63], there are Ugens that have the sole purpose of modifying parameters in the synthesis graph; these are the ones that run at control-rate. Periodic changes are usually accomplished with a low frequency oscillator (LFO) which oscillates at musical tempi or longer periods of time. Many synthesis designs make use of envelope generators (EGs) that generate functions of time of varying complexity. Envelopes consist of several consecutive segments that are parameterized in terms of rate (the duration of the segment) and levels (the value a segment outputs). The envelope generation is triggered by an external event, such as a key press on a keyboard. Most commonly, an envelope is used to shape the temporal development of the amplitude of a generated signal. In the simplest case this involves just an attack and a decay segment, where attack describes the time until the peak amplitude is reached and, vice-versa, decay how long it takes for the signal to fade back to zero amplitude. Plucked instruments, for example, have such an evolution of amplitude since after the excitation the instrument will start resonating up to a peak amplitude (attack phase) until the damped oscillation will eventually cease (decay phase). The typical amplitude of bowed or wind instruments involves more segments to be since the tone can be sustained. Here, we have an attack and decay phase followed by a sustain and release phase. The parameter for sustain

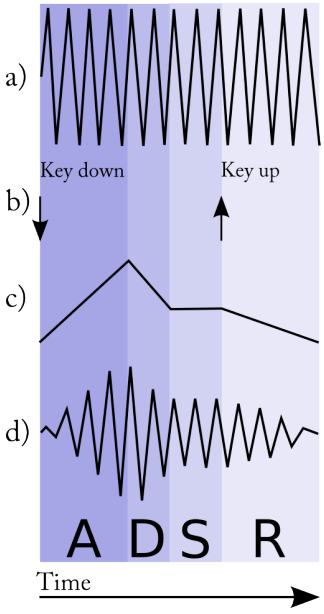


Figure 8.1: Amplitude modulation of a signal (a) by an ADSR (c) envelope that is triggered by a keyboard input (b) with its corresponding result (d)

is a (amplitude) level to which the decay segment fades to from the peak amplitude. In this case release is the segment describing how long it takes for the signal to die. These ADSR (attack, decay, sustain, release) envelopes are common and found in various synthesizer designs, since they give more control over how the sound is produced as the envelope remains in the sustain state for as long as a key is pressed. Figure 8.1 gives an example of how an amplitude of a sound is shaped with an ADSR envelope. Additional parameters include the control over the shape of the segments, such as linear, exponential or logarithmic slopes. Especially for the attack (and decay) phase this is crucial as summarized by Benson:

> " *It was not really understood properly until the middle of the twentieth century, when electronic synthesis was taking its first tentative steps, that the attack portion of a note is the most vital to the human ear in identifying the instrument. The transients at the beginning are much more different from one instrument to another than the steady part of the note.* "
> [Ben08, p. 257]

As will be shown, envelopes do not only have use for shaping the general amplitude of a signal but also its spectral components.

8.2 Additive Synthesis

The core concept of additive synthesis relates directly to the Fourier Series in that it makes use of the principle that all periodic functions can be formed by the summation of sinusoidal functions. More rigorously, one can express the time dependent periodic waveform $y(t) = y(t+P)$ with period P as sum of K sinusoidal components (partials):

$$y(t) = \sum_{k=1}^{K} a_k \cos(2\pi k f_0 t + \phi_k) \tag{8.1}$$

with a_k being the amplitude, $k f_0$ the frequency relative to a fundamental frequency f_0 and ϕ_k the phase offset, respectively to a k-th harmonic. The fundamental frequency f_0 corresponds usually to a musical frequency (*e. g.* an A, 440Hz). This is the most basic form of additive synthesis since only static waveforms can be created this way. In general, the partials for $k = 0$, forming the DC offset, and $k \gg 100$ are not included as they are inaudible. As K is usually fixed, the only property that affects the resulting waveforms is the amplitude weighting a_k. If k is restricted to integer values, $k \in \mathcal{Z}$, then only harmonic spectra can be created. In this form, the harmonic additive synthesis, it is possible to synthesize most basic musical waveforms (including filter phenomena), some of them which are again the building blocks for subtractive synthesis (cp. fig. 8.2).

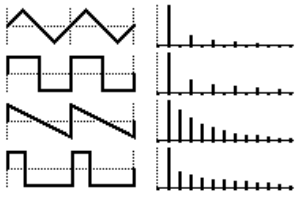


Figure 8.2: Simple waveforms that use an integer-relationship for partials and their spectra (taken from [Szw14])

Acoustic instruments whose overtone series (partials) is harmonic are, for example, the oboe, clarinet, violin and cello [Bea07, p. 18-30]. However, the spectra of certain acoustic timbres are to a large degree inharmonic, this includes bells (cp. fig. 8.3) or mallets, for example. In this case, for $k \in \mathcal{R}$, the above formula (8.1) , is usually modified to the form of:

$$y(t) = \sum_{k=1}^{K} a_k \cos(2\pi f_k t + \phi_k) \tag{8.2}$$

where f_k is the frequency of the k-th partial. So far, only periodic waveforms can be synthesized; acoustic phenomena, of course, also consist of a temporal development of the spectrum. In this regard, the amplitude

weights a_k and the frequencies f_k have to be time dependent:

$$y(t) = \sum_{k=1}^{K} a_k(t) \cos\left(2\pi \int_0^t f_k(u)\, du + \phi_k\right) \tag{8.3}$$

It becomes clear that the quality of this synthesis method to convey a certain timbre and its temporal evolution is dependent on the number of partials used (*e.g.* for inharmonic spectra) and the quality (and amount) of control data for $a_k(t)$ and $f_k(t)$ [Roa96, p. 140f]. [1] While the former is nowadays a negligible issue as band-limited sine oscillators can be effectively computed using vectorization schemes to the thousands on the CPU or up to a million on the GPU [SVS10] to even allow for polyphonic sounds, the latter is conceptually a difficult task. This is because this synthesis form is to its entirety dependent on these control parameters such that pinpointing spectral phenomena requires a careful construction of high dimensional time series data.

[1] To get a perspective on the amount of computing power and data is needed, one can look at two trivialized examples (taken from [Roa96, p. 142f]): for sounds with 24 partials and a polyphony of 16 at 48kHz sampling rate about $1.8 \cdot 10^8$ samples per second are needed; similarly, with a piece containing 10^4 events, around $2.4 \cdot 10^5$ amplitude and frequency envelopes are necessary. More modern additive synthesizers, such as the Native Instruments RAZOR, use up to 320 partials.

· ◇ ·

How is this control data generated? Especially for early compositions, this was entered manually, possibly using combinations of mapping data from different domains (Charles Dodge, *Earth's Magnetic Field*, 1970), programs that constrain the spectral microstructure (John Chowning, *Stria*, 1977), interactive composition systems, using high level descriptors such as the POD system (stochastic generation of events) [Tru77] or low level construction of envelopes such as Xenakis' UPIC [MSR93] [Xen92, p. 329]. The comparatively well-known UPIC system was a significant development insofar as it allowed the construction of the spectrum by drawing the temporal development of the sinusoidals using a specialized tablet/pen digitizer system with a dedicated realtime synthesis engine. This metaphor of *drawing* the spectral microstructure of a piece has been adopted in several modern applications such as MetaSynth [UI 99]. But, interestingly, this approach has been developed before in the context of photoelectronic instruments, such as the ANS Synthesizer by Evgeny Murzin (developed between 1938 and 1958 and used in *e.g.* Andrei Tarkovsky's movies Solaris, The Mirror, Stalker) or the Oramics by Daphne Oram (1962).

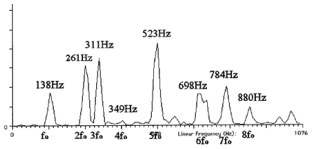


Figure 8.3: Spectrum of a bell sound showing a characteristic inharmonic spectrum (taken from [Dir07])

· ◇ ·

More modern approaches to generate the control data rely on high level transformations to mimic and alter spectral phenomena. In most cases, this is done by first analyzing recordings of sounds of interest to generate their sinusoidal multi-component representation and then modifying these by means of a pipeline of spectral operators. Research regarding such Synthesis by Analysis led to several extensions of the basic time-variant additive synthesis, most notably is the Spectral Modelling Synthesis [SS90; Ser97] which added a (filtered) noise component to the otherwise non-stochastic output signal. An alternative analysis method has been contributed in [Klü12].

· ◇ ·

Due to its inherent universality, additive synthesis provides a generic framework to synthesize arbitrary sounds without the need to model the original causative sound generating systems. However, as pointed out before, the central issue for synthesizing *realistic* timbres is the generation of control data. The alternative to this generic approach is to make use of specialized synthesis methods and -topologies that specifically model the various necessary acoustic phenomena and instrument behaviours.

· ◇ ·

Apart from the early examples of additive synthesis[2], additive synthesizers were made available commercially with the introduction of digital synthesis technology due to the added precision and lowered costs in comparison to analogue technology. However, there was a comparatively low adoption of additive synthesis outside of academic and experimental music establishments until the arrival of cost effective personal computing. Especially the exploration of different timbres without additional high level controls is more feasible using other synthesis techniques such as Frequency Modulation Synthesis where two parameters are enough to gradually shift between harmonic and inharmonic spectra of varying composition. A comparatively primitive additive synthesizer was implemented for the prototype in chapter 13.

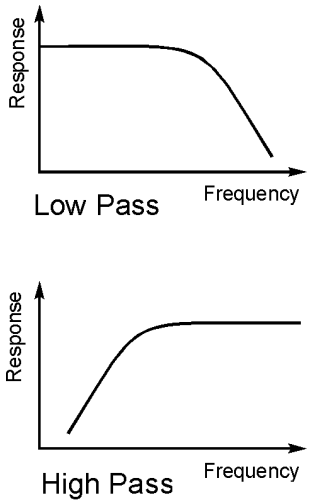


Figure 8.4: Amplitude responses of common filter types (taken from [Poo14])

8.3 Subtractive Synthesis

As the name suggests, subtractive synthesis is antipodal to additive synthesis as a harmonically rich signal is filtered in order to sculpt the desired output spectrum [DJ97, p. 155f]. The source signal can be arbitrary, but generally, certain signals are prevalent:

- Noise and impulse trains: in combination with a formant filter or a more sophisticated vocal tract model, this is commonly the basis for the source-filter model in speech synthesis [HC99, ch. 3]. Impulse trains are used to simulate the mostly periodic glottal excitation that is apparent in voiced sounds (vowels) while the filter mimics their spectral transformations that occur dependent on tongue position and lip protrusion. Unvoiced sounds, on the other hand, such as the fricatives 'f' and 's' have their source in noise and a constriction of the oral cavity. Voiced fricatives, such as 'z' or 'sh' are a mixture of both (ibid).

- Simple, non-sinusoidal waveforms: as previously mentioned, certain acoustic instruments exhibit physical behaviours of excitation that can be, to a simplifying degree, seen as being derivative of primitive periodic waveforms. For bowed instruments, the stick-slip phenomenon of the bow that excites the string is approximately similar to a saw-tooth wave [Ben08, p. 43]. Likewise, for wind instruments that function as a closed pipe (*e.g.* clarinet), the pressure differential created by blowing into it leads to odd modes of vibration. This behaviour can be simulated with a square (or triangle) waveform, as it contains only odd harmonics. For open pipes (*e.g.* recorder) the excitation behaviour is again more similar to a saw-tooth wave. The filter in such cases is generally used for "harmonic enveloping" [DJ97, p. 174] to change the strength of harmonic partials with time. In this way naturally occurring (decaying) resonances of the instrument body and other physical phenomena (*e.g.* when overblowing a recorder) are mimicked.

8.3.1 Filters

Filtering is a linear operation that modifies the spectrum of a signal by changing the amplitudes (and phases) of partials according to their frequency. The related amplitude response (also called magnitude $G(\omega)$ of the complex valued frequency response $H(e^{j\omega T})$) of a filter describes the applied gain. Different filters are usually distinguished by the shape of

their amplitude response. Common filter types are low-pass, high-pass, band-pass and band-reject (cp. 8.4). The cutoff frequency (f_c) is a parameter that defines the boundary in the amplitude response where the filter begins to attenuate partials (transition band) in the spectrum. The stop-band then defines the region where the attenuation is equal (or larger) than that required by the design of the filter and, accordingly, the pass-band is the one where no modification of the partials' amplitudes takes place. In case of a low-pass filter, frequencies below f_c are permitted with little change but spectral components above it will significantly reduced. Conversely, the passband of the high-pass filter is above f_c. Band-pass and band-reject, too have an antipodal relationship. Both can be constructed by connecting a high- & low-pass filter in series. For such filters another characterization is the bandwidth which is a measure of the selectivity of the filter as it defines the width of the passband around f_c. The steepness of the curve in the transition band (called roll-off) is measured as function of logarithmic frequency. In most cases of musical applications, this is given in units of decibels per octave (dB/8ve) or filter order, where a first order filter network is approximately equal to 6 dB/8ve. While the preferred order is entirely dependent on the musical task at hand, most synthesizers (classic and modern) employ filters of order two to four.

· ◇ ·

For certain spectral phenomena it is necessary to emphasize the partials around f_c above a unity gain. Therefore, such resonant filters can not only "carve out" parts of the spectrum of the source signal but also increase the amplitude of frequencies.

8.3.2 Formants

The aforementioned formant filters in speech synthesis is for example such a use-case where the filter is required to create acoustic resonances of the vocal tract. If the terminology of formants is applied to generic resonances outside the frequency range of human speech, it becomes apparent that this is a characterization of spectral phenomena where overtones are dependent on the harmonic structure of the source signal but static in relation to its fundamental frequency. This is a requirement for modelling the acoustic effect of exciting instrument various bodies (strings, winds, percussion). In section 8.6, we will see other methods of creating formant structures which can generate multiple spectral peaks.

8.3.3 Legacy

Even today, subtractive synthesis probably remains the most common form of musical audio generation, presumably, this is because of the comparatively low component and parameter count. This makes the method feasible to implement but also traceable for beginners with regard to parameter setting and the resulting sound. The basic architecture for subtractive synthesis makes use of one or more oscillators producing harmonically rich waveforms that can be mixed together. The signal is then routed to a filter which is followed by an amplifier. Filter and amplifier are in general modulated by envelopes. Figure 8.5 shows the presented structure.

· ◇ ·

The synthesizers in chapters 9 and 13 are based on this architecture in varying degrees.

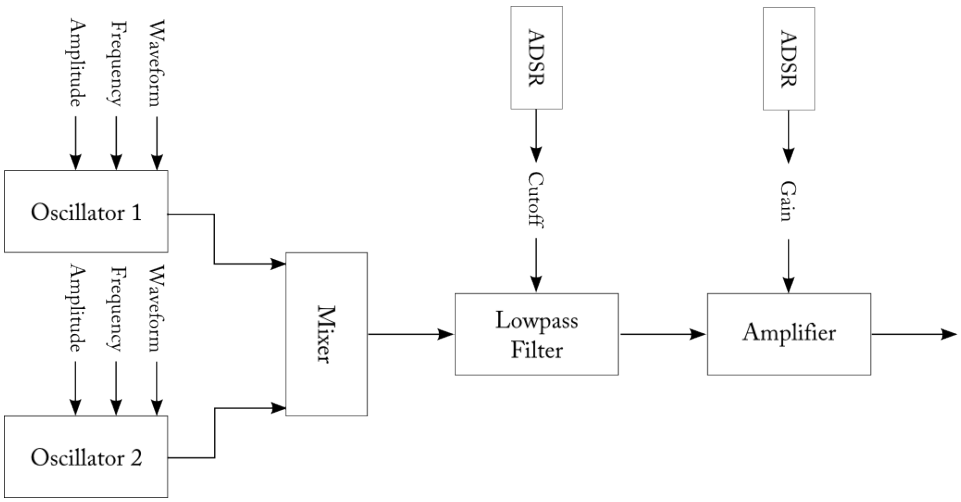


Figure 8.5: Common structure of a subtractive synthesizer.

8.4 Special Filters and Basic Effects

As for now, filters have been only presented in the context of the frequency domain, but they can also be considered in the time domain. This leads to the family of delay based effects that can be used to model spatial phenomena and time varying modulation [ZA02, p. 70f]. In the most simple case such a network simulates a single delay and is thus called Finite Impulse Response (FIR) comb filter. Such a delay has two tuning parameters, the amount of time delay τ and the relative amplitude g of the delayed signal that is mixed back with the original (reference) signal. Expectedly, the effect of the network can only be heard in relation to the reference. As the name implies, a comb filter has the frequency response of a series of regularly spaced spikes, giving the appearance of a comb (cp. 8.6). The parameter g determines the level and reference of attenuation of frequencies at multiples of $1/\tau$: at positive values the filter amplifies all frequencies at these multiples and attenuates the ones that lie in between; at negatives ones, the opposite behaviour holds.

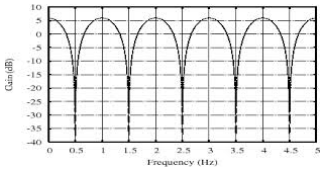


Figure 8.6: The frequency response (linear) of a comb filter (taken from [Smi02, ch. 4])

· ◇ ·

The perceptual effect of this FIR network is dependent on τ, for values $\leq$ 10ms this results in altering the spectral structure (phasing), values between 20ms and 100ms create a doubling effect and values larger than 100ms are typical for echos. As before, this network can be extended, in this case by adding a feedback path. In this case, g is used as a weight for the mixture of the original and the feed-back signal. Thus, the time response of the filter is infinite. Accordingly this is called an Infinite Impulse Response Filter (IIR). The frequencies that are affected by the IIR comb filter are similar to those of its FIR pendant but the gain ratio can be significantly higher and the slope of the peaks change in curvature (the peaks become more spiky). Thus, a comb filter can be used to create harmonic attenuation while the feedback configuration allows to emphasize the spectral peaks as harmonic resonances.

· ◇ ·

A special building block for many artificial reverberation models is the all-pass filter that is in these cases usually based on a specialized comb filter. In theory, all-pass filters have a unity frequency response, they are ”colourless”, but they introduce a modification of the phase content of the signal,

thus, practically, this only holds for single sinusoid input. While they do not model a physical diffuse reflection, a single non-zero input sample can be thought of expanding into an infinite impulse response, which is qualitatively what is desired for artificial reverberation designs. As such, all-pass filters are also called (impulse) diffusers.

· ◇ ·

In most digital signal processing context, the delay element of the filter network is also called a delay line. Its implementation usually makes use of a ring buffer [DJ97, p. 228] that can be extended by an interpolation mechanism to allow non-integer values of the sampling interval in order to be able to tune $1/\tau$ to specific frequencies. Such fractional delay lines (cp. [ZA02, p. 78f]) then can be modulated in delay time to build effects that rely on a specific pitch manipulation. An example is the Doppler effect, where the pitch variation is due to the changing distance between source and listener. This is equivalent of appropriately increasing or decreasing the length of the delay line. Depending on the interpolation scheme the result is similar to altering the speed of playback on a tape machine or vinyl player. Thus the effect of this modulation is a change in pitch of the delayed signal. If this modulation happens periodically, *e.g.* use a low-frequency oscillator to drive the delay parameter, then a vibrato effect is created. For both effects it is necessary to listen to the delayed signal only.

· ◇ ·

The parameters range and depth of the modulation as well as the mix of source and delayed signal give raise to classic effects that are frequently employed in modern audio production. Using short delay lines, a flanger can be produced that to the ear sounds similar to a phaser but produces a more complex harmonic sound as the constructive and destructive interference of mixing the original signal with the time delayed one always creates an exact harmonic series. For a phaser, on the other hand, peaks and troughs are generally not in a harmonic series. Slightly longer delay lines without feedback lead to the design of a chorus effect that gives the impression of multiple recordings or voices of a similar (detuned) signal being stacked on top of each other (such as in string sections). Because of the longer delay time, the interference of phase components is only weak such that the overall timbre of the sound is preserved, yet it is short enough not to be recognized as an echo. In many cases the overall effect of the chorus can be increased by placing multiple instances of these delay lines with non-correlating modulation parameters (to suppress the creation of flanging artefacts) in parallel. This later approach was implemented for the synthesizer in 9.4.1. Cascaded all-pass and comb-filters have also been used for the synthesizers in chapter 13.

8.5 Simple Reverberation

Natural reverberation is the result of sounds reflecting off surfaces as a process of dispersion. This enriching by overlapping the original sound imparts a change of timbre, which is to a large degree due to the dispersion being frequency dependent, thus altering the waveform. Therefore, as a compositional tool, long reverberations with its extensive overlapping are a method to build up dense textures [DJ97, p. 224]. In this context shorter sounds can be transformed into ones that slowly build up in density and then decay over a much longer period of time compared to the original. In most cases, the reverberation is described in the form of an impulse response, which is a representation of the temporal evolution of this

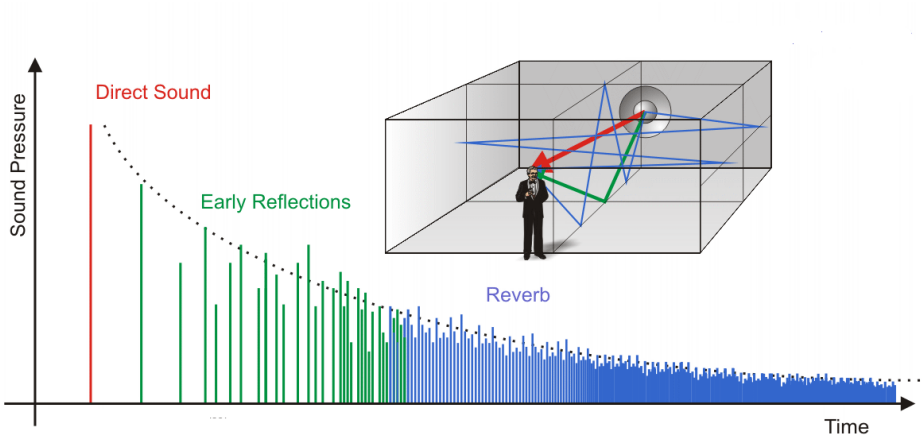


Figure 8.7: The phases of reverberation in an impulse response (taken from [Kar00])

acoustic phenomenon from a static source to a listening position. From an engineering perspective, this can be divided into three consecutive parts, the direct sound reaching the listener first, the early reflections of the first sparse echoes and the late reflections part that is composed of indiscernible individual reflections [ZA02, p. 164]. Figure 8.7 shows these partially overlapping segments of an impulse response. Early reflections are in most cases integrated with the direct sound by the human hearing, in this regard, after the direct sound response a listener does not perceive reflections as individual. An important physical quantity for the perception of the size of a room is the time difference between the direct sound and its first reflection. Most good concert halls have a delay in the 10 to 20 ms range [DJ97, p. 225]. Another measure is the reverberation time, commonly known as RT60, is the amount of time required for a sound to cease to $-60dB$. The choice of the amount of decibels dates back to measurements made by W. Sabine in the late 19th century for a sound to be roughly inaudible, as such it does not necessarily represent how long a listener will be able to hear a reverberation until it dies away. There are many approaches to generate artificial reverberations. Apart from convolving the input sound with a recorded impulse response from a real space [RM95], there are generative methods that may be used. However, for the most general case, these do not rely on complete physically accurate simulation of the acoustic phenomenon since this would be far to expensive (in the order of 10^{10} operations per second) and since the human hearing is unable to perceive the full complexity of it [Smi06, ch. 3]. In this way, most reverberation effect designs are only concerned with simulating what is only audible. The most widely adopted design is the reverberator by Schroeder [SL61] due to its relative simplicity and acceptable fidelity. It makes use of the delay based filters that have been introduced in section 8.4, simple all-pass and comb filters.

· ◇ ·

Since the synthesizers in this thesis, most importantly the one in section 9.4.1, use *Freeverb* by Jezar Wakefield, the general functioning of a Schroeder reverberator (on which *Freeverb* is based on) will be exemplified by means of this implementation. Schroeder proposed that the input signal is first fed into a parallel stack of comb-filters which are then summed into a series of cascaded all-pass filters to efficiently simulate the behaviour of late reverberations [SL61]. As mentioned before, an all-pass filter can be seen as impulse diffuser with a natural decay (exponential). The serial composi-

tion increases the density of this diffusion. To reduce the superimposition of echoes in the impulse responses, the delay lengths are chosen to be coprime. The comb-filters are used to impart the distinctive colouration of early reflections and thereby define the size of the room [Sch62]. In this case the comb-filters have low-pass filters in their feedback loop in order to achieve an effect similar to air absorption, which is frequency dependent and attenuates high partials. Generally, this emphasis of low frequencies is preferred as it is a common characteristic of concert halls (ibid.). The parallel stacking of comb-filters creates dense early reflections with varying spectral attenuation due to the incommensurate lengths of delay lines. Feeding the summed comb-filters into the all-pass stages then results in increasing the number of artificial reflections without re-colouring their output. In a final stage the original signal can be mixed into the output of the effect again to be able to gradually shift the focus of the reverberation into the background again, if desired.

8.6 Abstract Audio Synthesis

Abstract audio synthesis techniques focus on the use of mathematical functions that do not have a direct physical interpretation [Kle+13, p. 13]. However, this does not mean that such methods are unable to mimic the *effects* of physically informed processes as we will see later on. In many cases, abstract synthesis allows to create computationally efficient control over a dynamic spectrum with only a few parameters. If we look back at additive synthesis, we can see that this is exactly antipodal to it as an explicit specification of a spectrum (with all its implications on control data) is circumvented by establishing high level couplings and transfer functions that only implicitly describe the spectral outcome. As such, single parameters of these systems can have a great impact and transform sounds from simple to complex spectra or harmonic to inharmonic ones. One could argue that certain techniques attributed to subtractive synthesis or the application of the delay based filters presented earlier can also be seen as some form of abstract synthesis. The techniques shown in this section have been employed comparatively frequently in the prototypical implementations.

· ◇ ·

As mentioned, certain abstract techniques make use of a coupling between components of a system. A common concept is that of a modulator and a carrier in which the former changes a parameter of the latter usually in the audible spectrum. In most cases modulator and carrier are both oscillators, so that the modulator has direct influence on some parameter of the waveform generation of the carrier. A method of classic synthesizers is, for example, hard sync where the carrier is set to restart the period of its oscillation triggered when the modulator output reaches a certain threshold or phase. In this way modulator and carrier have the same base frequency but because of the introduced discontinuities the output will contain significantly more harmonics depending on the frequency ratio between both oscillators. Another classic method is pulse width modulation where the duty cycle of a square-wave is modulated.

· ◇ ·

Two common and versatile methods are frequency modulation (FM), or respectively phase modulation, and ring modulation (RM). We will take a closer look at these since they have been implemented as sound synthesis techniques in the prototypical applications in various degrees. However, for the sake of brevity some of the subtle details will be omitted.

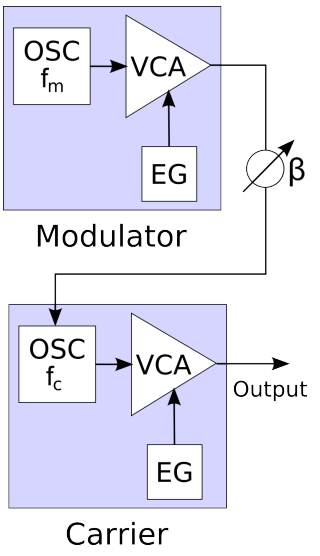


Figure 8.8: The most common configuration of two operator FM synthesis, including envelopes for modulation

8.6.1 Frequency Modulation

There are numerous variations of basic FM synthesis methods, the most prevalent (and simplest) form is the two operator serial configuration as shown in figure 8.8, where the two oscillators generate sinusoids (although different waveforms can be used). The first mathematical analysis and implementation of this in digital form is attributed to John Chowning [Cho77]. However, early implementations of this technique (again in the musical context) date back to the mid-1960s. FM can be thought of a non-linear distortion of the carrier waveform in accordance with the amplitude of the modulating signal. The modulation leads to a deviation of the frequency of the carrier f_c. If the amplitude of the modulator is zero, expectedly no modulation occurs, but if it is above or below zero, f_c is accordingly offset with the deviation β. The peak frequency deviation therefore centers around f_c, spreading the power of the original oscillation into new partials, called sidebands [DJ97, p. 117]. Thus, increasing β is a control of the bandwidth of the FM spectrum. However, the positions of the sidebands and consequently the width of the spectrum is also a function of the modulator frequency: in the simplest case these are positioned with $f_c \pm k f_m$ with k being a positive integer (the carrier is $k = 0$), this is depicted in figure 8.9.

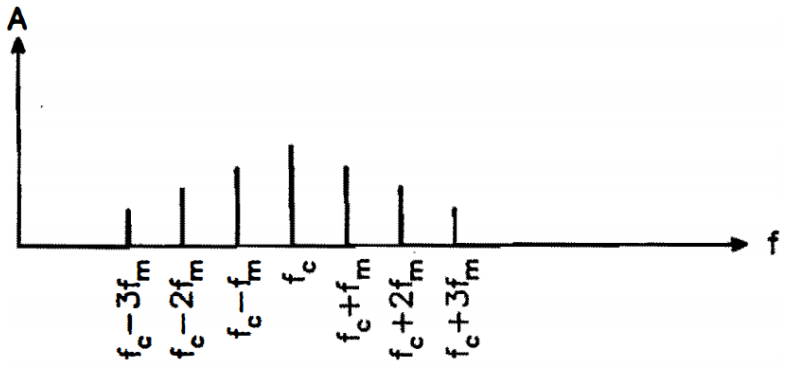


Figure 8.9: Spectrum of simple FM exhibiting sidebands through $k = 3$, taken from [DJ97, p. 107]

How is the amplitude of each partial determined? In [Cho77] Chowning shows that the absolute amplitude of each sideband is related to a Bessel function of integer values. These functions have various fields of applications that include problems of wave propagation and static potentials. In figure 8.10 we can see that a Bessel function describes a cyclic and damped system behaviour that eventually converges to zero with increasing x.
In general, when one speaks about modulation systems, it is useful to relate deviation and modulation frequency in order to see how the strength of the sidebands change over time as the strength of the modulator is varied. In most cases this is simply expressed as their ratio and called modulation index:

$$I = \frac{\beta}{f_m} \tag{8.4}$$

Thus, the amplitude of the kth sideband is given by the Bessel function of the first kind and the order k at the point I, $J_k(I)$ (ibid.). In this way, we can describe the spectrum for the simple two operator FM synthesis that takes

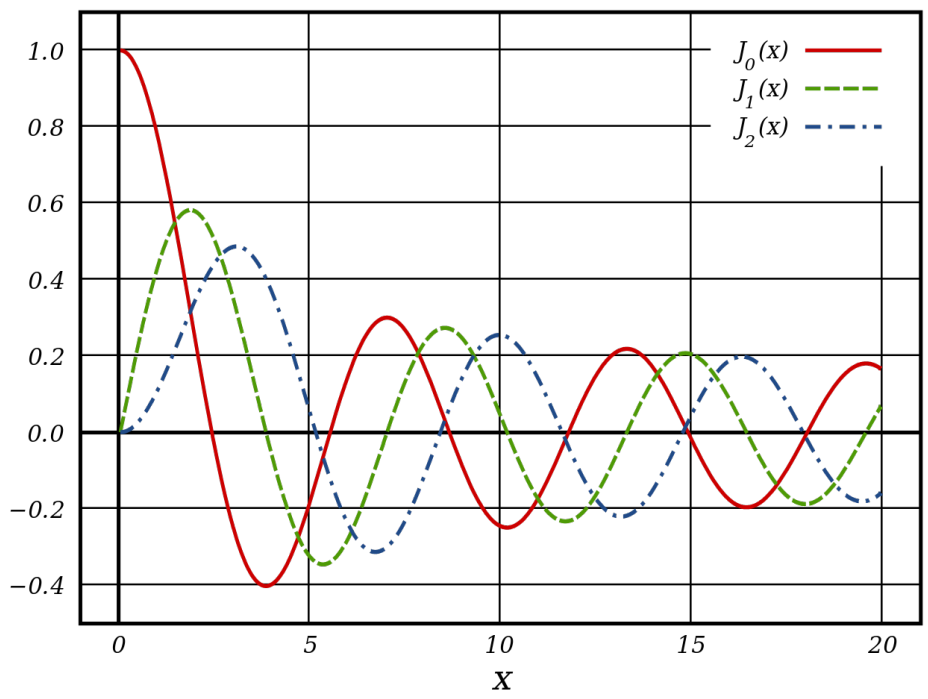


Figure 8.10: First kind Bessel functions $J_\alpha(x)$ for integer values of n (taken from [Wik08])

the parameters f_c, f_m and I:

$$\sin\left(2\pi f_c\, t + I\, \sin(2\pi f_m\, t)\right) = \; J_0(I)\, \sin(2\pi f_c\, t) + \sum_{k=1}^{\infty} J_k(I)\, \left[\sin(2\pi(f_c + k\, f_m))\, t \; + \; (-1)^k \sin(2\pi(f_c - k\, f_m))\, t\right] \tag{8.5}$$

When looking at the progression of Bessel functions in figure 8.10 again and taking into account which of these functions equation 8.5 uses it becomes apparent that negative amplitudes are generated. This means that negative frequencies exist within the spectrum and therefore such components have a phase of -180 degrees.

· ◇ ·

While these "out-of-phase" partials are not perceived by themselves, they do have an audible effect as soon as other partials are present at the same frequency. In this case amplitudes will either add or subtract from each other depending of their relative phases [DJ97, p. 109f]. Components below the carrier frequency tend to have more of these negative amplitudes. As partials can be spread into the negative side of the spectrum (below 0 Hz) they effectively fold back as positive frequencies with their phase inverted. This is a cause of constructive or destructive interference of sidebands and can have a great impact on the generated timbre but also on the perceived pitch. In figure 8.11 one can see the net result of interference for $f_c = 400$Hz and $f_m = 400$Hz that results in a spectrum similar to a sawtooth wave. If the interference was disregarded one could argue that the fundamental frequency in this FM spectrum shifted up to $f_c = 1200$Hz.
Further analysis of the spectrum can be performed when examining the ratio of the carrier and modulating frequency [Cho77]:

$$\frac{f_c}{f_m} = \frac{N_1}{N_2} \tag{8.6}$$

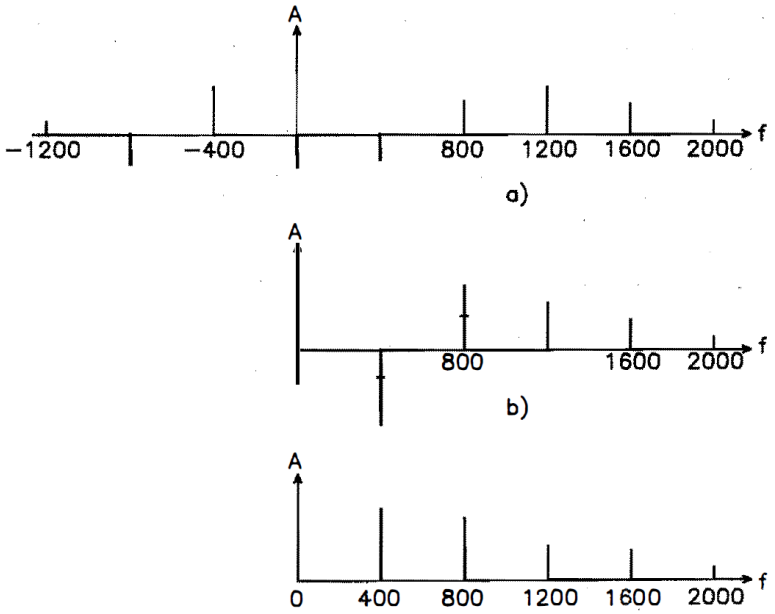


Figure 8.11: Example of constructive interference, taken from [DJ97, p. 110]. Subfigure (a) shows an FM spectrum which is folded back in (b), (c) displays the resulting net spectrum of magnitudes

The spectrum will be harmonic if N_1 and N_2 are integers without common factors. The resulting fundamental frequency f_0 can then be derived as:

$$f_0 = \frac{f_c}{N_1} = \frac{f_m}{N_2} \tag{8.7}$$

and we can furthermore determine the position of the sidebands as $k = N_1 \pm nN_2$ for n being a positive integer. There are other observations that can be derived for harmonic spectra:

- If $N_2 = 1$, then the spectrum contains all the harmonics of f_0
- If $N_2 = M$ and M is an integer greater than 1, every Mth harmonic will be missing. For the case of $M = 2$ the spectrum will contain only odd harmonics similar to a filtered square-wave.
- For values of $N_2 \geq 2$ the folded back components will not coincide with the positive ones, thus the spectrum is denser and more complex producing a richer harmonic sound.

N_1 and N_2 are undefined if f_c or f_m is an irrational number. This results in inharmonic spectra which are not only more difficult to analyze but which can also result in shifting the fundamental frequency and therefore the perceived pitch such that the carrier is not the lowest partial anymore. Picking up the property of folding back from before, we can construct cases that are perceived as inharmonic although N_1 and N_2 fulfil the requirements for harmonic spectra. Based on the example presented in [Has13, ch. 4], consider the following: even if the sidebands are fully harmonic, we can construct cases where the partials are reflected back in between the harmonics. This is true for large integer ratios that are close to irrational ones such as $5 : 7 = 1 : 1.4$ and $1 : \sqrt{2} = 1 : 1.414\ldots$ as these relationships imply high harmonics with a low fundamental so that for listeners no "tonal fusion" takes place [DJ97, p. 110]. Thus, Hass concludes:

> " *The result for the listener [...] will for practical purposes be inharmonic. The nature of these inharmonic spectra, which have at least twice the frequency components of the harmonic spectra with no phase cancellation, give FM synthesis a wide palette of bright, vibrant timbres, including many bell-like possibilities. Many of these inharmonic spectra*

can have sidebands that reflect close to, but not on top of existing sidebands, providing the opportunity for shimmering, chorusing-type effects [...]" [Has13, ch. 4, part 5]

In this way, even with integer ratios only, FM synthesis can create a wide array of timbres that mimic acoustic instruments.

· ◇ ·

For now we have only looked at oscillations that are purely sinusoidal. With the tool-set at hand, it is easy to see that more complex waveforms as either carrier or modulator lead to much more complex spectra[Rus04, p. 164]. In such cases all harmonics from the base oscillations have to be treated separately, each combination results in a spectrum. This set of spectra is then composed while accounting for the relative phase relationships. Since the spectral richness of a sound can be controlled by the index, this is usually utilized to dynamically alter the spectrum. In most cases this is accomplished with an additional modulating envelope generator.

· ◇ ·

In most cases the two operator FM synthesis method shown here is able to produce satisfactory formants [DJ97, p. 109f] and to imitate certain filter phenomena known from subtractive synthesis. However one can imagine that this basic topology can be further enhanced to allow for the creation of more realistic (or complex) formant structures and to better allow a targeted construction of certain spectra [Ben08, p. 274]. Indeed, throughout the years since its musical application, FM synthesis has undergone various developments and there are numerous contributions to the field. In [Roa96, p.236-250] a review up to the year 1996 of cascaded and parallel carrier and modulator topologies is given. This also includes feedback FM systems which allow to create more linear evolutions of the spectrum with increasing index. Besides these academic ones, perhaps the most practical and well known contributions to the field were done by Yamaha in collaboration with John Chowning [Ben08, p. 269] throughout the 1980s and early 1990s. Especially with the introduction of their digital DX7 synthesizer (1983), which is up to this date the best selling hardware synthesizer (180000 units). The DX7 featured pre-programmed with preset sounds, many of which can be heard in popular music hits of the 1980s. More importantly though, is the array of different topologies (32) that were available on the DX7 that included feedback and multiple carrier/modulator setups. Because of these, the synthesizer was able to create a large variety of acoustic instrument emulations of sufficiently high fidelity that were for the first time available to a larger public. Since then, several of these topologies were adopted as a standard ingredient in other synthesizer designs. One can imagine that such complex oscillator couplings are difficult to describe mathematically. In fact, not all topologies are analyzed mathematically thus far. For example, the cross-modulation of two exponential coupled oscillators (as opposed to the linear coupling which is described here) has been derived only recently [TL11]. It should be noted that it is also possible to generate percussive sounds that are based on noise sources with FM, since very complex inharmonic spectra without a strong fundamental are perceived as noise. This can be achieved with complex base waveforms for the oscillators and a high modulation as mentioned above or feedback systems which can be constructed to behave similar to chaotic dynamic systems [Ben08, p. 275].

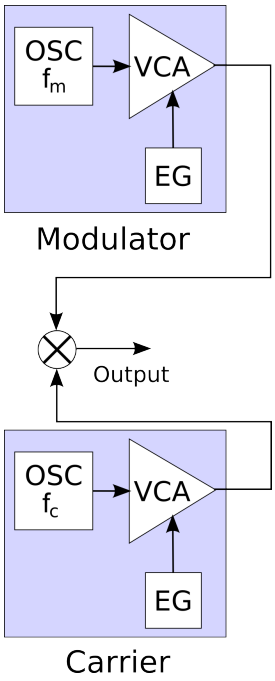


Figure 8.12: Ring modulation can be expressed as a multiplication of two signals, in this case oscillators. Additional envelope modulation for a typical topology has been added.

8.6.2 Ring Modulation

Ring modulation can be seen as a bipolar amplitude modulation which is simply the multiplication of two signals [Roa96, p. 216] as shown in figure 8.12. Again, we are making use of a modulator, $m(t)$, and carrier, $c(t)$. However, when seen as a multiplication, these roles can be swapped. The differentiation stems from the classic definition of amplitude modulation which is unipolar.

· ◇ ·

Using sinusoids as basic functions and making use of a standard trigonometric identity we can see how the spectrum is altered:

$$\begin{aligned} rm(t) &= c(t) \cdot m(t) = \cos(2\pi f_c\, t) \cdot \cos(2\pi f_m\, t) \\ &= \frac{1}{2}\left(\cos(2\pi(f_c - f_m)) + \cos(2\pi(f_c + f_m))\right) \end{aligned} \tag{8.8}$$

We can see that around the carrier frequency f_c the two new partials are centred with the distance $|f_c - f_m|$. Interestingly, neither f_c nor f_m exist in the resulting spectrum directly. Only two side bands have been created, this is why this form of modulation is also called Double Sideband Modulation [DJ97, p. 83]. Because of this, the pitch of the output signal can be perceived to be quite different. Figure 8.13 gives an example of the input and output spectra for RM. Similar to FM, for the exact analysis of the resulting spectrum one has to account for the interference of partials as soon as parts of the lower sideband moves into the negative part of the spectrum. For more complex input spectra, we can also see that the lower sideband is "flipped". If f_c and f_m are in an integer ratio, the sidebands are harmonic, otherwise they are inharmonic [Roa96, p. 217].

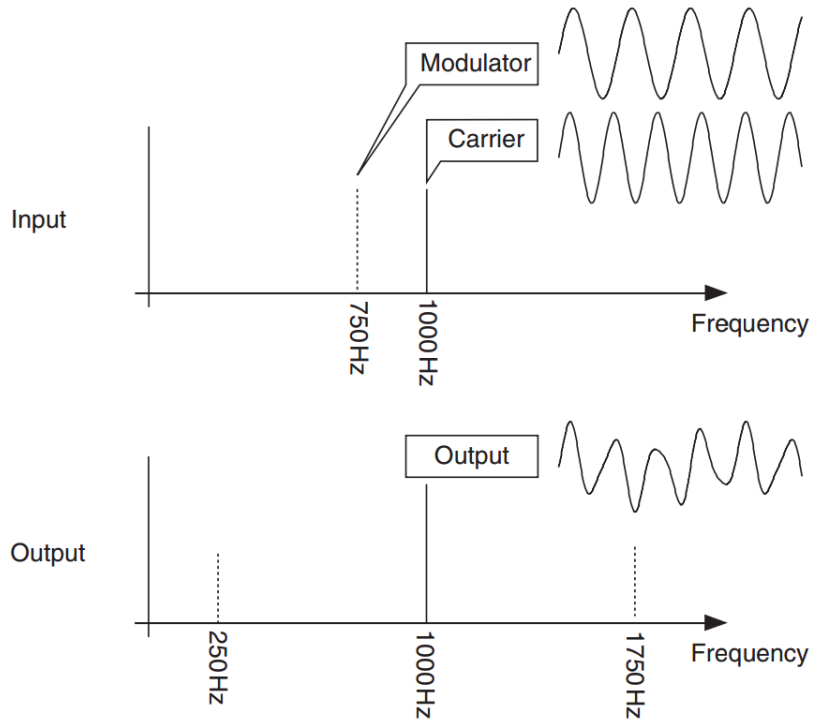


Figure 8.13: Ring Modulation of two sinusoids, the output spectrum only contains the sum and difference of the input frequencies (taken from [Rus04, p. 159])

· ◇ ·

Thus one can use RM modulation as simple means to completely alter the spectrum of the carrier signal (*e.g.* spectral inversion). Compared to FM, it produces less sidebands and may therefore be "less destructive" with regards to the original input in some cases. Holmes describes the effect of RM in the following:

> "*If the input signals have many harmonics, such as a guitar or the human voice, the resulting output signal is complex and rich, a kind of ghost of the original sound.*" [Hol02, p. 25]

As RM has a rich history in electronic music, it has been applied in many compositions, especially those of early electro acoustic music *e.g.* in Karl Heinz Stockhausen's *Gesang der Jünglinge* (1953).

8.6.3 Vector Phaseshaping Synthesis

Recent developments in non-linear synthesis techniques led to new methods for wave- and phaseshaping and feedback systems (cp. [Kle+13]). We can see how the former two ones stem from a natural extension of the digital oscillator if we do not regard it as an atomic entity. In figure 8.15 (a), the classic oscillator is depicted with the common Music V parameterization [DJ97; Kle+13]: f_0 is the fundamental frequency, ϕ_0 the initial phase, $F(\cdot)$ the waveform function that describes the shape of the signal and thus contributes to the timbre and A the peak amplitude. In this way it is apparent that Frequency Modulation and Ring Modulation (as a form of Amplitude Modulation) shown previously are methods that rely on cascading these unit generators. The internal structure of this oscillator in figure 8.15 (b) encapsulates two components [Kle+13]:
The phase generator (PG) generates the periodical phase function sample-by-sample, based on the pitch f_0 and the phase offset ϕ_0,

$$\phi(n) = [\phi(n-1) + \phi_\Delta] \mod 1, \phi(0) = \phi_0 \tag{8.9}$$

with ϕ_Δ being dependent on the fundamental frequency and the sampling frequency (ibid.). The $\mod 1$ operator limits $\phi(n)$ to the $[0, 1($. In this way, $\phi(n)$ creates a simple, periodic ramp function as shown in figure 8.14.

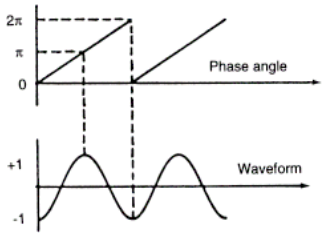


Figure 8.14: Phase function $\phi(n)$ (upper figure) and the resulting output waveform if $F()$ is an inverted cosine function (lower figure)

The wavegenerator (WG) takes output $\phi(n)$ and evaluates the waveform function $F(\cdot)$ to produce the preliminary output signal $w(n)$ before scaling by A. Similar to $F(\cdot)$, we can define the functions $P(\cdot)$ and $W(\cdot)$ that deform $\phi(n)$ and $w(n)$ in a non-linear way. These function mappings will preliminarily introduce timbral effects (ibid.). $P(\cdot)$ is then a functional mapping for phaseshaping while $W(\cdot)$ is the one for waveshaping. Phaseshaping can be understood as both as a form of phase modulation (similar to frequency modulation) and non-linear waveshaping.

· ◇ ·

Vector Phaseshaping Synthesis (VPS) applies phaseshaping by defining one (or more) two-dimensional breakpoints to deform the otherwise linear phase function $\phi(n)$ [Kle+11]. We can define a breakpoint p as

$$p = (d, v) \tag{8.10}$$

with d being the horizontal and v the vertical position; $d \in \Re$ is constrained to $0 \leq d \leq 1$, so it is normalized by the length of a period, while $v \in \Re$ can be arbitrary. With this, the phaseshaping function can be given as:

$$\phi_{vps}(x) = \begin{cases} \frac{vx}{d} & 0 \leq x \leq d \\ (1-v)\frac{x-d}{1-d} + v & d \leq x \leq 1 \end{cases} \tag{8.11}$$

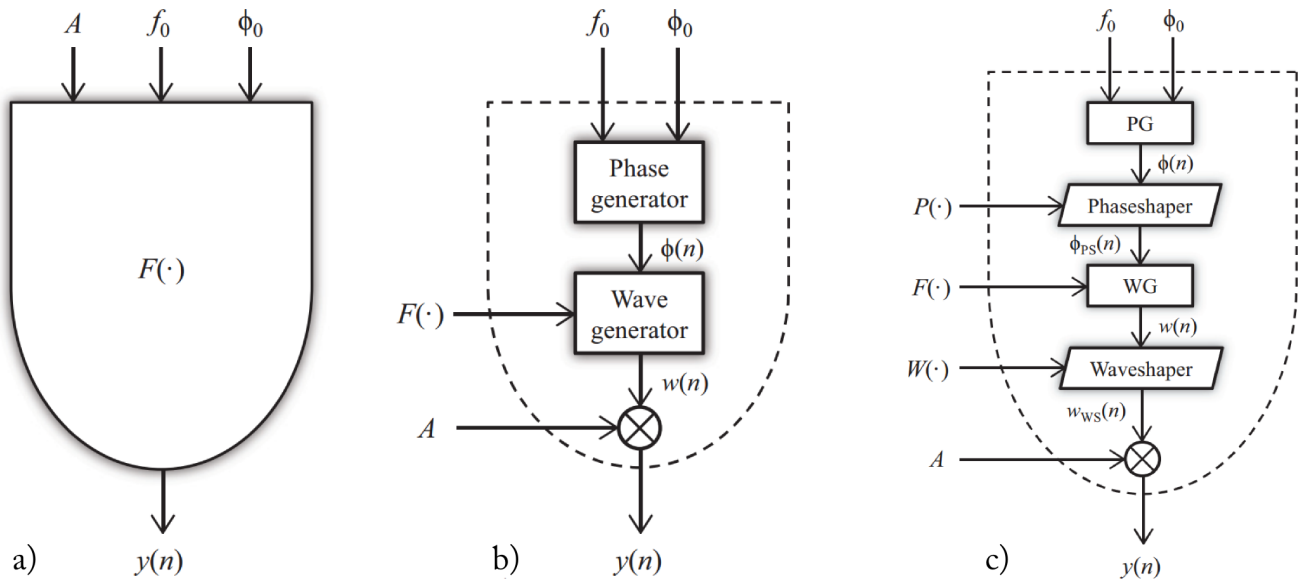


Figure 8.15: (a) Classic digital oscillator, (b) its internal structure and (c) extension with wave- & phaseshaping (taken from [Kle+13, p. 20])

The wavefunction in VPS is given as an inverted cosine. In the given form of ϕ_{vps}, the phase function $\phi(n)$ is transformed to be divided into two segments with different but constrained slopes. In case of $p = (0.5, 0.5)$ the original phase function is not altered at all, if v is raised towards the unity then the slope of the first segment is increased and the second one is decreased until it reaches a constant value slope of 1.0 (ibid.), figure 8.16 illustrates this. One can see that in these cases the bandwidth of the spectrum grows from a single harmonic and the shrinks again to form a half-wave rectified sinusoid consisting of mainly odd harmonics with a strong second one (ibid.). Leaving $v = 1$ and decreasing d will lead to impulse-like waveforms with varying duty cycles. When d is in this case modulated this allows to model pulse width modulation as mentioned in the beginning of this section. The combinations of d and v where pure sinusoids are created can be generalized as

$$|\frac{v}{d}| = |\frac{1-v}{1-d}|, \text{for} v, d \in \mathbb{Z} \tag{8.12}$$

for all cases where the absolute derivative at the infliction point is equal, so that the produced pitch will be equivalent to $|v/d|f_0$ (ibid.).
As mentioned before, VPS is able to synthesize formants some of which have a very vocal quality and allow to create effects similar to the resonant filtering used in subtractive synthesis. This is achievable if $v \geq 1.5$. In such situations d functions as a brightness control, spreading the spectrum and creating multiple formants as soon as it deviates from the centre position 0.5. An example for the effect can be seen in figure 8.17.
While it is possible to describe the VPS spectrum containing one breakpoint in closed form, some of the effects can also be stated as simple waveform morphologies (ibid.) as shown in table 8.18. One issue is that for $v > 1$ severe aliasing can occur due to incomplete periods of the cosine waveform. In [Kle+11] a method is introduced to remedy this by modifying $\phi_{vps}(x)$ to create piece-wise full-cycle sinusoids. The method implemented for this thesis involved using bandlimited cosine oscillators that are less prone to produce aliasing artefacts.

· ◇ ·

In conclusion, VPS allows to generate a wide range of spectra that are useful in the context of emulating classic waveforms and filter effects. The

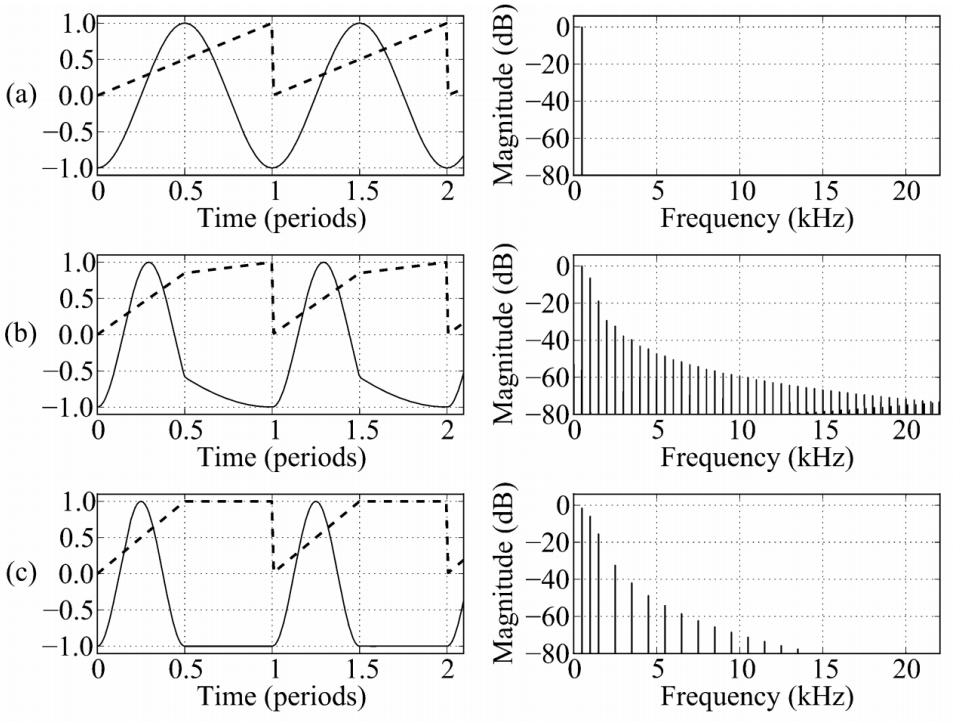


Figure 8.16: VPS waveforms and spectra for (a) $p = (0.5, 0.5)$, (b) $p = (0.5, 0.85)$ and (c) $p = (0.5, 1.0)$ (taken from [Kle+11])

low number of input parameters that describe these makes it suitable for use-cases where a low dimensionality of control parameters is needed as is the case for the prototypical implementation in chapter 9.

8.7 Perceptually Salient Audio Features to Describe Timbre

This section is intended to give a quick overview over predominant audio features and their extraction, an in-depth introduction to this matter is given in [Ler12]. It is also recommended to consult the contribution by Eerola *et al.* [EFA12] where audio feature analysis was performed to gain similar insights into timbral microstructures. In this thesis the same toolbox (MIRToolbox) [LTE08] used with congruent feature sets.

· ◇ ·

Audio feature (or content) analysis is a subfield of research in music information retrieval (MIR) with the goal to extract descriptive and discriminative properties from audio signals [Ler12, p. xiii] such as speech or music recordings. Low- and mid-level features can be grouped into five classes [Ler12, p. 4]:

- statistical / technical: derived from audio data such as the amplitude distribution
- timbre / sound quality
- intensity-related: properties of envelope and loudness
- tonal: pitch and harmony relationships
- temporal: rhythmic properties and temporal evolution

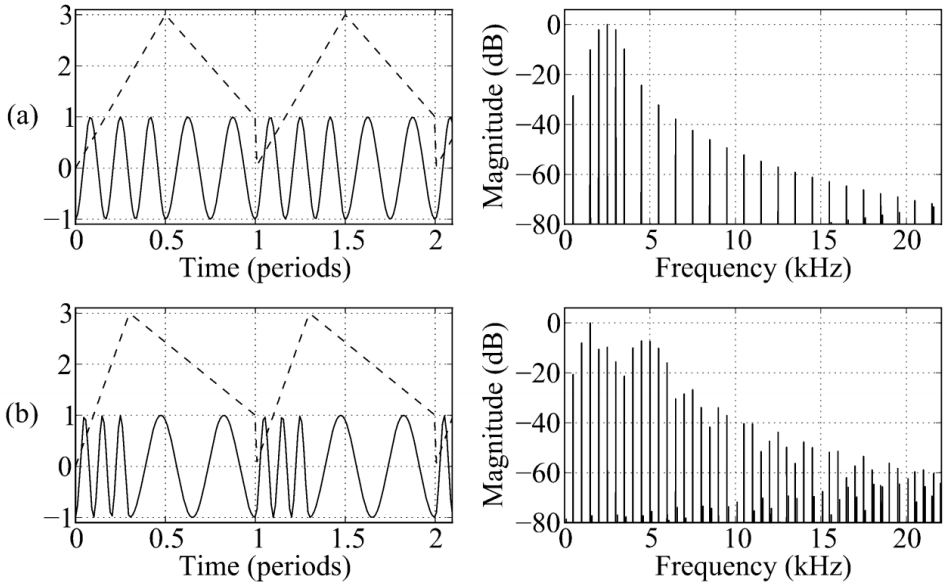


Figure 8.17: VPS waveforms and spectra creating a single formant, (a) $p = (0.5, 3)$, and multiple ones for $d \neq 0.5$ (taken from [Kle+11])

$p(d, v)$	Waveform	Spectrum
$(0.5, 1)$	half sinusoid	missing some even harmonics
$(0.5, \leq 1)$	distorted half sinusoid	steep spectral slope
$(0.5, \geq 1)$	varying-period sinusoid	peaks at $(2v - 1)f_0$ (single formant)
$(\leq 0.5, 1)$	pulse-like	spectrum gets richer with $d \to 0$
$(\leq 0.5, \geq 1)$	varying-period distorted sinusoids	multiple formant peaks
$(\leq 0.5, 0.5)$	sawtooth-like	more gradual spectral slope

Figure 8.18: VPS waveform morphologies (taken from [Kle+11])

Especially low-level features are not necessarily meaningful to the human perception, therefore these are usually grouped or enriched with additional meta-information in a second processing pass.

· ◇ ·

Within this thesis, audio feature analysis is broadly used in the context of synthesis by analysis (chapter 9 and chapter 11). Therefore only timbral, statistical and intensity related features are of interest that are also perceptually salient. Figure 8.19 depicts a summary of mainly timbre related audio features and their technical extraction methods that have been applied for adaptive audio effects in [VZA06]. It also highlights the relationships between signal features, related redundancies and perceptual correlates and shows that the extraction of certain features rely on a pipeline of consecutive processing steps in order to retrieve high-level features.

· ◇ ·

It is also apparent that audio features can be correlated. With certain machine learning approaches this can be problematic. Another issue may be of practical nature: since certain features bear little additional information but the number of examples in a test set is usually limited the accuracy of a classifier may decrease (course of dimensionality). If the feature set is comparatively large, this cannot be trivially solved by employing a feature selection.

· ◇ ·

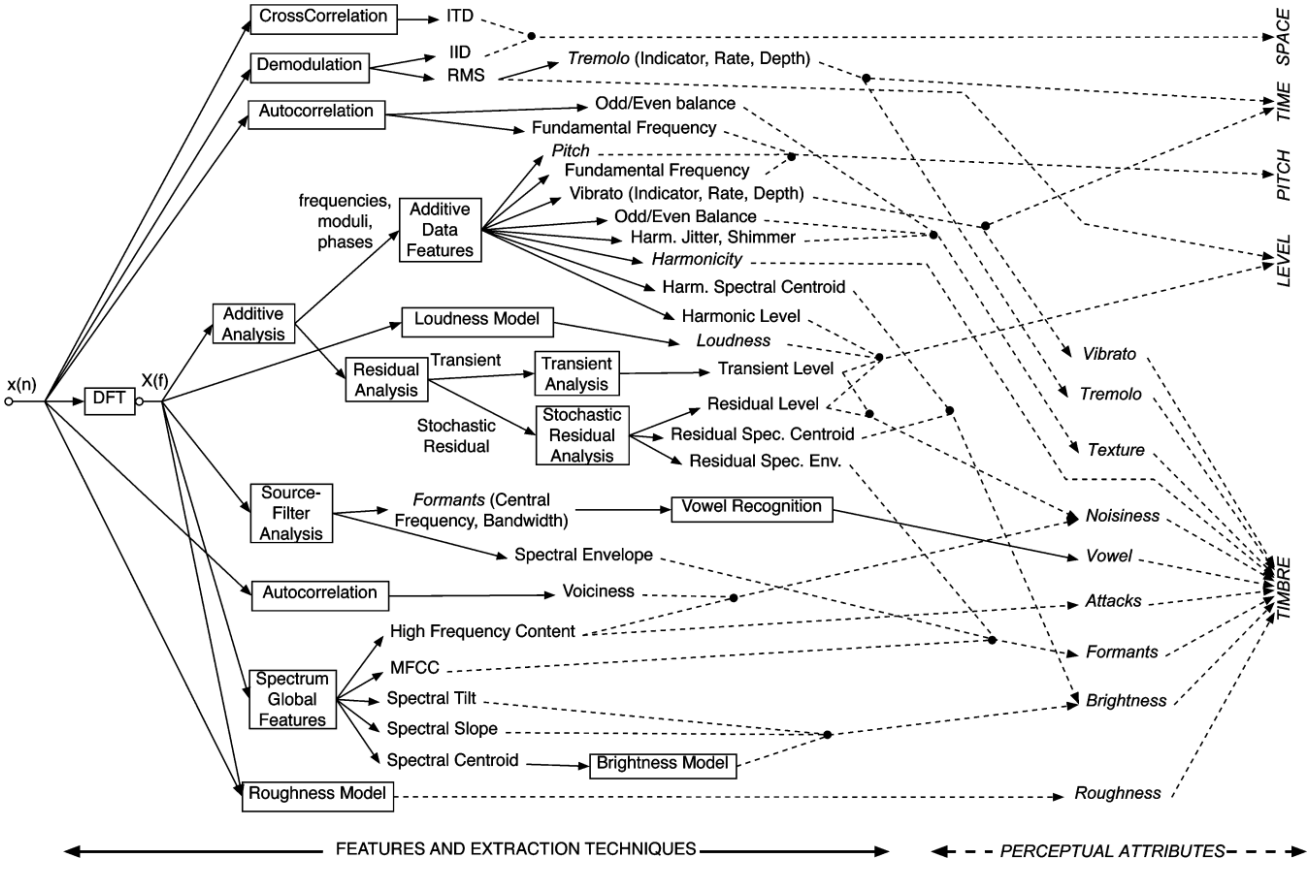


Figure 8.19: Timbral audio features and their extraction techniques (taken from [VZA06])

Which features are perceptually salient? One of the most prevalent approaches to get insight into this is based on dissimilarity ratings by human listeners for sound pairs [Mul+11; Pee+11]. These studies involved projecting the ratings into a lower dimensional representation by means of multidimensional scaling (MDS) such that the distance between ratings is matched in the MDS projection. After that, acoustic correlates are determined such that the projection yields a perceptual space. In [Bea07, ch. 8], Donnadieu gives a review of the related research up to the year 2005 and shows some general discrepancies in the results of studies that have been conducted. However, in most cases the predominant perceptual dimensions derived have been attack rate, spectral flux, spectral centroid and spectral roughness with varying degrees of importance. In most cases stimuli from acoustic (isolated) instruments have been used, however for the scope of this thesis, several confirmatory studies also approached synthesized timbres (albeit informed from acoustic instruments) [Cac+05; MC07] (cited from [Pee+11]).

· ◇ ·

Regarding the role of the temporal evolution to discriminate timbres, some perceptual salience is admitted but a holistic consensus does not exist so far, thus it is hypothesized that more appropriate high level descriptors may exist [Bea07, ch. 8]. In this way Hajda concludes:

> "*Even in these contrived contexts, it is clear that a single rule will not apply between classes of instruments. Given the above caveat, it seems that, for sustained continuant tones, the timevariant steady state usually provides sufficient and necessary information for the identification of an instrument. The co-evolution of the amplitude and spectral centroid seems important here, but the direction (i.e., regular vs reverse playback) does not.*" [Bea07, p.

256]

Correspondingly, in a study analyzing various additional audio features for mood classification of music, Eerola *et al.* observed that although a small subset of these was capable of explaining the variance in the emotion ratings, this subset was genre specific [Eer11].

· ◇ ·

To summarize, the ability of such audio features to give a meaningful, perceptually salient representation of the audio data may be highly dependent on the data itself. This issue may be aggravated as the results in a related study by McAdams [McA99] suggests that the variance among professional musicians performing the discrimination is more coherent than among nonmusicians and, even more, there are latent groups among all listeners where the inter-group coherence is larger than the intra-group coherence. In this way, certain perceptual attributes may have been acquired by experience. Audio features that have been arguably proven to be valuable for discrimination may not be perceptually salient for certain tasks so it becomes clear that no generic set of audio features exists. For the use cases of this thesis, these tasks are not as well defined as the ones in the related studies with regards to the basic audio source material, which consists of recordings of mainly classical acoustic instruments, and the nature of discrimination. In this way, rather than defining a strict low-dimensional set of audio features *a-priori*, emphasis is given to classifiers or projection methods that support high dimensional data (a large set of acoustic features) and that allow to make certain assessments about the likelihood of explaining the variance in the data. An approach for this is shown in chapter 9.

Chapter 9

Controlling Audio Synthesis by a Perceptually Motivated Mapping

Chapter Abstract

In this chapter, a prototype will be discussed that allows a group of users to design sound collaboratively in real time using a multi-touch tabletop. This approach relies on a machine learning method to generate a mapping from perceptual audio features to synthesis parameters. This mapping is then used for visualization and interaction. Finally, the results of a comparative evaluation study will be discussed.

This chapter is based on the contribution [KBG14], "Designing Sound Collaboratively Perceptually Motivated Audio Synthesis" by Klügel, Becker and Groh.
Sections 9.2, 9.3, 9.7, 9.5, 9.8 and 9.9 have been significantly expanded. Sections 9.4 and 9.6 are largely new.

The implementation of the interface (UI and HCI paradigms) for the prototype was done in the scope of Timo Becker's master's thesis and the interface concept was derived in several iterations and discussions between him and me.

The overall concept of the underlying framework including the machine learning approach and synthesis method was devised and implemented by me. The study was conducted by us both.

The source code and data sets for the prototype have been made available:
T. Becker and N. Klügel. *SoundDesigner*. Application, includes data sets, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/SoundDesigner

9.1 Perceptually Motivated Synthesis

Sound design is generally seen as the process of specifying, generating or manipulating audio elements in a non-compositional context. If we look at

it more from the musical perspective - sound design as designing timbres - then sound design is part of the daily routine in modern electronic music composition and performance as it is a form-bearing dimension of music (ch. 7).

· ◇ ·

The question of how to support the necessary exploration of timbre implies two viewpoints. The first one is the engineering (DSP) perspective that considers the capabilities of a synthesis method to generate manifold sounds. In this regard the aim is to have a generic synthesis architecture in which the choice of parameters leads to large sonic variation. Parameters are therefore *structural*. The other perspective is the HCI one in which the aim is to have synthesis parameters readily map to perceptible sonic attributes [Sea13b], thus human cognitive manageability. In this case, parameters are *perceptual*. Compared to the engineering perspective, the HCI perspective has not been studied to the same extent (ibid). In fact, if we look at commercial hardware or software synthesizers, the common interface scheme is to have a direct relationship between control affordances and structural parameters. Such a mapping is problematic because it makes the change of timbral qualities non-intuitive without knowledge about the sound generating structure and its technical functioning [Ris03; Röb10]. This is especially true for novices. Seago gives an example for this deficiency: *"[FM] is a synthesis method that may be viewed as essentially an exploration of a mathematical expression, but whose parameters have little to do with real-world sound production mechanisms, or with perceived attributes of sound. However, all synthesis methods require a significant degree of understanding of the approach being employed ..."* [Sea13a].

· ◇ ·

Apart from structural control over the synthesis, there are other approaches. For example, in certain sample-based or wave-table based synthesis environments such as the synthesizer *Serum* [XFe14] or the DAW *Renoise* [Mül+02], waveforms can be drawn directly in the user interface. In contrast, the UPIC system which was introduced in section 8.2 allows to draw the spectral evolution of a sound or a composition. However, an interface that relies on a visual representation of the sound directly in the time or frequency domain is at too low a level of abstraction because *"there is no 'semantic directness' (Hutchins et al. 1986) for the purpose of specifying any but the most crudely characterized sounds"* [Sea13a]. Alternatively, there are also approaches that connect language with synthesis parameters (cp. [Mil14; Ste06]) or which organize synthesis parameters using tags. Especially the latter is prominent in commercial applications such as *Maschine* [Nat10]. or *Bitwig Studio* [Bit14]. As pointed out in chapter 7, language is problematic because there is a *"complex and non-linear relationship between timbre space and verbal space"* [Sea13a], and there is ambiguity since there are valid alternative mappings based on cultural or social background.

· ◇ ·

Apart from schemes that more or less implement direct control of the synthesis parameters, it is also possible to rely on an "interpretative layer" that is informed by perceptual attributes from a feature analysis which then controls the audio synthesis. Therefore, the issues of controls that exhibit non-linear behaviour and inter-dependencies regarding the perceptual qualities of the output are circumvent. Overall, it is possible to abstract from the concrete synthesis model and thus generalize the interaction for arbitrary synthesis models. We see value in this generalization, specifically because it reduces peripheral tasks for electronic music creation and therefore helps

including novices into the collaboration. More generally, in [Sea13b] Seago proposes a set of criteria for such a timbre space (for the sake of clearity, we just define it as the space of possible timbres the *synthesizer* can generate) through which the user can navigate to find or generate a sound (paraphrased, only the relevant ones are kept):

S1 It should have large coverage to encompass a wide and musically useful variety of sounds.

S2 It should have sufficient resolution and precision.

S3 It should reflect psychoacoustic reality. The perceived timbral differences of sounds in the space should be broadly proportional to the Euclidean distance.

S4 It should have predictive power. A sound between two others should be perceived as hybrid of those.

We will use these criteria as guidelines for devising the system.

· ◇ ·

With respect to electronic music creation, we see the collaborative use-case of designing sound a largely untapped territory for fostering creative endeavours. The structure of this chapter is further divided into three parts, essentially: after discussing the relations of this contribution with the body of related work, we will introduce the main elements of our approach. We will then elaborate on a comparative user study that we conducted to evaluate our approach in comparison to generic direct synthesizer control and our prototypical implementation.

9.2 Related work

The majority of research related to mapping from timbre to synthesis parameters focuses on re- or cross-synthesizing the original sound sources from timbral features. For our use-case, however, we are interested in applying this mapping foremost as a paradigm for interaction, so the main issue is human cognitive manageability. This field of Human Computer Interaction has not been studied extensively [Sea13b]. Regarding the cross-synthesis methods, we can identify three common approaches, the signal processing one, which formulates explicit timbral transformations (*e.g.* [Ser97]), the machine learning one, where the timbral model is inferred from audio data (*e.g.* [LV08]) and the concatenative one, where timbre is constructed using the sound recordings themselves. A well known representative of the latter is the CataRT system [Sch12]. It is especially relevant as the navigation in timbre space is its core concept for interaction. However, for this, CataRT and subsequent developments use the high-level timbral features as orthogonal axes directly. This means that only a low (2-3) number of features span this space, discarding timbral qualities that may add valuable information and possibly reducing the representative quality of the space since features may be correlated. To circumvent these problems, CataRT allows the user to re-define the space at run-time. For the collaborative use-case, this introduces the conceptual challenge how a shared navigational space can have user-specific views without hampering awareness and articulation. In case of the collaborative application *Nuvolet* [CBC11], which is based on CataRT, this space is static. It uses a 3D camera system as sensor for gesture driven Audio Mosaicing. *Nuvolet*

aims to support a collaborative, virtuosic performance that enhances the performer-to-audience relationship. Consequently, it lacks features that provide affordances for the collaboration itself with respect to shareability and awareness. The cognitive load and imprecision that are initially induced by the 3D control method are also problematic for novices.

· ◇ ·

Apart from high-level timbral features, also low-level features that are reduced in dimensionality can be used to span the timbre space [NBG04; LV08]. However, the constructed timbre space by this machine learning approach may not have an obvious relationship to the human perception [LV08], so in order to remedy this, further processing steps than the reduction itself are necessary. Moreover, any mapping to synthesis parameters *and* the inter-relationship between timbral features are bound to be highly non-linear [HC06]. Non-linear mappings such as Self-Organizing Maps (SOMs) are therefore preferable. These have already been applied in numerous fields including sound-design. Especially the approach in [EP10] overlaps with our use-case as the generated 2D-SOM representation of the timbre-space is used as interaction metaphor. Due to some beneficial mathematical properties, we will use Generative Topographic Mapping (GTM) instead of SOM, as will be explained in more detail in section 9.6. Furthermore, we will use high-level features since otherwise we would have the additional problem of defining a proper mapping that is computationally expityressive enough to abstract from low-level features to perceptually more meaningful ones.

9.3 Concept and Preliminary Tests

We need to be able to synthesize a large variety of different timbres in order to not severely limit expressiveness, apart from the constraints imposed by the sound generating method (guidelines S1 & S2). As argued before, from this viewpoint an organizational structure for the various sounds is necessary. Thus, one has to rely on a corpus of sounds in which such a structure(s) are represented. Thus, similar to CataRT, it is the goal to generate a corpus $\mathcal{S}$ such that each sound $s_i \in \mathcal{S}$ can be analyzed for its timbral features $t_i \in \mathcal{T}$ to generate the mapping $s_i \leftrightarrow t_i$ (guideline S3). Later on, we would like to synthesize s_i again in real time within a tonal context. A common approach would be to use and re-synthesize recorded sounds. However, it is non-trivial to analyze all time-frequency relationships reliably for an *arbitrary* sound source, therefore possibly rendering its original timbre-space relationship invalid during re-synthesis (*e.g.* when the fundamental pitch is altered). Therefore we opted to develop our own synthesis model such that all parameters of the synthesis (*e.g.* pitch) are known.

· ◇ ·

To be more precise, the core process consists of sampling different parameter settings of a synthesizer and rendering the resulting audio which is then feature analysed. The feature space created by all sounds (parameter settings) is then projected down onto a two dimensional plane for visualization. If we also create a mapping from the two dimensional plane back to the feature space and from there to the original generating parameters, it becomes also feasible to control the audio synthesis based on this inherent timbral structuring. Figure 9.1 depicts this approach.

· ◇ ·

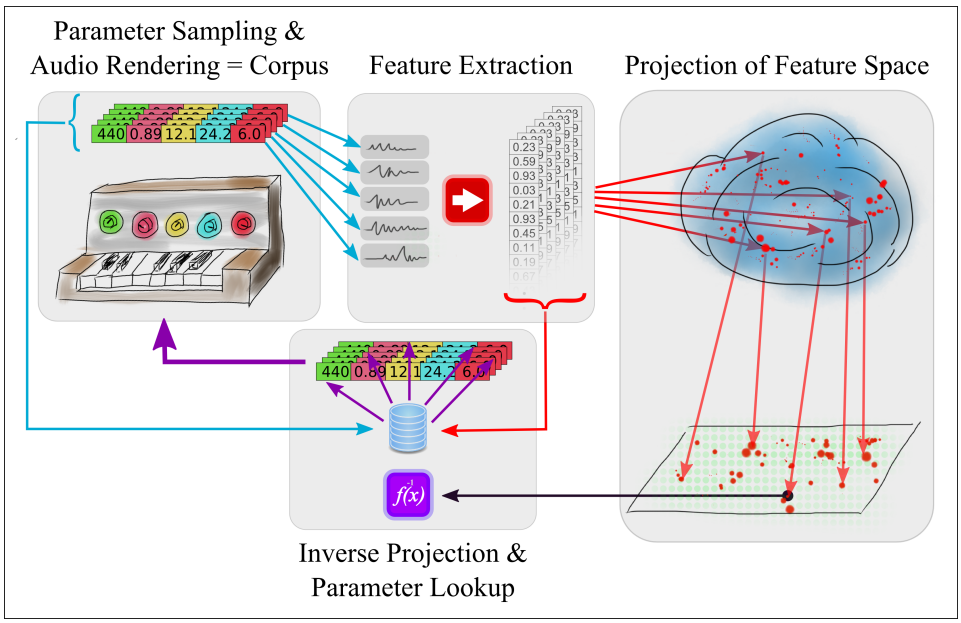


Figure 9.1: Visualization of the proposed process which shows the sampling of a synthesizer, the feature extraction and projection as well as the inverse mapping for lookup of the original synthesizer parameterization for interaction.

Two preliminary experiments were performed to evaluate the validity of a feature space that is projected down with the dimensionality reduction method described in section 9.6. These were not intended to be a rigorous examination of the method. Unlike the prototype, for both test cases time series feature data was created. This not only allows to depict the evolution over time in the projection as some sounds change timbre but also inflates the number of feature vectors in order to see whether the projection method can cope with a comparatively large amount of data points.

· ◇ ·

Test case *A* involved sets of six musically meaningful classes of sounds which were taken from the standard library of instruments delivered with *Renoise* [Mül+02]: acoustic instruments, bass sounds, atmospheric (pads), keyboard sounds, percussion and effects. However, these groups are neither musically nor spectrally disjunct. For example, acoustic plucked or percussive instruments can have similar spectral or amplitude envelopes at the beginning (exponential chirp with high energy). The data set as a whole consists of roughly 4500 feature vectors (17 dimensions) with a class membership that is highly uneven - 70% of the samples belong either to the class acoustic or bass. Thus, making specific qualitative or quantitative assessments would be inappropriate. In figure 9.2 the result of the projection for test case *A* is shown. The exact value of the third dimension in both plots can be ignored here, they show how probable one point on the two dimensional grid represents a specific point in the high dimensional feature space (responsibilities). We can see that the instrument groups are mostly represented as clusters with little perturbation. Upon looking closely at the plot, one can see that the responsibilities of a grid point are frequently shared for several classes but oftentimes they are dominated by the membership to one specific class. Hence, there must be some topology in feature space (the projection method is topology preserving for guideline S4) induced by the similarity of certain feature vector classes. It is interesting to see that although the class of acoustic samples is dominant, it is limited in terms of spread and mainly shows dominant responsibilities in these regions. Along the lines of the previous argumentation, this indicates that

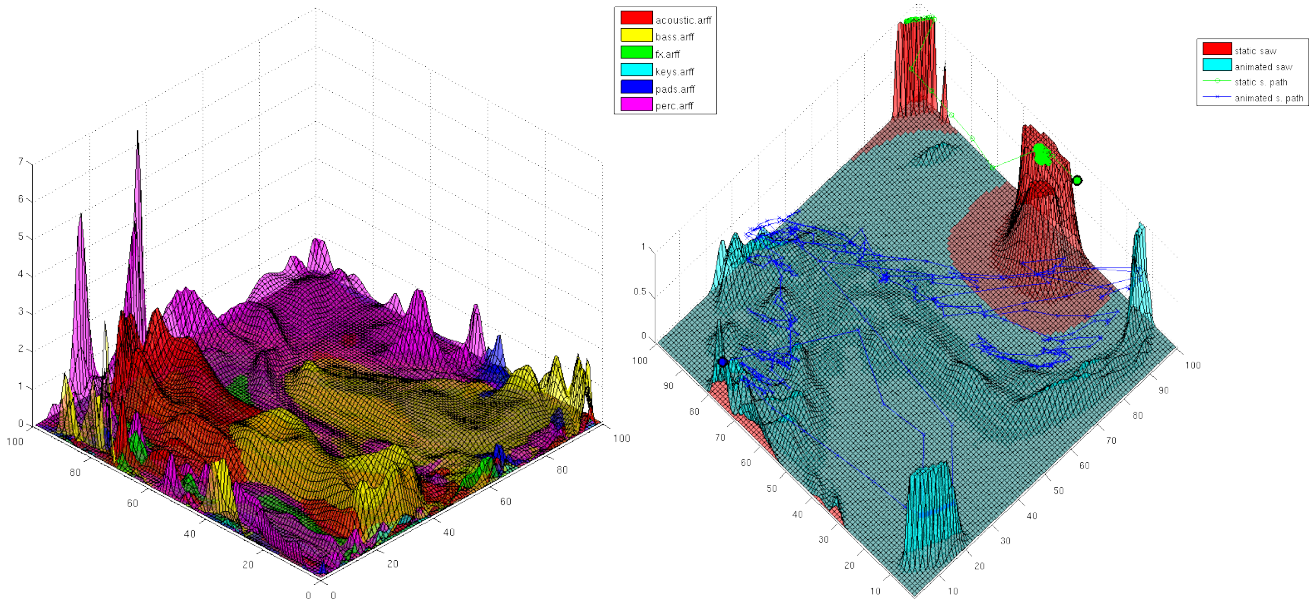


Figure 9.2: Example plots of preliminary data sets for test case *A* (left) and test case *B* (right)

these samples have some inherent similarity. Especially given the closeness of this cluster to grid points with a high responsibility for parts from the class of percussive samples this underlines the plucked string character of many sounds from this acoustic class. The large spread of the percussive class can be explained by the large variety of instruments in the set (bass drum, harmonic and inharmonic resonances, high-hats) and the spectral feature analysis which uses a Short Time Fourier Transform with 1024 samples per block. Such a block size means that at the common sampling rate of 44.1kHz, spectral changes happening faster than ≈ 0.75Hz will not be registered but instead will smear or mix the spectral content within a block. Fast spectral changes are, however, a characteristic of percussive sounds. Thus, the variance of spectral content on an analysis block basis may be increased. This is also an argument why appending the parameter settings of the synthesis to expand the feature data with information about the temporal evolution of a sound is sensible. However, more concrete statements would have needed a more thorough examination of the samples and their feature representation. Test case *B* simply examines the representation of two sounds and their temporal evolution. For this, two sounds were created, the first one is a static saw waveform continuing for approximately 9 seconds. The second one, named "animated saw", is the same basic sound that is slowly faded in and subsequently distorted. The temporal evolution can be seen in 9.2 on the right hand side. Apart from the two peaks in the opposite corners of the plot, which can be attributed to the abrupt ending of the sounds followed by a long silence, the visualization shows what is to be expected: the static sound is represented as a singular cluster while the animated one moves along a path without significant jumps.

· ◇ ·

To conclude, as the goal of this preliminary study was to see whether reasonable cluster memberships emerge in the projection based on sounds that are approximately related, the results are satisfactory.

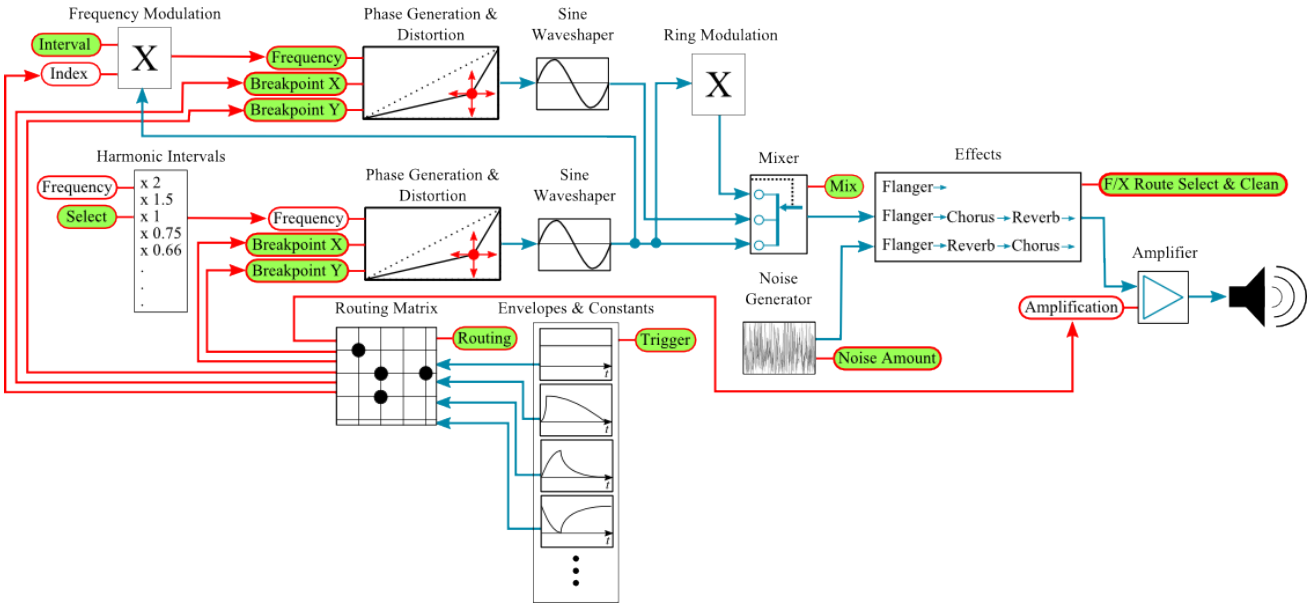


Figure 9.3: Final design of the synthesis topology (some parts are omitted), parameters in a green box indicates that they are control parameters, the individual routing parameters have been grouped

9.4 Synthesis & Corpus

To create the corpus it is necessary to sample various parameter settings $p_i \in \mathcal{P}$ from the possible parameter combinations $\mathcal{P}$ to generate the sounds s_i for the analysis. Hence, we are interested in having a low number of parameters.

9.4.1 Synthesis

The synthesis model makes use of Vector Phase Shaping (VPS) as described in section 8.6.3 and allows low frequency modulation of the two dimensional breakpoint position. The motivation for the design is that it should be feasible to create certain timbres known from subtractive synthesis which can also be made more complex by means of frequency and ring modulation. This choice of VPS allows for a reduction of control parameters compared to common topologies of subtractive synthesis, if we admit that certain waveforms or phenomena are only approximated by VPS:

- many basic waveforms and mixtures of them can be created by this form of phase distortion, this omits a parameter for waveform selection or mixing
- effects such as pulse width modulation are possible without the need for the two additional parameters of LFO frequency and modulation amount
- a filter can be completely omitted and thus also the parameters for cut-off frequency and resonance

Thus, the VPS oscillator has three parameters: frequency and the two modulation amounts for the breakpoint position instead of the six for a fully subtractive topology (frequency, waveform, pulse-width modulation amount and frequency, modulation amount for cut-off frequency and resonance). The synthesizer consists of two VPS oscillators where the second one acts as a modulator that is tuned into harmonic intervals of the main fundamental frequency (up and down two octaves). Each of these pairs is then

instantiated three times, so that at a summing stage one can cross-fade between an oscillator-mix, a ring-modulated, a frequency-modulated and simple carrier signal. After that, white noise can be mixed into the sum before finally going into a variable effects processing chain and an amplifier for amplitude modulation. The sole purpose of the effects chain is to either emphasize certain part harmonically in the spectrum (flanger) or to act as a colouring diffuser (flanger-chorus-reverb or flanger-reverb-chorus). The reverberator is set to long decay times. The reason for this is that it allows transforming any input material into long evolving and sustained sounds. Placing the amplifier after the effects chains therefore allows to regain amplitude control over the evolving sounds (modify decay times). Figure 9.3 shows this topology. Each parameter that can be controlled is marked as green, the multiple instances of the carrier and modulator pair have been omitted. The blue lines indicate the signal flow and the red ones the modulation destinations. The modulation system can be thought of a switching matrix that allows to route modulation sources to any modulation destination. Parameters such as the FM-index or breakpoint position act as weight for the modulation amount. The modulation sources are 18 unipolar envelopes of different shapes and lengths (9 are normal and the others are inverted and offset) and the two constant values zero and one. With the latter it is possible to switch the modulation off or on. For example, in this way, the resonance effect (v-value of the VPS oscillator) can be set fixed to a certain value. In this way the synthesizer is capable of creating a wide range of simple and complex timbres. The basic subtractive synthesis inspired architecture is useful for creating bass sounds or other timbres for mainly melodic use. Generating percussive timbres is possible since envelopes with a short attack phase are available: these can modulate the white noise amount (*e.g.* for high hats) or, for example, the emulated filter frequency which in combination with a strong resonance can create the short sine sweep that is a typical building block for drum synthesis. Similar to this, long envelopes enable synthesizing atmospheric timbres that slowly evolve, especially in combination with the post-processing effects. To conclude, the synthesizer design shown has 16 parameters that can be controlled externally.

9.4.2 Corpus

Except for the routing, all parameters can be set at a floating-point precision, thus, to generate the corpus and overcome computational problems as well as to reduce the amount of sample points to make the projection feasible, it was necessary to coarsely discretize the parameter ranges. These ranges have been chosen after initial listening tests that weighed large timbral ranges against useful variations of similar timbres. As such, the parameters have been individually discretized to at most 20 steps, which resulted in $|\mathcal{P}| \approx 10^{15}$ possible parameter combinations.

· ◇ ·

Ideally, we would like to cover the whole extent of timbral varieties in the discretized parameter space for the corpus. However, given the cardinality of $\mathcal{P}$, it becomes clear that even synthesizing these parameter configurations is not feasible. Hence, the corpus can be created either by an expert, automatically by performing some high dimensional search method or a mixture of both. We applied the latter, as a heuristic method which makes use of audio similarity.

· ◇ ·

$\mathcal{P}$ forms a hypercube if we interpret the normalized parameters as orthogonal axes. Thus, within the volume of such a hypercube the set of points represent valid parameter configurations. A hypercube can be further split along any axis, creating two siblings with a smaller volume and therefore less included parameter configurations. Furthermore, we can determine an estimate of the similarity of the sounds within a hypercube if we measure the similarity for the *synthesized* audio that two parameter configurations at opposing vertices on each side of such a split represent.

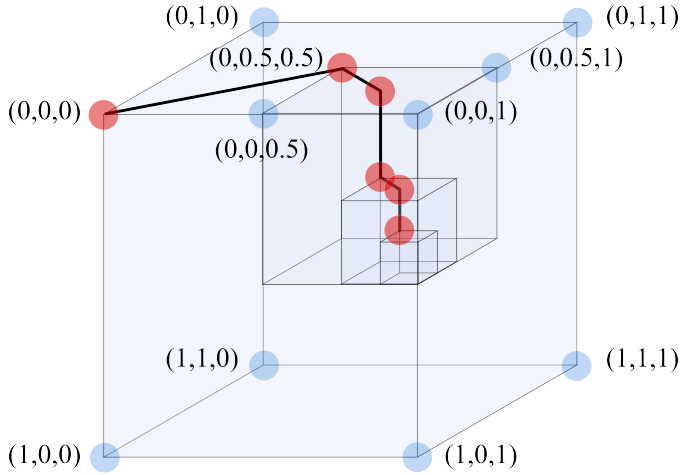


Figure 9.4: Example plot of the search method for three dimensions (three parameters)

Using this heuristic, we are essentially performing a multi-dimensional binary search, creating dissimilar sounds along a path that eventually leads to a recursively generated volume containing mostly similar ones. Figure 9.4 shows this scheme for three parameters (three dimensions). At every sample point of a value a new sound is generated and added to the corpus. The path itself is only relevant for the traversal algorithm.

· ◇ ·

We can gain a speed-up for the search method as the produced siblings are independent and can be processed in parallel. Given a set of parameter configurations - or presets - that an expert has created, the method can search for sounds in between these according to the similarity measure; each pair of presets is then interpreted as two extreme vertices describing a hypercube uniquely. We used the similarity measure proposed in [Pam04], as its robustness to arbitrary sounds has been shown. The nature of VPS to predictably generate, for our ears, mostly musical spectral effects makes this method applicable. Note that the fundamental pitch for each sound has been normalized to the note *C*, differing only in the octave (-5 to +5 from middle $C \approx 261.6$Hz) for the whole data set. The length of the generated samples for the subsequent feature analysis has been set to 4 seconds since longer samples significantly increase the processing time for the similarity analysis.

9.4.3 Implementation Details

The implementation made use of the functional programming language Scala and the aforementioned *Akka* toolkit for actor based programming (section 6.1.4). The system consists of two types of actors, Worker and Coordinator: the former performs the rendering of the audio based on parameter settings and the distance measurement, the latter generates the parameter settings. The parameter settings are themselves represented as infinite but stateful streams of coordinate vectors since generating

coordinates *a priori* would either exceed the heap or stack memory. These represent the volumetric coordinates (two endpoints of a hypercube). The coordinate data (parameter resolution) is discretized to a certain grid size. Workers request a stream from the Coordinator and perform their processing until, by the distance measure, two parameter settings are seen as similar. Then a new stream is requested. The partitioning of the stream by the Coordinator was performed as described in the previous section. Workers communicate bidirectionally to instances of MATLAB and SuperCollider (running in non-realtime rendering mode) via standard Unix FIFO queues. For a parameter setting, the Worker generates the source code of the specifically parameterized synthesizer definition code which is then received, interpreted and executed by the SuperCollider language client. When both parameter settings are rendered, MATLAB is informed to perform the distance measurement and send back the results. To allow for a roll-back of the system in case of a failure, the already processed coordinate data and file locations are saved into a database (*MongoDB* [Mon09a]). Additionally, MATLAB instances were re-spawned online when they became unresponsive[1].

[1] During development several bugs in the MATLAB implementation were encountered

· ◇ ·

The final configuration involved running the Workers in a thread-pool of size 32 and 16 instances of MATLAB and SuperCollider. The time spent generating roughly 1000000 parameter configurations was 23 days on a 12 core dual Intel XEON 2630 with 64GiB main memory. The smaller data set used in the evaluated prototype took around two days to calculate. Future work would benefit from a global gradient or one that takes the neighbourhood of parameter settings into consideration to speed up the search. As a final remark, it should be noted that the feature analysis performed by MATLAB was the most time consuming step in the processing chain. Therefore, a distance measure that is more efficient to calculate but which also is perceptually meaningful should be preferred in future iterations.

9.5 Features

We performed the analysis of the corpus to derive only high-level timbral features. First the samples were normalized, the preceding and subsequent silence was trimmed, and finally they were resampled to 22kHz sample rate. Then all audio files were analyzed using the *MIRToolbox* [LTE08] to generate audio features (30) in accordance with the presented focus, namely features that relate to perceptually relevant timbral and dynamic qualities of these sounds only (guideline S3). These generated features are largely time series data, which, in our use-case, are difficult to integrate: First technically difficult, as the synthesis uses modulators whose state would have to be saved with the generated audio in order to properly represent every frame of a time-series feature in a $t_i \leftrightarrow p_i$ mapping. Second, conceptually difficult, as each sound in the timbre space is then represented as a path. A visualization of this may overburden users, but it also requires a more complex user interface and more interaction as path operations are now the canonical way of exploring and designing sound. And third, practically difficult, as the memory requirements for the visualization method is $\mathcal{O}(n^2)$ and our feature data set exceeds 100GiB. Thus we collapsed the originally 30 dimensional time series data onto a single 368 dimensional feature vector by extracting statistical properties of the features. These statistical properties cover *inter alia*: variance, kurtosis, entropy, or periodicity were calculated for each time-series feature. Feature vectors containing undefined

values were removed while the remaining ones were standardized by Z-Score transformation as suggested in [SEL11]. Since the projection method, which will be introduced shortly, is not well suited for this high dimensionality of input data, we performed a greedy forward feature selection yielding 50 features. Furthermore, the original synthesis parameters were added to the feature vectors in order to include to some degree information about the temporal evolution of a sound. Initial experiments by listening tests using a first iteration of our prototypical application revealed that this has a positive impact on the quality of the mapping. We allot this finding to reducing the amount of ambiguous information for the entire feature vector, as the statistics based conversion reduced the temporal descriptiveness for some of the original features (*e.g.* the statistical properties of the spectral centroid for a sound being played back in forward or reverse are similar).

9.6 Mapping & Visualization

A reduction of the high dimensional timbre space to two dimensions is deemed especially helpful in the context of multi-touch applications since the lower dimensional representation can be directly used for visualization [Kir+12] and interaction. As pointed out earlier, the disadvantage of using SOMs for this task is that they do not guarantee certain properties for our use case, namely, the convergence of the algorithm and the preservation of topology (guideline S4). Therefore we preferred Generative Topographic Mapping (GTM) [BSW98], which exhibits these characteristics. As a probabilistic method, the GTM defines a probability density function modelling how well a set of low L-dimensional latent variables, x_k, is mapped to high D-dimensional data-points t_i by the function y. In our case, L is the 2D visualization space and D is the feature space.

· ◇ ·

The mapping y defines a transformation carried out by weighting non-linear basis functions, $\phi(x)$, which are centred uniformly on a grid in latent space. As with the presented use-case, these basis functions are typically radially symmetric Gaussians with a common width parameter σ. We can express y in the form of a generalized (linear) regression model that makes use of a weight matrix W, such that

$$y(x; W) = W\phi(x) \tag{9.1}$$

So far, the intrinsic dimensionality of the mapping in data space is L. Only using this strictly confined L-dimensional manifold does not allow for some variance between the observed variables (the feature data) and the images of the latent variables. Therefore, the manifold is convolved with an isotropic Gaussian noise distribution with an inverse variance β which gives it some volume. Thus, for a data point t given a latent point x_i a probability of fitting can be given as:

$$p(t|x_i, W, \beta) = \left(\frac{\beta}{2\pi}\right)^{D/2} \exp\left(-\frac{\beta}{2}\|t - y(x_i, W)\|^2\right) \tag{9.2}$$

As stated before the Gaussians are positioned at regular points on the grid, this prior distribution $p(x)$ can be expressed as a sum of delta functions:

$$p(x) = \frac{1}{K}\sum_{i=1}^{K}\delta(x - x_i) \tag{9.3}$$

Thus, the distribution in data space,

$$p(t|W,\beta) = \int p(t|x,W,\beta)p(x)\mathrm{d}x \tag{9.4}$$

takes the form of

$$p(t|W,\beta) = \frac{1}{K}\sum_{i=1}^{K} p(t|x_i,W,\beta) \tag{9.5}$$

In this way, the GTM can be seen as a constrained Gaussian Mixture Model. The mapping method is shown schematically in figure 9.5.

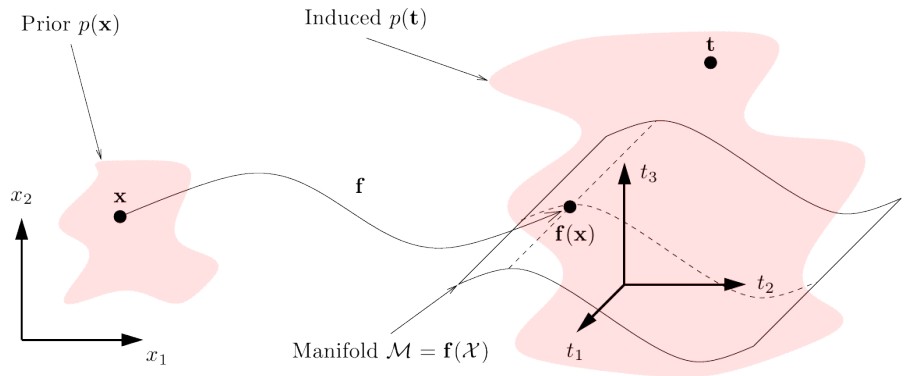


Figure 9.5: The non-linear mapping of the GTM defines a manifold that is embedded into data space (taken from [BSW98])

· ◇ ·

The probabilistic properties allow to evaluate the quality of the mapping as data log-likelihood

$$\mathcal{L}(W,\beta) = \ln \prod_{n=1}^{N} \frac{1}{K}\sum_{i=1}^{K} p(t_n|x_i,W,\beta) \tag{9.6}$$

Therefore parameters β and the weightings W can be determined by Maximum Likelihood using the Expectation Maximization algorithm [DLR77], which generates the projection. The continuous and smooth nature of y enables the topology preserving property of the GTM - neighbour points in the latent space remain neighbour points in data space. This smoothness can be controlled by the parameters of the Gaussian basis functions set beforehand. At the end of the algorithm, the responsibility (probability) of each latent variable having generated each data point can be evaluated. As latent variables are arranged on a grid, the position of each down-projected data point can be determined by weighting the latent variables' grid positions with their responsibility for it. In our case it is further necessary to perform a Z-Score transformation [SEL11] of the feature data since the variance of the features diverge vastly (by the factor $\approx 10^9$) but the global parameter β applies to all dimensions.

· ◇ ·

To help users differentiate more easily between the clusters of points in the projection, we performed a rough colour coding to indicate cluster membership. Since the GTM is topology preserving, one can apply K-Means Clustering in the feature space ($K \approx 50$) and assign colours accordingly. As we know which data point is projected onto which 2D position, we can construct an element-wise pseudo-bijective mapping. For interaction, in

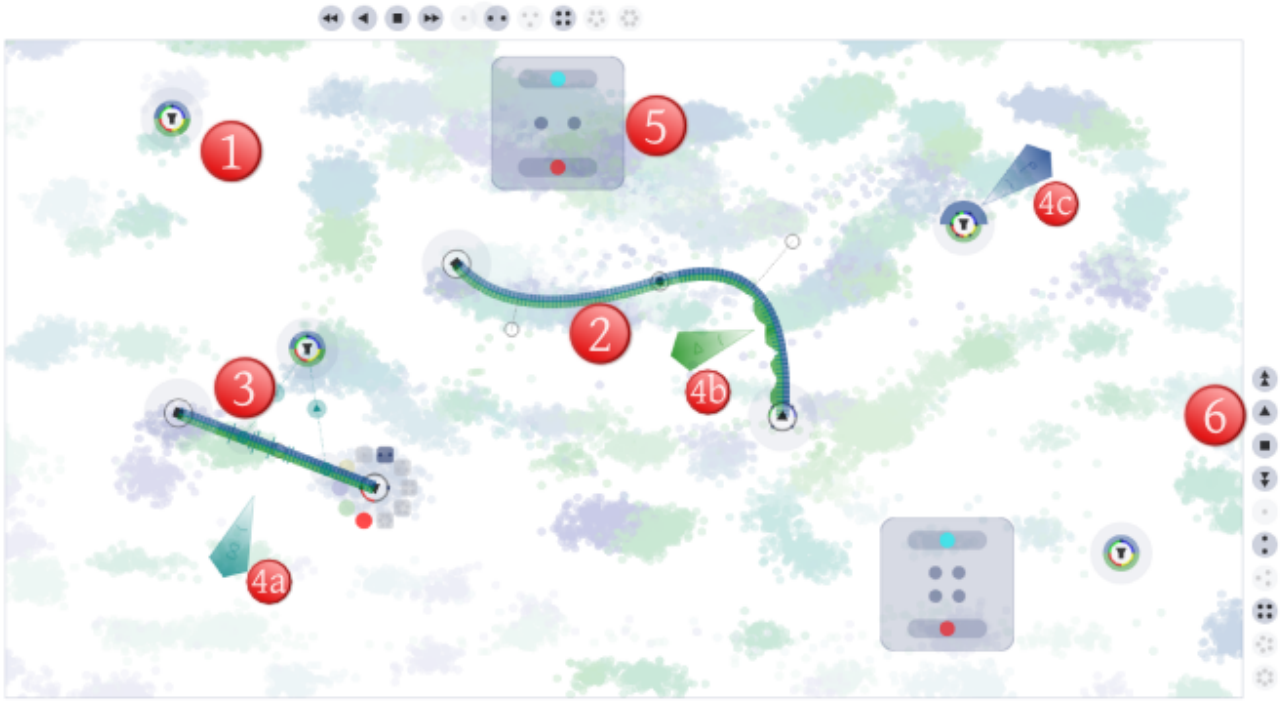


Figure 9.6: Screen-shot of our prototype showing Nodes (1), Paths (2-3) and tools (4a-4c) as well as global controls for tonal parameters (5) and playback (6)

order to find the nearest projected point in latent space from an arbitrary point (*e.g.* a user's touch) in the 2D space in real time, K-Nearest Neighbour search with $K = 1$ using a KD-tree was applied. The resulting time-complexity of $\mathcal{O}(k \log n)$ with $\approx 10^6$ points permits comparatively quick look ups. In section 9.9 we will give an example of how we can benefit from more neighbours. Finally, we applied a generic hash map to associate a feature vector with the respective set of generating parameters. We will call the visualization, $p_i \rightarrow s_i \rightarrow t_i \rightarrow x_i$, and the mapping, $x_i \rightarrow t_i \rightarrow s_i \rightarrow p_i$, *Timbre Surface* for future reference.

9.7 Prototype

We formulated the following constraints and requirements to frame the conceptual design of our prototypical application:

Number of users: The physical dimensions of our multi-touch table is 56", thus we limit the number of simultaneously collaborating users to five.

Expertise of users: As argued in the introductory section, the focus is to include novices but also experts into the collaborative sound design process.

User objective: We'd like to provide a tool for the explorative engagement with sounds but also their purposeful construction.

Situational context: Rather than to casually engage with the application, we expect users to get together purposefully. So we deem some explanatory exposition with respect to the interaction language and expressive possibilities permissible (section 1.2).

Mode of Collaboration: Possible modes among others are distributed leadership, turn-taking or a fully democratic process (section 5.2). We opted for the latter since this allows us to evaluate whether the proposed synthesis and interaction method aid users in comparison to a more conservative approach.

Facilitation of Collaboration: Besides using methodologies that aim to foster awareness, shareability and articulation, we aim at supporting the concept of public and private (acoustic) spaces and different levels of task coupling. Private spaces have been shown to give users more creative freedom to formulate contributions [FB13].

As stated earlier, users can design their sounds in a tonal context. For this, streams of harmonically fitting note events are generated and can be set to transpose sounds within the limit of one octave. This way, we aim to preserve the validity of the timbral mapping [Mar+03] while providing a useful tonal extension. The note events were generated from another computer based on distinct chord progressions that were pre-programmed and running in phase-synchronous loops in the DAW *Live* [Abl02]. The loops were then arpeggiated using stock MIDI effects and sent as distinct note stream via a dedicated MIDI-channel to the synthesizer instances in the prototype. The generating computer and the one running the prototype application were linked using a bidirectional MIDI control setup (two connected MIDI interfaces), such that the prototype could alter the arpeggiators' step and gate lengths on the generating computer while also receiving the note data streams. This setup is shown in figure 9.7. Private acoustic spaces were supported by having separate headphone output channels for every user that can be routed freely to hear any sound being played. Thus users are able to tune into the sound that collaborators design or set their sound public for the others to hear. This was facilitated by an additional switching matrix for each synthesizer instance (not discussed and shown in section 9.4.1) to send the synthesizer output to the dedicated hardware audio channels where the headphones were connected to.

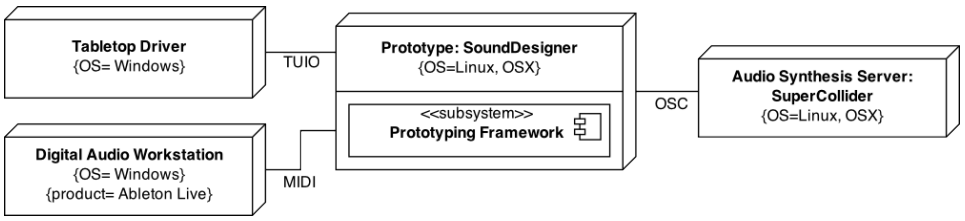


Figure 9.7: UML deployment diagram of the presented prototype

· ◇ ·

The Timbre Surface is at the core of the operational design. It spans the whole background of the user interface. Furthermore, we use the concept of a *Node* (fig. 9.6, item 1), a visually represented, draggable entity to facilitate the selection and playback of sounds according to its 2D position on the Timbre Surface. Nodes can be created and removed by simple gestures as seen fit and manipulated to change the sound's pitch and volume but also to set the incoming note stream and headphone output. These properties are directly visually indicated to support awareness.

· ◇ ·

Dragging a Node updates the synthesizer's parameters according to its position; the respective sound will be immediately heard. Continuous playback can be toggled with an additional button. To create sounds that

change over time, Nodes can be connected to form Paths (fig. 9.6, item 2). This expresses a timed motion over the Timbre Surface whose slope in between two Nodes can be altered with control points of a Bezier curve. This is the Timbre Space equivalent of parameter automation common in digital audio workstations. However, every Path represents a separate time-line. The segments of a Path are used for visualization, showing the progress of playback but also for representation of the properties duration, volume and pitch at a certain point in time. By means of additional playback nodes and the possibility to chain paths (fig. 9.6, item 3) more complex sequences of sounds can be created. To modify the volume, pitch and time of a Path or Node, we followed a tool-based approach. This circumvents issues with ambiguous input (multi-touch gestures and parallel input) and complex menu-driven command chains. For each of these three properties tools can be instantiated which can then be dragged and "docked" to the item to be modified. Then, by subsequent dragging of the separately highlighted tip of the tool, the property is modified accordingly. Figure 9.6 4a, 4b, 4c show the tools for changing the duration of a segment, the volume and the pitch.

· ◇ ·

Apart from feedback that indicates the playback state of a Node or Path, also feedforward has been applied. This was implemented to show users which Nodes or Paths could be connected while the respective gesture was performed.

· ◇ ·

A global menu can be used change the playback of all Nodes and Paths but also to show the widgets that manipulate the note streams. They show two parameters that modify *e.g.* the note length and frequency of an arpeggiated sequence of pre-programmed chord progressions. Changing these parameters may sometimes be necessary to emphasize a desired effect, *e.g.* use longer notes for atmospheric sounds and shorter ones for percussive sounds. We see this as suboptimal as this information could be derived from a more complete audio feature analysis, this is open for improvements of future works. We also did not implement the synchronization of the playback of paths to the note events themselves, so their playback is independent of incoming events. We see this as permissive as such synchronization can be interpreted in various ways, thus it would require additional complexity to the user interface.

9.8 Evaluation

We created a data set containing 60,000 sounds, which is based on roughly 1,500 presets chosen by an expert and on the generative method described above. For the evaluation, we conducted a user study lasting one week that pursued the goal of comparing our approach to a classical one and to evaluate the practical design of the application with respect to user experience and collaboration.

9.8.1 Organization of the study

The study was subdivided into three consecutive parts: comparison, experimentation and questionnaire. 22 people took part it in groups of 2-4 collaborators. With respect to their demography, their age was between 19 and 43 years (mean 25.5) while 27% were female and 73% male. Apart from

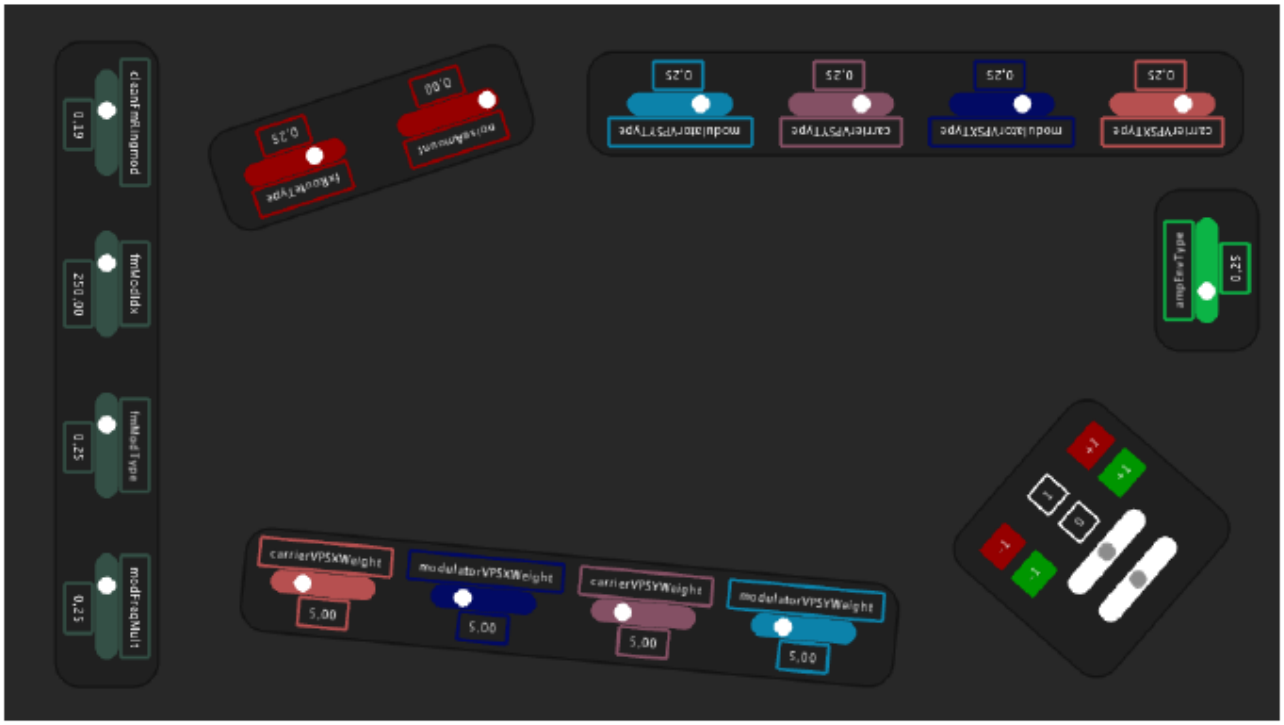


Figure 9.8: Screen-shot of the alternative prototype

one person, all participants enjoyed listening to music, most of them frequently, with a preference for Electronic and Rock/Pop. 59% of them had used music software before and 55% played an instrument - 91% also knew what a synthesizer was although only 27% had used one before. 14% had composed a musical piece and 5% designed sound. Regarding the technical knowledge, 59% had previously used a multi-touch table. Only a small fraction of participants had prior ideas of what to accomplish creatively as the vast majority did not indicate any expectations towards the application or (collaborative) sound design.

· ◇ ·

For the comparative part, we developed an application that models a classic approach to sound design (cp fig. 9.8) in a collaborative setting. We use the metaphor of a shared instrument where users can change only technical parameters of our synthesis method using simple sliders. These are grouped by functionality as widgets that can be moved and rotated freely.

· ◇ ·

After a short explanation, users were given 5 minutes to experiment with the classic approach and after that, again after an introduction, for the same duration to experiment with the proposed one using only equivalent activities (node creation and dragging). We are well aware of the different approaches to collaboration that both applications offer, however with respect to the functionality, we regard this comparison as permissible. For the next part, we presented all of the remaining functionality and gave participants 25 minutes to delve deeper into the application but also to get accustomed to each other in hope that they would gradually focus on the collaborative aspects of sound-design (9.9). A computer-mediated questionnaire concluded the study. Alongside questions regarding demographic data, we addressed the comparison-, usability-, interaction- & information design, visual design and finally the collaboration itself. Most questions used a 5-level Likert scale.



Figure 9.9: Participants of the user study evaluating our prototype

9.8.2 Results

The questionnaire showed mixed results regarding the overall ease of use, mental effort and the perceived ability to *purposefully* execute ideas for our new application in comparison to the classical one. Corroborating the latter, participants were not always able to realise their musical and timbral ideas to a large part because they felt that finding specific sounds quickly was rather hard. With respect to collaboration, it was stated that sharing and continuing work of others was not sufficiently *facilitated* while it was *perceived as easy*. This means that articulation was perceived to be fostered, as opposed to shareability. Reasons given for this were first that wearing headphones was seen as a complication for communication, thus being detrimental to the process of collaboration.

· ◇ ·

Apart from these mixed results, the feedback for the application was very positive. The comparison between the classic and our approach showed that ours was deemed more musically inspiring and incited people more to experiment and collaborate. It was further stated that it helped to obtain more musical results both alone and in the group. The relationship between input and auditive output was found to be more understandable. Somewhat surprisingly, but in line with the previous discussion, people perceived the new application as being able to provide more freedom in creating timbres although technically this is not the case. Regarding utility, most participants were able to find interesting timbres and create interesting complex ones. Most interestingly, a minority stated that they had been shown new vistas with respect to music and timbre. In relation to engagement and pleasure, the majority experienced Flow stating that they felt immersed. Collaboratively designing and experimenting with sound were rated as providing fun while it was felt that the collaboration in general had been fostered with enough access points to join in. In terms of awareness and shareability, the possibility to experiment in a private auditory space was received very positively. Finally, the application was favorably reviewed concerning the interaction and information design, as well as concerning aesthetics. The main UI control components, the timbre visualization, nodes and paths but also animations representing transitions, feedback and feedforward were assessed to be meaningful and aesthetically pleasing.

9.9 Discussion & Conclusion

The evaluation revealed shortcomings with respect to Navigation and Awareness. Regarding the first shortcoming, users stated that they found Timbre

Surface incoherent as small changes in position did not translate to small changes in timbre and that the clusters in the projection did not always have a comprehensible inter-relationship. The quality of the Timbre Surface depends crucially on the quality of the features. In this way the issue can be remedied with a different set of timbral features or different encodings thereof, such as the correlation of features for a sound. Additionally, the negative evaluation results with respect to the precision of the Timbre Surface led to a subsequent experiment investigating the influence of more thorough feature selection methods via observing the likelihoods involved in the GTM. The results showed that these methods can improve the quality of the projection significantly. The GTM method itself also provides parameters that can be further adjusted. Real-time interpolation of the sample points could lead to a smaller data set and therefore disentangle the visualization but can also lead to a homogenization of parameter settings depending on their neighborhood. For a quick evaluation of this method we used the nearest 8 neighbors of a position and weighted their parameters according to the Euclidean distance from that position. This created a sufficiently smooth mapping where gradual changes in the 2D position led to gradual changes in the aural output. However, this removed many of the original timbres from the Timbre Surface since this linear interpolation does not inherit the non-linear nature of the GTM. Hence, a more complex interpolation method is needed which takes the GTM into account. For example, the gradient of the generated responsibilities could be used for weighting. Another approach is to introduce complementary UI tools, that can pull clusters apart (*e.g.* a "magnifing glass").

· ◇ ·

With respect to the second shortcoming, participants stated that the issue is mainly the control over the private space and the related default settings used. We have seen in the previous section that this resulted in a degradation of directedness of communication and shareability. Similar concerns have been raised in [Xam+14] and observed in [BP00]. Another related study revealed that the use of headphones was detrimental for awareness at the individual and group level, and thus for coordination because there was not enough overlap in the interface representation of private and acoustic spaces and the way audio was delivered to users [FB13]. The causes for these issues are therefore related to control, representation and feedback in the application interface and conceptual with respect to audio delivery. In this way, there may be a multitude of possible solutions. However, removing private spaces and therefore relying on public acoustic spaces only does not appear to be an option as this was regarded a *necessity* by a large majority of the participants. As pointed out in [BP00], audio distribution with a public channel is problematic for novices since they have difficulty in identifying their own contributions in the mix. Another problem is that private spaces better support iterative development of ideas and experimentation since the user decides when to commit contributions to the global mix. On the one hand, this may reduce disruption of the global musical output because experimentation can happen independently of it and therefore minimize conflict potential in group collaboration. On the other hand, it can foster involvement with the group task for individuals, especially shy ones, because the intermediate results of experimentation are free from assessment of collaborators. As suggested by many participants themselves, the main issue with respect to control could be resolved by providing users the option to mute specific co-users altogether or to provide a user-specific mixer functionality. With respect to feedback, the artefacts (Nodes and Paths) could offer more means to indic-

ate information about the sound generation such as amplitude levels. This way the visualization would correlate more immediately to the sonic output. Regarding audio delivery, in [Xam+14] and [BP00] ambisonic speaker arrangements were used to give each user's sonic output a spatial quality in order to make the distinction of the various contributions more easily discernible and attributable. A similar suggestion was made at the NIME conference 2014 where this contribution was presented. For our prototype, each user may have a dedicated speaker output channel and speaker placed nearby the individual around the table. In combination with private spaces using headphones this could represent an optimal solution since the benefits of private acoustic spaces would still be accessible while contributions in the public space would be attributable acoustically.

· ◇ ·

On the positive side, our application was received as a valuable alternative to the classic approach to sound-design, being perceived as more musical, expressive, inspiring, comprehensible and inciting towards collaboration. Participants stated that they were able to find interesting and design complex timbres. This activity was perceived as pleasurable and Flow inducing. Furthermore, the collaborative use-case is seen as fun. The application as been reviewed as supporting and fostering collaboration.

Although the named shortcomings conflict with the goals set at the beginning of this chapter, we do not assess them as overly severe since they are not conceptual issues but rather technical ones that can be approached systematically within further research. To conclude, given this positive feedback and the amount of committed suggestions for improvements by participants, we see the results as satisfactory. In this regard, it is worthwhile noting that this application has been developed with the affect model presented in chapter 11 in mind, namely the automatic emotional affect estimation of timbres. Here, sounds can be given an affect value (Valence and Arousal) that can be included in the feature vector for projection. Thus allowing to design sound not only in relation to timbral descriptions but also to emotional affect related ones. As a closing remark, it should be noted that the presented approach is generic as it can be extended or replaced with other synthesis methods. In this way the system can not only be seen as abstracting over the synthesis parameters but also over synthesis architectures: similar sounds generated by different synthesis architectures would have similar feature vectors and therefore two dimensional mappings.




Part

III

Emotion




Chapter 10

Introduction to Emotion

"O music
In your depths we deposit our hearts and souls
Thou has taught us to see with our ears
And hear with our hearts"

Kahlil Gibran (1883-1931)

"[...] we say "I hear a sad violin" when it is the sound we hear, and we who are sad. Our entire auditory world, it seems, is based on a substitution of the indexed for the indexical and the effect for the cause."

Greene and Porcello, *Wired for Sound*, [GP05, p. 165]

Chapter Abstract

The following aims to define emotion in terms that are relevant to this thesis. This also includes what emotions can be evoked through music in listeners. Furthermore affective algorithmic composition is introduced which is the underlying theme for the subsequent chapters. The chapter concludes with an overview of the related contributions.

One of the most prevalent and ancient notions about music is that it expresses and conveys emotions that are perceived by the listener. These effects are certainly the reason why people listen to music to change mood or evoke emotional memories or why they engage in musical activities [EV13]. Similarly, music plays an integral part to convey the emotional experience of film [Coh01] or computer games [Mun07].

10.1 Emotion and Music

The differentiation between representing (expression) and creating emotions (inducting) for the perceiver is central in the context of this thesis. It is a viewpoint that has been advocated by Gabrielsson in [Gab02] and is

strongly supported in current empirical research [SZ01; EM13]. The former refers to the causal effect of emotion that is the result of compositional elements or the performance which are eventually perceived by the listener [Sch+13], who is thus perceiving emotional meaning. The latter is the induction of real and felt emotions (ibid). Historically, this distinction *for the listener* between perception and eliciting emotions has been controversial and a source of debate between 'cognitivist' and 'emotivist' viewpoints. Empirical research, however, suggests that the underlying processes are more complex [JV08; EV13]. For example, in a study by Kawakami [Kaw+13] it was shown that the experience listening to sad music frequently evokes positive emotions such that this process may neither completely depend on the "emotional content" or personal events. Accordingly, recent research, acknowledges the interaction of a variety of factors or mechanisms that come into play to manifest (various forms of) emotions and consequently extend the interdisciplinary notion of music psychology such as emotional contagion [EM13], brain stem reflexes, evaluative conditioning, visual imagery, musical expectancy [JV08] or aesthetic emotions (neuroaesthetics). Hence, although very recent contributions to this field propose models that unify such viewpoints (*e.g.* [Sch+13], [Sie+13] & [Jus13a]), they do currently not depict a consistent and holistic framework [EV13]. In this way, because of the multi-faceted nature of such models, it makes sense to focus mainly on the processes of expression rather than models of an emotive listening perspective. Indeed, some of these mechanics have been already operationalised such that applying them in the context of this thesis is feasible. For example, in [SZ01], Scherer argues that the emotions experienced by a piece is a multiplicative function of structural features, performance feature, contextural features and listener features. Additionally, several mechanisms of musical structure that convey emotional qualities have been proposed and confirmed empirically; summaries are given in [GL01], [Liv+10] and [GD07].

· ◇ ·

In general, emotions such as *soothing, aggressive, happy, sad* can be characterized in terms of valence and arousal. Here, valence refers to the pleasantness and arousal to the activation or intensity of an emotion. Research regarding this two dimensional structure indicates that it is a fundamental representation of musical emotions [GD07]. Furthermore, there is evidence that valence and arousal are *"pancultural [...] and present in young children"* [Bar98]. Historically, this model goes back to Hevner [Hev36] and has been elaborated upon by Russel [Rus80] in his well known circumplex Valence / Arousal model. While the former is a discrete or categorial representation of emotions on a two dimensional plane, the latter is grounded in a continuous or dimensional representation (see fig. 10.1). However, both viewpoints have been pursued in subsequent research. Making a distinction between these two is a source of debate and also depends on the perspective or individual evaluating emotions [Bar98]. In this context, Juslin makes valid points against a dimensional model:

> " *[...] dimensional models appear too reductionist. In the circumplex model, two emotions that are placed in the same position in the circular matrix may be very different. For example, anger and fear are two emotions that are highly correlated within this model because they are both high in arousal and unpleasantness. Yet they are very different in terms of their implications for the organism (Lazarus, 1991). Furthermore, musical expressions of the two emotions are quite different [...]* " [Jus13b]

Nevertheless, in the scope of this thesis the dimensional model will be used. The specific motivations for this will become more apparent in chapter 12

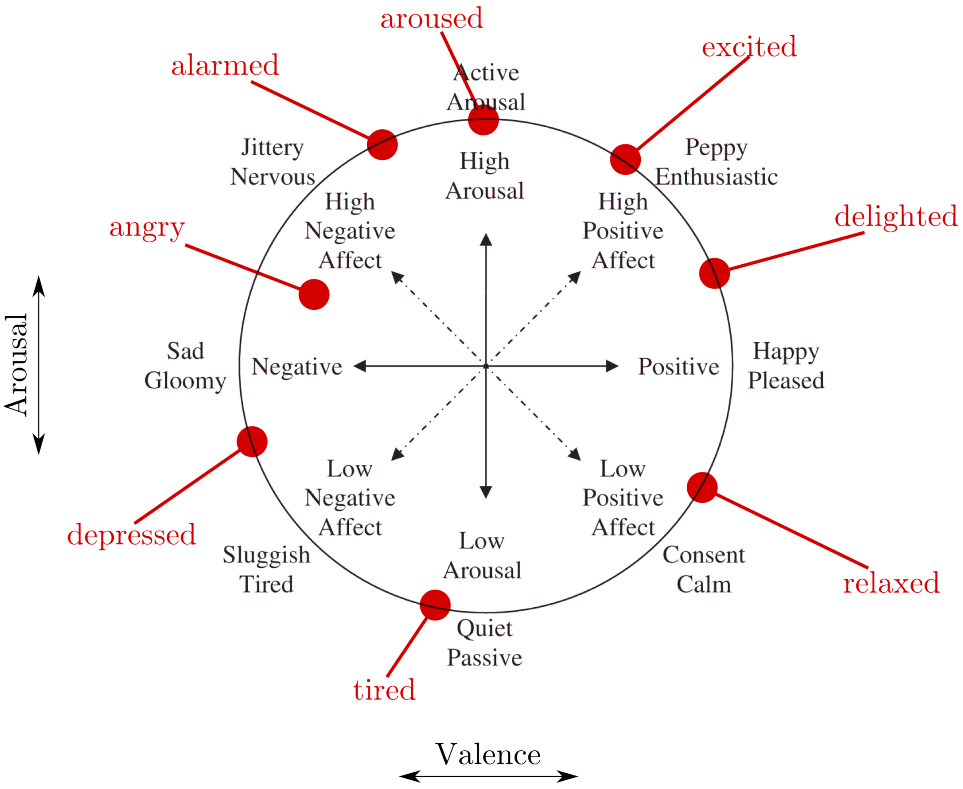


Figure 10.1: Russel's circumplex model of Valence / Arousal that locates emotions on a two dimensional plane [Rus80]

and 11, the general ones will be laid out in the following. Several transformative rules for tonal content of a composition can be defined as a continuous (or interpolating) function for a dimensional model. Thus, for the generative, rather than the analytic use-case, such continuous functions allow a degree of freedom with respect to their application in creative contexts. Disregarding the specific properties of such functions, the outcome emotions in this representation can be mixed, for example (cp. 12). Similarly, for a machine learning approach, basing the error measure on distance on the two dimensional plane rather than hard class membership allows to relax the error measure (cp. 11).

· ◇ ·

Instead of further distinguishing between different definitions, contexts and implications of the term *emotion*, the term *affect* is preferred here because it is an *"[...] umbrella term that covers all evaluative - or valenced (i.e., positive/negative) states such as emotion, mood, and preference"* [JV08, p.10]. Thus, this simplifies but also diffuses the role of emotion in the following chapters. On the other hand, however, this diffusion allows to keep the complexity of certain computational model manageable. This is especially the case for chapter 11, where a model for the affective analysis of timbre is derived. Otherwise, such a model would also have to cover the different viewpoints of emotions (*e.g.* the listener and composer perspective, perception and induction of emotion, emotion versus mood).

· ◇ ·

Hence, a Valence / Arousal (V/A) space is considered useful for our purposes since it eliminates ambiguity and provides a consistent (global) model of affect. As such, it has been applied to a large body of related work (*e.g.* [EFA12; Sco+12; SSK12]).

10.2 Which Emotions Can Music Express?

If we are to support -in some abstract way- the expression of emotion, then the question is, what emotions can be communicated. Or, put differently, what are the contents of this communication that listeners may perceive in music?

	Kreutz (2000)	Lindström et al. (2003)	Juslin and Laukka (2004)
Subjects	**50 students**	**135 expert musicians**	**141 volunteers**
No. of emotions Subjects	**32**	**38**	**38**
RANK ORDERING			
1.	**Happiness**	**Joy**	**Joy**
2.	**Sadness**	**Sadness**	**Sadness**
3.	Desire	**Anxiety**	**Love**
4.	Pain	**Love**	Calm
5.	Unrest	Calm	**Anger**
6.	**Anger**	Tension	**Tenderness**
7.	**Love**	Humour	Longing
8.	Loneliness	Pain	Solemnity
9.	**Fear**	**Tenderness**	**Anxiety**
10.	Despair	**Anger**	Hate

Figure 10.2: Highly rated emotions from several studies evaluating (musically trained) listeners' opinions about what emotions music can express. Taken from [Jus13b]

· ◇ ·

In a recent contribution by Juslin [Jus13b], which summarizes previous research, it is argued that, per-se, any subjective emotion can be interpreted by a listener[1]. However, regarding the objective differentiation of communicated emotions, there are core emotions that may be extended by additional layers of expression (ibid). The latter are less cross-culturally invariant. Previous research has shown that certain emotions are easier to express in music than other ones (ibid) as supported by listener experiments and the respective studies' inter-agreement. Figure 10.2 depicts an overview of these listener experiments and the resulting highest rated emotions. Although the results are based on subjective assessment, their quantity and quality (agreement) is argued to support the position that these can be regarded as basic or core emotions. Furthermore, they are acknowledged in various other sources of related research disciplines (ibid). These include the emotions *happiness, anger, fear, surprise* and *sadness*. Basic emotions are the result of iconically coded expressions, thus expressions that are the responses based on formal similarities between the music and some other signal (ibid). Generally, these are evolutionary evolved, physiological signalling mechanisms [SZ01], *e.g.* which are primed by intonation of emotional expressions in language. Hence, the aforementioned extensions to basic emotions are either symbolic (intrinsic) or associative (indexical) and have more influence on the structure [Deu12, p. 601]. As stated before, these are not culturally invariant.

[1] It should be noted that Juslin argues from a categorial perspective.

10.3 Emotion and Timbre

The previous sections have mostly emphasized the structural and tonal aspects of music in the context of emotions. However, when regarding expression, timbral qualities also constitute a central building block of music.

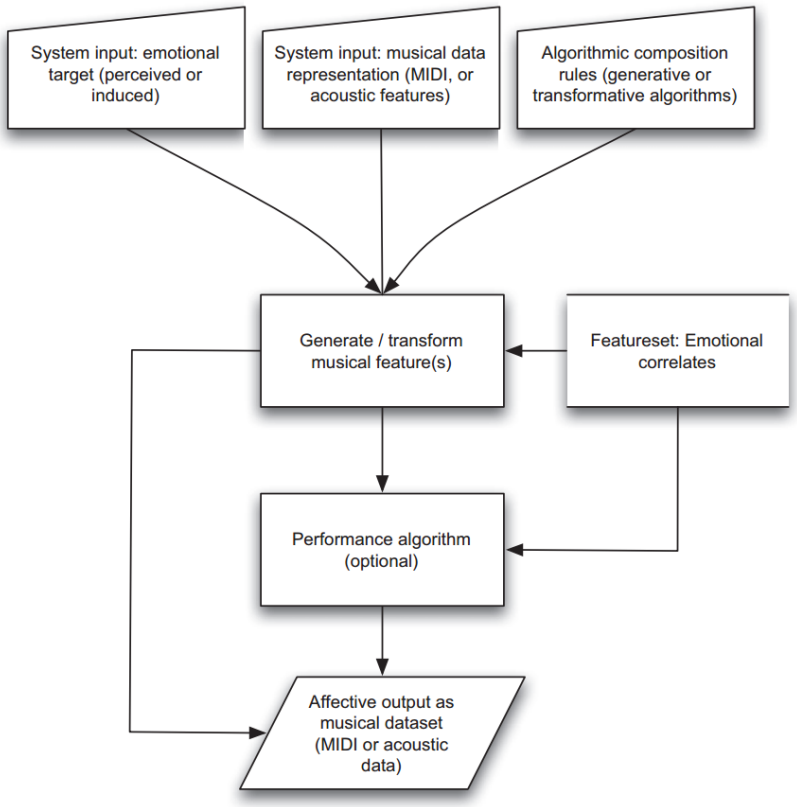


Figure 10.3: Overview of an affective algorithmic composition system with its three inputs. Taken from [Wil+14]

While it has been shown that timbre contributes to emotion judgements in larger structures of music [Hai+09], also alterations in musical emotional expression can be detected at the level of single notes [Goy+04], such that isolated instrument sounds contain cues that indicate affective expression independently of the presence or absence of other cues such as melodic ones [EFA12; Hai+09]. Because of the high inter-consistency of listeners' affects in judgements for both [EFA12; Hai+09] and the property of timbre that it is form-bearing dimension for a composer or performer, timbre can be seen as *"[...] central to interpretative decision-making and projection in performance"* as argued by Holmes [Hol11]. Thus, it is a medium of affective expression that has a large influence on how a piece or performance is received.

10.4 Affective Algorithmic Composition

Affective computing is aimed at studying and developing systems that are able to sense and recognize affect and react to it, model affect or simulate it - generally add human qualities to computational decision making [Pic95; Hud08]. From the music related perspective, affective computing is also about *stimulating* affect. Affective algorithmic composition (AAC) combines the fields of computer aided composition and emotional assessment to build perceptually motivated affective computing strategies for music generation [Wil+14]. AAC systems target an intended affective response on the basis of some generative or tranformative system. They rely on three inputs: the affective target, the algorithmic composition rules and musical data as stimulus or source material (see fig. 10.3). The system relies on musical features as emotional correlates and computational rules

that are computable or at least operationalisable and thus, most of the related research is concerned with identifying, deriving and confirming these in empirical studies.

10.5 Presented Research

The projects of the following chapters are concerned with mediating affective intent and thus they are closely related to the field of AAC. In chapter 12 a working AAC system is devised that relies on transformative rules to alter the emotional expression of user generated melodies. The prototype allows a discursive collaboration for constructing simple compositional structures. It is intentionally kept simplistic since the corresponding user study focuses not solely on the prototype, but instead also evaluates the validity of such rules in the creative context because of the lack of related studies.

· ◇ ·

Chapter 11, proposes a machine learning approach that can be used to estimate the affective content of sounds. It can be seen as a model in AAC that embeds a musical feature selection. Such a model is integral to add support for automated or user-guided selection of sounds that fit to a composition or the design of timbres. To follow the line of the argumentation in chapter 9: if such a model was available, it would be possible to control the technical synthesis parameters using alternative representations that are based on intended affect. More examples are given in section 11.2. For the purposes of this model, no user studies were performed.

Chapter 11

Emotional Affect of Sounds

Chapter Abstract

This chapter proposes a computational model that can represent a relationship between timbre and estimated emotional affect. The aim of the model is to steer or augment the control of audio synthesis processes in future prototypes. The model is based on a Machine Learning approach that is trained on a large corpus of user generated sounds and metadata.

This chapter is based on the contribution [KG13], "Towards Mapping Timbre to Emotional Affect" by Klügel and Groh.
Sections 11.1, 11.3, 11.4, 11.6 have been revised. The main section 11.5 is largely new.

11.1 Introduction

As pointed out in the previous section 10.5, one goal of this chapter is to establish systems that target a user-defined affect. In consequence, this would be an augmentation of the musical scope of individuals so that they are put in the position to implement a specific emotion they have in mind without the necessity of being aware of the various related musical construction parameters. In the contribution laid out in this chapter, the primary aim is to model a relationship between timbre and affect. As apparent from section 7.1, research concerned with this relationship from a perceptual viewpoint has not yet yielded many conclusive or holistic results. Indeed, an analytic understanding of the affect of timbre is still missing as most research in the context of music so far has focused on the effect of tempo, dynamics and mode [EFA12]. Because of this missing theoretical foundation, a Machine Learning (ML) based approach may yield productive results. The main idea is to learn these relationships from user generated data. Afterwards, individual sounds that may be part of a larger musical context can be automatically given a label or value of an expected affect for the composer or listener[1]. Since our model is able to label data autonomously after training, the labeling process can be applied to large databases of sounds with different timbres. In this way the tedious task of supervised labeling for sets of sounds with various timbres regarding the emotional affect may be avoided (*e.g.* similar to tagging sounds in section

[1] To simplify the model, we do not distinguish between the effective perspective of composer and listener.

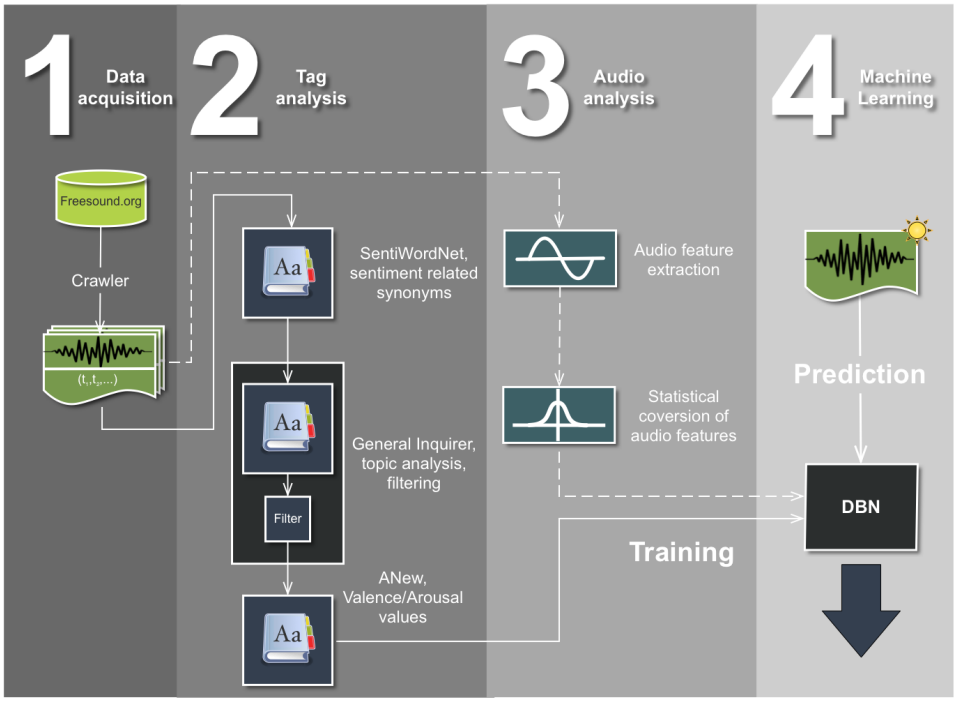


Figure 11.1: The processing data-flow of the proposed method: 1) starting with the acquisition of sounds that have been tagged by users in an online community, 2) tags are analysed for affective content. 3) Sounds are feature analysed. 4) A deep belief network is trained with this data to predict the affective content of previously unknown sounds.

9.1). Thus, this provides an approach to create new applications based on Synthesis by Analysis methods as will be exemplified in the next section.

· ◇ ·

The proposed model will be created via training a Deep Belief Network [HOT06] using a large data set created from the Freesound.org [Akk+11] database and text-mining. Figure 11.1 shows the processing dataflow of the proposed approach. It should be emphasized that the difference to the discipline of mood classification of songs in Music Information Retrieval (MIR) is the type of data analyzed and therefore the implied model: here we do not use a mixture of sounds but isolated, singular ones that are predominantly short (compared to a recording of a large musical structure) and detached from a harmonic and rhythmic context. Since songs themselves communicate emotion on additional layers (structurally) [Hai+09] and since this also implies the utilization of additional feature data for the audio analysis, a solely timbral model, such as ours, is in comparison expected to perform *a-priori* worse. Even more, since we are interested in a global model that abstracts away from subjective evaluations of affect, such a model is also bound to perform worse than individualized models (such as shown in section 11.3). Similarly, the use of a global data set that incorporates many varying affective perspectives (see section 11.4) will result in noisy training data.

· ◇ ·

The following sections are organized as follows: first potential applications of the proposed model are shown, second related work is referenced and third the acquisition of the training data set is laid out. Then the performance on a simplified training set of various common ML classifiers is evaluated which subsequently leads to the reason to employ a regression method based on Deep Belief Networks. After the presentation of this method, the chapter concludes with an outlook of future work.

11.2 Potential Applications

One use case scenario for our research is to combine such a timbre-affect model with an audio synthesis model, primarily to help novices shape sounds with respect to a desired emotional affect. Therefore, we see music creation rather than pure analysis as the main application field of our contribution, such as controlling the audio synthesis by using the emotional affect as hyper-parameter. Such work would expand upon the model and prototype shown in chapter 9. It could be implemented in the following way:

- establish a corpus of sounds generated by an audio synthesizer instantiated with permutations of its structural parameters
- retain the generating parameter settings and label each sound using the proposed model
- use the labels as meta-mapping for the structural parameters (depending on the audio synthesizer these structural parameters can be interpolated)

In this way exemplifying use cases in regards to control timbre could be:

- present the user(s) with a pre-selection of sounds fitting to the mood they want to create with a song (static mood)
- help user(s) to create transitions from one mood to another (dynamic mood)
- help users create a static mood (*e.g. calm*) but offer a set of different timbres with a similar emotional affect so that the music is still dynamic / varying

11.3 Related Work

There are several timbral qualities that have been studied and are known to relate to or to be fundamental to affect [SDP12; Pad+03]. These can be roughly grouped into spectral energy, -structure and -variation. Depending on the method, they can be quantitatively or qualitatively measured in the spectral, temporal or spectro-temporal domain; examples are the spectral centroid (geometric center of the spectrum; brightness), spectral spread (standard deviation of the spectrum), HF-LF ratio (high to low energy ratio), attack slope or inharmonicity (deviation of partials from the harmonic frequencies). In [EFA12], Eerola *et al.* performed an extensive study to relate such audio features to the affect dimensions. The analyzed sounds were orchestral samples (105) with various articulations. Within this study correlations between affect and various audio features have been found for both affect dimensions. Furthermore it was possible to construct a model to predict the affect from feature data with reasonable results via linear regression. As will become clear in the section Machine Learning, a linear model is not sufficient for our data set. It was also shown in [EFA12] that the two affect dimensions led to the most consistent results when compared to an affect model employing the three dimensions of valence, arousal and tension [SG00]. Because the aim of their study not to rely on artificial sound generation schemes is antipodal to ours, we decided not to use their provided data set. It is of particular interest for our model not to

emphasize a certain set of timbres but to allow the classification of inherently inharmonic or initially "non-musical" sounds (w.r.t. electronic music, guideline S1 in section 9.1).

· ◇ ·

In [TP13] Thorogood *et al.* present a system that can be used for such sounds[2]. The systems works as a mobile application that learns the relationship between a continuous user response in the V/A space and audio features from a collection of sounds that are played back in real time. The trained model is entirely user specific and therefore makes use of data that has been gathered reliably - from the perspective of the affective evaluation. A linear regression model was then created for every user in order to facilitate a user study to measure the models' prediction quality. In both affect dimensions, the 71% of the variance could be explained, thus indicating that these models were of practical use. However, it is concluded that the affective evaluation relies on other (latent) variables that have not been measured and therefore other modelling methods may be more suitable. While this contribution serves as an example of the feasibility of creating such affective models, the evaluating remarks of the authors and the fact our goal is to create a more generalizing, non-subjective model, our conclusion is that for our model, a deeper machine learning approach is a necessity.

[2] This contribution and the one this chapter is based on were presented at the same conference, New Interfaces for Musical Expression 2013.

· ◇ ·

Similar to our goals, Oliveira *et al.* [OC08] developed a model that used a ML approach to control the selection of timbres in relation to their affect in an automated music composition system [OC10]. Their model yields a correlation of 75% between timbre (audio features) and affect labels. The affect labels (discrete V/A space) were acquired in a previous listener study and are based on short orchestral pieces. With respect to our own research, we especially see the derivation of affect from musical pieces as problematic since a model informed by *timbral* qualities will then be biased towards the affect of the pieces themselves. Le Groux *et al.* [LV10] follow a different path by first designing a physically informed synthesizer (modal synthesis) to generate sounds in accordance to perceptually relevant features and then evaluating their impact on the affect in a listener study. They show that the spectral centroid as well as the spectral flux are directly related to arousal. However, no significant relationship to valence could be established. This and the comparatively simple synthesis model (percussive sounds) may reveal the issue that more variation in timbre is necessary to cover the wealth of affects. The low number of participants (10) may also contribute to this.

· ◇ ·

In summary, a data set for our purpose would have to consist of a variety of different timbres, musical (acoustic, synthesized) and non-musical ones, as well as covering a large spectrum of emotional affects. Furthermore, as a rule of thumb, it is usually recommended to have at least 10 to 20 times more observations than predictors (*e.g.* audio features) to be able to perform a meaningful multivariate analysis of the data. Getting hold of such a data set is therefore an important item for our work. There are related data sets available, but to our knowledge they have either limited musical relevance such as *e.g.* The International Affective Norms for Digitized Sounds (IADS-2) [BL07] which focuses on affects of real-life stimuli, or do not meet the requirement to have isolated timbres as the audio data consists of a mixture or sequence of several timbres like the large data sets used in the

MIR community (*e.g.* MTurk [Spe+11]) for song emotion (and genre) classification. To our knowledge, the only exception is the data set used by Scott *et al.* [Sco+12]. It consists of the individual tracks of a song labeled with continuous V/A labels. However, this data set is biased in timbre since it contains only 50 songs from the genre of Rock music only. This issue may be prevalent in other related MIR data sets as well (USPop2002; orchestral film scores).

11.4 Data set for Training

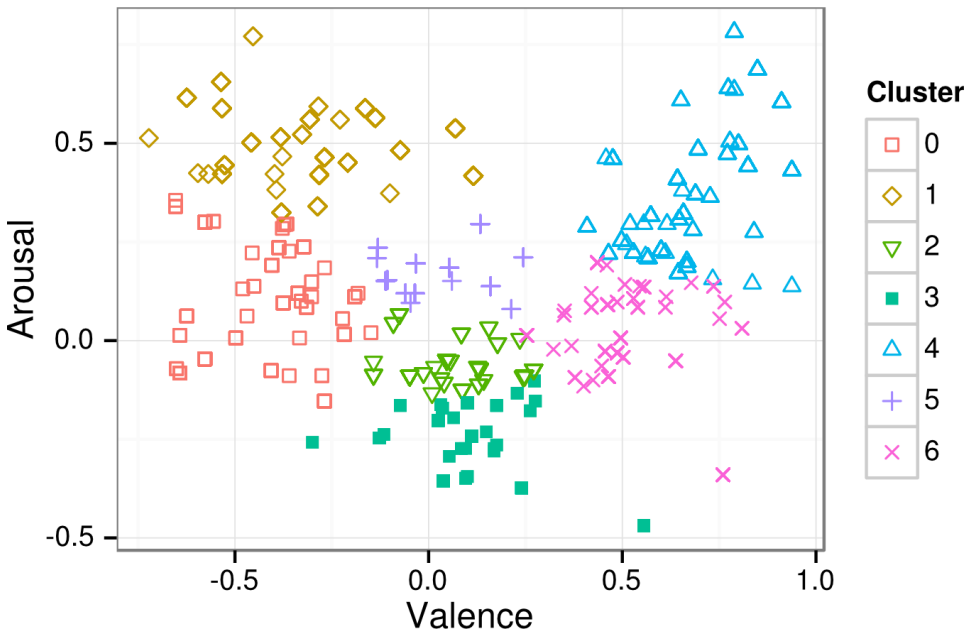


Figure 11.2: V/A values of all samples in the data set; shapes/colors specify the cluster membership for the preliminary performance analysis

Because of the general lack of a suitable data set, we decided to construct one based on a large set of samples and related meta-data downloaded from the Freesound.org [Akk+11] library. The insight into the samples' emotional affect is gained by processing the meta-data. The predictor data for training is generated by audio analysis. Freesound.org is an online collaborative sound database where people can share recorded audio clips (royalty free) and, among other things, tag these sounds. This database focuses to a large degree on the creative use of the material by sound and video artists. The main idea for the data acquisition is that the audio data can be used as source for audio features and the linked tags as source for a folksonomy discerning the emotional affect of the audio data. Thus, we downloaded 139155 samples (May 2012) with the accompanying meta-data as first step.

11.4.1 Dictionary Analysis

The tag data are based on a narrow folksonomy [Fon+12], so a single tag can be assigned only once to each sound. For the 139155 sounds, 39337 unique tags have been used, yielding an average of 6.64 tags per sound. As pointed out by Font and Serra [Fon+12], the Freesound.org folksonomy is quite noisy and therefore suffers from inconsistencies such as synonymy and polysemy which complicates the extraction of structural information.

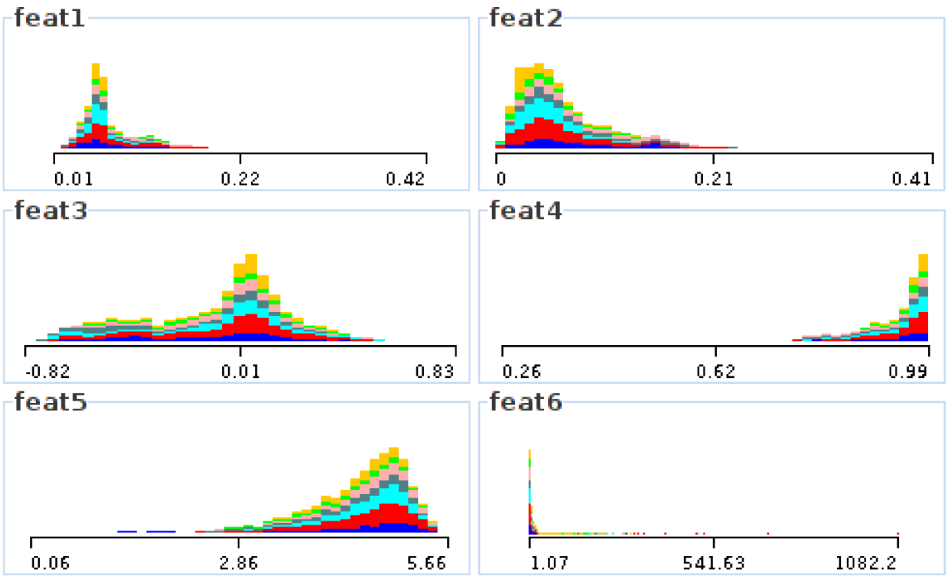


Figure 11.3: Histogram of values of top six features according to clusters (cluster membership color encoding as in fig. 11.2)

An issue of properly deriving V/A values for the tags is that the tags are part of different semantic categories (*e.g.* subjective, context, content). The goal is to estimate the affect from all categories (ideally emphasizing the *subjective* category) and disregarding all information that can lead to inconsistencies. In view of that and to relax the previously mentioned issue of noisiness, a combination of finding sentiment related synonyms for each tag and of filtering tags based on the content of the words was employed. Furthermore, tag data and dictionary data were stemmed (using SentiWordNet [BES10]).

· ◇ ·

The dictionary analysis per sound is roughly accomplished as follows: first sentiment related synonyms for each tag are retrieved, then for each of these synonyms it is evaluated whether a V/A value pair exists. If it exists, the tag and the synonym are compared against a blacklist containing terms of context and a whitelist containing terms of musical affect. The V/A values are kept if the tag or synonym are both not blacklisted or if either one of them is whitelisted, otherwise the V/A value is discarded. Each sound is associated with the mean V/A value of all tags. The sentiment related synonyms are generated with SentiWordNet3 [BES10]. It is based on a dictionary that holds 117684 synonyms. The blacklist dictionary is based on the *General Inquirer Augmented Spreadsheet* [Var02] and was specifically compiled to contain only words of matters that are considered unrelated to music and its affect. Furthermore it includes terms that are deemed to create inconsistencies, such as the word 'piano' which has a V/A value but refers to the content of the sample, therefore it would colorize the valence arousal value for an accordingly tagged sample. The dictionary of the *Musical Adjectives Project* [Dr 12], serves as whitelist. It consists of 690 collaboratively collected adjectives that describe and categorize emotions in music. Finally, The Affective Norms for English Words (ANEW) dictionary [BL99] was applied to derive V/A values. It holds 1034 English words including verbs, nouns, and adjectives whose emotional affects have been evaluated in a large study. The originally included domain of dominance has been ignored in our musical context (singular timbres).

· ◇ ·

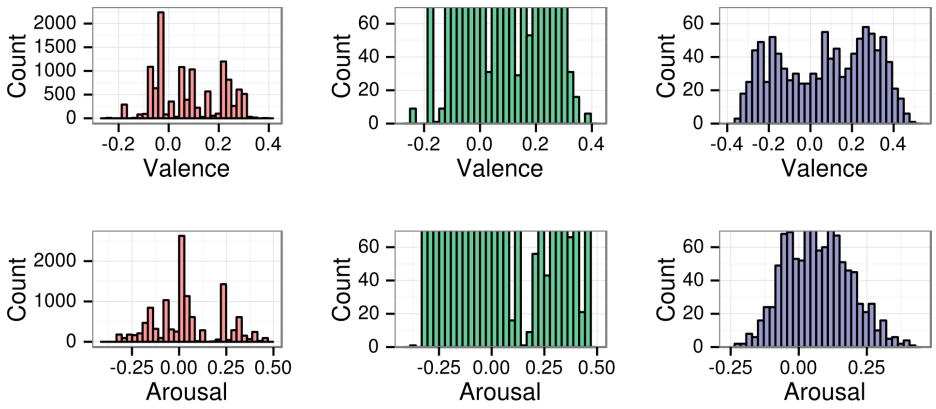


Figure 11.4: Histogram of V/A values (left to right) in our data set, the same magnified and in ANEW

In the end, 11324 (12.3%) sounds were kept from the original data set for further processing, each having a V/A value associated. Figure 11.2 shows the V/A values for all sounds in the final data set. When comparing these to the ones in the original ANEW dictionary (cp. figure 11.4), one can see that the distribution is slightly biased as singular coordinates have a disproportionately high occurrence. In the data set 203 V/A coordinates are unique which contrasts to the 813 in the dictionary. The average number of tags with a V/A value in the data set is 1.32 per sound. The number of unique tags with V/A value is 445 (an excerpt is given in table 11.1) which were resolved into 405 sentiment related synonyms, thus it may cautiously be concluded that the original tags describe differing concepts at large. Concluding, aliases have been created because different concepts (in the dictionary) have a similar emotional affect. 2.2% (*e.g.* table 11.1) of the most frequently used V/A valued tags make up 44.5% of the complete set of V/A valued tags which shows that this transformed folksonomy is less diverse in vocabulary than the original one. Nevertheless, using the data set seems plausible as the *spread* of affect in the data set is comparable to the one of the ANEW dictionary (as indicated by figure 11.4), so diversity and structure of affects can be considered to be adequately represented. It should be noted that structurally this is also true in comparison to the IADS-2 and the International Affective Picture System [LBC97] data sets.

· ◇ ·

Given our representation of affect as coordinates in the V/A space, we assume that these coordinates can be linearly combined, including even V/A coordinates of *opposite* affects (*e.g.* happy and sad), leading to a composition of affect coordinates which is representationally unambiguous. On the other hand the combination of affects in a discrete model (*e.g.* bags of labels) may not be as trivial since a simple union of labels might either lead to ambiguity (with possibly severe impact for the ML model) or sparsity of the affect model (*e.g.* when using majority voting).

11.4.2 Audio Analysis

The applied audio feature analysis is similar to the one shown in section 9.4. This resulted in 684GiB of feature data, most of them being time-series data. The size of the data had to be reduced in order to circumvent memory constraints for the applied ML-algorithms. A further justification to reduce the features to a single descriptor per sample is that the available V/A data is static. Statistical properties such as variance, kurtosis, entropy,

User tag with V/A value	Count	User tag with V/A value	Count
'dark'	1722	'dream'	294
'spooky'	832	'smooth'	274
'evil'	515	'terror'	261
'dirty'	472	'cry'	245
'happy'	339	'frightening'	236

Table 11.1: Excerpt from the 25 most frequently used sentiment related tags with V/A value

or periodicity were calculated for each time-series feature, thus the time series data was again collapsed onto a single feature descriptor. This transformation reduced the data set to 48MB. All feature vectors where then concatenated to create the observation / feature matrix. The final observation / feature matrix represented 11324 observations with 368 features each. The downside of the noisiness of our data (*e.g.* due to the empirical nature of the applied dictionaries, variations in recording quality, etc.) is that training a representative model may be difficult or impossible to achieve, the benefit however may be the applicability of such a model to real world data.

· ◇ ·

In conclusion, the data set contains 30.8 times more observations than predictors, more importantly the requirements postulated beforehand have been met inasmuch as the wealth of emotional affects and a variety of timbres from unrelated sound sources is represented.

Classifier	368 feat. error %	top 25 feat. error %	top 4 feat. error %
Naïve Bayes	70.61	67.82	74.75
Bayes Net	65.73	65.85	71.84
RandomForest	**55.59**	**52.59**	71.09
LibSVM	76.37	76.20	**70.97**

Table 11.2: Preliminary performance (classification error) of common ML-methods analysis with simplified data set importance ordered features

11.5 Machine Learning

To establish a model, the task is to find a mapping between audio features and emotional affect values. For a *preliminary* overview of how various established ML algorithms perform, the aspired mapping is *simplified*: the former continuous V/A coordinates were clustered into 7 distinct emotional affect classes which roughly resemble the discrete emotional affect classes when basic affects are represented on the V/A plane [LBC97; BL07; BL99]. Hence, the ML task was relaxed to creating a mapping from audio features to these 7 classes. Figure 11.2 shows the color encoded cluster correspondence of each sample on the V/A plane. To overcome the issues of many ML classification algorithms (such as SVM) of being sensitive to ambiguous or inconsistent features [SEL11; SDP12] a feature selection is usually applied. Although generally suggested [SEL11], we did not perform a wrapper-based feature selection since that would have involved testing all 2^{368} combinations of feature sets for each classifier. Instead, feature performance was evaluated with three sets of importance ordered features where the importance of the features was estimated using ReliefF [Sik03]. Table 11.2 shows the results of this performance evaluation

for various standard classifiers. The measure was the misclassification rate of the predicted class and the observed one. In spite of the fact that some classifiers (*e.g.* Naïve Bayes) make the assumption that the features are independent which is certainly not true for audio features, the initial idea is to see how, in general, prevalent classifiers perform. The SVM has been applied using Radial Basis Functions with a grid search to find the optimal parameters for the kernel.

· ◇ ·

One central issue regarding the feature data is shown in figure 11.3, namely that, on a per feature basis, the distribution of the features in relation to cluster membership is not discriminative. This is in accordance with observations made in other works as pointed out in [KSM10]. Although the presented algorithms have already been successfully applied in similar contexts [KSM10], for our data set the results are disappointing as all algorithms show unreliable classification performance. The results also suggest that the data set may be very noisy and may contain inconsistent labeling. Otherwise, from the empirical evidence as shown in [KSM10], the SVM based approach should have performed better. Another indicator is that RandomForest, which is generally considered to be less susceptible to these issues, performed best. Indeed, models that correspond to a distributed representation of feature data may be more appropriate, *e.g.* a vector of computational units as will be shown in the context of the Restricted Boltzmann Machines. Here, these units are not mutually exclusive such that, in comparison to models that cluster or partition the input, factors of variation can be separated [Lar+07]. It is also possible that the tested ML algorithms are, from a computational point of view, not able to find a generalizing pattern in the feature space for discrimination, or that the feature data actually originate from a higher dimensional manifold. In particular, the measurement of error may be inappropriate for the task as an error of absolute misclassification introduces an artificial discrete segmentation of the V/A space without accounting for "near misses". The latter part of this section introduces the *final* ML method and an improved measure of error, and concludes with a performance evaluation.

11.5.1 Deep Belief Networks

Informally, Deep Belief Networks (DBNs) [HOT06] share architectural similarities with neuronal networks. They belong to the class of deep architectures, which also includes Deep Convolutional Neural Networks or Deep Autoencoders. Their output is produced by consecutive layers of computational units. As a result, they allow for hierarchical learning such that a model is created progressively from a low to a high level structure throughout the layers. With this, a deep architecture may be able to model highly complex functions with only a limited number of parameters whereas the few layers of shallow models (*e.g.* linear models, single layer neural networks, kernel SVMs) may require an exponential amount of computational units [Lar+07]. A DBN is a probabilistic generative model whose building blocks (layers) are Restricted Boltzmann Machines (RBMs) which themselves model stochastic latent variables as part of a two-layer neuronal network. In this context, they are used as unsupervised feature learners. The difference to conventional networks is that RBMs are energy-based and have undirected, symmetrical connections between nodes. The restriction allows to devise an efficient training procedure [CH05]. It denotes the property that the variables in each layer are conditionally independent

from each other. Thus, represented graphically, there are no connections between nodes at the same layer as shown in figure 11.5.

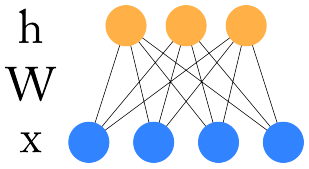


Figure 11.5: A graphical representation of a simple Restricted Boltzmann Machine

Training Restricted Boltzmann Machines

The generic RBM makes use of binomial input and hidden units. Real-valued input data can be applied with the interpretation that they form a probability of a binary event. An example for this is the encoding of grey-scale pixel values of images as binomial input in [HOT06]. Subsequent contributions proposed to extend the standard RBM to employ other types of input units which handle real valued data such as the Gaussian-Bernoulli RBM in [MDH12]. However, in this contribution the standard model will be used.

· ◇ ·

In energy-based models, the configuration of the variables of interest are associated with a scalar energy. In turn, an energy function defines a probability distribution as

$$P(v) = \frac{e^{-\text{Energy}(v)}}{Z} \tag{11.1}$$

where Z is a normalizing factor, the partition function which is in the generic case intractable since it involves an exponential sum over all configurations of v. As a consequence of the lack of interactions between units in the same layer in an RBM, the energy function is bilinear (ibid):

$$\begin{aligned} \text{Energy}(v,h) &= -\textstyle\sum_i a_i v_i - \sum_j b_j h_j - \sum_i \sum_j h_j w_{i,j} v_i \\ &= -a^\mathsf{T} v - b^\mathsf{T} h - h^\mathsf{T} W v \end{aligned} \tag{11.2}$$

The connections between visible units v_i and hidden units h_j are weighted by a matrix W, additionally the visible and hidden units can be given a bias, a_i and b_j respectively. For such a latent variable model, the probability $P(v)$ of a visible input v depends on the hidden variables h such that $P(v)$ is the marginal of $P(v,h)$ as the sum over all possible hidden layer configurations:

$$P(v) = \sum_h \frac{e^{-\text{Energy}(v,h)}}{Z} \tag{11.3}$$

As before however, since Z is intractable, calculating $P(v)$ for the RBM is not feasible.

· ◇ ·

Instead of considering all configurations of (v,h), sometimes one is interested in the conditional probability, *e.g.* the configuration of h for a given configuration of v when h can only take values from a small discrete set [Ben09]:

$$P(h|v) = \frac{e^{-\text{Energy}(v,h)}}{\sum_h e^{-\text{Energy}(v,h)}} \tag{11.4}$$

In case of Boltzmann Machines, this is tractable so that it allows us to efficiently compute the activation function for inference for each unit. In fact, because of the conditional independence of visible and hidden units given one-another, it factorizes. As such the full conditional distribution of the whole hidden layer can be written as

$$P(h|v) = \prod_j P(h_j|v) \tag{11.5}$$

which is the product over the conditional distributions for each individual hidden unit. Because of the property of symmetry of the RBM, the same can be stated for $P(v|h)$.

$$P(v|h) = \prod_i P(v_i|h) \tag{11.6}$$

As v_i and h_j are binary ($v_i, h_j \in \{0, 1\}$), the units in this RBM model Bernoulli distributions such that the probability of a hidden unit having the value one given v can be expressed as follows with help of the local Markov property:

$$\begin{aligned} P(h_j = 1|v) &= \tfrac{1}{1+e^{-(b_j + \sum_{i=1}^{m} w_{i,j} v_i)}} \\ &= \sigma(b_j + \textstyle\sum_{i=1}^{m} w_{i,j} v_i) \end{aligned} \tag{11.7}$$

and vice versa:

$$\begin{aligned} P(v_i = 1|h) &= \tfrac{1}{1+e^{-(a_i + \sum_{j=1}^{m} w_{i,j} h_j)}} \\ &= \sigma(a_i + \textstyle\sum_{j=1}^{m} w_{i,j} h_j) \end{aligned} \tag{11.8}$$

The function $\sigma(x)$ denotes the logistic sigmoid function. In this way, one can see that $P(h_j = 1|v)$ and $P(v_i = 1|h)$ describe a probabilistic version of the generic neuron activation function.

· ◇ ·

For training the RBM, one is interested in adjusting the model's parameters $\theta = (a, b, W)$ in order to maximize the probabilities assigned to instances of a training set V:

$$\arg\max_{\theta} \prod_{v \in V} P(v) \tag{11.9}$$

Equivalent to this is minimizing the average negative log-likelihood,

$$\theta_{\text{MLE}} = \text{argmin}_{\theta} \mathcal{L}(\theta)$$

$$\mathcal{L}(\theta) = \frac{1}{|V|} \sum_{v \in V} l(\theta, v) \tag{11.10}$$

$$l(\theta, v) = -\log P(v)$$

which can be performed by stochastic gradient descent. Taking equation 11.3 into account, one can see that the log-likelihood consists of two terms, l^+ and l^- [Tie08]:

$$\begin{aligned} l(\theta, v) &= -\log P(v) = -\log \textstyle\sum_h \frac{e^{-\text{Energy}(v,h)}}{Z} \\ &= -\left(\log \textstyle\sum_h e^{-\text{Energy}(v,h)} - \log Z\right) \\ &= -\left(\log \textstyle\sum_h e^{-\text{Energy}(v,h)} - \log \textstyle\sum_{x,y} e^{-\text{Energy}(x,y)}\right) \\ &= -l^+(\theta, v) + l^-(\theta, v) \end{aligned} \tag{11.11}$$

The result for the stochastic gradient of the negative log-likelihood,

$$\frac{\partial \mathcal{L}(\theta)}{\partial \theta} = \frac{1}{|V|} \sum_{v \in V} \frac{\partial l(\theta, v)}{\partial \theta} \tag{11.12}$$

$$\begin{aligned}
\frac{\partial l(\theta,v)}{\partial\theta} &= -\frac{\partial l(\theta,v)^{+}}{\partial\theta} + \frac{\partial l(\theta,v)^{-}}{\partial\theta} \\
&= \underbrace{\sum_{h} P(h|v)\frac{\partial \text{Energy}(v,h)}{\partial\theta}}_{\text{positive phase}} - \underbrace{\sum_{x,y} P(x|y)\frac{\partial \text{Energy}(x,y)}{\partial\theta}}_{\text{negative phase}} \\
&= \underbrace{\mathbb{E}_{h}\left[\frac{\partial \text{Energy}(v,h)}{\partial\theta}\middle| v\right]}_{\text{positive phase}} - \underbrace{\mathbb{E}_{x,y}\left[\frac{\partial \text{Energy}(x,y)}{\partial\theta}\right]}_{\text{negative phase}}
\end{aligned} \tag{11.13}$$

is that the positive phase is tractable, as only a sum over the hidden configurations has to be made, while the negative one is not, since an exponential sum has to be performed for visible *and* hidden configurations. Because of this, a stochastic approximation of the maximum likelihood gradient is performed by replacing the expectation by a point estimate. For this, one can apply a Gibbs sampler, which is a form of Markov-Chain Monte-Carlo (MCMC) sampling, to the joint distribution $P(v,h)$ and obtain estimates of $P(v)$ [Hin02].

· ◇ ·

Gibbs sampling is an iterative algorithm that performs N sub-sampling steps to sample the joint distribution of N random variables, $X = (X_1, \ldots, X_N)$. Such a step samples $X_i^{(j)}$ (the ith sample of the jth update step) by drawing a sample from the conditional distribution in the form of $X_i = P(X_i|X_1^{(j)} \ldots X_{i-1}^{(j)}, X_{i+1}^{(j-1)} \ldots X_N^{(j-1)})$. $X_i^{(0)}$ is chosen randomly. In case of the RBM this means alternating between sampling $h^{(n+1)} \sim P(h|v^n)$ and $v^{n+1} \sim P(v|h^{n+1})$ where each sample is randomly chosen to be 1 with the respective probability. Since the visible and hidden units are per layer conditionally independent, Gibbs sampling can be performed very efficiently for the RBM since the updates can be performed in parallel. Although the samples v^t, h^t guaranteed to be unbiased samples of $P(v,h)$ when $t \to \inf$, for training this computation is prohibitively expensive [Fis10].

· ◇ ·

Contrastive Divergence (CD) is an approach that only performs k steps of Gibbs sampling instead of waiting for the equilibrium to be reached, thus introducing a bias. This method makes a second approximation, namely that the number of steps are sufficient to derive the direction of the gradient (although the size of it may not be accurate) which leads to good parameter estimates [Ben09]. An additional modification to generic Gibbs sampling is that the start samples are taken from an empirical distribution, the data samples themselves. The idea is that the empirical distribution is similar to the one of the model as it is expected to model the true underlying distribution of the data. Thus, from this point of view, initializing the MCMC with a training example leads to a state of the chain that is already close to convergence (ibid). In practice it has been shown that even a single iteration step, $k = 1$ works sufficiently well. To summarize, given a training sample v_t one can obtain $\tilde{h}_t$ to compute a point estimate for the positive phase,

$$\mathbb{E}_{h}\left[\frac{\partial \text{Energy}(v_t,h)}{\partial\theta}\middle| v_t\right] \approx \frac{\partial \text{Energy}(v_t,\tilde{h}_t)}{\partial\theta} \tag{11.14}$$

and similarly, with k steps of Gibbs sampling $\tilde{v}$ and $\tilde{h} \sim P(h|\tilde{v})$ can be generated for the point estimate of the negative phase,

$$\mathbb{E}_{v,h}\left[\frac{\partial \text{Energy}(v,h)}{\partial\theta}\right] \approx \frac{\partial \text{Energy}(\tilde{v},\tilde{h})}{\partial\theta} \tag{11.15}$$

Looking at the gradient procedure, this means that the probability of the observation v_t with its hidden layer $\tilde{h}_t$ is increased (positive phase) while the probability of observing $\tilde{v}$ under the model's distribution is decreased. Accordingly, when this procedure is part of an iteration adjusting the model parameters, then the gradient will become smaller until an equilibrium has been reached. In this case the model has learnt the distribution of the data set.

· ◇ ·

For the sake of completeness, the learning (update) rules are given as:

$$W \Leftarrow W + \alpha(h(v_t)v_t^T - h(\tilde{v})\tilde{v}^T)$$

$$a \Leftarrow a + \alpha(v_t - \tilde{v})$$

$$b \Leftarrow b + \alpha(h(v_t) - h(\tilde{v})) \tag{11.16}$$

with α being the learning rate and $h(x) = \sigma(b + Wx)$.
It can be shown that adding hidden units to a RBM always improves the log-likelihood unless it already perfectly models a distribution [Ben09]. Additionally, they can represent any discrete distribution, although it may be represented more compactly with other models such as the unrestricted Boltzmann machine (ibid).

Training in Deep Belief Networks

As mentioned before, a Deep Belief Network consists of several layers of representations that model the observed data [Hin07]. Informally, this architecture is based on stacking several RBMs on top of each other. As we will see, this way the learning methods allows to circumvent several problems that are otherwise present when training deep architectures.

· ◇ ·

The training of deep architectures encounters two main difficulties (hypotheses). The first one is that here the training poses a harder optimization problem in comparison to more shallow architectures such that models may under-fit. For example, when training deep Neural Networks using Back Propagation, the gradient is progressively getting more dilute such that for lower layers the correction signal is minimal (vanishing gradient problem) [GB10]. In this way, historically, training deep Neural Networks has been practically infeasible. The second one is the issue of over-fitting. With the addition of layers not only the number of parameters is increased but also the space of complex functions that can be modelled. Thus, for such models with a high capacity and low bias, there may be many fitting functions that correspond to a minimal error for a specific training set. As a result the model may not sufficiently generalize from the training data. The latter can be understood as a regularization problem. In case of the DBN this is addressed using unsupervised pre-training to initialize the parameters of the model in some form already represents the latent structure of the input distribution.

· ◇ ·

A DBN mixes undirected and directed connections between variables. The top layers always form a RBM and the others a Bayesian Network (Sigmoid Belief Network) such that the joint distribution between the input v and the l layers can be expressed as:

$$P(v, h^1, \ldots, h^l) = \left(\prod_{k=1}^{l-2} P(h^k | h^{k+1}) \right) P(h^{l-1}, h^l) \tag{11.17}$$

where the conditional distribution $P(h^k | h^{k+1})$ is:

$$P(h^k | h^{k+1}) = \prod_i P(h_i^k | h^{k+1}) = \prod_i \sigma \left(b_i^k + \sum_j W_{ji}^k h_j^{k+1} \right) \tag{11.18}$$

In this way, the DBN is a special kind of Sigmoid Belief Network (SBN) with the top level prior being derived from a RBM. Building this network is part of the unsupervised pre-training procedure which is performed in a greedy and layer-wise fashion. Layers are added one at a time and their respective parameters are trained locally. In fact, these layers are RBMs whose parameters are tied throughout training, thus successively leading to the directed layers of the SBN. As such, one can think of the training as a recursive procedure that takes the activities of the hidden units of one RBM and feeding it as data for learning in the next, higher-level RBM.

· ◇ ·

For the initial case of a single hidden layer DBN, a RBM is trained on the empirical input data only. Here, $P(x|h^1)$ and $P(h^1) = \sum_x P(x, h^1)$ depend only on the parameters of the first layer [Ben+07]. When the second layer is added, $P(h^1)$ is modelled using a separate set of parameters involving the parameters for the second hidden layer h^2. Thus, $P(h^1)$ is now the marginalization of the second hidden layer as $P(h^1) = \sum_{h^1} P(h^1|h^2)$. This process can be repeated for the following layers and is exemplified in figure 11.6. In this way a DBN learns increasingly abstract intermediate representations in a hierarchical manner from originally low-level feature data. This greedy layer-wise training of RBMs is justified using the variational bound [LB08; HOT06] which also guarantees that adding a new layer that is initialized appropriately will yield an improved likelihood with respect to the previous layer [Ben+07].

· ◇ ·

After pre-training, fine-tuning can be performed in a supervised manner in order to adjust the representation to be more discriminative. For this, usually an additional output layer is added to the network. Supervised training can then be applied using Backward Propagation or the Up-Down Algorithm to the whole network [HOT06; Hin07]. The aforementioned issues related to under-fitting are here less severe since latent features/variables are only perturbed in this step and not created.

· ◇ ·

As stated before, pre-training is a regularization procedure to force encourage hidden layers to encode the latent structure of the input distribution. In this way the aim is to favour from the variety of possible networks (functions) those that already "understand" the data structure to improve generalization and reducing variance of the estimating procedure. In recent works it has been shown empirically that even with different computational modules, *e.g.* Kernel PCA [CS09], Independent [Le+11] or Autoencoders

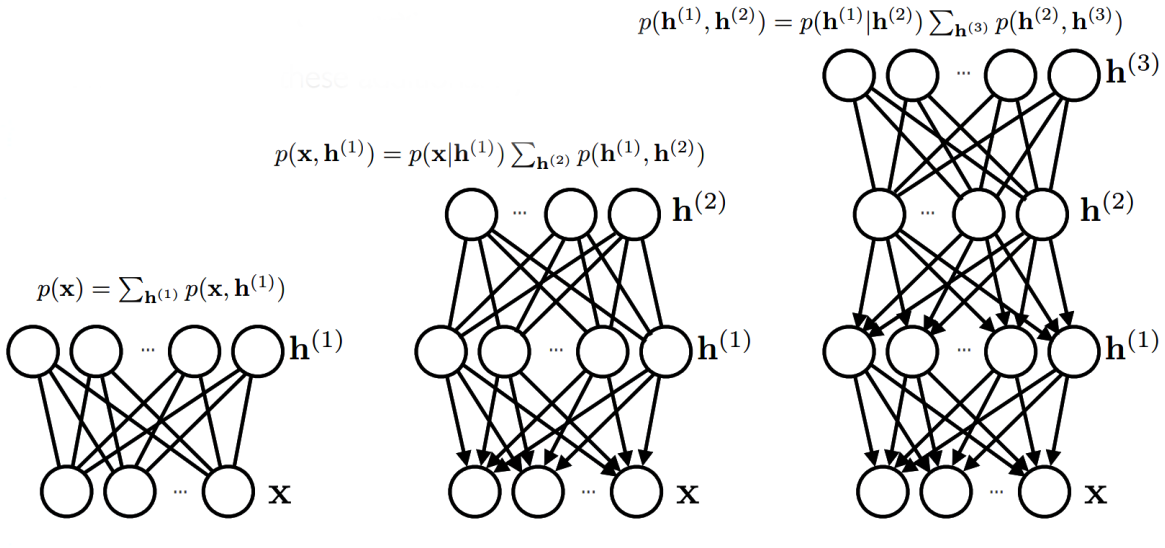


Figure 11.6: Learning a DBN from layer-wise extension of a single Restricted Boltzmann Machine (taken from [Lar13])

[Lar+07] to name a few, unsupervised pre-training or initialization in combination with stacked architectures led to improved representations (*e.g.* in terms of classification error or quality of generated samples or invariance properties) [BCV13]. In [Erh+10] Erhan *et al.* perform an empirical evaluation of the impact of pre-training with regard to the depth of a deep belief network and the number of the hidden units (see figure 11.7). The regularizing property of the pre-training is evident in networks with a large capacity as less over-fitting is apparent. Conversely, it is shown that the approach leads to under-fitting for networks with a small capacity. In

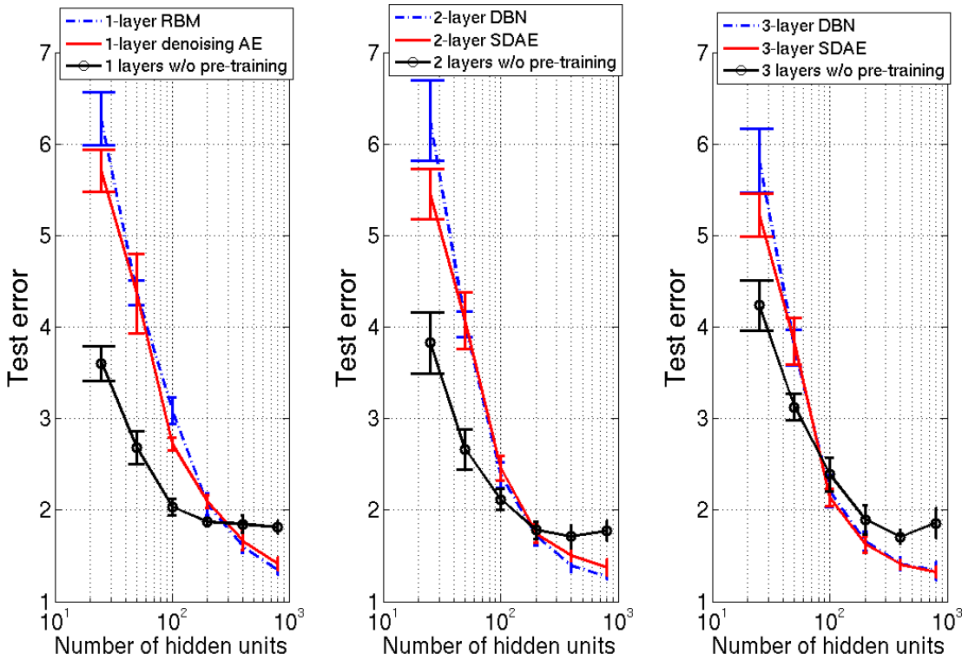


Figure 11.7: The effect of layer size and pre-training of Deep Belief Networks with different depths (taken from [Erh+10])

another contribution, Larochelle *et al.* [Lar+09], evaluated the impact of different hidden layer sizes on the classification error of a task involving a deep belief network with a fixed number of layers. Here, the options con-

sidered either increasing, decreasing or constant widths with regard to the hidden layers. The results showed that constant width performed at least or better than the other options. While decreasing the width can be seen as some form of dimensionality reduction of the input data, it also reduces the computational complexity. Which in our use case is not negligible.

Applying Deep Belief Networks for the Use-Case

In our work, the learning objective of the specialization phase is to minimize the error between observation and prediction of V/A coordinates given the feature vector. Similar to [HOT06], the real-valued input data was interpreted as probability values of an underlying Bernoulli distribution for input. Thus, to alter the value ranges accordingly, the input data was per feature Z-Score transformed. For the mapping from the last RBM layer onto the V/A plane, a logistic regression was employed whose output logit variables were interpreted as encoding of grid tiles of a uniformly discretized grid of the V/A plane (cp. figure 11.9). In this 1-of-K encoding each output variable corresponds to exactly one tile of the grid. Note that this differs from the previous representation in that it is more general than the limited number of generated affect clusters and hence embodies a more detailed representation of affect. Employing logistic regression allows for fine-tuning the whole network via supervised gradient descent on the negative log-likelihood cost function. Furthermore, the combination of 1-of-K coding and the softmax saturation in the logistic regression proved to be beneficial as early attempts to use a linear regression method similar to [SSK12] led to disappointing results. This means, instead of a continuous representation of the V/A space, the model uses a discretized one. However, with increasing grid resolution it converges towards a continuous solution. As will be shown, it is possible to create a model that performs well while making rather fine granular predictions. Figure 11.8 schematically shows the architecture of the presented model.

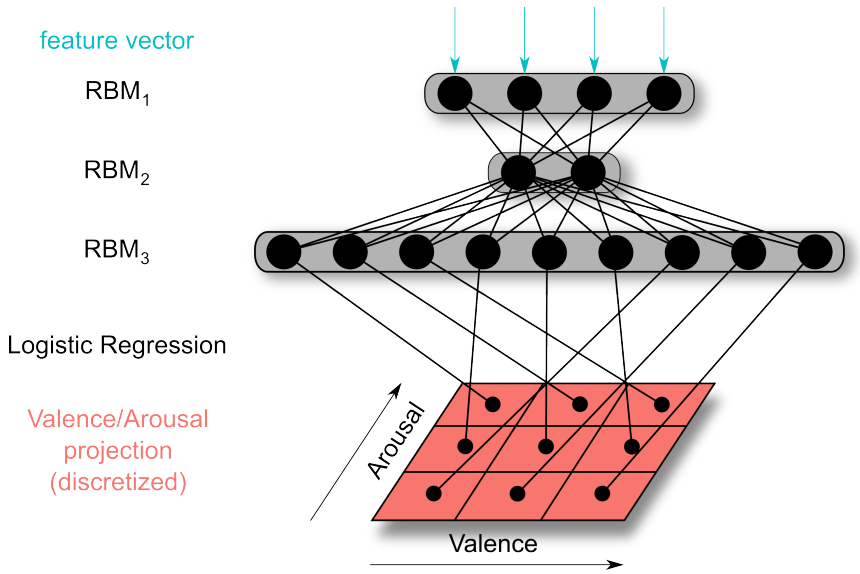


Figure 11.8: Schematic of the applied model (number of layers/nodes is exemplary)

· ◇ ·

The question remains what a suitable error measure for supervised training may be and, respectively, how the overall performance ought to be evaluated. As previously mentioned, a distance-based measure would be

preferable, yet a suitable baseline is missing due to the generated nature of the training data. Hence we use the expected value of the distance of two random points *on the grid* based on the following reasoning: Given a feature vector and its corresponding observation on the V/A grid, the worst prediction of the model is a randomly chosen coordinate on the grid. As the error measure is the Euclidean distance between these points, namely observation and prediction, this *expected* random distance serves well as baseline.

· ◇ ·

All distance measures are themselves normalized against the chosen grid size to overcome to a large degree the quantization error. Based on our initial experiments we chose to use N input nodes (N features) and three layers with N, $N \cdot 10$ and $N \cdot 10$ nodes, respectively. This expanding topology performed better than a reducing one suggesting that the feature data may live on an even higher dimensional manifold. All performance evaluations were done with 10-fold cross-validation using stratification w.r.t. the V/A coordinates (which was not applicable in all cases due to the uneven distribution of samples). To evaluate the impact of the grid size we computed the model for varying grid sizes given the top 200 of the ReliefF importance ordered features. Due to the complexity of the model and the DBN training algorithm itself, this calculation took 83 days on two NVIDIA GTX 580 GPUs with 3 gigabytes of memory.

· ◇ ·

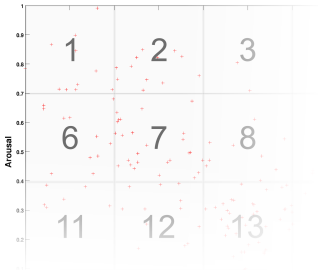


Figure 11.9: An excerpt of a 25 tile grid with the V/A values from the coordinates mapped into it.

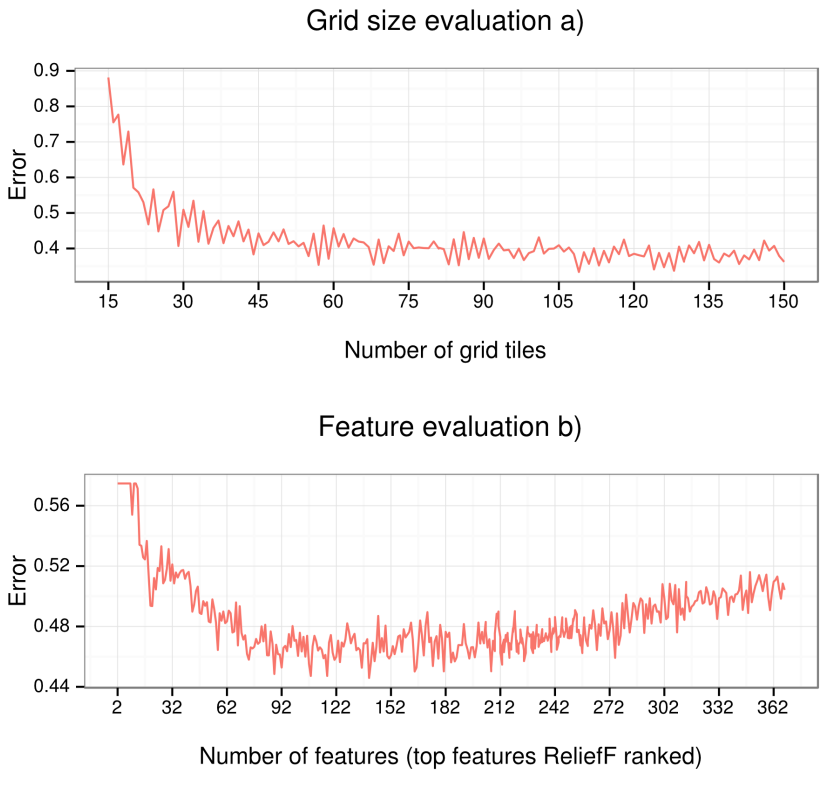


Figure 11.10: Performance evaluation w.r.t. to a) grid size and b) number of importance ordered features (fixed grid size: 25)

Figure 11.10 a) shows the regression error in relationship to the grid size; for the examined range, it can be seen that the error decreases with increasing resolution (increasing number of grid tiles ; the minimum is 124). The comparatively large oscillations are attributed to the uneven distribution of V/A points in combination with the uniform grid. Here, small clusters

of V/A points may be merged on a single grid tile or spread across a small neighborhood of grid tiles according to the boundaries of a grid tile. Furthermore, we evaluated the impact of the selection of ReliefF ranked features for an exemplary chosen fixed 25×25 grid (cmp. 11.10b). In our case we see how the error decreases with the number of features until the tipping point (the top 158 features) of model complexity versus training time is reached. Nevertheless, with 124^2 grid tiles and 158 features, the error approaches 31%, which is, given the potential issues of the data set and learning algorithms as pointed out before, very reasonable. This corresponds to a distance of ∼18 tiles (15% of the grid in one dimension), therefore we regard this estimated performance as satisfactory for the use-case as introduced at the beginning.

11.6 Conclusion

In the scope of this chapter it was argued that from the performative and compositional point of view, insight into the emotional affect of timbre would be beneficial to music creation, especially for novices. Furthermore, it was previously pointed out that in order to gain this insight and for practical applications, a model representing this relationship between timbre and affect would be necessary. Within the framework of this thesis, especially the approaches shown in chapter 12, the perceptually motivated synthesis, and chapter 9 could benefit from such a model. For the former, the feature analysis for the generated audio when sampling parameters of the synthesizer could be extended to include the analysis by the proposed model. As such, the feature data could be extended with the V/A values generated by this model. After the dimensionality reduction step, the synthesis parameters would also be arranged according to their estimated affect. If this is also supported by additional visual feedback (*e.g.* affect dependant colour-coding), then user would have cues to design and alter sounds within the bounds of a certain mood. Figure 11.11 illustrates this extension to the already established mapping.

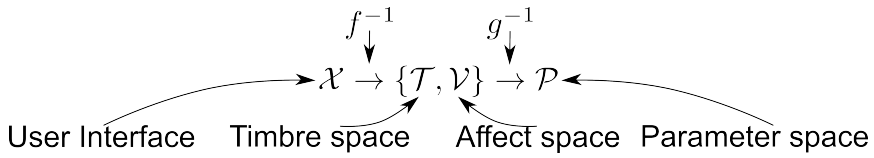


Figure 11.11: Extension of the mapping proposed in chapter 9 with affect features

· ◇ ·

For the prototype which is presented in more detail in chapter 12, it would allow for presenting the user with different timbres that are automatically adjusted to fit the mood of the generated melody. Here, a similar parameter sampling approach could be employed for different synthesis models. The difference to the previous method would be that the user has no control over the exact transformations of a sound but rather, on a generic level, choose a certain prototypical sound from a set of possible timbres that is altered solely by the machine. Both approaches would then represent approaches to a full related audio synthesis environment. The largest part of this contribution dealt with the development of such a model using a ML based approach. For this, we established a suitable new data set based on real world data and applied recent developments in ML to this task of finding a mapping from audio feature data to affect. This included the development of a respective error measure so as to estimate the quality of the model, yielding promising results.

· ◇ ·

Future work, should aim at improving the model by re-assessing the various parameters of the involved ML algorithm and the feature data themselves. For example, instead of the current high level features, the model can be trained to extract relevant features implicitly, *e.g.* from the magnitude spectrum of the audio data as shown in [SSK12].

Chapter 12

Controlling the Emotional Affect of Simple Melodies

Chapter Abstract

This chapter evaluates an approach to support creating simple melodies with respect to a desired emotional (outcome) affect using transformative production rules. The rules have been derived from related works in the field of music theory and music psychology. The accompanying prototypical implementation is evaluated with a user-study which shows the soundness of the approach. An analysis of user interactions with the system allows to deduce user interaction patterns and quantifications about the music produced. This way, connections between valence, arousal, melody properties and melody contours and emotions could be revealed. The results of a questionnaire show that the incorporated methods for creating music associated with certain emotions is appreciated by the users, prevalently by novices.

This chapter is based on the contribution [KHG14a], "FugueGenerator - Collaborative Melody Composition Based on a Generative Approach for Conveying Emotion in Music" by Klügel, Hagerer and Groh.
All sections have been revised; section 12.4.2 is new.

The implementation of the prototype was done in the scope of Gerhard Hagerer's guided research project. The concept was derived in several iterations and discussions between him and me.

I implemented the underlying framework and synthesis method and devised the respective transformative rules. The study was conducted by us both.

The source code has been made available:
G. Hagerer and N. Klügel. *FugueGenerator*. Application, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/FugueGenerator

---

12.1 Introduction

As argued in the chapter 3, for the collaborative use-case, novices have to be integrated in the music composition or performance process. Producing musical content with respect to harmony and melody creation that follows a specific affective intent, however, involves a significant amount of music theoretical knowledge. In the following sections it will become apparent that certain rules within this context have been already established in related research to describe or transform musical structures with respect to affect. Such rules include the harmonic and rhythmic structure as well as the articulation of a melody. Furthermore, we establish additional rules to change the timbre of generated sounds. On the basis of these, we will devise an Affective Algorithmic Composition (AAC) system (section 10.4).

· ◇ ·

Although approaches to interpretatively modify existing musical material exist (cp. [Wal+11; OC10]), not many efforts have been made to specifically support the (collaborative) creation of new music with respect to the desired emotional affect. Thus, it is also necessary to evaluate the validity of such rules for the creative use-case, since it is possible that discrepancies between affective intent and percept arise.

· ◇ ·

Conceptually, the presented prototype supports users to generatively make music by drawing melody contours and have the software system interpret these with respect to an affective intent. As pointed out before, our generative approach is rule based inasmuch as it applies transformations that we have formulated beforehand to modify the users' input [1]. The method of drawing melody contours is inspired by Xenakis' work on the UPIC (section 8.2).

[1] Compare [Nie09, p. 236-244] for a review of related systems

· ◇ ·

The user study presented in 12.5 investigates whether the use case indeed benefits from a discursive collaboration. Moreover, it shows that the employed rules can successfully alter the tonal content of simple musical phrases such that certain emotions can be expressed productively with regards to an emotional intent. Furthermore we will highlight what connections come up between desired emotions and melodies.

· ◇ ·

The implementation work has been titled "FugueGenerator", which, upon later reflection, is misleading since it implies the automatic generation of fugues. The name was originally chosen because the generated music resembled qualities of simplistic Baroque fugues. More appropriately, as one reviewer suggested, it could be called "CollaborativeArpeggiator".

12.2 Related Work

Regarding recent research that applies the construction of musical structures with respect to emotion, one can differentiate between a transforming and a generative approach. In the transformative approach, existing pieces of music are modified either in terms of the score or performative features. The KTH Performance System [BFS02], for example, modifies mainly performative (interpretative) aspects of a piece. This system provides a large set of sophisticated rules (initially focusing on violin performances) covering micro-level timing, itonation or tonal tension among

others. These have been partly derived from analysis-by-synthesis methods and additional measurements. Most of them have been confirmed empirically by listening tests [FBS06]. However, the system does not include rules for altering the original score with respect to harmonic content. Although the performance parameters of the system can be controlled in real-time, the score data has to be preprocessed in an additional authoring environment (Director Musices). Because of this, the specialization on violin performances and the missing rule system for score manipulation, we decided to look into alternative rule based systems.

· ◇ ·

In the Computational Music Emotion Role System (CMERS) [Liv+10] structural (score) parameters of a piece are also altered. Importantly, the research regarding CMERS involved comprehensive studies about correlations between structural and performance features of music on the one hand and emotions on the other. These insights were applied in this contribution to formulate rules based on evidence to manipulate melodies towards their inherent valence and arousal values and their respective emotions. Complementary to this, Oliveira et al. [OC10] focus on a knowledge based system that makes use of empirical data to derive a classification model for musical segments that can be then recombined and transformed towards a desired emotion.

· ◇ ·

In the completely generative class of AACs music is produced entirely from scratch using a combination of production and transformation rules. The Affective Music Generator [WIC08] takes a coordinate within the two dimensional valence and arousal space as input parameter and then uses a purely generative method. Figure 12.1 shows the graphical user interface. The construction takes structural and performative parameters into account which are related to rhythm, melody contour, harmony and articulation. Studies showed that users assessed the produced music correctly [Wal+11] and thus that the generative mappings may be regarded functional. As Wallis bases his rules on Livingstone's work, we used his implementation as guideline for our prototype.

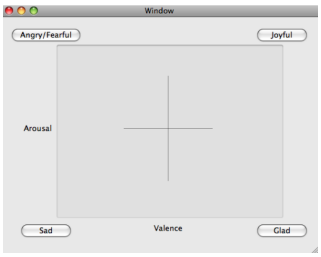


Figure 12.1: The user interface in [WIC08] to produce generative music based on a valence and arousal vector

· ◇ ·

To conclude, some algorithms and consolidated knowledge to *systematically elicit emotional responses in users* [WIC08] exist. Nonetheless, we see lack of applied use cases and evaluations thereof, so the question whether such concepts offer a suitable approach to hide musical complexity for the end-user by means of a metaphor for high-level musical expression while still offering an experience that gives users joy in musical expression can be regarded as still not being answered fully.

12.3 Conceptual Framing

The main idea is to support users to construct affect related musical ideas within some representation of the valence arousal space. Straightforwardly, to move within that space, users could be presented a 2D surface to control their respective valence arousal vector similar to the Affective Music Generator (see 12.1). However, the question remains on what basis the musical events themselves are generated.

· ◇ ·

An important issue for collaborative creativity support systems, especially for musical applications, is fostering the users' engagement in the creative endeavour [BHL07; Vas07], ideally leading to a direct connection to the musical output in form of an embodiment of music (section 3.4). In this regard, we see value in giving the user the ability to shape musical events in a way that directly reflects an intended melody contour and that especially respects this melodic intent even when transforming it in respect to the emotional intent. Thus we focus on simple means to create melodic content which is then represented in alternative emotional contexts.

· ◇ ·

Regarding the collaborative aspects, the application is designed for two interlocutors since this allows us to perform a focused evaluation towards aspects such as assessing the users' agreement on perceived emotion or turn taking patterns.

12.4 Implementation

The previously discussed shaping of melody contours is realized by drawing them directly on a 2D surface. These are continuously repeated and constrained by a chord progression. The valence and arousal values can be controlled manually. Accordingly, they modify the underlying scale, tempo, articulation and timbre. Each collaborator can create an own melody contour. The two resulting melodies are played back in parallel for polyphony. Furthermore, controls for valence and arousal are not only part of the shared workspace but also part of a private workspace for the respective user: valence is a global (shared) control and arousal is a private control for each user. This decision will elaborated later in this section. In summary, the system makes use of three internal step sequencers, two for each melody contour and one that represents the chord progression.

12.4.1 Harmony and Melody Generation

While there is a large corpus of suitable theories and approaches to formally analyse harmony relationships in music, such as Neo-Riemannian theory (*e.g.* [Tym11]), not all offer the operational formalization suitable for application in computing. The implementation relies on transformations and operations on pitch class sets based on an algebraic theory of chord structures as presented in [Lad99] and pitch transformations that are performed relative to such a pitch class set[2].

· ◇ ·

A pitch class represents all pitches that are related to each other by octave transpositions. For example, the pitch class C includes all transpositions $C_{-1} \ldots C_8$. A pitch class set describes a scale or a chord as a collection of pitch classes without duplication [Col04]. It can also be expressed as a binary string of pitches belonging to the set (or not). To use the example in [Col04], the pitch class set $[C, C\#, E, F\#]$ or simply $[0, 1, 4, 6]$, can be expressed as $(1100101000000)_2 = (3232)_{10}$ for a 12-tone musical system. We can find equivalence classes for this by rotation of the binary representation. For all possible (bit) combinations of a 12-bit string, we can define scales that are prime as scales that can not be generated by bit rotation of other scales.

[2] Ladma's algebraic constructs (G-systems) are similar to those of Fripertinger's [Fri99] which are referenced by Collins [Col04]. Both contributions were published at the same time. Ladma provides further algorithms to calculate pitch set equivalence classes.

Thus, it is possible to generate all prime scales (or chords) in the 12-tone musical system and express other scales simply as their rotations[3].

[3] This can be done either using G-systems, the method proposed in [Col04] or simply by brute force which is feasible for the 4096 possible bit combinations as in our library implementation.

· ◇ ·

Pitches in the library implementation are represented not as absolute pitches but as relative ones to a scale. For example, the first relative pitch will always represent a root note. Chromatic pitches can only be generated if they are paired with a concrete scale instance. This way, melody contours of relative pitches can be transposed and fit into a target scale. The library implementation provides a DSL for conveniently performing such operations.

· ◇ ·

For our prototype, we use the Ionian scale $(2773)_{10}$ as generator scale which is then simply rotated several times to obtain the other scales. All melody related operations are performed on relative pitches which are then transformed into concrete (chromatic) ones before synthesis.

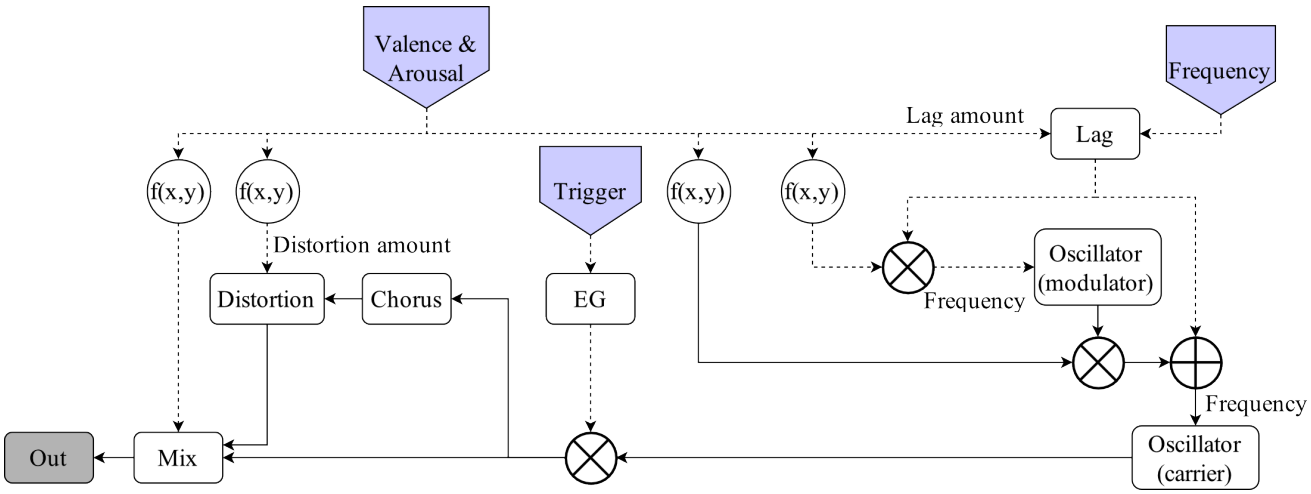


Figure 12.2: Architecture of the underlying synthesizer including control inputs used by the generative process; $f(x, y)$ denotes a weighting or linear mapping of two dimensional input parameters. The dashed lines indicate control modulations for the respective Ugens.

12.4.2 Synthesis

The sound generation is based on a simple two operator FM synthesis with an additional effects chain which is depicted in figure 12.2. Ugens that are labelled $f(x, y)$ denote a weighting of the input parameters for valence and arousal. In this design, the modulator frequency is restricted to multiples of the base frequency such that only harmonic spectra can be created. The lag-Ugen acts as smoothing interpolator in the time domain. The rate of change can be set. When the shared frequency control signal is smoothed various slurs can be generated to create a legato effect. The frequency modulated oscillator output is then processed with a chorus and distortion effect in parallel. The distortion is based on waveshaping ($\tanh$ & $\cos$) after the input signal has been equalized to emphasize the mid and high frequency range. In combination with the chorus, this creates a wide, slightly resonating distortion, similar in tone to a typical guitar fuzz effect. Both signal paths are then mixed together in a way that the output can be faded from one to another. With this design pure sine tones and simple waveforms with more harmonic content can be generated. The effect chain allows to gradually shape these into harmonically rich sonorities and further into comparatively harsh tones. The articulation between staccato

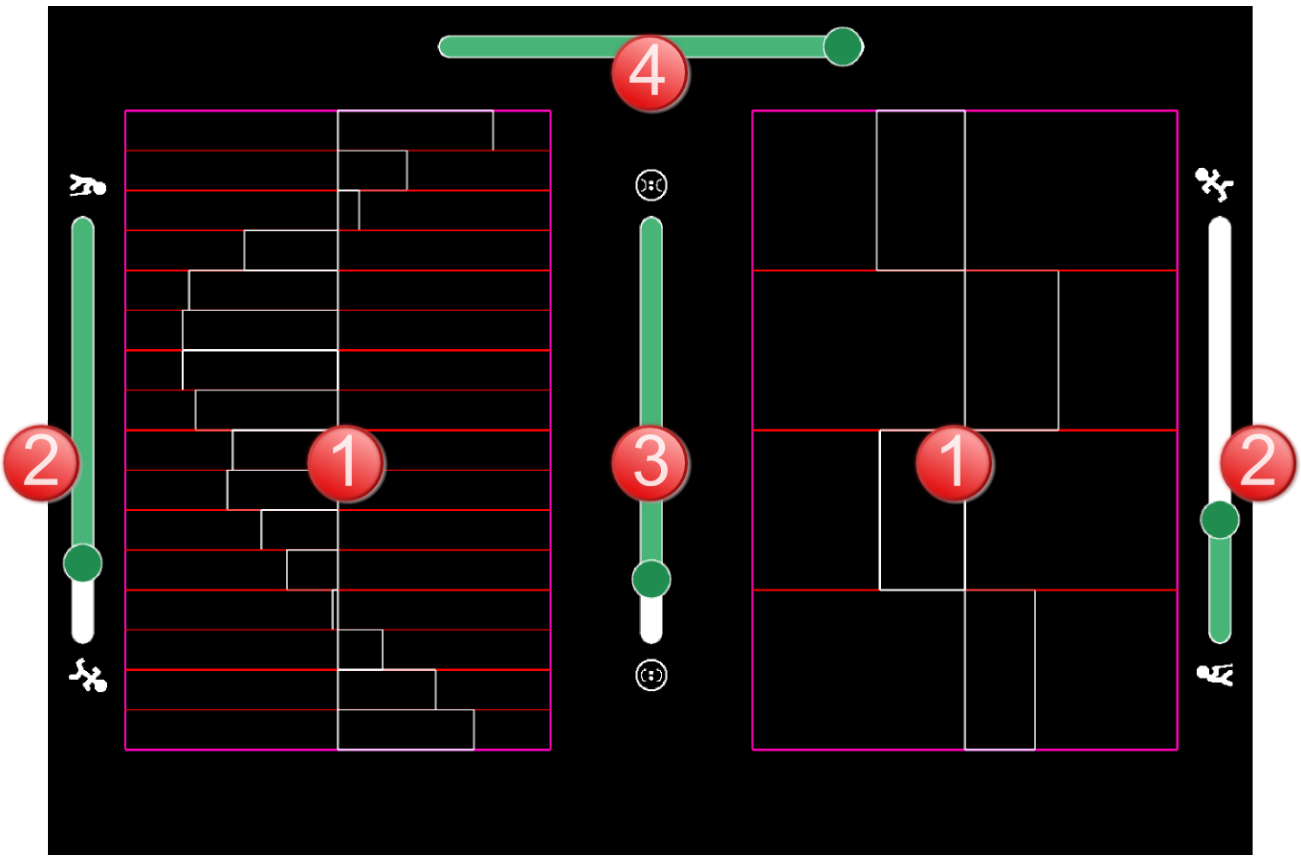


Figure 12.3: Screen-shot of the prototypical implementation with labelling of the interface controls

and legato can be continuously varied by modifying the amplitude envelope decay times with respect to the global tempo.

12.4.3 Concept and User Interface

The graphical user interface consists of the elements shown in figure 12.3: surfaces for the melody contours (fig. 12.3,1) , controllers for the local arousal settings (fig. 12.3,2) , a global valence control (fig. 12.3,3) , a register control that indicates which melody is playing in a lower register (fig. 12.3,4). Users operate the application on the short sides of the tabletop, facing each other. Drawing the melody contour is done by touching and dragging within the melody surface (fig. 12.4). The height of each bar indicates the pitch relative to the underlying harmonic. The set of bars forms a sequence of notes. The bar that is currently played is highlighted in the user interface.

12.4.4 Melody Generation

The number of bars in a sequence depends on the local arousal setting (see paragraph 12.4.6). The tempo is the same for both melody contours, while the divisions of a beat (the number of bars) are equivalent to note lengths; *e.g.* a sequence with two bars is made of two half notes.

· ◇ ·

The tonal system of the melodies is based on the fundamental note, which lies at the center line of the surface. The upward deviation is equivalent to a positive deviation in pitch from the root note and vice versa (cp. fig. 12.4).

The amount of pitch deviation is also dependent on the local arousal (see paragraph 12.4.6) and the global valence settings (see paragraph 12.4.5). Such relative pitches then guide the selection of absolute pitches as an arpeggio of the currently playing chord. The system uses a repeating chord progression which is common in many song forms: Tonic, Sub-dominant, Dominant, Tonic. The common tonal basis is a mode, which changes in accordance with the global valence value as described in paragraph 12.4.5. Nevertheless, the fundamental note always stays the same.

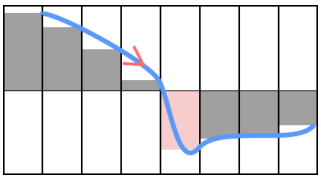


Figure 12.4: Illustration of drawing a melody contour inside a melody container. The blue line shows the path of a drag gesture, the red arrow indicates the direction. The bars adjust to the last touch position within the respective row.

12.4.5 Controlling Valence

As previously mentioned, the valence value steers the selection of the mode. Livingstone [Liv+10] empirically showed a correlation between valence and happy major or sad minor modes, and Persichetti [Per61; CF06] suggested orderings of modes reaching from darkest to brightest. Following this, we implemented a correlation between valence and modes which is influenced by the work of Wallis [WIC08]. This ordering of modes by ascending valence or brightness is: Phrygian, Aeolian, Dorian, Mixolydian, Ionian, Lydian.

12.4.6 Controlling Arousal

We based the structural and articulation related transformations again on the work of Livingstone [Liv+10]. These are shown in table 12.1. The rhythm complexity and the tempo are controlled by arousal. An increase of arousal also increases the number of bars (and notes) on the melody surface. The lowest arousal setting has two half, the highest 16 sixteenth notes. This results in fast and agitating melodies at a high arousal setting. These local parameters also influence the global tempo (weighted mean), such that agreement of users in the extremes of the arousal scale can further reinforce the effect.

	Low	High
Rhythm	simple	complex
Articulation	legato	staccato
Tempo	slow	fast
Pitch Range	narrow	wide

Table 12.1: Influence of arousal on properties of music

12.4.7 Cross-Relations between Valence and Arousal

So far, the transformation rules for valence and arousal only apply to separate concerns of the shared composition. In this section we will discuss additional cross-relations concerning the timbre of the synthesized audio and rhythmic complexity. These are shown in table 12.2.

· ◇ ·

As stated in section 12.4.2, the synthesis has several parameters that are controlled by a weighted sum of the valence and arousal values. The FM index and modulator frequency is adjusted proportionally to arousal and valence with the latter having more weight. This leads to the highest amount of frequency modulation and thus harmonics for the V/A value representing *happiness*. In this setting, the sound generated sounds familiar to a square-wave, the quality can be described as bright and bell/organ-like with a characteristic similar to the high energy lead-melody sound of late 1980s / early

1990s chiptunes. Decreasing the arousal will reduce the modulation index and therefore further limit the bandwidth of the spectrum such that at the setting for the mood *calm*, the harmonics have been significantly reduced. If the valence is then also set to a low value, the synthesized sound is a pure sine tone. Such a dark mood in combination with decreasing arousal introduces more glide (slur) between notes, emphasizing a legato effect. Complementary, a distorted timbre is made increasingly dominant on low valence and high arousal settings (representing aggression) by mixing the processed signal from the effects chain into the output sum.

Valence Arousal	Low	High
Low	Timbre: slurred & (low/no) harmonics Rhythm: straight	Timbre: low harmonics Rhythm: swing
High	Timbre: distorted & bright Rhythm: straight	Timbre: clean & bright Rhythm: swing

Table 12.2: Influence of valence on properties of music

· ◇ ·

Regarding the aspect of rhythm, rhythmic complexity is related to arousal and high valence [Liv+10]. The *Weighted Note-to-Beat Distance* (*WNBD*) has been shown to be a measure for rhythmic complexity that agrees with human perception [Thu08]. This measure will be explained in more detail in section 13.4 as a tool for analysis of rhythmic complexity. However, in this use case, it will be applied as a generative method. Therefore, we created the inverse function of this complexity measure to modify the note durations and onsets to exert control over the swing added. An increasing inverse WNBD value leads to an increasing shift of every second or forth note onset while reducing its duration by the same shift distance. It was implemented such that the increase of a local arousal value consequently increases the inverse WNBD of a melody but only for high valence values.

12.5 Evaluation

For the evaluation we follow two objectives: first, we would like to know whether the devised systems helps users construct simple music with a specific emotional intent. Thus, the question is whether the transformation rules hold, and if they overlap with the users' (subjective) perception. Second, it is of interest to evaluate to what degree this approach supports collaborative creativity and how collaborative turn taking patterns manifest when the emotive interpretation of melodic content is guided by the system.

· ◇ ·

For this we performed a user study which involved 16 people being paired into 8 groups. The age of the participants was between 20 and 35 years. Their background was predominantly academic with varying fields of study, i.e. equal in humanities and technical subjects. They assessed their own musical talent as well as their knowledge and experience in music theory as being neither very advanced nor very bad on average, while extremes on both sides of the scales occurred occasionally.

· ◇ ·

The deployment of the application for evaluation is shown in figure 12.5.

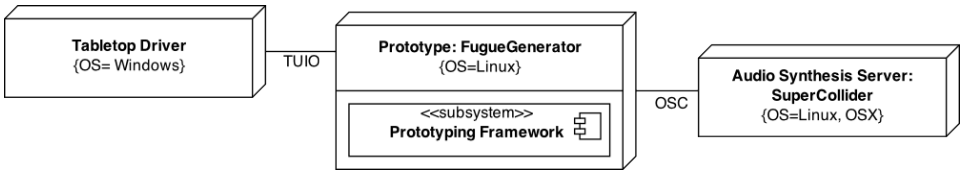


Figure 12.5: UML deployment diagram of the presented prototype

12.5.1 Procedure

We started each evaluation with a short introduction followed by two parts. In the first, we asked the group to construct specific moods. The second part was unrestricted, giving users time to experiment. For the specific construction, users were asked to realize compositions that, based on their own perception, represent the following moods: aggression, happiness, despair, being chilled, solemn, annoyed, graceful, relaxed exhilaration. Testers were given about two minutes for each emotion and had both to agree on the result. Afterwards, the participants answered the first part of a questionnaire that asked about the users' inter-agreements and the agreement to the emotive content of the created generated music. The following free session took ten minutes so that users had time to produce musical output. Again, this part was concluded with a questionnaire, which included the general assessment of the application and gathering demographic data of the users.

12.5.2 Logging Data

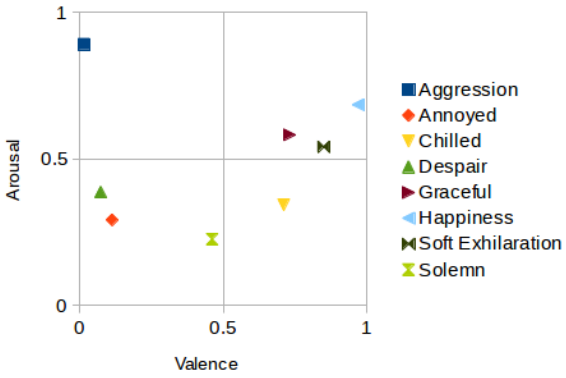


Figure 12.6: Median values of user generated valence / arousal positions from the evaluation

Apart from the questionnaire another data source was logging data generated by the application as discussed in section 5.1.3. These contained time series data about the characteristics of the melodies (the minimum, maximum and average heights of the melody bars and their mean deviation) and the controls for valence and (local) arousal values. With this, gaining insight into the following aspects is enabled:

- Comparison between V/A values to the specified emotions in the first part of the evaluation.
- Correlation of V/A values with the shape of melody contours.

- Reconstruction of turn taking patterns.
- Analysis of the emotive and melodic development within a session

12.5.3 Emotions and Valence / Arousal

The valence and arousal positions for the specific construction of emotions in the first part of the evaluation can be seen in figure 12.6. These have been gathered using the logged data. The emotions shown represent the median value for all groups with valence being represented on the x-axis and arousal on the y-axis. This result corresponds to Livingstone's comprehensive outline. Emotions like aggression, annoyance, happiness, gracefulness and solemnity show high degrees of similarity in their coordinates in the valence arousal space. Differences are most likely explained by the deviation in word meaning throughout the translation (to German) of the words describing these emotions. This was especially the case for "Annoyed" (ambiguously translated as "verdrießlich") and "Soft Exhilaration" (translated as "sanfter Rausch"). The latter was used as an German alternative to serene to fit better into the respective position within the valence arousal space. Chilled (translated as "beschwingt") was chosen as alternative to tender, which worked out very well in comparison to Livingstone.



Figure 12.7: The median values of the pitch heights for each emotion. Emotions are sorted by descending valence (see fig. 12.6).

· ◇ ·

To conclude, the results on average comply with the outline of the experiments collected by Livingstone. The transformation rules and the basic application concept can be regarded as suitable for users to construct music from scratch rather than transforming existing pieces for an altered listening experience. Moreover, given the general musical skills of our participants, it can be seen as a valid approach to augment the users' musical skills.

Livingstone	FugueGenerator
Pitch Height	Average Height
Pitch Range	Maximum Height – Minimum Height
Pitch Contour	Upwards: AHD>0 & Pitch Range>0 Downwards: AHD<0 & Pitch Range>0 Oscillating: AHD==0 & Pitch Range>0 Even: AHD==0 & Pitch Range==0

Table 12.3: Comparison of definitions as they appear in [Liv+10] and in our contribution. AHD means Average Height Deviation.

12.5.4 Emotions and Melody Properties

We now turn to the analysis of the logging data for the melody contours to see whether connections between melody properties and the implemented emotions exist. To answer this, it is first necessary to define several characteristic properties of melody contours. For this purpose Livingstone suggests properties listed in table 12.3 along with their adaptations to our logging data.

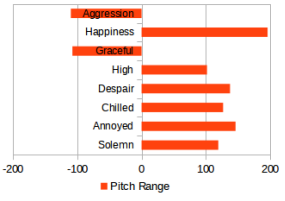


Figure 12.8: The median values of the pitch ranges for each emotion. Emotions are sorted by descending arousal (see fig. 12.6).

· ◇ ·

The definitions as shown in table 12.3 were applied to the logging data in the same manner as for valence and arousal previously, i.e. the pitch height, pitch range and pitch contour are calculated from the data setting at that time when both team members were in agreement about the implemented

emotion. Thus, it is possible to group the median of all properties of the melodies by emotion. The pitch heights are depicted in figure 12.7 and sorted by descending valence. The pitch ranges are shown in figure 12.8 and sorted by descending arousal.

· ◇ ·

For both pitch heights and pitch ranges a certain trend can be identified. Concerning the former the pitch heights mostly decrease with decreasing valence, with aggressive emotion forming an exception. In contrast, the pitch ranges are generally relatively high for low arousal and and low for high arousal, while happiness is an exception.

· ◇ ·

Regarding the direction of the melody contour one can differentiate between upwards, downwards, oscillating and even motions. For this, we used the rules given in table 12.3. In figure 12.9 it is apparent that the melodies tend to be directed upwards for low and downwards for high valence emotions, while oscillating or even melodies rarely occur in general. Compared to Livingstone, these results are somewhat surprising as it was argued that melody contours are expected to have opposing directions compared to our findings [Liv+10, p. 46, Table 1]. Nonetheless, he also makes clear that this relationship is not deemed to be strong because many deviations for different arousal settings exist and the outcome of his experiments regarding melody contours is generally ambiguous. However, if future research confirms relationships between melody contour and affect, then AACs may incorporate them as additional rules.

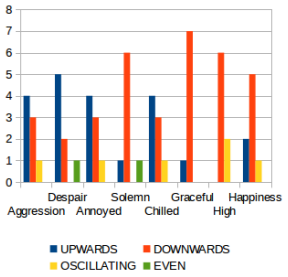


Figure 12.9: Plot showing how often which type of melody contour was created in relationship to targeted emotions. The emotions are sorted by valence (see fig. 12.6).

· ◇ ·

To summarize, it is apparent that certain trends can be observed in the statistics, i.e. low arousal leading to high pitch ranges as well as high valence leading to low pitch heights and melody contours directed downwards. Nevertheless, these figures have to be treated with caution, since the exceptions themselves need further investigation and due to the size of the data set the results are not fully representative.

12.5.5 Free Play

In the second part of the test sessions, participants were given the opportunity to use the application freely, so observations concerning questions about if and how the collaboration and the interaction between interlocutors takes place and how users value the application as a tool for expression.

· ◇ ·

First, we look again at the time series data from the logs to see whether common interaction patterns between participants of a group are apparent. In figure 12.10 the average heights of the melody bars of an example group are shown. With respect to the interaction flow, we highlighted situations in yellow, when simultaneous changes appeared for both melodies, and in green when the changes on both melodies occurred in direct succession. This plot shows that most melody changes are subject to a temporal correlation. The same applies to figure 12.11, where the global valence and local arousal values for each collaborator are depicted (the recorded interaction was performed by another group). The observed temporal correlations are interesting as such behaviours, namely prompt reactions to each other, are typical for turn-taking patterns in music improvisation when interactions are conversational [Mon09b, p. 80-92]. In this regard, solely

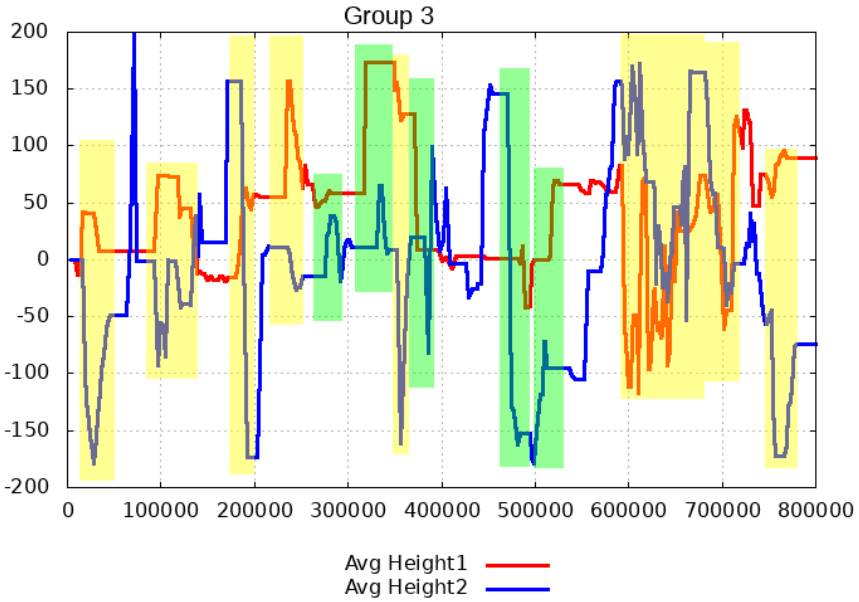


Figure 12.10: Plot of the average height (y-axis) of the bars in each melody container over time (x-axis). The green background indicates successive and the yellow one simultaneous changes performed by the users to show user interaction.

focusing on this data, it can be said that the application supports musical collaboration at least in terms conversational patterns which is consistent with the initial goal of supporting conversational collaboration. Nonetheless, a quantifiable analysis of mutual influences in terms of statistical dependencies of changes in time would help to gain more reliable results by considering all test sessions in an objective way.

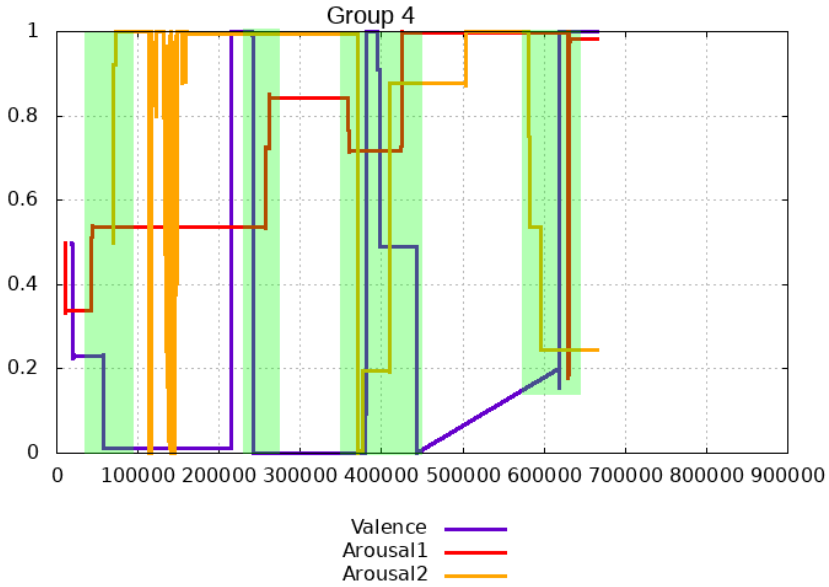


Figure 12.11: Plot of the valence and arousal values (y-axis) for each user over time (x-axis). The green background indicates successive and the yellow one simultaneous changes performed by the users to show user interaction.

12.5.6 Questionnaires

We now turn to the questionnaire. It consisted of 40 multiple-choice questions. Most answers to the questions are expressed on a five-point Likert scale. We used the arithmetic mean of the answers for the evaluation. Besides this, some of these results also take the user groups into account: the group of users that described themselves as musically talented and the others. This distinction by self-assessment is based on their ratings

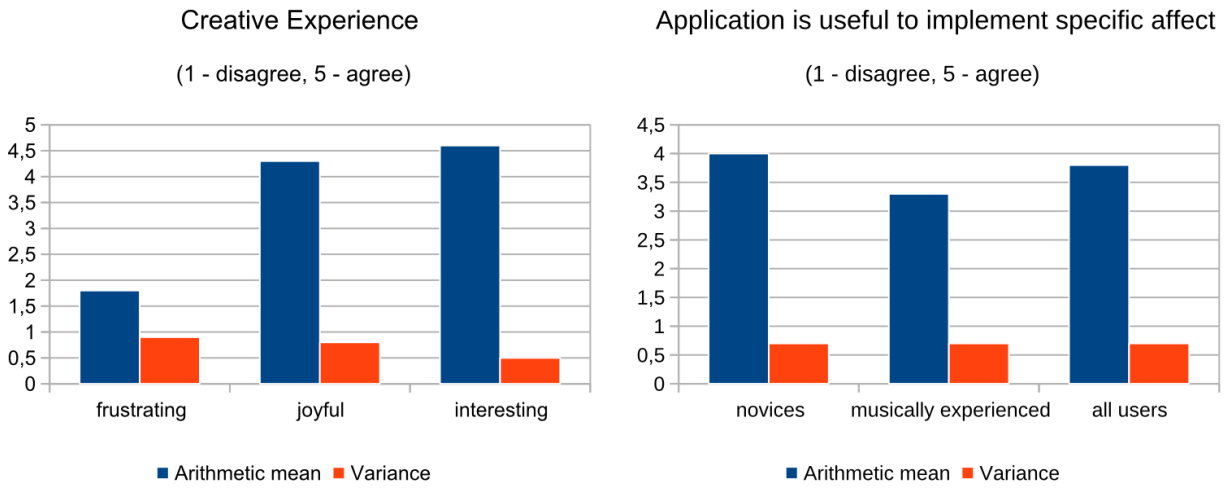


Figure 12.12: Answers in the questionnaire that relate to appreciation

of knowledge and practice in music theory (i.e. harmony, circle of fiths, modes, rhythm); a user that rated himself with four or five in the respective fields is regarded as musically talented.

12.5.7 Appreciation

With respect to the appreciation of the application, we can distinguish between two aspects, whether the software was found helpful to implement a given emotion and how the creative part of the session was experienced. These aspects are depicted in figure 12.12. It is apparent that the application was assessed as helpful. Yet, musically experienced people stated a lower consent than novices. In this context a loss of control of the musical construction parameters may be responsible. This will be elaborated in more detail in the next section. Concerning appreciation for the free and creative part of the sessions, all users evaluated the experience as interesting and joyful and only marginally frustrating.

12.5.8 Collaboration

For assessing the collaboration we can again distinguish between the two parts of the user study. For the first test part it is not clear if interaction between two people is helpful to perform the musical task of implementing an emotion, since different opinions, approaches and musical skills between two individuals could lead to discrepancies that would distort the previously presented logging data regarding emotions. Figure 12.13 shows that in the present case this can be ruled out as collaborators found each other to be helpful solving the task. Furthermore, their agreement on the assessment of affect in their music and their mutual influence of this assessment were rated as positive. In addition, users stated that they value the application as suitable to fulfil the given task. This indicates that the results of the previously discussed logging data are consistent.

· ◇ ·

With respect to the free and creative part of the test sessions, it is of interest whether creative exchange between the collaborators took place and how this was experienced. The users regarded the exchange as positive, active, meaningful and inspiring (cp. figure 12.13). This is also evident in their assessment that they and their collaborators influenced the music generated

via interacting with the other person. Again, this is consistent with the observations made in section 12.5.5, where changes in melody contours, valence and arousal values have a strong temporal relationship. Similarly, this discursive collaboration could also be observed in the evaluations. Another aspect is that the interaction was evaluated as positive and inspiring and thus beneficial for creativity support.

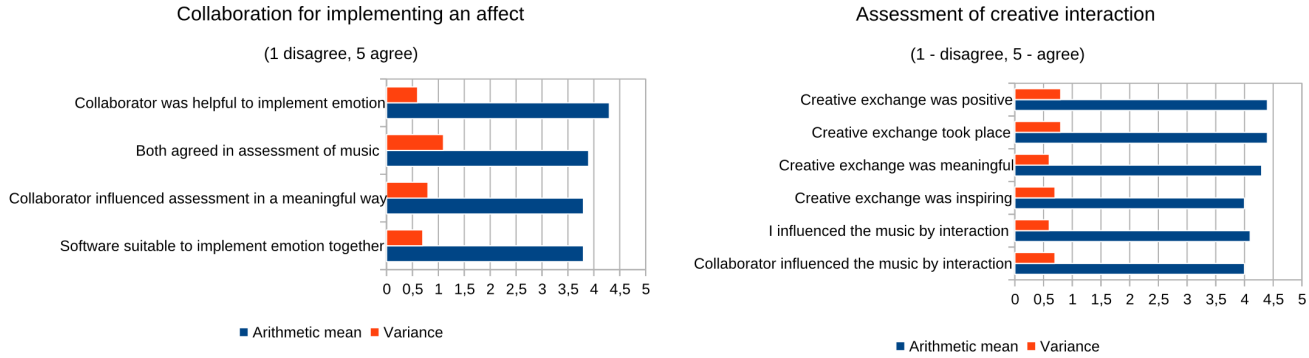


Figure 12.13: Answers in the questionnaire that relate to collaboration

12.6 Conclusions

As the evaluations showed, the prototypical implementation was appreciated by users as offering means for creativity support, specifically for conversational collaboration. It can be claimed that the approach to delegate the music theoretical background for the construction of music with respect to affect to the computer is promising. Most importantly, for individuals that are not entirely musically talented. Collaboration was found to be a beneficial, making the creative experience more valuable.

· ◇ ·

While previous research focussed on listening tests to validate affect related rules and generative constructs, we applied these in a creative and collaborative use case and confirmed these to be suitable. Moreover, the implementation allowed to gain various insights into correlations between affect and melody contours. However, more empirical research is necessary to reliably confirm or dismiss our findings. The same applies for the analysis of interaction patterns, which in the current state mainly represents an exemplary procedure.




Part

IV

Structure




Chapter 13

Groove Production

Chapter Abstract

The creation of rhythmic structures is an important part of electronic music-making. Thus, in order to support this task computationally, studies are necessary that help to gain insight into the relationship between rhythm complexity, degrees of freedom for expression and users' satisfaction and skill. In this chapter three prototypical applications are presented which aim to support the collaborative creation of rhythmic structures. They give progressively more degrees of freedom to create rhythmic complexity. By means of a user study the impact of this on the users' satisfaction is analyzed. Furthermore, the data logged during the experiments allows to measure the factual rhythmic complexity created during the user evaluation. These results are then compared with the responses of a questionnaire.

This chapter is based on the contribution [KHG14b], "TreeQuencer: Collaborative Rhythm Sequencing A Comparative Study" by Klügel, Hagerer and Groh.
All sections have been revised; section 13.3.1 is new.

The implementation of the prototype was done in the scope of Gerhard Hagerer's bachelor's thesis project which was later published in the named conference paper. The concept was derived in several iterations and discussions between him and me.

I implemented the underlying framework and synthesis method and devised the respective transformative rules. The study was conducted by us both.

The source code has been made available:
G. Hagerer and N. Klügel. *TreeQuencer*. Application, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/TreeQuencer

13.1 Introduction

The importance of rhythm has undergone a remarkable transformation from an equal to the most visceral element in the musical trinity of melody,

rhythm and harmony in the last century - in both the high art and popular music [Nei02]. As discussed in section 3.2.2, rhythm is social as it binds communities through dances they may share, such as in tribal or rave cultures, and also through participation in drum circles [Par98]. This participatory facet of rhythm, the collaborative creation of music, especially if intrinsically motivated, can be a highly engaging activity, *the joy of the groove*:

> *"[..] it is immensely satisfying to be a part of a group that's really grooving. Group flow is that experience of participating in a complex activity that stretches one s skills and completely envelopes one's consciousness. To be absorbed in the groove - not for any external motivation or reward but because of sheer rush inherent in it - that is what keeps many musicians coming back to jam sessions"* [Swi11]

· ◇ ·

Focussing on the creative task itself, we aim at investigating the extent of rhythmic complexity and expressiveness that users would want to pursuit without being either underwhelmed or overwhelmed (appropriate difficulty, section 4.4), as this is detrimental to the experience of group Flow. Especially regarding the support for heterogeneous groups, this question is essential.

· ◇ ·

However, to our best knowledge, sufficient empirical results do not exist. Therefore we aim to perform a comparative study with several prototypical applications that successively give more degrees of freedom to the users with regard to shaping rhythmic structures and then evaluate their responses. We will furthermore measure the rhythmic complexity of the created rhythmic structures and compare them to these responses. In this regard, we try to shed some light into this issue such that later applications may be able to perform their role of the mediator actively or more intelligently. Apart from research into user experience in HCI this study is therefore also tangent to musicology.

· ◇ ·

There have been several contributions that addressed the collaborative creation of rhythmic forms. One can distinguish between real time input driven systems such as BeatBugs [WAJ02] which allow users to directly perform percussive patterns that are then recorded into the system and sequencer based approaches such as the reacTable [Jor+06] or The Planets [MS10]. In case of the latter, users can modify the musical events asynchronously to the produced audio. We consider this temporal decoupling as beneficial since it conceptually allows for a larger variety of access points for collaboration and the mediating role of the IT-system. These benefits include reduction of perceived latency in feedback and lowering thresholds for instrument control, and for following protocols in music improvisation and collaboration (section 5.3). Furthermore, this is more natural for the domain of electronic music. Therefore we design our prototypical applications as collaborative sequencing environments.

· ◇ ·

Additionally, we want to support melodic elements (*e.g.* bass sounds, simple harmonics) rather than primarily focusing on percussive ones, since we believe embedding the shared composition in a more musically comprehensive context reflects a more realistic use case. Furthermore, we seek to allow the modification of timbres in real time as timbre itself is an important part of the musical structure, especially for electronic dance music (section 5.3).



Figure 13.1: Screen-shot of the first prototype (p1)

13.2 Concept

One of the most prevalent sequencing concepts in electronic music production is the step sequencer which dates back to the early 20th century (*e.g.* Raymond Scott's Circle Machine) and has been employed in various drum machines (*e.g.* Roland TR-808, first picture in figure 5.1) but also in modern Digital Audio Workstations (*e.g. FL Studio* [Ima97]). In general, a musical duration is divided into a discrete set of units, the steps. At every step a musical event can be set to be triggered. This step sequence pattern is then played back in a loop. In many cases a 4/4 bar is divided into 8 or 16 steps, meaning that events can happen at most at granularity of 1/8th or 1/16th. In this regard, a step sequencer is very limited in terms of rhythmic expressiveness since only one time line exists that is fixed in length and because the musical subdivisions of a bar can only be even multiples of the duration of a step (*e.g.* triplets can not be used). We will make use of this core concept and successively extend it to allow for more degrees of freedom in regard to rhythmic expressiveness for our prototypes.

· ◇ ·

The first extension allows to create polymeters such that looping rhythmic structures with even or odd overlapping elements can be constructed. This differs from the original concept as now several time lines (step sequencers) with the same pulse but different length would be necessary.

· ◇ ·

The second extension concerns the duration of subdivisions in order to construct polyrhythms. This can be enabled by allowing each of the time lines in the polymeter to have a different pulse frequency. This makes it possible to create complex rhythmic patterns.

· ◇ ·



Figure 13.2: Screen-shot of the second prototype (p2)

If we disregard the pulse frequency for a moment, then the step sequencer metaphor can be expressed as a graph construct in which nodes represent audible musical events (such as a hit of a bass drum) and edges express the succession of these events. In this way, a step sequencer with 8 steps can be seen a simple linear list of 8 nodes that are connected by edges. Similarily, a polymeter can be expressed as a tree that is to be traversed starting at the root. We will make use of this perspective on rhythmic constructs as interaction metaphor for our prototypes because in this way we can coherently express time-event relationships for all of our prototypes.

13.3 Prototypes

In our prototypes, nodes represent sound generators directly, we chose to use different abstract three dimensional forms and colors to differentiate between them in the visualization (cp. fig. 13.1). These forms can be interacted with in the following way to change the graph structure and timbre:

- Dragging a form across the shared workspace (re-) connects it to the nearest form, thus this is used to construct the graph. Dragging forms outside of the workspace deletes them.
- A pinch gesture on the forms scales it and modifies its loudness. Forms can be muted when scaled to a minimum value, thus constructing rests.
- Rotation (2D or 3D using a multi-touch gesture) modifies synthesis parameters. Each form has 3 modifiable parameters associated. The global amount of a rotation of a form is indicated as one of three arc segments around it.

Edges of the graph are indicated as simple lines, the progression of the sequence(s) is indicated by small circles that travel along the edges from

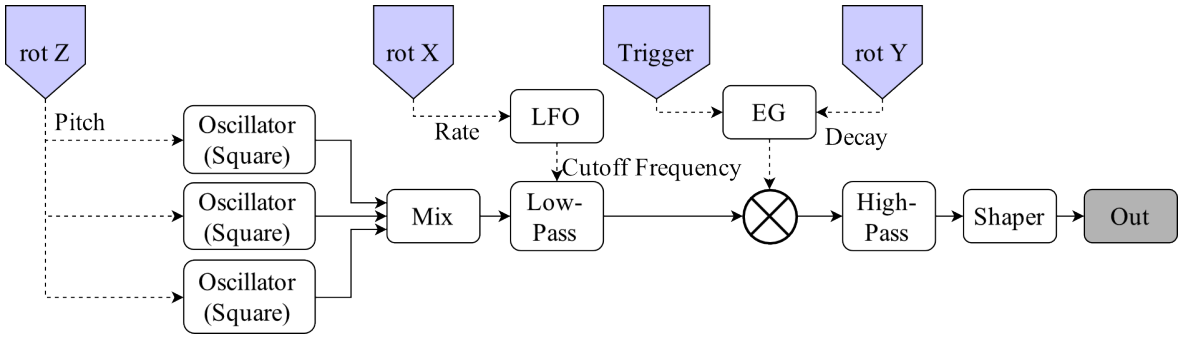


Figure 13.3: Topology of the implemented subtractive synthesizer. The dashed lines indicate control modulations for the respective Ugens.

node to node at the pace of the pulse duration of the sequence. In this way it is visually indicated which forms are about to be triggered next. If this indicator reaches a form, the sound generator is triggered to produce an aural result. This is also shown visually by changing the brightness of the form according to the amplitude level of the generated sound. In this way there are two visual feedback mechanisms that inform about the state of the current composition and which are tightly coupled with the aural events. Furthermore, interacting with the graph structure has immediate results such that modifications are applied while actions are transient. This feedforward method allows users to see how their interactions alter the shared composition.

· ◇ ·

We used six different sound generators. They can be controlled in real time therefore influencing the sonic output immediately. In this way users are able to alter the synthesized timbres and alter the musical result even when the rhythmic structure is static.

13.3.1 Synthesis

The synthesis methods differ in each implementation, thus although each is based on a simple structure in comparison to commercial all-purpose synthesizers, the combination of these allows to create a large palette of different timbres.

· ◇ ·

The most simple one is a subtractive synthesizer for generating bass sounds, the signal flow is shown in figure 13.3. It makes uses of three detuned oscillators to generate a thick chorusing effect as basis. The mixture is then send through a common topology for subtractive synthesis, a low-pass filter followed by an amplifier. The filter is set to resonate slightly to emphasize the overtones of the oscillator mixture. It is controlled by an LFO with a user specified frequency to create rhythmic variations of the bass tone. Since the filter resonance may create a DC offset in certain cases, the synthesizer output is high-pass filtered. Finally, the signal is distorted by a tanh wave-shaper. The resulting sound textures are comparable to those bass sounds found in many ”bass-music” genres (e.g. drum and bass, dub-step, UK garage).

· ◇ ·

A similar architecture is employed to play fixed chords. Here, a bank of oscillators tuned into an harmonic interval is pulse-width modulated by an LFO. Additionally, to have control over the noisiness of the generated timbre, the user can mix a white-noise signal in. Instead of an LFO control, the low-pass filter is controlled by a fixed envelope and the amount

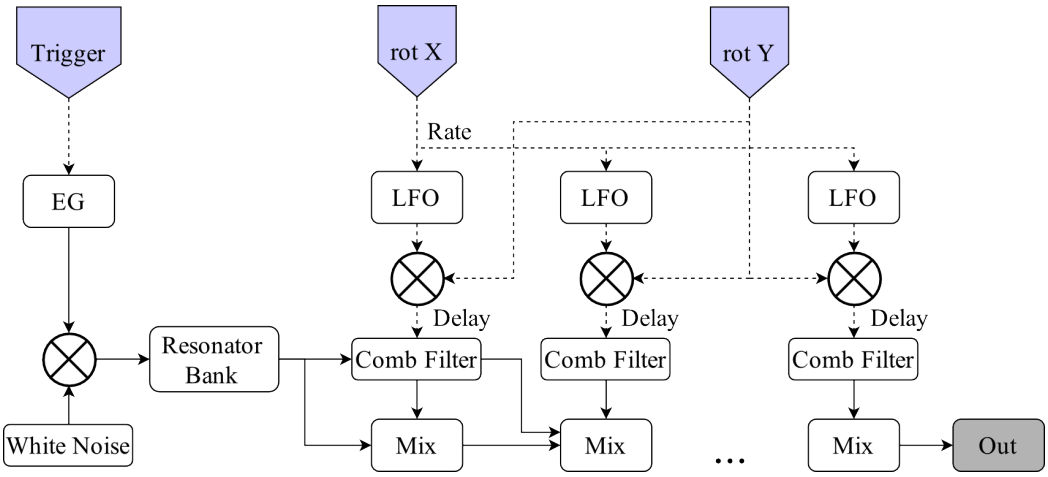


Figure 13.4: A synthesizer inspired from physical modelling to generate resonating impacts. The dashed lines indicate control modulations for the respective Ugens.

of modulation is user defined. The whole signal is then fed through cascaded delays and finally a reverberator (Freeverb). Depending on the filter modulation setting, the generated sound can change from an organ-like to a brassy quality. The combination of the delays and a reverberator adds a rhythmic ping-pong echo that slowly fades away and either retains a tonal character or dissolves into muted dominant reverberation tail, depending on the amount of white-noise in the oscillator-mixture. Similar chord sounds can be found in various Dub-Techno related genres. White-noise dominant textures are frequently used in such dance-floor oriented genres to increase tension in a bridge.

· ◇ ·

Three additional synthesizers make it possible to create varying degrees of tonal or percussive effects. The first one derives its topology from additive synthesis but with complex oscillators as stacked voices that are tuned into an harmonic interval producing rich overtones. Each voice is built as an subtractive synthesizer with a fixed envelope controlling the low-pass filter. The user can gradually fade-in the voices of the stack from the bottom (the fundamental) up to the highest harmonic. In this way it is possible to have fine control over the harmony created. The second synthesizer performs simple FM synthesis with two oscillators. The modulator is a saw-oscillator that can be swept over a large frequency range. Furthermore the user has control over the modulation index and the amount of delay effects that are mixed into the output sum. While the synthesizer can be used to create generic timbres common in FM synthesis, it is also intended as a rhythmic tool when the modulator is set to low frequencies. In this way the saw waveform acts similar to a pitch envelope with an extremely short attack and a frequency dependent decay (and repetition) time. Such frequency sweeps, albeit with a more complex structure, are also used in drum-sound synthesis. Last, a form of physical modelling synthesis was implemented to generate resonating impacts. Figure 13.4 shows an overview this structure. Here, a short broadband burst which is generated by an amplitude modulated white noise generator is connected to a bank of resonators. The sum is then sent into a cascade of time-varying comb filters which have a length below the audible range in order to create a short train of overlapping echos with varying spectral colouring. The length and the amount of modulation is user controlled.

· ◇ ·

The last sound generator is a drum synthesizer that makes use of three different synthesis algorithms which can be seamlessly mixed in order to

perform a simple means to morph or layer percussive sounds. The details of the topology will be omitted here since it is the most complex one and a description would be comparatively lengthy. In general, three dedicated sound generators can synthesize kick-drum, snare-drum and clap sounds. Since the user has additional control over the fundamental frequencies, intermediate percussion such as tom-drums can be synthesized as well. The clap sound is generated by various filtered and wave-shaped noise signals that are amplitude modulated by multi-stage envelopes. The snare and kick-drum generators have similar architectures; they use pitch modulated sine generators which are also amplitude modulated and mixed with white noise. They differ mainly with regards to the employed intermediate filtering and parameterization.

13.3.2 Prototype 1

The first prototype p1, which is an adoption of a simple step sequencer, consists of 8 subdivisions represented as static, non-audible forms to which forms can be attached to. Only for this prototype no other graph constructions are allowed. Thus, solely rhythmic constructions equivalent to that of a step sequencer can be manifested.

13.3.3 Prototype 2

The second prototype p2 allows the construction of tree structures. The root of the tree is represented as non-audible form. Sub-trees leading from the root are looped according to the longest path from root to leaf node (each with the same pulse). This construction allows for parallel events to happen in parallel paths.

13.3.4 Prototype 3

The third prototype p3 is a modification of the second one, where the pulse interval can be changed per sub-tree to a multiple of 1/8th. This is set by changing the distance of the first node of the sub-tree to the center of the workspace.

· ◇ ·

The global tempo in each prototype can be altered with one of four sliders placed around the workspace.

13.3.5 Deployment

All prototypes use the same environment (figure 13.5) and make use of the framework that has been laid out previously.

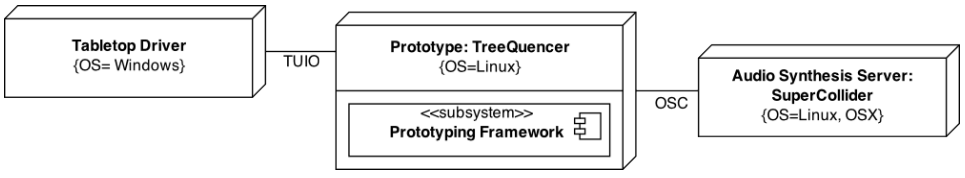


Figure 13.5: UML deployment diagram for the presented prototypes

13.4 Evaluation

For the evaluation we will make use of both a user study as well as tracking the use of the application by logging the interaction. With the latter it is possible to reconstruct the rhythmic patterns that have been created and therefore to apply measures for rhythmic complexity but also to analyse whether users alternatively preferred to shape the timbre of sounds via measuring the duration of gestures. 31 persons took part of our user study in groups of 3-4 people. With respect to their demography, their age was between 19 and 43 years, while 24 were male and 7 female. A large portion of our participants had a musical background (21 had performing experience). This is also reflected in the average time per week a test person makes music (9.4 hours). We performed the study by first presenting the second prototype and then the first followed by the third. Each of these sessions took 10 minutes after a short introduction about the basic usage. We tried to minimize any necessary instructions as we wanted to force participants to explore the prototypes. After the evaluation, participants were asked to fill out a questionnaire that contained questions regarding personal information and self-assessment, individual experience, group and interaction workflow as well as general feedback. Most questions used a 5-level Likert scale.

· ◇ ·

Using the logging data from the interaction with the applications, the graph structure at every time instant of the session can be reconstructed. This can be further transformed into a pattern notation in which every sound generator class is given its own staff and where all the trigger onsets of their respective multiple of a 1/8th are recorded. The length of the pattern is determined by the number of steps it takes until the whole rhythmic structure repeats (the smallest common multiple of the constituent patterns). Figure 13.6 gives an example of the resulting reconstructions. The transformation behaves similar to the real-time play-back algorithm. It uses the graph structure and begins from the start node with a breadth first search that progresses parallel and layer-wise while accounting for the local pulse divisions of a tree. The algorithm terminates when all parallel iterations loop back simultaneously to the start node.

```
----X----------X----------    ----X----------X----------
-----X-----X-----X-----X-----X    -----X-----X-----X-----X-----X
----------X----------X    ----------X----------X
-----X-----X-----X-----X-----X    -----X-----X-----X-----X-----X
--X-----X-----X-----X-----X---    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
---------X----------X-----    ---------X----------X-----
Length: 30                    Length: 30

  ----X----------X----------X----------
  -----X-----X-----X-----X-----X-----X-----X-----X-----X-----X
  ----------X----------X----------X-----
  -----X-----X-----X-----X-----X-----X-----X-----X-----X-----X
  --X-----X-----X----XX-----X-----X-----XX----X-----X-----X--X
  ---------X----------X----------X----------
  Length: 60
```

Figure 13.6: Excerpt from the transformed log-output. This notation follows a simplified time unit box system (TUBS) or drum tablature where each synthesis class is given an own staff. Each character in a row represents a fixed length note (*e.g.* 8th note), 'X' indicates that an event takes place while '-' indicates a rest. The excerpt shows that the rhythm synthesized by the fifth synthesizer class is modified such that first a highly repetitive onset pattern is created (every time unit an event takes place) which is then altered into an asymmetric pattern that results in an inflation of the global pattern (length).

· ◇ ·

After this transformation it is possible to apply rhythmic complexity measures for the whole pattern. We used the *Weighted Note to Beat Distance*

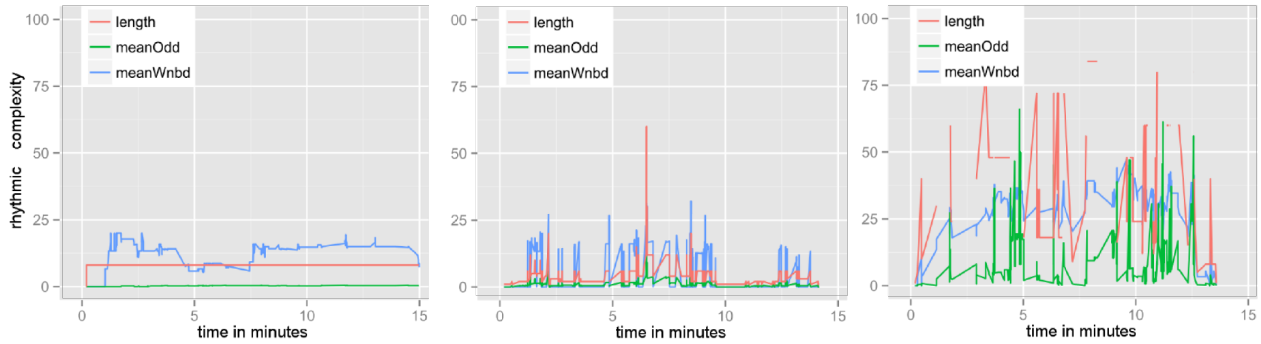


Figure 13.7: These plots show the running means of the rhythmic complexity based on WNBD and oddness of the meter (another perceptually agreeable rhythm complexity measure) as well as the length of the sequence for all three prototypes (left to right) of a user session. Especially for the first prototype it is apparent that WNBD gives more insight into the rhythmic structure (the first prototype uses a fixed pattern length). It is also visible that the third prototype easily creates patterns of high complexity due to the inflation of pattern length caused by the polyrhythms. Here the WNBD is less susceptible to abrupt changes, which is indicative of more reliable data.

(*WNBD*) measure since it was shown in [Thu08] to agree with human perception and because it reflected more of the dynamic variation of the rhythmic structures in our data set compared to other measures (cp. fig. 13.7). WNBD is based on the distance between the start and end of a note in relation to the pulse of a pattern (strong beats) [Góm+05]. According to this distance a note x is given a weight $D(x)$: it gives more weight in the case that a note crosses over a strong beat or ends just before the next one in order to emphasize such rhythmic constructs that are connected to a strong feeling of syncopation. Furthermore, it gives less weight in case the note occurs exactly in the middle of two strong beats and some weight in the case of an onset happening off the strong beat and none otherwise. More rigorously, let e_i be a song beat and $T(x) = min\{d(x, e_i), d(d, e_{i+1})\}$ with d being the fractional distance based on the meter. Then $D(x)$ is defined as 0 for $x = e_i$, $\frac{1}{T(x)}$ if $x \neq e_i$ ends before or at e_{i+1}, $\frac{2}{T(x)}$ if note $x \neq e_i$ ends after e_{i+1} but before e_{i+2} and $\frac{1}{T(x)}$ if note $x \neq e_i$ ends after e_{i+2}. The WNBD of a rhythm is then the sum of $D(x)$ for all notes of that rhythm.

13.5 Results & Conclusion

In general, the feedback to all of our prototypes was very positive (cp. fig. 13.8c) with some of the groups playing up to one hour longer with the prototypes after the experiments. Several observations were made during the experiments themselves or when watching the corresponding recorded video material:

- Flow: Participants frequently moved to the beat by nodding head, clapping hands, tapping feet on the ground or the fingers on the edges of the tabletop. Even though people sometimes showed how much they enjoyed the rhythms by facial expression, many of these actions were done apparently unconsciously. The participants were especially engaged with the application when modifying timbres and discovering sounds (as they were shown only a small subset of the sound palette in the introduction). This was accompanied by showing emotions of amusement via verbal statements facial expressions (*e.g.* grinning). This shows that the exploration of sounds further caused constructive involvement leading to more engagement and curiosity, which are characteristics of a flow experience.

- Group Flow: Participants frequently showed not only positive emotions with regard to their own actions but also in regard to actions of their

collaborators. We observed that the articulation of Flow of several participants at the same time led to a more extroverted expression of engagement such as vividly dancing to the produced music, collectively clapping to the beat or energetic laughing. In this sense it can be claimed that at times collaborators felt to be 'in the groove' with each other. Figure 13.9 shows some frames of a video where such behaviour was captured.

- Styles of interaction: In all groups a division of labor was observable. In many groups a single participant took the role of a coordinator, expressing his ideas while others helped executing them or assigning roles (as in a band setting) to his collaborators. This was done in an equally directed and undirected manner. The degree of verbal communication varied, some groups debated about the next steps while others did not talk at all but merely used gestures and eye-contact.
- Frequent Processes: We frequently observed the same sequence of steps that groups performed repeatedly to create their compositions. This is strongly indicative of an iterative and collaborative creation process. The steps were *exploration*, where several persons experiment in parallel in a loosely coupled way. Followed by *restriction*, where after complaining about the non-musical quality of the aural output the group decides to remove all nodes and to restart the construction. *Planning* after a consulting conversation about improvement suggestions with the division of roles and working methods to avoid earlier mistakes. *Constructing* while the pace of the conversation fades and the interaction and collaboration becomes goal oriented. And finally *reception* of the accomplished work if the musical result was felt worth listening to while the group shows positive emotions such as joy (fig. 13.9).

We now look at the data generated by logging the users' interaction with the prototypes. Regarding the question whether the additional degrees of freedom for rhythmical complexity have been used in a consequent way, the answer is negative on average (fig. 13.8a). The median values of the reconstructed rhythmical complexity values for each prototype show that the complexity generated by prototype two is less than for prototype one, although the opposite result would have been expected: for prototype one the maximum amount of rhythmic complexity that is producible is limited due to the fixed amount of sequencer-steps and prototype two allows more degrees of freedom. However, in accordance with expectation, the rhythmic complexity created with prototype three surpasses the results of the other two. These results agree with the results of the questionnaire elements asking for the perceived complexity of the created music per prototype (fig. 13.8c). One can conclude that the user's perception and the measurement of rhythmic complexity match, which not only indicates the validity of WNBD as an appropriate measure but also that the subjective user's assessment is also valid objectively. However, one question is whether the rhythmical complexities are voluntary or involuntary results. For this we will take a closer look at the results of the questionnaire.

· ◇ ·

The duration of gestures can be used as measure for whether users preferred to alter the timbre of sounds in real time rather than changing the rhythmic structure of the sound form, which would account just for short re-positioning of nodes. Figure (fig. 13.8b) shows the mean of the duration of gestures for each prototype. We chose the mean here since it is susceptible for outliers and thus may better represent extended gesture

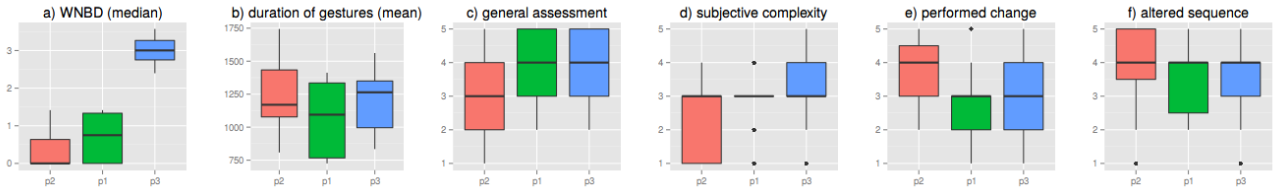


Figure 13.8: Plots of the logged data: a) Weighted Note To Beat Distance, b) the duration of gestures during interaction and the questionnaire: c) subjective assessment, d) perceived musical complexity, e) manner of changing objects (high means experimental, low means targetted) and f) the changed attributes (high means timbre, low means rhythm) for the respective prototypes. p1 (green), p2 (red), p3 (blue) denominate the respective prototypes

input. Here it is apparent that in all applications timbral transformations of the sound forms were performed. However, regarding prototype one it is also visible that users may also have performed a large amount of rhythmical modifications.

· ◇ ·

For the evaluation of the questionnaire, we calculated Spearman's R as correlation coefficients for pairs of answers. Regarding the collaboration itself, a high correlation is apparent between the ability to implement own ideas and fostering one's own creativity (0.71) as well as between the ability to implement own ideas and the preference to use the applications collaboratively (0.52). This means that users saw their collaborators and the applications as aids to pursue their own ideas. A highly positive correlation also exists between enjoying collaboration, on the one hand, and either evaluating the three apps as fostering creativity (0.55) or discovering new musical means (0.56) on the other hand. Users that expressed curiosity when working with the applications also expressed being inspired by the work of other users (0.61). Furthermore users who evaluated the musical outcome as harmonious or danceable liked not only the musical outcome in general (0.62, 0.52) but also liked the experimentation with sounds in a collaborative setting (0.36, 0.50). This may also lead to the conclusion that the collaboration may foster experimentation rather than the targeted construction of sound forms. This is apparent in the correlation between musical skills (a score accumulated from several biographical questions assigned to every user) and more critical evaluations of our prototypes: the higher the musical skill the more disappointment (0.37), confinement (0.45), and overstrain (0.46) for all applications were expressed, especially regarding the general assessment of prototype two (0.38). However, it seems unlikely that musically skilled users evaluated the applications negatively because of the actions of their collaborators, since the higher the musical skill the more positively collaborators and the interaction with them was seen (0.25).

· ◇ ·

Many of our higher musically skilled participants attended a conservatory and thus may have substantial experience regarding musical collaborations. Given the correlation (0.32) that musically skilled users preferred to use the application three for a more targeted construction of sound forms and that users having stated that the applications exceeded their expectations especially assessed application one positively (0.65), it can be stated that for skilled users, application two is seen as too constrictive (correlation between skills and general assessment: -0.34) and application three as too difficult to comprehend (correlation between skills and comprehensibility: -0.23). The assessment of application one correlates positively with enjoying the experimentation with sounds (0.52). Additionally, there are positive correlations between the comprehensibility of the musical results



Figure 13.9: A sequence of frames captured from a video recording (20 seconds) made during the evaluations. Previous constructive actions led to a prompt reception phase in which all users spontaneously started dancing and joyously laughing after presumably having achieved a collectively felt successful collaboration. Similar behaviour was recorded with other groups as well.

generated with application one, on the one hand, and being able to implement own ideas (0.31) or experiencing pleasure (0.30) on the other hand. This comprehensibility of prototype one also correlates with perceiving the music as rhythmic (0.40) and with being inspired by other collaborators (0.39). This leads to the conclusion that application one in general is seen as more comprehensible and therefore traceable for collaboration. Additionally it seems suitable for sound-forms whose articulation is more rooted in timbre than in rhythmic (complexity) as can be found in today's dance music.

· ◇ ·

We assume that musically skilled users are adept to cope with and to create more rhythmically complex music. In this regard, a good prototype would allow such users to create rhythmically complex music compared to less skilled users. To gain more insight into such a relationship between musical skills and created rhythmical complexity, we performed a linear regression for these variables for each prototype. For prototype two, the slope of the best fit straight line is negative (coefficient -0.159), which may indicate that skilled users are not be able to express themselves accordingly while less skilled users may have created the complexity involuntarily. For prototype three the slope has no clear trend (coeff. 0.05), which may indicate that skilled users were unable to use their expertise for more targeted exploration. For prototype one, the slope has a stronger upwards trend (coefficient 0.24). Thus, from the viewpoint of the assumption previously made, this shows that this prototype is the one that scales best with users' musical skills. This more targeted interaction with prototype one, especially regarding the construction of the rhythmic structure, is also apparent in the users' evaluation of how they performed their changes (fig. 13.8e) and what was changed (fig. 13.8f). Furthermore this is also supported by the observation that the duration of gesture interaction includes a large portion of short events.

· ◇ ·

It is to say that it is important to gather more empirical data in order to be able to make further reliable statements investigating the appropriate level of expressiveness for collaborative composition support. The current results indicate that the first prototype is the most appropriate generic choice and that for exactly this type of musical engagement active mediation is not a necessity for facilitating group collaboration. The overall feedback on the application has been very positive as the experience of group Flow has been frequently observed. This furthermore supports the initial assumption in section 5.3 that the choice of electronic dance music as musical domain is appropriate to imply shared knowledge preliminarily.

Chapter 14

The Shared Composition as Optimization Problem

Chapter Abstract

This following chapter looks at achieving the shared musical intent as an optimization problem that can be ”solved” via active mediation from the IT system. A mixture of genetic algorithm and generative process is used to evolve grooves as shared composition in the collaborative setting. These grooves contain rhythmic, melodic and accompanying music material. A subsequent user study shows that this approach is feasible.

This chapter is based on the contribution [KLG14], “A Genetic Algorithm Approach to Collaborative Music Creation on a Multi-touch Table” by Klügel, Lindström and Groh.
All sections have been revised.

The implementation of the prototype was done in the scope of Andreas Lindsröm’s guided research project. The concept was derived in several iterations and discussions between him and me.

He implemented the generative music system and devised the additional musical patterns. The framework and user interface was implemented by the present author. The study was conducted by us both.

The source code has been made available:
A. Lindström and N. Klügel. *SoundDesigner*. Application, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/Mutator
N. Klügel. *Genetic Algorithm Toolbox for Scala*. Framework / Library, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/mutant5000

14.1 Introduction

In the previous chapters we have seen examples of applications that are embedded into the situative context. These applications have a mediating role inasmuch as they are used to consolidate different intents by the

collaborators and facilitate the expression of intents by augmenting the users' skills which is desirable especially for novices. This can be provided by either making use of compositional rules as in chapter 12. By guessing the users' intent function, compositional recommendations can be made or specific derivations of the shared composition can be generated. In this way, the application can be seen as taking part in the situative context as it engages in the creative act and becomes an active collaborator. Another viewpoint is that this procedure - guessing the shared intent and transforming the composition towards it - can be seen as an optimization problem where neither the domain specific knowledge nor the optimization function have been stated explicitly. In this chapter, a genetic algorithm will be used to establish such a form of mediating functionality. More specifically, the application makes use of both, evolutionary and generative processes, thus it falls into the category of *Evo/Gen* approaches [MSO13]. The aim is to have it adapt to the users' modifications of the shared composition in order to converge towards a collectively agreeable goal.

· ◇ ·

To our knowledge, there are no comparable contributions in a *collaborative* context that have evaluated such functionality in relationship to the group acceptance and *perceived* support in the creative task by the machine. Therefore the main contribution of this chapter is evaluating the feasibility of the approach. However, the main difficulty in solely evaluating the additional benefit is to avoid or at least minimize the bias that the expressiveness of direct controls themselves impose upon the application such as familiar control metaphors for controlling musical events or overtly self-evident gestures encoding musical event mappings. This may otherwise skew results by favouring proficient users for this use case or simplify the compositional process. The reasoning is that musical novices would benefit the most from a mediating functionality given that they presumably have no prior knowledge on how to approach the task of music composition. Therefore they may not be able to *musically* understand the effect of their interactions with the application in view of an intended goal. Thus, regarding the user interface and interaction metaphors, we decidedly put the users into a position where they have to rely on the mediating functionality, thus enforcing a loss of control at the expense of *perceived* expressiveness. This will be elaborated in more detail in section 14.4. It should be emphasized that for a less experimental prototype this approach is suboptimal because of the implications that the loss of control entails and because, generally, mediation should be transparent (cp. chapter 4).

· ◇ ·

To summarize, the objective of this contribution is to evaluate, whether active and adaptive mediation, specifically using a Evo/Gen process, adds additional value for supporting the creative task of collaboratively composing music. In this regard, the evaluation does not focus on expressiveness. We will devise a system that is capable of creating a variety of short musical forms (”beats”) that include bass and melody lines with accompaniment including chord progressions and drum patterns. We see the music composition system underlying the application as capable to create a realistic subset of music forms common in modern popular music.

· ◇ ·

The outline of this chapter is as follows: first we will give an introduction to genetic algorithms and review related work, especially with regard to the application of genetic algorithms. We will then motivate the concept and elaborate the prototypical implementation. This is followed by the

presentation of a user study that aimed to evaluate the prototype. The contribution concludes with the discussion of the results thereof and gives starting points for future research.

14.2 Genetic Algorithms

There have been many attempts to mediate or automate music composition tasks in order to have computers aid composing music that is aesthetically pleasing. Different approaches in the general field of algorithmic composition are described in [Mit98]. One prominent approach uses genetic algorithms to generate music. Genetic algorithms draw inspiration from evolution in nature in order to solve computational problems and were invented by John Holland in the 1960s [Mit98, p.2].

· ◇ ·

No domain specific knowledge about the problem is necessary and they can be applied to a vast array of optimization problems. Given a well-defined problem, candidate solutions to the problem are encoded as chromosomes or *genotypes*, usually as bit-strings. In order to evaluate the fitness of a genotype, a fitness function is used that assigns a score to it. In a simple form, a genetic algorithm starts by randomly generating a starting population of genotypes and assigning scores to each one using the fitness function. New chromosomes are then spawned from the initial population using crossover and mutation operations. Crossover between two ”parent” chromosomes is done by concatenating divisions of the bit-strings of the chromosomes in order to create a new one. Mutation of the resulting ”child” chromosome is performed by changing the value with a small probability at each bit position. This is done to ensure variance in the population over time. By giving highly fit individuals a bigger probability of being chosen for reproduction by crossover, the aim is to obtain high quality solutions after several generations of evolution. The overall procedure is depicted schematically in figure 14.1. Apart from the genotypes, a GA usually also defines *phenotypes*. These are in most cases the actual solutions to the problem which have been mapped from the genotypes.

· ◇ ·

When applying genetic algorithms in a musical composition context, the search problem can be defined as ”from the space of all possible compositions, find one that sounds good” [Jac95]. In a Evo/Gen system the genotypes represent sets of parameters for the generative system [MSO13]. The phenotype is then the concrete instance of a musical piece, phrase or in our case a beat that was generated. This contrasts to a direct representation which would assign elements of the bit-string directly to a single note. Two obstacles need to be overcome here in order to be able to apply genetic algorithms: first, an encoding of the musical structure or parameterisation of the generative processes must be defined so that genotypes can be created and evolved. Second, a fitness function must be defined that can rate a genotype or, indirectly, the quality musical output. Of these two obstacles, the second one is definitely the hardest one to overcome due to the difficulty of objectively rating the quality of music by automatic means. Nevertheless, quite a few applications have emerged using this approach. An overview of some of these can be found in [Nie09, ch. 7.4]. Many of the applications described usually perform a specific task (such as harmonizing a melody line) with comparison to human compositions as basis for the fitness function. Others use rules grounded in musical theory to rate the chromosomes.

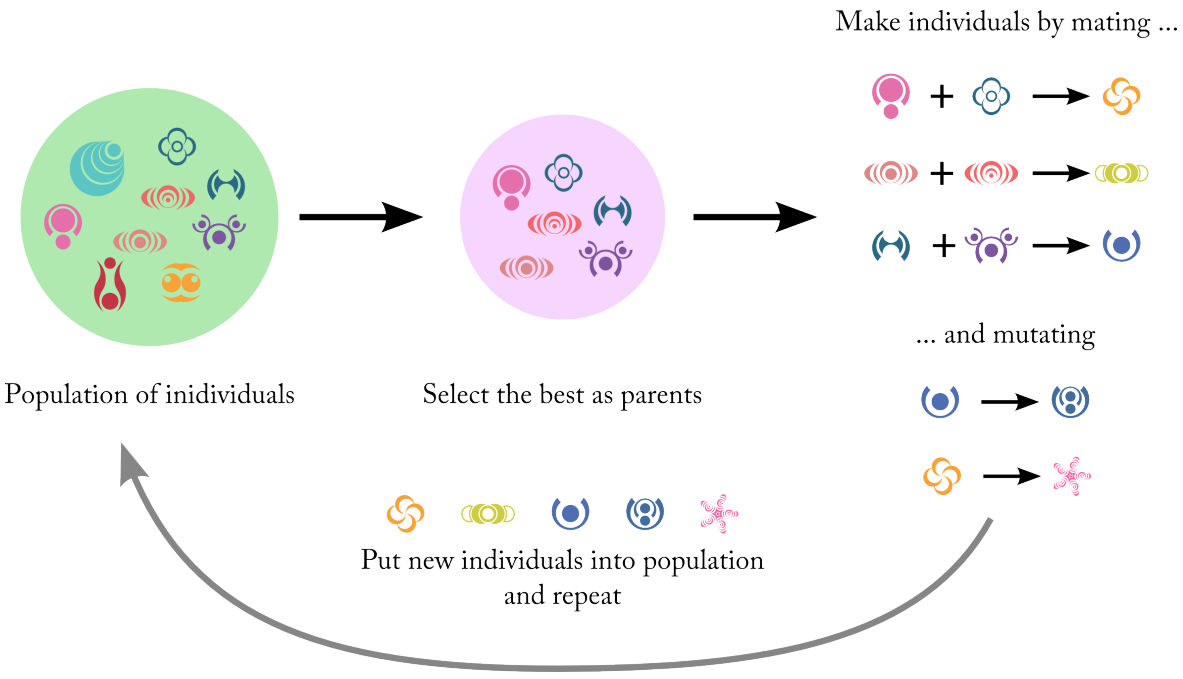


Figure 14.1: Schematic of a GA algorithm, which includes population, selection, mutation and iteration (based on figure 13.1 in [MSO13])

· ◇ ·

Another approach is to have a human-based fitness function by letting a human evaluator rate the output of the program. The evolutionary computation is therefore interactive. This is used for example in the application GenJAM [Bil94], which can generate jazz solos over a given chord progression. This circumvents the difficult problem of algorithmically rating musical quality. Similar approaches have been presented in [Sha+10] for interactive grammatical evolution of user controllable generator functions, in [MO11] for evolutionary created executable graph structures that generate music autonomously and in [McD+10] for deriving context free grammars that generate a piece. However, McDermott *et al.* identify several issues in applying evolutionary algorithms for interactive composition (paraphrased):

GEP1 Human evaluation of fitness is much slower than computer evaluation, this fitness evaluation bottleneck is especially problematic when music is being evaluated since audition times are proportional to the length of a piece.

GEP2 Users become bored, fatigued, and annoyed over long evolutionary runs.

GEP3 Users may encounter many bad individuals especially in early generations (cold start problem), when the initial population is created randomly.

GEP4 The typical fitness-evaluation interaction paradigm is closed as it does not afford flexibility and creative use.

GEP5 Populations can converge prematurely, good individuals can be lost or users can change their minds. This may lead to an evolutionary dead end.

GEP2 could lead to non-consequent rating from the evaluator which in turn gives a poor performance from the algorithm. Therefore such issues have

to be remedied for a practical real-time application of a genetic algorithm approach. With respect to GEP4 & GEP5, it is necessary that users have direct influence on the behaviour of the algorithm such that it can adapt to new composition objectives (*e.g.* an abrupt change in the group's stylistic preference). GEP1 can be circumvented if the mutation process itself is user-guided and controlled in real-time as will be explained in the next section.

14.3 Concept

There are some studies and applications that explore the possibility of using genetic algorithms in an interactive setting. One example is described in [FM02], where a genetic algorithm guided by the position of two users in space generates layered melodies. However, this application lacks a fitness function and only generates new genotypes based on the users' actions.

14.3.1 Controlling the Genetic Algorithm

Studies and applications of this kind are interesting for several reasons. It might be possible to overcome the previously mentioned fatigue problems when using human evaluators by letting the evaluators themselves interact with and affect the genotypes linked to the music creation. This can be seen as guided mutation operations, as opposed to the random mutations performed by the program. In fact, it is argued in [Gar02, p. 2-3] that development of musical structure can be seen as directed mutation; new musical ideas are generated by mutating other musical ideas.

· ◇ ·

Such an approach can be implemented using a turn-based composition strategy that comprises of repeated turns of active modification of the composition by the users (*e.g.* changing the scale or patterns of the shared composition) and evaluation by the users after which the genetic algorithm performs one cycle and then presents the users with an updated version of their previous composition. The evaluation can be implemented using a voting scheme, thus it corresponds directly to a fitness function in generic genetic algorithms.

· ◇ ·

Splitting the composition process into several cycles of phases furthermore has not only the advantage of users being able to directly alter the composition in real-time to steer the genetic algorithm but also to hear the results of their interactions immediately as feedback (first phase). As argued before this immediacy is necessary for any musical engagement to support iterative creation. Moreover, it alleviates the problems associated with GEP1.

· ◇ ·

To summarize, our approach implies a direct analogy between the human composition process and the compositional process performed by a genetic algorithm. This also serves as a motivation for the choice of using genetic algorithms with guided mutation from the users as an additional element, since the user input can directly be applied to the algorithm at run-time. Therefore, the approach involves the users in the music generation instead of just letting them passively listen and rate the automated music creation, in order to achieve faster convergence of the genetic search and to solve the previously discussed listening fatigue problem. With this,

we hope that this creativity support system converges towards the collaborators' shared musical intent.

14.3.2 Generative Processes

In order to limit the musical space that the users can explore, the application was designed to produce musical beats with a length of four bars. The beats have a granularity corresponding to 16th notes, not unlike that of a 16th-note step sequencer. This approach of course further limits the musical abilities of the application. The application can only produce note lengths that are integer multiples of a 16-th note, which means that for instance triplets can not be achieved. However, for the purpose of this experiment, this simple approach should be sufficient while it can be safely assumed that the application can be generalized to contain more complex rhythmic combinations and to support music that is not beat based, if the basic concept should prove to be successful. Overall, the application can be regarded as a simple generative music composition system because it makes use of parameterisations and transformations of basic musical structures instead of directly encoding notes.

· ◇ ·

A beat in the application consists of a drum part, a bass line, a melody and a chord sequence that defines the harmonic environment for the bass and melody parts. The drum part in turn consists of hi-hat, bass, and snare drum rhythms that can vary individually over the four bars. The bass and melody are monophonic with notes extracted from the current chord. Each chord consists of four notes that are extracted from a global seven note scale. One chord is applied to each bar, which means that the beat can be seen as a chord sequence of four chords. Apart from controlling the root note of each chord, the users can also control the specific voicing used for each chord as well as the rhythm of the chord playback. Other global parameters apart from the scale used are the tempo for the beat and a binary shuffle switch that applies to all instruments on a 16-th note level. By interacting with the interface of the application, the users can change the parameters of each part individually. An important question in this context is how to map the settings of the interface controls to patterns (rhythm and pitch) for each of the instruments. Also, in order to apply a genetic algorithm to the beat generation, an encoding is needed to represent different beats.

Pattern Mappings

Regarding the mapping from user input to musical patterns, an initial approach was to allow all possible combinations of beats by representing the rhythm of an instrument by four 16-bit numbers (one for each bar) where the value of the bit in position i decided if the i:th 16-note in that bar should be a hit (1), or a rest (0). By setting the range of the user controls for the rhythm of that instrument in that bar to the interval $[0, 2^{16} - 1]$, all possible 16-th note rhythms can be achieved. For each note in the rhythms of the melodic instruments (bass and melody), all scale pitches would be available. However, this approach turned out to be problematic for two main reasons. First, the encoding of a 4 bar beat would need more than 300 bits for its representation which might lead to very long convergence times, especially in an interactive setting with human evaluators. Second, a large subset of the possible beats would probably be considered non-musical by most users. Thus, this approach needed some modification.

To reduce the search space and to render more musical beats, predefined patterns were introduced to each instrument. The rhythmic patterns for each instrument still used the bit-extraction technique described above, but now the only a small subset of integers in the interval could be used (between 4-16 rhythmic patterns per instrument). For the melodic instruments, common arpeggiator patterns[1] were used. Eight different scales could be chosen as the harmonic environment for each beat, ranging from common scales such as Ionian (major) and minor pentatonic to more exotic scales such as diminished and whole-tone scales. The tempo, expressed in beats per minute, could be set to any integer in the interval $[60, 300]$ and the shuffle switch was indirectly defined as a modulo-2 operation of the sum of the drum parameters, resulting in a shuffled beat if they add up to an odd number. To imitate the shuffling of a human drummer, the shuffle ratio was decided by the tempo of the beat as described in [FS02]. The advantages of this pattern-based approach include more musically coherent beats and radically shorter beat encodings (more details about the encoding are given in section 14.3.2). The patterns can be ordered with respect to their rhythmic complexity [Thu08], which allows for more musically sensible mutations. An obvious disadvantage is that the musical space is totally defined by the pre-defined patterns which reduces the generality of the application and the musical freedom of the users.

[1] Ascending, descending, ascending/descending and alternating.

Beat Encoding

With this pattern based approach, the genotype of a beat could be encoded as a string of concatenated integers, with each integer giving the pattern index of a specific instrument. Thus, the integers specifying which pattern to apply for each instruments correspond to a gene in the chromosome. The pattern encoding for each instrument is given in table 14.1.

Instrument / Parameter	# patterns
Hi-Hat	8
Snare drum	4
Kick drum	4
Chord root note	7
Chord voicing	8
Chord rhythm	4
Melody	16
Bass line	16
Scale	8

Table 14.1: Patterns used for the various instrument types or compositional parameters and their respective number of alternate instances

As an example, the gene integer that corresponds to the melody pattern in a bar takes a value in the interval $[1, 16]$. A beat contains four bars in which the patterns can vary independently (with the exception of the scale, which is global for all four bars). Disregarding global tempo and pitch parameters, the cardinality $|S|$ of the set of unique beats that the application can produce is

$$|S| = 8 \cdot (8 \cdot 4 \cdot 4 \cdot 7 \cdot 8 \cdot 4 \cdot 16 \cdot 16)^4 \approx 2.3 \cdot 10^{28} \tag{14.1}$$

Half of these beats will be shuffled beats. The tempo and pitch of each beat can be altered for further variation. This result ensures that the musical space available to the users to explore is still very large, despite using the limiting pattern approach.

14.3.3 Fitness Function

With the encoding of beats defined, all that remains is to specify a fitness function that grades the quality of a beat. Due to the difficulty of algorithmically grading musical quality (see section 14.2) and due to having the goal of a highly interactive setting, the choice was made to implement a voting system for the fitness function. After modifying a beat, each user can grade the current beat by supplying a decimal value in the interval $[0, 1]$, where high scores correspond to high musical quality. The average of all votes is then fed to the genetic algorithm and used as fitness. By using this approach, the aim is to map the group's creative will into a singular value, thus allowing the beats to converge to the group's preference over time.

14.4 Interface & Prototype

In order to investigate how genetic algorithms can support and enhance the creativity of group composition, an application was developed that realizes the concept of applying user input as directed mutation to a genetic algorithm that generates musical beats. Regarding the emphasis on the mediating functionality presented in the introduction, we decided to use abstract shapes to control the various parameters inherent in the composition which deliberately do not suggest concrete musical meaning. Thus, for a user not knowing the underlying pattern based system, there is no immediate connection between interaction and the *exact* musical result. However, all parameters of the shared composition are represented with a direct one-to-one mapping: each individual instrument or parameter was represented as a shape, changing their size or three dimensional rotation (using multi-touch gestures) corresponded to the pattern selected for that particular instrument of parameter. Furthermore, besides the scale factor of a shape the amount of absolute rotation was indicated by illustrating arcs surrounding it, giving visual feedback that the interaction had been registered by the application. In this way the state of the application is always visible and not artificially obfuscated. Instead, it is expressed in terms that are not making use of established musical vocabulary and that only allow a rough guess of what the parameter values represented by the position of a shape are (*e.g.* the visualization of a rotated shape does not indicate which pattern exactly has been chosen).

· ◇ ·

A screen-shot of the interface of the prototype can be seen in figure 14.2. The four blue and yellow shapes correspond to chord and drum parameters, respectively. The grey pyramid shape at the top of the screen controls the melody and the cube to its right controls the bass parameters. The two red shapes to the right controls global scale, pitch and tempo. The three voting panels can be seen, as well as a centered circular progress indicator that completes one lap in the four measures of the beat.

· ◇ ·

When the application was started, an initial population of six beats were randomly generated and given low random fitness scores. One of these randomly generated beats was chosen as the initial beat for the users to modify. The shapes were automatically resized and rotated according to the randomly generated patterns corresponding to each shape. This beat could now be heard by the users and they could modify it by moving and rotating the shapes. The users were free to vote at any time, and when all

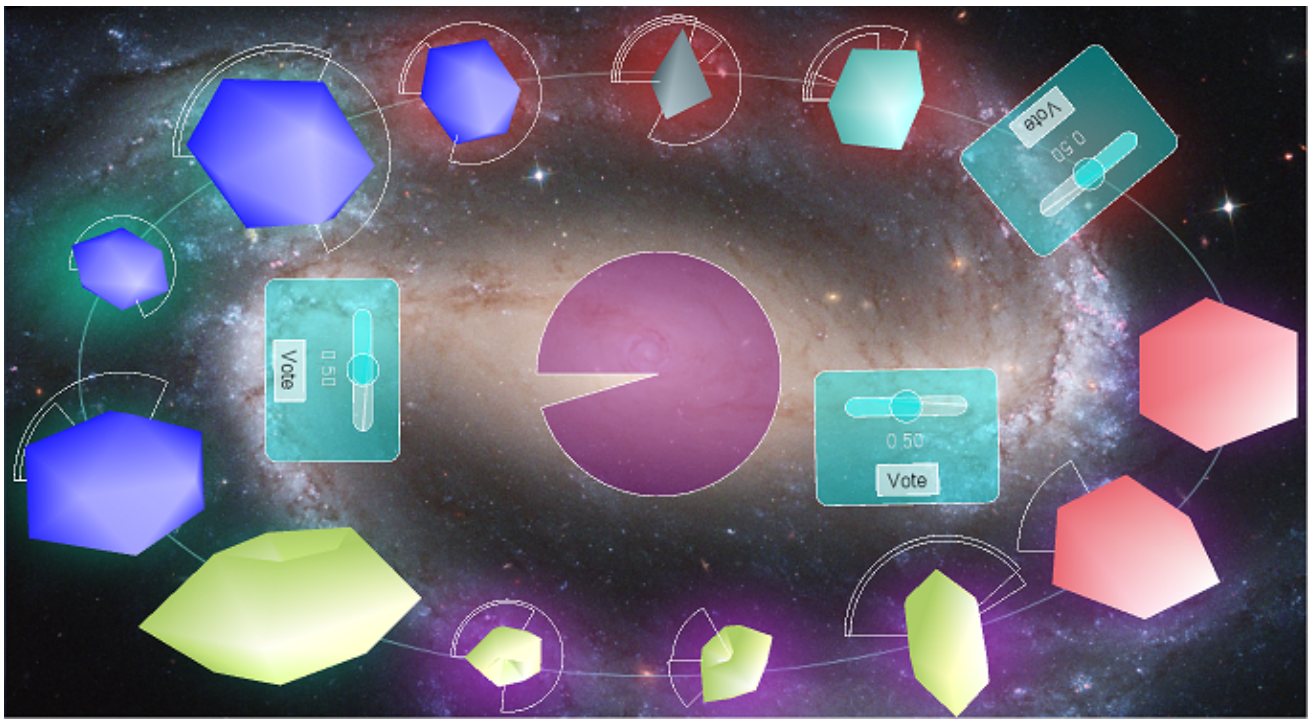


Figure 14.2: A screen-shot showing the interface of the prototypical implementation; the implementation re-used UI components of the prototype 13

three users had supplied a vote, the average score was supplied to the genetic algorithm. Two beats were then selected for mating from the entire population. A new offspring was created using crossover of the genotypes and the resulting child subjected to random mutations (usually by adding or subtracting a small number to the pattern index) and the child beat was given to the users for modifications. The chance for a beat of being selected for mating was proportional to the fitness rating of the beat. To further speed up convergence, we performed additional modifications to the genetic algorithm, namely the reduction of the population size to at most ten genotypes and a reduction of the fitness score of old beats after each new generation to premier new beats added to the population.

· ◇ ·

We used a 55 inch multi-touch table to interface with our application. The number of parameters available to modify the beats as well as the dimensions of the multi-touch board made the prototype suitable to use in groups of three, with the intention that one person controls the drum parameters, a second the chord parameters and a third the melody and bass. The control over the global parameters scale, tempo and pitch could be shared between the users. This can be seen as an analogy to a traditional band environment. However, for the evaluation we explicitly told the participants that they are free to change position as they wish during the testing.

· ◇ ·

The interface and genetic algorithms were implemented using the aforementioned framework (chapter 6) and sending OSC messages to a PureData [Puc+96] patch that realized the patterns. From this patch, MIDI messages were sent to a running instance of the DAW *Ableton Live* [Abl02] which played back the beat using stock instruments. The deployment diagram of this setup is shown in figure 14.3.

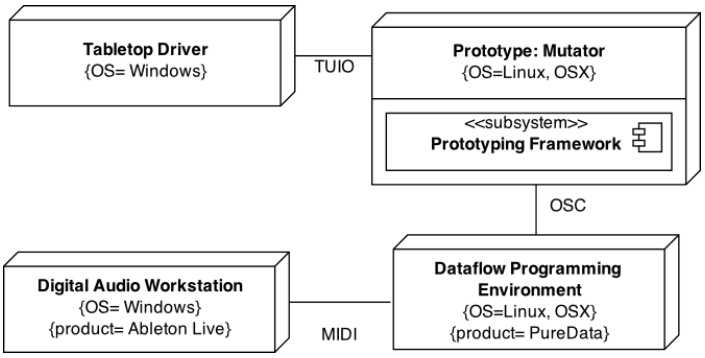


Figure 14.3: UML deployment diagram of the presented prototype (Mutator)

14.5 Evaluation

18 test subjects were divided into groups of three for the evaluation. The test subjects were engineering students of which 61% played an instrument or sang in their free time. 56% had received musical training in some form. The test subjects' amount of previous experience with multi-touch interfaces varied substantially. The evaluation was divided into two consecutive parts: first, after a brief explanation of the functionality of the application, the test subjects were allowed to create beats with the application for 25 minutes. During this session, the users were allowed to experiment and use the voting system freely. They were instructed that they could use any means of collaboration available to them, verbal as well as non-verbal. They were also told that they could move around the multi-touch table to rotate the control of the beat parameters within the group. In the second part of the evaluation, the test subjects answered a computer-mediated questionnaire containing questions regarding collaboration aspects, expressive capabilities of the prototype and the user experience. Most questions used a 5-level Likert scale.

14.6 Results

As expected, the application was rated low to mediocre regarding control and expressiveness. Only 45% of the participants agreed that they found it easy to control specific parts of the composition, 67% to control their parts of the music such as instruments or and only 50% control over the music as a whole. Furthermore, they felt that they could not express their creativity properly (44% agreement), but express themselves musically (63% agreement). Still, and in accordance with our assumption in the introduction that unacquainted users would still be able to make out the effect of their interactions, 71% agreed that their actions affected the music. Additionally, concerning the potential for exploring the musical possibilities, 84% of users thought there was a lot to discover with the application.

· ◇ ·

Regarding collaboration, 71% saw their collaborators as enrichment for their creative endeavours, 67% agreed that the application supported their collaboration and furthermore 72% felt that the application helped them being creative as a group.

· ◇ ·

With respect to the most important part of the evaluation, the mediating functionality was voted favourably. The average number of voting rounds was 11, indicating that this part of the system was frequently used during

the comparatively short time for evaluation. In accordance to this, only 24% of the participants thought of the voting system to be unnecessary. 83% of users stated that the music got better over time, and 66% stated that after completing a single voting round, the musical quality of the new offspring beat that was presented was higher than the parent beat while still being recognizable (70%). Half of the users said they were helped to new musical ideas and also stated that their group was mostly in agreement about the voting (73%).

· ◇ ·

Given the low ratings in regards to control, it is surprising that only 38% percent of the users stated that they got frustrated using the application. According to the favourable reviews of the collaborative and mediating aspects, 80% of the participants felt that it was fun to make music with the application and, equally, that they enjoyed using it. This is surprising, especially since the results were mixed with respect to liking the music that has been produced.

· ◇ ·

Regarding verbal communication, it was noted that groups where at least one member did not know the other members used a lot less verbal communication in the form of directives or discussions as opposed to groups where all members knew each other from before. Furthermore, some groups frequently changed positions around the table, which may hint at their more democratic approach to using the application.

14.7 Discussion & Conclusion

The problems revealed in the evaluation are related to the usability of the prototype. The difficulty for users to understand the functionality of the controls and inability to modify specific parts of the beat to their liking can be traced to the abstract layout of the interface. Given this contrast to the fact that the majority of users thought the quality of the resulting beat improved (after single or multiple voting round) compared to the original beat indicates that the genetic algorithm approach may indeed fulfill its role in supporting the users in their creative endeavours. Furthermore, the aim of having software inspiring new musical ideas is rather ambitious and the fact that half of the users got new ideas from the genetic approach can therefore be seen as positive. However this result should be regarded as preliminary and therefore taken with caution, since additional experiments are necessary to differentiate between beneficial social effects of the collaboration, the efficiency of the genetic approach, or possible placebo effects. Therefore we argue that a possible approach would be to evaluate the application with two modifications: completely randomized offspring and offspring that are not altered at all. With these results it should be possible to remove the bias from the evaluations. We credit the mixed results regarding the produced music to both the instrumentation (stock synthesizer patches) and the use of patterns. As the space of possible compositions is still confined to these and that some users may have preferred other musical styles.

· ◇ ·

To conclude, we nevertheless see these results as satisfactory, especially regarding the favourable feedback to the collaborative aspects in general, indicating that there is additional value in supporting collaborative creativity actively by taking part of the decision process is a viable option. For

more complete applications, however, the aim should be not only to add expressive or musically meaningful controls for the composition but also to the synthesis of sound for shaping timbre.




Part

V

Conclusion




Chapter 15

Summary and Future Work

Chapter Abstract

The following chapter concludes this thesis. It summarizes the presented contributions with respect to the research question posed in the introduction. Furthermore, starting points for future work are proposed which include additional studies, and extensions to the presented approaches and prototypes.

Based on an extensive literature overview, it was argued that music, music-making and meaning-making are embedded into socio-cultural constructs and make use of embodied practices. As argued in chapter 2, modern practices in electronic music production do not integrate collaboration as a foundational aspect or are even diametric to it (*i. e.* the bedroom producer as target group for music making devices). In practice, the beneficial effects of collaborative music-making are therefore not facilitated for electronic music. These insights motivated the focus on creativity support for collaborative electronic music-making. It was pointed out that physical interaction with artefacts and the physical space itself are beneficial to social interaction and knowledge building. In consequence, an interactive tabletop interface is seen as facilitator to meaningfully support such embodied practices. To summarize, the central theme was to embed technology into a social (or socio-cultural) situation. The reason why multi-touch tabletops were focussed instead of tabletops that also feature tangible interaction is that derived interaction paradigms are more widely applicable.

15.1 Summary

The methodological basis for the presented research is a design science approach. As highlighted in the introduction (section 1.4.2), approaching the presented research topics is motivated by a critique on a previous contribution [Klü+11]. In this regard, the aim was to improve upon it - true to the iterative search approach in design science. The critique in the introduction (section 1.4.2) stated that the previous work lacked focus on contemporary practices to create music, and that it mixed musical domains which prevented establishing such focus. Contemporary practices should include means that support exploration of a musical domain, reduce the

necessity to perform peripheral tasks and lower thresholds for musical engagement using alternative domain representations (or models). Central to this is the notion of mediation as established throughout this thesis that derived high-level models to control musical processes. Electronic music was approached within the three foundational domains, timbre, rhythm and melody. For each domain, prototypes were developed that are based on practice informed considerations on how they are made effective:

- *SoundDesigner* (chapter 9) focussed on sound design using a perceptually informed model.
- *FugueGenerator* (chapter 12) allowed the creation of melodies from an affect related viewpoint.
- *TreeQuencer* (chapter 13) facilitated groove production with emphasis on expressive rhythm patterns and simple sound design.
- *Mutator* (chapter 14) supported groove production using pattern based rhythm, bass-line and melody creation.

To summarize, the main research question on *"How can the performative collaborative creation of electronic music in a Single Display Groupware environment be supported for novices and experts alike?"* is decomposed and approached in the following way:

Support for novices and experts alike: Model building and high-level control over musical processes.

Electronic music: Foundational musical domains embedded into practice informed prototypes.

Performative creation: Real-time control over musical processes.

Collaboration: Design traits derived from musical HCI, creativity support systems and embodied practices.

Further iterative steps with respect to the presented topics are seen as (partly divergent) prospects for future work and are laid out in section 15.3.

· ◇ ·

Apart from the various quantitative and qualitative measurements used to gather insights in the evaluations, one central aspect of the success of a prototype was estimating the user experience, specifically the occurrence of (group) Flow. The experience of Flow or at least joy in using the application was observable in all evaluations in varying extent. For the prototype *TreeQuencer* a Flow experience was perhaps most clearly demonstrable because apart from self-reports the sessions were recorded on video, and because the evaluation featured the largest count of participants. For all prototypes the experience of making music together was appreciated favourably and seen to feature additional benefits over (hypothetical) solo use. This was also the case for the prototype *Mutator* where control over musical processes have been made deliberately difficult. Thus, for the prototypes, collaboration brought in additional hedonistic value. This is a very satisfactory result, given that music-making itself is mainly a hedonistic activity.

· ◇ ·

Another viewpoint in this thesis was to see the Creativity Support System as a mediator between different musical intents of the collaborators apart from the mediating role that it is given naturally as technological artefact included or embedded in a social situation. Two approaches were presented, passive mediation where high-level parameters were derived from a modelling activity and active mediation where the computer takes the role of a participating collaborator (cp. table 15.1). For the former the primary aims were to reduce peripheral tasks and create mappings for control of musical processes that lower thresholds for novices while still facilitating the exploration (expression) of the respective musical domain for intermediates. Specifically the argumentation in chapter 4 corroborated this reasoning such that the theoretical background was put into effect in the modelling activities and their implementations. This approach can be seen as effective for the prototypes since all evaluations included mixed groups (intermediates and novices) and there was no severe discrepancy in the reports with respect to the support of collaboration depending on skill-level. However, designing for both, exploration and low thresholds is difficult and did not always succeed fully. In case of *FugueGenerator*, the application was regarded as too simplistic for extensive expression by intermediates. The study related to the prototype *TreeQuencer* revealed that mediation may not be *necessary* for certain musical domains. However, the operational concept of this application already relies marginally on peripheral tasks. From this viewpoint, it could be argued that future iterations may not focus on refining the concept but rather explore alternatives to it. In case of the comparative study related to the prototype *SoundDesigner*, it was on the other hand revealed that such a modelling approach can not only successfully reduce peripheral tasks but also help understanding the musical domain. Finally, chapter 11 introduced another model which could be applied for tasks where timbre selection or sound design are peripheral but depend on it with respect to an intended affect. Since these results are diverse, future work is needed to gain further insights as will be discussed in section 15.3. Active mediation, where the computer participates in the collaboration and also alters the shared composition, was implemented in the prototype *Mutator*. Here, the approach was focussed on supporting the manifestation of intent directly by guessing "the users' intent function" and adapting the shared composition accordingly. In this regard, the aim of the modelling activity is not necessarily to reduce peripheral tasks but to abstractly model intent or behaviour (of the users). While in the case of *Mutator*, the experiment was contrived since it forced users to rely on this functionality, it nevertheless showed that this approach to mediation is a viable alternative.

· ◇ ·

Apart from the additional hedonistic value that collaboratively making music offers, we have seen that it also facilitates bonding and inclusion of people in the social context, given that group Flow was experienced (and sometimes signalled extrovertedly) while collaborators were mixed with respect to background and frequently did not know each other before the evaluations. Similarly, we have seen that a group of diverse people could be integrated in the creative task.

15.2 Approaches

The following section summarizes the main contributions in this thesis, it includes the derived design paradigms and highlights key aspects. This is

Prototype / Model	*Focussed musical domain*	*Modelling*	*Method*
SoundDesigner, ch. 9	sound design, exploration of timbres	perceptually motivated timbre space; audio feature analysis and data driven model building	probabilistic model as shared navigational space
Affect related timbre analysis, ch. 11	sound design, exploration of timbres, automatic timbre selection	affect informed timbre space	probabilistic model
FugueGenerator, ch. 12	harmony, composition of simple melodies	affect related composition rules	generative
TreeQuencer, ch. 13	groove production	complex rhythmic structures	semi-generative
Mutator, ch. 14	groove production	shared compositional intent	generative, evolutionary

Table 15.1: Overview of the devised prototypes, their respective musical domains, underlying models and methods of implementation

done in tabular form to allow for a quick overview.

· ◇ ·

The thesis included five main contributions which are either pure computational models or prototypical applications that make use of exemplary modelling of a musical domain. Table 15.1 summarizes this and indicates what the main focus and characteristic of the respective model is. Apart from studying user experience, several prototypes also implied further musicological evaluations. This and the focus of the studies is outlined in table 15.2. Finally the various abstractions from the standpoint of interaction, artistry and signal processing that have been employed in the prototypes are shown in table 15.3.

15.3 Prospects for Future Work

Design science refines research artefacts and the focus of the research in iterative steps and thus it is inherently bound to test prospects for future work. The prospects for this thesis are shown in the following sections and include musicological studies, and alternatives and extensions to the presented methods, human computer interfaces, audio synthesis and approaches to collaboration.

15.3.1 Studies

There are several questions that have yet to be conclusively answered or at least be addressed to facilitate decisive design practices. Specifically, establishing more insight into the following areas would be invaluable for framing the contexts of mediation methods, thus making mediation more effective. With respect to collaboration schemes it is widely argued that true democratic collaboration holds most benefits for the creative process. However, there is a lack of comparative studies that evaluate this form of

Prototype / Model	*Study focus*	*Methodology*
SoundDesigner, ch. 9	UX/HCI: human cognitive manageability, mediation of technical control musicology: comparison to classic / prevalent synthesis control	user study, 22 people
Affect related timbre analysis, ch. 11	computational: feasibility of model	cross-validation (machine learning)
FugueGenerator, ch. 12	UX/HCI: human cognitive manageability, mediation of compositional rules musicology: appropriateness of established compositional rules for creative context	user study, 16 people application event logging and analysis
TreeQuencer, ch. 13	UX/HCI: group Flow depending on rhythmic complexity musicology: comparison of perceived and performed rhythmic complexity depending on degrees of freedom	user study, 31 people application event logging and analysis
Mutator, ch. 14	UX/HCI: active mediation of shared compositional intent	user study, 18 people

Table 15.2: Overview of the methodologies used in evaluating the prototypes and the respective study focuses

collaboration against alternatives such as a hierarchic one and put it into the context of a specific musical task or in a relation to the skill-level of the involved interlocutors. Therefore the dependencies in this triangular relationship are not clear such that the design space and the resulting implications are diffuse. Likewise, there is a lack of understanding which musical tasks are difficult with respect to musical background. While it was pointed out in the introductory chapters that such a holistic understanding cannot be achieved, an empirically established consensus on these issues is necessary to focus further research. More consensus is also needed in musicology related research that is concerned with timbre, its description and perception as was pointed out in chapter 7. Cross-sectional studies could at least establish agreeable vocabulary for research and point out salient timbral features for non-musicians as opposed to studies performed that principally focussed on musicians. In the evaluation of *FugueGenerator* several relationships between melody contour and intended affect were discovered which do not wholly overlap with the referenced literature. If further studies reliably elicited and proved similar rules, these could be employed in future affective algorithmic composition systems as empirically informed rules. Thus overlap with user expectation could be enhanced.

· ◇ ·

Analogous to the above, there is a necessity to gain more insights into interface related issues. In chapter 4 it was argued that tangible interfaces are advantageous over purely multi-touch based ones because they allow for a richer exercise of embodied practices. There are studies that corroborate this line of argumentation, showing that tangible tabletops have beneficial effects for understanding and learning in certain (collaborative) use-cases (cp. [Sch+11a]). However, more systematic experiments are necessary to conclusively frame the effects that tangible interaction has. More import-

antly, it is not clear, what the specific effects are in the context of music related tasks and collaboration[1]. If more information in this regard was available, a comparison between the gains of tangible interaction and the added interface or application complexity would be feasible.

[1] Questions are for example: Does it allow more fine grained control? Does the increase in physicality of interaction equal more hedonistic value? Does it allow more transparent feedback?

· ◇ ·

One problem is the audio delivery in application scenarios where users need to distinguish their own contributions from the musical whole. In case of the study related to *SoundDesigner*, the motivation for this was to provide private acoustic spaces that allowed users to experiment without disrupting the other contributors. More generally, one issue is to make contributions in the audio stream apparent without interfering with the musical whole. For the named prototype, relying on separate headphone channels introduced awareness issues. As discussed in section 9.9, one solution may be to use ambisonic speaker arrangements or binaural (spatializing) audio encoding for headphones to indicate from whom the audio stream is coming from. While the reasoning for these alternatives is sound, apart from the study in [Xam+14] where ambisonic audio delivery has been shown to positively affect awareness and immersion in the musical experience, there is a lack of further evidence comparing these different approaches for the collaborative use-case. Specifically, there is need to understand the impact the audio delivery method has on awareness with respect to the various forms of and domains in music making in order to make suitable design choices.

· ◇ ·

Finally, with respect to the prototypes, specifically *Mutator*, and the role of applications supporting the collaborative creation of music in general, additional studies are necessary to differentiate and investigate the relationships between the beneficial social effects of collaboration, successful mediation and placebo effects.

15.3.2 Improvements and Alternatives

Apart from studies that generally help framing the scope of the presented research area or establish insights to directly inform design traits, there are also incremental improvements or alternatives to the presented methods that may be employed for these or future prototypical implementations. While approaches to minor improvements have been laid out in the respective chapters before, the ones shown here can be regarded as more invasive deviations from previous concepts.

· ◇ ·

In chapter 4, it was argued that there is a dichotomy in HCI with respect to safety and serendipity or inspiration. This meant that human error may lead to the deviation of a musical artefact from the original intent on the one hand, and inspire new ideas on the other. This duality may be resolved using technical means that embrace branching and versioning of the composition process in a confluently persistent system. Rutz presented an approach for this in [Rut12] which performs temporal traces of the composition process and further allows their confluent transformations. While there are difficulties that have to be overcome on how to access the functionality in a multi-user environment from the interface perspective, the integration of such a system would allow to promote and embrace human error while providing safety.

· ◇ ·

Domain	*Abstraction from*	*Reference in thesis*
Interaction	Complex event automation paths in sound design	Projection of low-dimensional event paths to high dimensional synthesis parameters informed by audio features, chapter 9
Music / Artistry	Compositional rules to induce emotion Harmony & rhythm in beat loops Complex musical event generation for rhythm composition	Generative composition & computer aided transformations of user input w.r.t. emotional intend, chapter 12 Generative, pattern based composition system, chapter 14 Extended step sequencing based on tree traversal, chapter 13
Signal Processing	Specific synthesis architecture Synthesis structure, parameter mapping & synthesis control (w.r.t. perception, affect)	Corpus based synthesis & sampling of synthesis parameters, section 9.4 Feature-based audio synthesis, chapter 9; Affect related audio analysis, chapter 11; (simple) affect related synthesis, section 12.4.2

Table 15.3: Overview of the abstractions in the domains interaction musical / artistic and signal processing and their applications within this thesis

With respect to audio synthesis, and timbre or affect space modelling, there are also alternatives to the currently presented methods possible. Instead of feature analysis of sample data and a machine learning based model, another approach would be to use a synthesis by analysis method that directly transforms the sample data in a musically meaningful way. For example, in case of the affect model shown in chapter 11, the data set contains audio samples which have been associated with a V/A value. Using a resynthesis method, such samples can be altered in fundamental pitch and length. Spectral modelling (see section 8.2) or phase vocoder synthesis [KLB06] are common methods to achieve this and have been adopted widely in studio practices and commercial products. There are also alternatives, such as the audio synthesis method using signal decomposition presented in a previous contribution [Klü12] or FM parameter estimation using genetic algorithms [Lai+06]. While transformations using audio resynthesis can only be applied within certain bounds because otherwise the original affect (or perception) related data may be invalidated (as discussed in 9 and [Mar+03]), and because such methods introduce artefacts for extreme transformations, they nevertheless constitute practical alternatives to the more generic models that have been presented. An extension to using plain resynthesis methods could be to learn transformations that are related to a certain affect or perceptual property as parameter modifications. For example, several high level features for audio analysis rely on spectral features, for such mappings an inverse mapping may be construed such that these can be embedded into an additive synthesis method. This way spectral components of the resynthesized signal could be altered to comply with spectral features that have been associated with a certain affect or perceptual property.

· ◇ ·

It has been suggested that a perceptually informed timbre space could be constructed which uses orthogonal features and maintains a perceptual ordering of timbres [McA+95]. Such a model would afford to analyse

and modify timbres within relationships similar to notes and scales (*e.g.* timbre inversion or transposition). This would provide new starting points for mediation in which sounds are adapted in relationship with each other or which are based on common compositional rules that are derived from such a model.

· ◇ ·

Finally, this thesis has not approached the topic of creating song structures as part of a musical activity. For example, the computer could assist in creating build-ups of a piece by selecting and arranging user generated building blocks or transform these to fit into the structure of a piece. Approaches to this could rely on a music theoretical background[2] or on machine learning to better accommodate modern structures in electronic music.

[2] *e.g.* based on Gestalt theory, derivations thereof [Rey97], or computational models for Schenkerian analysis [Mar10]

Bibliography

[AB99] A. Adler and C. Brett. *Social interest: Adler's key to the meaning of life*. Oneworld Publications Ltd, 1999 (cited on page 37)

[Abl02] Ableton AG. *Live*. [Online; accessed 28-March-2015]. 2002. url: https://www.ableton.com/en/live/ (cited on pages 120, 193)

[Adv06] Advanced Media Workflow Association. *Advanced Authoring Format*. [Online; accessed 28-March-2015]. 2006. url: http://www.amwa.tv/ (cited on page 19)

[Akk+11] V. Akkermans, F. Font, J. Funollet, B. De Jong, G. Roma, S. Togias and X. Serra. "Freesound 2: An improved platform for sharing audio clips". In: *Proceedings of the 12th International Society for Music Information Retrieval Conference*. International Society for Music Information Retrieval (ISMIR). 2011 (cited on pages 136, 139)

[Alg] Algorave (international, independently organized). *Algorave Homepage*. [Online; accessed 28-February-2015]. url: http://algorave.com/ (cited on page 64)

[Ans+13] M. G. Ansdell, M. Pavlicevic, C. Elefant and B. Stige. *Where music helps: Community music therapy in action and reflection*. Ashgate Publishing, Ltd., 2013 (cited on page 7)

[App10] Apple Incorporated. *iPad*. [Online; accessed 28-February-2015]. 2010. url: https://www.apple.com/ipad/ (cited on page 11)

[Arm10] J. Armstrong. "Erlang". In: *Communications of the ACM* 53.9 (2010), pp. 68–75 (cited on page 63)

[Ass+99] G. Assayag, C. Rueda, M. Laurson, C. Agon and O. Delerue. "Computer-Assisted Composition at IRCAM: From PatchWork to OpenMusic". In: *Compututer Music Journal* 23.3 (1999), pp. 59–72 (cited on page 64)

[Ass60] A. S. Association. *American Standard Acoustical Terminology*. American Standards Association, New York, 1960 (cited on page 77)

[Aud15] Audiofanzine.com. *Akai MPC60*. [Online; accessed 10-November-2015]. 2015. url: http://img.audiofanzine.com/image.php?lang=en&identifier=id&size=normal&module=product&product_id=90459 (cited on page 53)

[Bai+13] E. Bainomugisha, A. L. Carreton, T. v. Cutsem, S. Mostinckx and W. d. Meuter. "A survey on reactive programming". In: *ACM Computing Surveys (CSUR)* 45.4 (2013), p. 52 (cited on pages 60–62)

[Bai07] F. Bailes. "Timbre as an elusive component of imagery for music". In: *Empirical Musicology Review* 2.1 (2007), pp. 21–34 (cited on page 83)

[Bam77] J. Bamberger. "In search of a tune". In: *The arts and cognition* (1977), pp. 284–319 (cited on page 23)

[Bar09] J. Bardzell. "Interaction criticism and aesthetics". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2009, pp. 2357–2366 (cited on page 45)

[Bar98] L. F. Barrett. "Discrete emotions or dimensions? The role of valence focus and arousal focus". In: *Cognition & Emotion* 12.4 (1998), pp. 579–599 (cited on page 130)

[BB07] B. Buxton and W. Buxton.
Sketching user experiences: getting the design right and the right design.
Morgan Kaufmann, 2007 (cited on page 27)

[BBP07] O. W. Bertelsen, M. Breinbjerg and S. Pold.
"Instrumentness for creativity mediation, materiality & metonymy".
In: *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*. ACM. 2007, pp. 233–242 (cited on pages 6, 47, 48, 55)

[BBP08] O. W. Bertelsen, M. Breinbjerg and S. Pold.
"Instrumentness in Clusters of Artefacts–a First Take on Collaborative Creativity".
In: *Proceedings of the extended workshop in Artifacts in Creative Collaboration* (2008) (cited on page 51)

[BCV13] Y. Bengio, A. Courville and P. Vincent.
"Representation learning: A review and new perspectives". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35.8 (2013), pp. 1798–1828
(cited on page 149)

[Bea07] J. W. Beauchamp.
Analysis, Synthesis, and Perception of Musical Sounds: The Sound of Music.
Springer Science & Business Media, 2007 (cited on pages 82, 87, 104)

[Bec13] T. Becker.
"Development of an Application for Colaborative Sound Design on a Multi-Touch Table".
MA thesis. Technische Universität München, 2013 (cited on pages 6, 52)

[Ben+07] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle et al.
"Greedy layer-wise training of deep networks".
In: *Advances in neural information processing systems* 19 (2007), p. 153
(cited on page 148)

[Ben08] D. J. Benson. "Music: a mathematical offering".
In: *The Mathematical Intelligencer* 30.1 (2008), pp. 76–77 (cited on pages 87, 89, 98)

[Ben09] Y. Bengio. "Learning deep architectures for AI".
In: *Foundations and trends® in Machine Learning* 2.1 (2009), pp. 1–127
(cited on pages 144, 146, 147)

[Ben10] J. Bennett. "Collaborative songwriting–the ontology of negotiated creativity in popular music studio practice". In: *Proceedings of the 6th Art of Record Production Conference*. Vol. 2. 2010, p. 1 (cited on page 25)

[Ber04] O. W. Bertelsen. "Transparency by Tertiary Artefactness".
In: *Aesthetic Approaches to Human-Computer Interaction* (2004), p. 1
(cited on page 48)

[Ber69] D. P. Berger. "The Shakuhachi and the Kinko Ryu Notation".
In: *Asian Music* (1969), pp. 32–72 (cited on page 78)

[BES10] S. Baccianella, A. Esuli and F. Sebastiani. "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining." In: *LREC*. Vol. 10. 2010, pp. 2200–2204 (cited on page 140)

[BF03] T. Blaine and S. Fels. "Collaborative musical experiences for novices".
In: *Journal of New Music Research* 32.4 (2003), pp. 411–428
(cited on pages 1–3, 11, 27, 31, 50)

[BFS02] R. Bresin, A. Friberg and J. Sundberg.
"Director musices: The KTH performance rules system".
In: *Proceedings of the 46th meeting of the IPSJ Special Interest Group on MUSic and computer (SIGMUS)* 46 (2002), pp. 43–48 (cited on page 156)

[BG14] J. Bresson and J.-L. Giavitto.
"A reactive extension of the OpenMusic visual programming language".
In: *Journal of Visual Languages & Computing* 25.4 (2014), pp. 363–375
(cited on page 65)

[BHL07] N. Bryan-Kinns, P. G. Healey and J. Leach.
"Exploring mutual engagement in creative collaborations".
In: *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*. ACM. 2007, pp. 223–232 (cited on pages 40, 158)

[Bia12] M. Biasutti. "Group music composing strategies: A case study within a rock band".
In: *British Journal of Music Education* 29.03 (2012), pp. 343–357 (cited on page 23)

[Bil94] J. Biles. "GenJam: A genetic algorithm for generating jazz solos".
In: *Proceedings of the International Computer Music Conference*.
International Computer Music Association. 1994, pp. 131–131 (cited on page 188)

[Bis+08a] M. Bischof, B. Conradi, P. Lachenmaier, K. Linde, M. Meier, P. Pötzl and E. André.
"Xenakis: combining tangible interaction with probability-based musical composition".
In: *Proceedings 2nd International Conference on Tangible Embedded Interaction (TEI '08)*. ACM, 2008, pp. 121–124 (cited on pages 10, 11)

[Bis+08b] M. Bischof, B. Conradi, P. Lachenmaier, K. Linde, M. Meier, P. Pötzl and E. André.
"Xenakis: combining tangible interaction with probability-based musical composition".
In: *Proceedings of the 2nd international conference on Tangible and embedded interaction*. ACM. 2008, pp. 121–124 (cited on page 58)

[Bis09] J. Bispham.
"Music's "design features": Musical motivation, musical pulse, and musical pitch".
In: *Musicae Scientiae* 13.2 suppl (2009), pp. 41–61 (cited on pages 26, 29)

[Bit14] Bitwig GmbH. *Bitwig Studio*. [Online; accessed 28-February-2015]. 2014.
url: https://www.bitwig.com/en/bitwig-studio.html (cited on page 108)

[BJ13] S. Brown and J. Jordania. "Universals in the world's musics".
In: *Psychology of Music* 41.2 (2013), pp. 229–248 (cited on page 1)

[BJT05] J. Bligh, K. Jennings and B. Tangney.
Designing interfaces for collaborative music composition. Tech. rep.
Centre for Research in IT & Education, Trinity College, Dublin, Ireland, 2005
(cited on page 83)

[BK13] T. Becker and N. Klügel. *SoundDesigner*.
Application, includes data sets, Distributed under GPL version 3. 2013.
url: https://github.com/lodsb/SoundDesigner (cited on pages XVI, 107)

[BL07] M. M. Bradley and P. J. Lang. "The International Affective Digitized Sounds Affective Ratings of Sounds and Instruction Manual". In: *Emotion* (2007), pp. 29–46
(cited on pages 138, 142)

[BL99] M. M. Bradley and P. J. Lang. "Affective Norms for English Words (ANEW): Instruction Manual and Affective Ratings". In: *Psychology*. Technical Report C-1 (1999)
(cited on pages 140, 142)

[Bla12] D. Blake. "Timbre as Differentiation in Indie Music".
In: *Music Theory Online* 18.2 (2012), pp. 1–18 (cited on pages 78, 80)

[Ble14] Blend.io. *blend.io*. [Online; accessed 28-March-2015]. 2014. url: https://blend.io
(cited on page 19)

[BMS05] F. Bevilacqua, R. Müller and N. Schnell. "MnM: a Max/MSP mapping toolbox".
In: *Proceedings of the 2005 conference on New interfaces for musical expression*.
National University of Singapore. 2005, pp. 85–88 (cited on page 64)

[BN01] F. Biocca and K. Nowak. "Plugging your body into the telecommunication system: Mediated embodiment, media interfaces, and social virtual environments".
In: *Communication technology and society* (2001), pp. 407–447 (cited on page 32)

[Bød06] S. Bødker. "When second wave HCI meets third wave challenges". In: *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*. ACM. 2006, pp. 1–8 (cited on page 48)

[Bou+00] R. Boulanger et al. *The Csound Book*. MIT press Cambridge, MA, 2000 (cited on pages 72, 86)

[BP00] T. Blaine and T. Perkis. "The Jam-O-Drum interactive music system: a study in interaction design". In: *Proceedings of the 3rd conference on Designing Interactive Systems*. ACM. 2000, pp. 165–173 (cited on pages 124, 125)

[Bri+11] P. Brinkmann, P. Kirn, R. Lawler, C. McCormick, M. Roth and H.-C. Steiner. "Embedding pure data with libpd". In: *Proceedings of the Pure Data Convention*. Vol. 291. 2011 (cited on page 64)

[BSW98] C. M. Bishop, M. Svensén and C. K. Williams. "GTM: The generative topographic mapping". In: *Neural computation* 10.1 (1998), pp. 215–234 (cited on pages 117, 118)

[BTT05] D. Benyon, P. Turner and S. Turner. *Designing interactive systems: People, activities, contexts, technologies*. Pearson Education, 2005 (cited on pages 5, 6, 44, 46)

[Buc92] R. Buchanan. "Wicked problems in design thinking". In: *Design issues* (1992), pp. 5–21 (cited on page 5)

[But06] M. J. Butler. *Unlocking the groove: Rhythm, meter, and musical design in electronic dance music*. Georgetown University Press, 2006 (cited on page 52)

[CA14] C. Canonne and J.-J. Aucouturier. "Play together, think alike: Shared mental models in expert music improvisers". In: *Psychology of Music* (2014) (cited on pages 40, 42)

[Cac+05] A. Caclin, S. McAdams, B. K. Smith and S. Winsberg. "Acoustic correlates of timbre space dimensions: A confirmatory study using synthetic tonesa)". In: *The Journal of the Acoustical Society of America* 118.1 (2005), pp. 471–482 (cited on page 104)

[Can13] C. Canonne. "Focal Points in Collective Free Improvisation". In: *Perspectives of New Music* 51.1 (2013), pp. 40–55 (cited on pages 40, 41)

[Car12] A. T. de Carvalho. "The Discourse of Home Recording: Authority of "Pros" and the Sovereignty of the Big Studios". In: *Journal on the Art of Record Production* 7 (2012) (cited on pages 17, 18)

[Car14] M. Carkci. *Dataflow and Reactive Programming Systems: A Practical Guide*. Matt Carkci, 2014 (cited on page 63)

[CBC11] J. M. Comajuncosas, A. Barrachina and J. O. Connell. "Nuvolet : 3D Gesture-driven Collaborative Audio Mosaicing". In: *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*. Vol. 11. June. 2011, pp. 252–255 (cited on pages 3, 109)

[CC13] E. Czaplicki and S. Chong. "Asynchronous functional reactive programming for GUIs". In: *ACM SIGPLAN Notices*. Vol. 48. 6. ACM. 2013, pp. 411–422 (cited on page 63)

[Cd07] N. Collins and J. d'Escriván. *The Cambridge companion to electronic music*. Cambridge University Press, 2007 (cited on page 86)

[CF06] N. D. Cook and T. X. Fujisawa. "The psychophysics of harmony perception: Harmony is a three-tone phenomenon". In: (2006) (cited on page 161)

[CG12] C. Canonne and N. B. Garnier.
"Cognition and Segmentation In Collective Free Improvisation: An Exploratory Study".
In: *Proceedings of the 12th International Conference on Music Perception and Cognition and the 8th Triennal Conference of the European Society for Cognitive Sciences of Music* (2012) (cited on pages 40, 41)

[CH05] M. A. Carreira-Perpinan and G. E. Hinton. "On contrastive divergence learning". In: *Proceedings of the tenth international workshop on artificial intelligence and statistics*. University College London, City University. 2005, pp. 33–40 (cited on page 143)

[Cha+04] C. Chafe, M. Gurevich, G. Leslie and S. Tyan.
"Effect of time delay on ensemble accuracy".
In: *Proceedings of the International Symposium on Musical Acoustics*. Vol. 31. 2004 (cited on page 46)

[Cha02] J. Chadabe.
"The limitations of mapping as a structural descriptive in electronic instruments".
In: *Proceedings of the International Conference on New interfaces Musical Expression*. National University of Singapore, 2002, pp. 1–5 (cited on page 2)

[Che14] E. Chew. "Music interaction with others as conveyor of relational intent: A response to Cross (2014)". In: *Psychology of Music* 42.6 (2014), pp. 826–838 (cited on pages 6, 35)

[Cho77] J. M. Chowning.
"The synthesis of complex audio spectra by means of frequency modulation".
In: *Computer Music Journal* (1977), pp. 46–54 (cited on pages 95, 96)

[CJ06] T. Coughlan and P. Johnson. "Interaction in creative tasks".
In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM. 2006, pp. 531–540 (cited on page 44)

[CJ95] A. Cockburn and S. Jones. "Four principles of groupware design".
In: *Interacting with Computers* 7.2 (1995), pp. 195–210 (cited on page 38)

[CN14] M. Cirelli and R. Nakamura.
"A Survey on Multi-touch Gesture Recognition and Multi-touch Frameworks".
In: *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces*. ACM. 2014, pp. 35–44 (cited on page 70)

[Coh01] A. J. Cohen. "Music as a source of emotion in film".
In: *Music and emotion: Theory and research* (2001), pp. 249–272 (cited on page 129)

[Col04] N. Collins. "An Algorithm for the Direct Generation of Set Class Representatives in Any Pitch Class Space". In: *Music Theory Online* 10 (2004) (cited on pages 158, 159)

[Col05] D. Collins. "A synthesis process model of creative thinking in music composition".
In: *Psychology of music* 33.2 (2005), pp. 193–216 (cited on page 23)

[Cot04] S. Cottrell. *Professional Music-Making in London : Ethnography and Experience*. Ashgate Publishing, 2004 (cited on pages 78, 81, 83)

[Cro05] I. Cross. "Music and Social Being". In: *Musicology Australia* 28.1 (2005), pp. 114–126 (cited on pages 26, 27)

[Cro08] D. Cronin. "Into the Groove: Lessons from the desktop music revolution".
In: *Interactions Magazine* 15.3 (2008), pp. 72–78 (cited on pages 44, 48)

[CS09] Y. Cho and L. K. Saul. "Kernel methods for deep learning".
In: *Advances in neural information processing systems*. 2009, pp. 342–350 (cited on page 148)

[Csi88] M. Csikszentmihalyi. *Society, culture, and person: A systems view of creativity.* Cambridge University Press, 1988 (cited on page 22)

[Csi90] M. Csikszentmihalyi. *Flow: The Psychology of Optimal Experience*. Harper and Row, 1990 (cited on page 31)

[Csi97] M. Csikszentmihalyi. *Creativity : Flow and the Psychology of Discovery and Invention*. Perennial, 1997 (cited on page 22)

[Cus12] L. A. Custodero. "The call to create: Flow experience in music learning and teaching". In: *Musical Imaginations: Multidisciplinary Perspectives on Creativity, Performance, and Perception* (2012), pp. 369–384 (cited on page 31)

[CW04] C. Cox and D. Warner. *Audio culture: readings in modern music*. Bloomsbury Publishing, 2004 (cited on page 16)

[Cza12] E. Czaplicki. "Elm: Concurrent FRP for Functional GUIs". In: *Master's thesis, Harvard* (2012) (cited on page 62)

[Cza98] K. Czarnecki. "Generative programming: Principles and techniques of software engineering based on automated configuration and fragment-based component models". PhD thesis. 1998 (cited on page 58)

[Dah12] L. Dahl. "Wicked Problems and Design Considerations in Composing for Laptop Orchestra". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by G. Essl, B. Gillespie, M. Gurevich and S. O'Modhrain. University of Michigan, May 2012 (cited on page 5)

[Dah14] L. Dahl. "Designing New Musical Interfaces As Research: What's the Problem?" In: Workshop organised by the Creativity and Cognition Studios, University of Technology, Sydney, 2014 (cited on pages 4–6, 46)

[DB92] P. Dourish and V. Bellotti. "Awareness and coordination in shared workspaces". In: *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*. ACM Press, 1992, pp. 107–114 (cited on pages 36–38)

[Dem10] J. Demers. *Listening through the noise: the aesthetics of experimental electronic music*. Oxford University Press, 2010 (cited on pages 15, 17, 78)

[Den74] J. B. Dennis. "First version of a data flow procedure language". In: *Programming Symposium*. Springer. 1974, pp. 362–376 (cited on page 63)

[Deu12] D. Deutsch. *The Psychology of Music*. Academic Press, 2012 (cited on page 132)

[Dew05] J. Dewey. *Art as experience*. Penguin, 2005 (cited on page 21)

[DHH06] A. C. Disley, D. M. Howard and A. D. Hunt. "Timbral description of musical instruments". In: *International Conference on Music Perception and Cognition*. 2006, pp. 61–68 (cited on page 80)

[Dir07] P. L. Dirks. *An Analysis of Jonathan Harvey's "Mortuos Plango, Vivos Voco"*. [Online; accessed 30-March-2015]. 2007. url: http://cec.sonus.ca/econtact/9_2/dirks.html (cited on page 88)

[DJ97] C. Dodge and T. A. Jerse. *Computer music: synthesis, composition and performance*. Macmillan Library Reference, 1997 (cited on pages 86, 89, 92, 93, 95–100)

[DJA93] N. Dahlbäck, A. Jönsson and L. Ahrenberg. "Wizard of Oz Studies: Why and How". In: *Proceedings of the 1st international conference on Intelligent user interfaces*. ACM. 1993, pp. 193–200 (cited on page 46)

[DK06] C. Dobrian and D. Koppelman. "The 'E' in NIME: musical expression with new computer interfaces". In: *Proceedings of the 2006 conference on New interfaces for musical expression*. IRCAM—Centre Pompidou. 2006, pp. 277–282 (cited on page 61)

[DLR77] A. P. Dempster, N. M. Laird and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1977), pp. 1–38 (cited on page 118)

[DMS07] T. Djajadiningrat, B. Matthews and M. Stienstra.
"Easy doesn't do it: skill and expression in tangible aesthetics".
In: *Personal and Ubiquitous Computing* 11.8 (2007), pp. 657–676 (cited on page 30)

[DNB05] M. Duignan, J. Noble and R. Biddle. "A taxonomy of sequencer user-interfaces".
In: *Proceedings of the International Computer Music Conference*. 2005, pp. 725–728
(cited on page 18)

[DNB10] M. Duignan, J. Noble and R. Biddle.
"Abstraction and activity in computer-mediated music production".
In: *Computer Music Journal* 34.4 (2010), pp. 22–33 (cited on page 25)

[Dou04] P. Dourish. *Where the action is: the foundations of embodied interaction*.
MIT press, 2004 (cited on page 30)

[DP97] G. De Poli and P. Prandoni. "Sonological models for timbre characterization*".
In: *J. New Music Res.* 26.2 (June 1997), pp. 170–197 (cited on page 77)

[Dr 12] Dr. Gail Fischler and students. *The Musical Adjectives Project*.
[Online; accessed 28-March-2015]. 2012.
url: https://themusicaladjectivesproject.wikispaces.com/ (cited on page 140)

[Dru09] J. Drummond. "Understanding interactive systems".
In: *Organised Sound* 14.02 (2009), pp. 124–133 (cited on pages 2, 3)

[Eag+08] B. Eaglestone, N. Ford, P. Holdridge, J. Carter and C. Upton.
"Cognitive styles and computer-based creativity support systems: two linked studies of electro-acoustic music composers".
In: *Computer Music Modeling and Retrieval. Sense of Sounds*. Springer, 2008, pp. 74–97
(cited on page 28)

[EDL97] EDLMax. *Open Media Framework*. [Online; accessed 28-March-2015]. 1997.
url: http://www.edlmax.com/FormatOmf.htm (cited on page 19)

[Eer11] T. Eerola. "Are the emotions expressed in music genre-specific? An audio-based evaluation of datasets spanning classical, film, pop and mixed genres".
In: *Journal of New Music Research* 40.4 (2011), pp. 349–366 (cited on page 105)

[EFA12] T. Eerola, R. Ferrer and V. Alluri. "Timbre and affect dimensions: Evidence from affect and similarity ratings and acoustic correlates of isolated instrument sounds".
In: *Music Perception: An Interdisciplinary Journal* 30.1 (2012), pp. 49–70
(cited on pages 102, 131, 133, 135, 137)

[EH97] C. Elliott and P. Hudak. "Functional reactive animation". In: *ACM SIGPLAN Notices*.
Vol. 32. 8. ACM. 1997, pp. 263–273 (cited on page 62)

[EK08] F. Echtler and G. Klinker. "A multitouch software architecture". In: *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges*. ACM. 2008,
pp. 463–466 (cited on page 70)

[Ell09] C. M. Elliott. "Push-pull functional reactive programming".
In: *Proceedings of the 2nd ACM SIGPLAN symposium on Haskell*. ACM. 2009, pp. 25–36
(cited on pages 62, 63)

[EM13] H. Egermann and S. McAdams. "Empathy and Emotional Contagion as a Link Between Recognized and Felt Emotions in Music Listening".
In: *Music Perception: An Interdisciplinary Journal* 31.2 (2013), pp. 139–156
(cited on page 130)

[Emm07] S. Emmerson. *Living electronic music*. Ashgate Publishing, Ltd., 2007
(cited on page 17)

[Emm86] S. Emmerson. *The Language of Electroacoustic Music*. Macmillan, 1986
(cited on page 15)

[Eno04] B. Eno. "The studio as compositional tool".
In: *Audio culture: Readings in modern music* (2004), pp. 127–130 (cited on page 17)

[EP10] A. Eigenfeldt and P. Pasquier.
"Real-time timbral organisation: Selecting samples based upon similarity".
In: *Organised Sound* 15.02 (2010), pp. 159–166 (cited on page 110)

[Erh+10] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent and S. Bengio.
"Why does unsupervised pre-training help deep learning?"
In: *The Journal of Machine Learning Research* 11 (2010), pp. 625–660 (cited on page 149)

[ESH02] H. Eden, E. Scharff and E. Hornecker. "Multilevel design and role play: experiences in assessing support for neighborhood participation in design".
In: *DIS '02: Proceedings of the 4th conference on Designing interactive systems*.
ACM, 2002, pp. 387–392 (cited on page 38)

[EV13] T. Eerola and J. K. Vuoskoski.
"A review of music and emotion studies: approaches, emotion models, and stimuli".
In: *Music Perception: An Interdisciplinary Journal* 30.3 (2013), pp. 307–340
(cited on pages 129, 130)

[F+07] B. Fry, C. Reas et al. *Processing*. 2007. url: http://processing.org (cited on page 71)

[Fal03] D. Fallman. "Design-oriented human-computer interaction".
In: *Proceedings of the SIGCHI conference on Human factors in computing systems*.
ACM. 2003, pp. 225–232 (cited on page 6)

[FB10] R. Fencott and N. Bryan-Kinns. "Hey man, you're invading my personal space! Privacy and awareness in collaborative music". In: *Proceedings of the 10th International Conference on New Interfaces for Musical Expression (NIME 2010)*. 2010, pp. 198–203
(cited on pages 39, 41, 42, 44)

[FB12] R. Fencott and N. Bryan-Kinns.
"Audio delivery and territoriality in collaborative digital musical interaction".
In: *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers*. British Computer Society. 2012, pp. 69–78 (cited on page 39)

[FB13] R. Fencott and N. Bryan-Kinns.
"Computer musicking: HCI, CSCW and collaborative digital musical interaction".
In: *Music and Human-Computer Interaction*. Springer, 2013, pp. 189–205
(cited on pages 5, 45, 46, 55, 120, 124)

[FBS06] A. Friberg, R. Bresin and J. Sundberg.
"Overview of the KTH rule system for musical performance".
In: *Advances in Cognitive Psychology* 2.2-3 (2006), pp. 145–161 (cited on page 157)

[FC05] P. Ferreira-Lopes and D. Coimbra. "Music and Interaction: Consequences, Mutations and Metaphors of the Digital Music Instrument".
In: *Artech 2005, 2nd Workschop Workshop Luso-Galaico de Artes Digitais* (2005)
(cited on pages 2, 32)

[FI10] A. Fischer and C. Igel. "Empirical analysis of the divergence of Gibbs sampling based learning algorithms for restricted Boltzmann machines".
In: *Artificial Neural Networks–ICANN 2010*. Springer, 2010, pp. 208–217
(cited on page 146)

[Fis05a] G. Fischer. "Creativity and distributed intelligence". In: (2005), pp. 71–73
(cited on page 24)

[Fis05b] G. Fischer. "Distances and diversity: sources for social creativity".
In: *Proceedings of the 5th conference on Creativity & cognition*. ACM. 2005, pp. 128–136
(cited on pages 22, 24)

[FKG10] M. R. Frieß, N. Klügel and G. Groh. "Lzrdm: Collaborative Multi-touch Sequencer".
In: *ACM International Conference on Interactive Tabletops and Surfaces*. ITS '10.
(Demonstration). ACM, 2010, pp. 303–303 (cited on page XV)

[Flu13] Flux:: Sound and Picture development. *Ircam Studio Pack*. [Online; accessed 28-March-2015]. 2013. url: http://www.fluxhome.com/products/packs/ircamstudio_pack (cited on page 18)

[FM02] S. Fels and J. Manzolli. "Interactive, evolutionary textured sound composition". In: *Multimedia 2001*. Springer, 2002, pp. 153–164 (cited on page 189)

[Fon+12] F. Font, G. Roma, P. Herrera and X. Serra. "Characterization of the Freesound online community". In: *Cognitive Information Processing (CIP), 2012 3rd International Workshop on*. IEEE. 2012, pp. 1–6 (cited on page 139)

[Fra12] A. Fraser. "The spaces, politics, and cultural economies of electronic dance music". In: *Geography Compass* 6.8 (2012), pp. 500–511 (cited on page 52)

[Fri+11] M. R. Frieß, G. Groh, N. Klügel and M. Kleinhans. "A tabletop application environment for generic creativity techniques". In: *International Journal of Computer Information Systems and Industrial Management Applications* 3.4 (2011) (cited on pages XV, 36–38)

[Fri+12] M. R. Frieß, G. Groh, N. Klügel and M. Kleinhans. "An Analysis Framework for Quantitatively Characterizing Collaboration in Tabletop Based CSCW Environments". In: *International Journal of Computer Information Systems and Industrial Management Applications* 4.2 (2012) (cited on pages XVI, 36, 37, 45)

[Fri13] M. R. Frieß. "Situative Creativity". PhD thesis. Technische Universität München, 2013 (cited on pages 7, 35–38)

[Fri99] H. Fripertinger. "Enumeration and construction in music theory". In: *Proceedings of the Diderot Forum on Mathematics and Music (Vienna)*. 1999, pp. 170–203 (cited on page 158)

[FS02] A. Friberg and A. Sundström. "Swing ratios and ensemble timing in jazz performance: Evidence for a common rhythmic pattern". In: *Music Perception* 19.3 (2002), pp. 333–349 (cited on page 191)

[Fut13] Futucraft. *Cotracks*. [Online; accessed 01-April-2015]. 2013. url: http://futucraft.com/cotracks/ (cited on page 11)

[Gab02] A. Gabrielsson. "Emotion perceived and emotion felt: Same or different?" In: *Musicae Scientiae* 5.1 suppl (2002), pp. 123–147 (cited on page 129)

[Gar02] A. Gartland-Jones. "Can a Genetic Algorithm Think Like a Composer?" In: *Proceedings of the International Conerence on Generative Art*. International Conference on Generative Art. 2002 (cited on page 189)

[Gat91] J. Gates. "Music participation: Theory, research, and policy". In: *Bulletin of the council for research in music education* (1991), pp. 1–35 (cited on pages 27, 28)

[GB10] X. Glorot and Y. Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *International Conference on Artificial Intelligence and Statistics*. 2010, pp. 249–256 (cited on page 147)

[GD07] P. Gomez and B. Danuser. "Relationships between musical structure and psychophysiological measures of emotion." In: *Emotion* 7.2 (2007), p. 377 (cited on page 130)

[Gho10] D. Ghosh. *DSLs in action*. Manning Publications Co., 2010 (cited on page 67)

[GKO11] N. Gillian, R. Knapp and S. O'Modhrain. "A machine learning toolbox for musician computer interaction". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by A. R. Jensenius, A. Tveit, R. I. Godoy and D. Overholt. 2011, pp. 60–66 (cited on page 65)

[GL01] A. Gabrielsson and E. Lindström.
"The influence of musical structure on emotional expression." In: *Music And Emotion*. Oxford University Press, 2001, pp. 223–248 (cited on page 130)

[Góm+05] F. Gómez, A. Melvin, D. Rappaport, G. T. Toussaint et al.
"Mathematical measures of syncopation".
In: *Renaissance Banff: Mathematics, Music, Art, Culture*.
Canadian Mathematical Society. 2005, pp. 73–84 (cited on page 179)

[Goy+04] K. N. Goydke, E. Altenmüller, J. Möller and T. F. Münte. "Changes in emotional tone and instrumental timbre are reflected by the mismatch negativity."
In: *Cognitive Brain Research* 21.3 (2004), pp. 351–359 (cited on page 133)

[GP05] P. D. Greene and T. Porcello.
Wired for sound: engineering and technologies in sonic cultures.
Wesleyan University Press, 2005 (cited on pages 18, 82, 129)

[Gre98] S. Gresham-Lancaster. "The aesthetics and history of the hub: The effects of changing technology on network computer music". In: *Leonardo Music Journal* (1998), pp. 39–44 (cited on page 64)

[GS09] S. Gelineck and S. Serafin. "From idea to realization-understanding the compositional processes of electronic musicians". In: *Audio Mostly* (2009) (cited on pages 22, 49)

[Gup12] M. Gupta. *Akka essentials*. Packt Publishing Ltd, 2012 (cited on page 66)

[Hai+09] J. C. Hailstone, R. Omar, S. M. D. Henley, C. Frost, M. G. Kenward and J. D. Warren. "It's not what you play, it's how you play it: timbre affects perception of emotion in music." In: *The Quarterly Journal of Experimental Psychology* 62.11 (2009), pp. 2141–55 (cited on pages 133, 136)

[Hal63] E. T. Hall. "A system for the notation of proxemic behavior1".
In: *American anthropologist* 65.5 (1963), pp. 1003–1026 (cited on page 36)

[Han+09] T. E. Hansen, J. P. Hourcade, M. Virbel, S. Patali and T. Serra.
"PyMT: a post-WIMP multi-touch user interface toolkit". In: *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. ACM. 2009, pp. 17–24 (cited on page 70)

[Har07] Harmonix Music Systems Inc. *RockBand*. [Online; accessed 28-March-2015]. 2007.
url: http://www.harmonixmusic.com/games/rock-band/ (cited on page 50)

[Har14] G. Harkness. "Get on the Mic: Recording Studios as Symbolic Spaces in Rap Music".
In: *Journal of Popular Music Studies* 26.1 (2014), pp. 82–100 (cited on page 17)

[Has13] J. Hass. *Introduction to Computer Music: Volume One*.
[Online; accessed 28-March-2015]. 2013.
url: http://www.indiana.edu/~emusic/etext/synthesis (cited on pages 97, 98)

[HB10] D. Hesmondhalgh and S. Baker. "'A very complicated version of freedom': Conditions and experiences of creative labour in three cultural industries".
In: *Poetics* 38.1 (2010), pp. 4–20 (cited on page 18)

[HC06] M. Hoffman and P. R. Cook. "Feature-based synthesis: Mapping acoustic and perceptual features onto synthesis parameters".
In: *Proceedings of the 2006 International Computer Music Conference*. Vol. 33. 2006 (cited on page 110)

[HC99] J. Harrington and S. Cassidy. *Techniques in speech acoustics*. Vol. 8. Springer, 1999 (cited on page 89)

[Hev+04] A. R. Hevner, S. T. March, J. Park and S. Ram.
"Design Science in Information Systems Research".
In: *MIS Q.* 28.1 (Mar. 2004), pp. 75–105.
url: http://dl.acm.org/citation.cfm?id=2017212.2017217 (cited on pages 5, 7)

[Hev36] K. Hevner. “Experimental studies of the elements of expression in music.” In: *The American Journal of Psychology* (1936) (cited on page 130)

[Hew05] T. T. Hewett. “Informing the design of computer-based environments to support creativity”. In: *International Journal of Human-Computer Studies* 63.4 (2005), pp. 383–409 (cited on pages 31, 39, 48)

[HG01] M. Hoegl and H. G. Gemuenden. “Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence”. In: *Organization science* 12.4 (2001), pp. 435–449 (cited on pages 36, 37)

[Hin02] G. E. Hinton. “Training products of experts by minimizing contrastive divergence”. In: *Neural computation* 14.8 (2002), pp. 1771–1800 (cited on page 146)

[Hin07] G. E. Hinton. “Learning multiple layers of representation”. In: *Trends in cognitive sciences* 11.10 (2007), pp. 428–434 (cited on pages 147, 148)

[HK13a] G. Hagerer and N. Klügel. *FugueGenerator*. Application, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/FugueGenerator (cited on pages XVI, 155)

[HK13b] G. Hagerer and N. Klügel. *TreeQuencer*. Application, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/TreeQuencer (cited on pages XVI, 171)

[HLB05] P. G. Healey, J. Leach and N. Bryan-Kinns. “Inter-play: Understanding group music improvisation as a form of everyday interaction”. In: *Proceedings of Less is More—Simple Computing in an Age of Complexity. Microsoft Research Cambridge* (2005) (cited on page 41)

[Hol+13] S. Holland, K. Wilkie, P. Mulholland and A. Seago. “Music Interaction: Understanding Music and Human-Computer Interaction”. In: *Music and Human-Computer Interaction*. Springer, 2013, pp. 1–23 (cited on page 42)

[Hol02] T. B. Holmes. *Electronic and experimental music: pioneers in technology and composition*. Psychology Press, 2002 (cited on page 100)

[Hol11] P. a. Holmes. “An exploration of musical communication through expressive use of timbre: The performer’s perspective”. In: *Psychology of Music* (Mar. 2011) (cited on pages 4, 77, 78, 81, 83, 133)

[Hol12] T. Holmes. *Electronic and experimental music: technology, music, and culture*. Routledge, 2012 (cited on page 16)

[Hom09] M. Homer. “Beyond the Studio: The Impact of Home Recording Technologies on Music Creation and Consumption.” In: *Nebula* 6.3 (2009), pp. 85–99 (cited on page 17)

[Hor05] E. Hornecker. “A design theme for tangible interaction: embodied facilitation”. In: *ECSCW’05 Proc. ninth Conf. Eur. Conf. Comput. Support. Coop. Work*. Springer-Verlag New York, Inc., 2005, pp. 23–43 (cited on pages 11, 29, 38, 50)

[HOT06] G. E. Hinton, S. Osindero and Y.-W. Teh. “A fast learning algorithm for deep belief nets.” In: *Neural Comput.* 18.7 (2006), pp. 1527–1554 (cited on pages 136, 143, 144, 148, 150)

[Hud+03] P. Hudak, A. Courtney, H. Nilsson and J. Peterson. “Arrows, robots, and functional reactive programming”. In: *Advanced Functional Programming*. Springer, 2003, pp. 159–187 (cited on page 62)

[Hud08] E. Hudlicka. “Affective computing for game design”. In: *Proceedings of the 4th Intl. North American Conference on Intelligent Games and Simulation (GAMEON-NA), Montreal, Canada*. 2008 (cited on page 133)

[Hur01] D. Huron. “Tone and Voice: A Derivation of the Rules of Voice-Leading from Perceptual Principles”. In: *Music Perception* 19.1 (Sept. 2001), pp. 1–64 (cited on pages 77, 78, 81)

[HW12] I. Hattwick and M. Wanderley.
"A Dimension Space for Evaluating Collaborative Musical Performance Systems". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by G. Essl, B. Gillespie, M. Gurevich and S. O'Modhrain.
University of Michigan, May 2012 (cited on page 4)

[ICK06] D. Ignatoff, G. H. Cooper and S. Krishnamurthi. "Crossing state lines: Adapting object-oriented frameworks to functional reactive languages".
In: *Functional and Logic Programming*. Springer, 2006, pp. 259–276 (cited on page 62)

[ICL] ICLC, International Conference on Live Coding.
International Conference on Live Coding Homepage.
[Online; accessed 28-February-2015]. url: http://iclc.livecodenetwork.org/
(cited on page 64)

[Ima97] Image-Line Software. *FL Studio*. [Online; accessed 28-March-2015]. 1997.
url: http://www.image-line.com/flstudio/ (cited on page 173)

[Ing01] T. Ingold. *Beyond Art and Technology: The Anthropology of Skill*.
University of New Mexico Press, Apr. 2001, pp. 17–31 (cited on page 30)

[Iwa99] T. Iwai. "Composition on the table". In: *ACM SIGGRAPH 99 Electron. art Animat. Cat.* SIGGRAPH '99. ACM, 1999, pp. 10– (cited on pages 11, 12)

[Jac+10] G. Jacucci, A. Morrison, G. Richard, J. Kleimola, P. Peltonen, T. Laitinen and S. Università.
"Worlds of Information : Designing for Engagement at a Public Multi-touch Display".
In: *New York* (2010) (cited on page 3)

[Jac95] B. Jacob. "Composing with genetic algorithms".
In: *Proceedings of the International Computer Music Conference*.
International Computer Music Association, 1995 (cited on page 187)

[Jär+08] J. Järvi, M. Marcus, S. Parent, J. Freeman and J. N. Smith.
"Property models: from incidental algorithms to reusable components".
In: *Proceedings of the 7th international conference on Generative programming and component engineering*. ACM. 2008, pp. 89–98 (cited on page 60)

[JGJ11] C. F. Julià, D. Gallardo and S. Jordà. "MTCF : A Framework for Designing and Coding Musical Tabletop Applications Directly in Pure Data". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*.
Ed. by A. R. Jensenius, A. Tveit, R. I. Godoy and D. Overholt. 2011, pp. 457–460
(cited on page 64)

[JGW05] A. R. Jensenius, R. I. Godøy and M. M. Wanderley. "Developing tools for studying musical gestures within the Max/MSP/Jitter environment".
In: *Proceedings of the International Conference Music Computing (ICMC)*. 2005
(cited on page 65)

[Joh+91] R. Johansen, D. Sibbet, S. Benson, A. Martin, R. Mittman and P. Saffo.
Leading business teams: How teams can use technology and group process tools to enhance performance. Addison-Wesley Longman Publishing Co., Inc., 1991
(cited on page 36)

[Joh11] A. Johnston.
"Beyond Evaluation : Linking Practice and Theory in New Musical Interface Design". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. 2011, pp. 280–283 (cited on page 46)

[Jor+06] S. Jordà, M. Kaltenbrunner, G. Geiger and M. Alonso.
"The reacTable: a tangible tabletop musical instrument and collaborative workbench".
In: *ACM SIGGRAPH 2006 Sketches*. ACM, 2006, p. 91 (cited on pages 2, 9, 58, 172)

[Jor03] S. Jordà. “Interactive music systems for everyone: exploring visual feedback as a way for creating more intuitive, efficient and learnable instruments”. In: *Proceedings of the Stockholm Music Acoustics Conference (SMAC03), Stockholm, Sweden*. Citeseer. 2003 (cited on page 44)

[Jor05a] S. Jord‘a. “Multi-user Instruments: Models, Examples and Promises”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. 2005, pp. 23–26 (cited on page 1)

[Jor05b] S. Jordà. “Digital Lutherie: Crafting musical computers for new musics performance and improvisation”. PhD thesis. 2005 (cited on pages 5, 25)

[Jus13a] P. N. Juslin. “From everyday emotions to aesthetic emotions: towards a unified theory of musical emotions”. In: *Physics of life reviews* 10.3 (2013), pp. 235–266 (cited on page 130)

[Jus13b] P. N. Juslin. “What does music express? Basic emotions and beyond”. In: *Frontiers in psychology* 4 (2013) (cited on pages 130, 132)

[JV08] P. N. Juslin and D. Västfjäll. “Emotional responses to music: the need to consider underlying mechanisms.” In: *Behav. Brain Sci.* 31.5 (2008), pp. 559–575, 559–575 (cited on pages 130, 131)

[Kal+05] M. Kaltenbrunner, T. Bovermann, R. Bencina and E. Costanza. “TUIO: A protocol for table-top tangible user interfaces”. In: *Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation*. 2005, pp. 1–5 (cited on page 60)

[Kan+98] K. C. Kang, S. Kim, J. Lee, K. Kim, G. J. Kim and E. Shin. “FORM: A feature-oriented reuse method with domain-specific reference architectures”. In: *Annals of Software Engineering* 5 (1998), pp. 143–168 (cited on page 58)

[Kar00] Karagioza Blog For Audio Recording, Production and Mastering. *Picture of reverbation phases*. [Online; accessed 25-March-2015]. 2000. url: http://karagioza.com/?p=676 (cited on page 93)

[Kaw+13] A. Kawakami, K. Furukawa, K. Katahira and K. Okanoya. “Sad Music Induces Pleasant Emotion”. In: *Frontiers in Psychology* 4.311 (2013) (cited on page 130)

[KBG14] N. Klügel, T. Becker and G. Groh. “Designing Sound Collaboratively Perceptually Motivated Audio Synthesis”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by B. Caramiaux, K. Tahiroglu, R. Fiebrink and A. Tanaka. Goldsmiths, University of London, June 2014, pp. 327–330 (cited on pages XV, 107)

[KBH12] N. R. Krishnaswami, N. Benton and J. Hoffmann. “Higher-order functional reactive programming in bounded space”. In: *ACM SIGPLAN Notices* 47.1 (2012), pp. 45–58 (cited on page 63)

[Ker13] S. Kerrigan. “Accommodating creative documentary practice within a revised systems model of creativity”. In: *Journal of Media Practice* 14.2 (2013), pp. 111–127 (cited on page 24)

[KG13] N. Klügel and G. Groh. “Towards Mapping Timbre to Emotional Affect”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by W. Yeo, K. Lee, A. Sigman, J. H. and G. Wakefield. Graduate School of Culture Technology, KAIST, May 2013, pp. 525–530 (cited on pages XV, 135)

[KHG14a] N. Klügel, G. Hagerer and G. Groh. "FugueGenerator - Collaborative Melody Composition Based on a Generative Approach for Conveying Emotion in Music". In: *Proceedings of the Joint International Conference Music Computing (ICMC) and Sound and Music Computing (SMC)*. Ed. by A. Andreopoulou, G. Kosteletos, N. Georgiou, A. Kolkou and A. Charoumenidou. National and Kapodistrian University of Athens, Sept. 2014, pp. 1043–1050 (cited on pages XV, 155)

[KHG14b] N. Klügel, G. Hagerer and G. Groh. "TreeQuencer: Collaborative Rhythm Sequencing A Comparative Study". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by B. Caramiaux, K. Tahiroglu, R. Fiebrink and A. Tanaka. Goldsmiths, University of London, June 2014, pp. 50–53 (cited on pages XV, 171)

[KHT06] S. R. Klemmer, B. Hartmann and L. Takayama. "How Bodies Matter: Five Themes for Interaction Design". In: *Proceedings of the 6th Conference on Designing Interactive Systems*. DIS '06. ACM, 2006, pp. 140–149 (cited on page 31)

[Kin+12] K. Kin, B. Hartmann, T. DeRose and M. Agrawala. "Proton++: a customizable declarative multitouch framework". In: *Proceedings of the 25th annual ACM symposium on User interface software and technology*. ACM. 2012, pp. 477–486 (cited on page 70)

[Kir+12] N. Kireeva, I. Baskin, H. Gaspar, D. Horvath, G. Marcou and A. Varnek. "Generative Topographic Mapping (GTM): Universal Tool for Data Visualization, Structure-Activity Modeling and Dataset Comparison". In: *Molecular Informatics* 31.3-4 (2012), pp. 301–312 (cited on page 117)

[KLB06] T. Karrer, E. Lee and J. Borchers. "PhaVoRIT: A phase vocoder for real-time interactive time-stretching". In: *Proceedings of the ICMC 2006 International Computer Music Conference*. 2006, pp. 708–715 (cited on page 205)

[Kle+11] J. Kleimola, V. Lazzarini, J. Timoney and V. Valimaki. "Vector Phase Shaping Synthesis". In: *Proceedings of the 14th International Conference on Digital Audio Effects*. Sept. 2011 (cited on pages 100–103)

[Kle+13] J. Kleimola et al. "Nonlinear abstract sound synthesis algorithms". PhD thesis. 2013 (cited on pages 86, 94, 100, 101)

[KLG14] N. Klügel, A. Lindström and G. Groh. "A Genetic Algorithm Approach to Collaborative Music Creation on a Multi-touch Table". In: *Proceedings of the Joint International Conference Music Computing (ICMC) and Sound and Music Computing (SMC)*. Ed. by A. Andreopoulou, G. Kosteletos, N. Georgiou, A. Kolkou and A. Charoumenidou. National and Kapodistrian University of Athens, Sept. 2014, pp. 286–293 (cited on pages XV, 185)

[Klü+11] N. Klügel, M. R. Friess, G. Groh and F. Echtler. "An Approach to Collaborative Music Composition". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by A. R. Jensenius, A. Tveit, R. I. Godoy and D. Overholt. 2011, pp. 32–35 (cited on pages XV, 8, 10, 39, 49, 52, 54, 58, 199)

[Klü09] N. Klügel. "Development of an Environment for Collaborative Composition and Production of Music". Diploma Thesis. Technische Universität München, 2009 (cited on pages 2, 8)

[Klü12] N. Klügel. "Practical Empirical Mode Decomposition for Audio Synthesis". In: *Proceedings of the International Conference of Digital Audio Effects*. University of York, Sept. 2012, pp. 15–18 (cited on pages XVI, 88, 205)

[Klü13a] N. Klügel. *Genetic Algorithm Toolbox for Scala*.
Framework / Library, Distributed under GPL version 3. 2013.
url: https://github.com/lodsb/mutant5000 (cited on pages XVI, 185)

[Klü13b] N. Klügel. *Library for scales and tuning*.
Framework / Library, Distributed under GPL version 3. 2013.
url: https://github.com/lodsb/scales (cited on page XVI)

[Klü14] N. Klügel. *Reakt - Reactive Event System*.
Framework / Library, Distributed under GPL version 3. 2010-2014.
url: https://github.com/lodsb/reakt (cited on pages XVI, 57)

[KLV61] J. L. Kelly, C. Lochbaum and V. A. Vyssotsky. "A block diagram compiler".
In: *Bell System Technical Journal, The* 40.3 (1961), pp. 669–678 (cited on page 63)

[KRL14] N. Klügel, C. Ruff and U. Laufs. *UltraCom - UI-Framework*.
Framework / Library, Distributed under GPL version 3. 2010-2014.
url: https://github.com/lodsb/UltraCom (cited on pages XVI, 57)

[KRR09] W. A. König, R. Rädle and H. Reiterer.
Squidy: A Zoomable Design Environment for Natural User Interfaces. CHI EA '09.
ACM, 2009, pp. 4561–4566 (cited on page 70)

[Kru+03] R. Kruger, S. Carpendale, S. D. Scott and S. Greenberg. "How people use orientation on tables: comprehension, coordination and communication". In: *Proceedings of the 2003 international ACM SIGGROUP conference on Supporting group work*. ACM. 2003, pp. 369–378 (cited on page 37)

[Kru+05] R. Kruger, S. Carpendale, S. D. Scott and A. Tang.
"Fluid integration of rotation and translation". In: *Proceedings of the International SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2005, pp. 601–610 (cited on page 11)

[KSM10] Y. Kim, E. Schmidt and R. Migneco.
"Music emotion recognition: A state of the art review".
In: *Proceedings of the International Society for Music Information Retrieval Conference*. 2010, pp. 255–266 (cited on page 143)

[KT10] S. Kirschner and M. Tomasello.
"Joint music making promotes prosocial behavior in 4-year-old children".
In: *Evolution and Human Behavior* 31.5 (2010), pp. 354–364 (cited on page 37)

[Lad99] V. Ladma. *G-Systems - An algebraic theory of musical structure*.
[Online; accessed 28-March-2015]. 1999.
url: http://vladimir_ladma.sweb.cz/english/music/articles/dide99.htm
(cited on page 158)

[Lai+06] Y. Lai, S.-K. Jeng, D.-T. Liu and Y.-C. Liu. "Automated optimization of parameters for FM sound synthesis with genetic algorithms".
In: *International Workshop on Computer Music and Audio Technology*. 2006
(cited on page 205)

[Lan+10] R. Laney, C. Dobbyn, A. Xambó, M. Schirosa, D. Miell, K. Littleton and N. Dalton.
"Issues and techniques for collaborative music making on multi-touch surfaces".
In: *7th Sound and Music Computing Conference*. July 2010 (cited on pages 39–41)

[Lar+07] H. Larochelle, D. Erhan, A. Courville, J. Bergstra and Y. Bengio. "An empirical evaluation of deep architectures on problems with many factors of variation". In: *Proc. 24th Int. Conf. Mach. Learn.* ICML '07 2006 (2007). Ed. by Z. Ghahramani, pp. 473–480 (cited on pages 143, 149)

[Lar+09] H. Larochelle, Y. Bengio, J. Louradour and P. Lamblin.
"Exploring strategies for training deep neural networks".
In: *The Journal of Machine Learning Research* 10 (2009), pp. 1–40 (cited on page 149)

[Lar13] H. Larochelle. *Neural Networks class (course material)*. [Online; accessed 25-March-2015]. 2013. url: http://info.usherbrooke.ca/hlarochelle/cours/ift725_A2013/contenu.html (cited on page 149)

[LB08] N. Le Roux and Y. Bengio. "Representational power of restricted Boltzmann machines and deep belief networks". In: *Neural Computation* 20.6 (2008), pp. 1631–1649 (cited on page 148)

[LBC97] P. J. Lang, M. M. Bradley and B. N. Cuthbert. *International Affective Picture System (IAPS): Technical Manual and Affective Ratings*. Vol. 77. 4. The Center for Research in Psychophysiology, University of Florida, 1997, pp. 1–5 (cited on pages 141, 142)

[Le+11] Q. V. Le, W. Y. Zou, S. Y. Yeung and A. Y. Ng. "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis". In: *Proceedings of the Conference on Computer Vision and Pattern Recognition*. IEEE. 2011, pp. 3361–3368 (cited on page 148)

[Lem+07] M. Leman, F. Avanzini, A. De Cheveigne and E. Bigand. "The Societal Contexts for Sound and Music Computing: Research, Education, Industry, and Socio-Culture". In: *Journal of New Music Research* 36.3 (2007), pp. 149–167 (cited on pages 29, 30, 82)

[Lem+09] M. Leman, M. Demey, M. Lesaffre, L. van Noorden and D. Moelants. "Concepts, Technology, and Assessment of the Social Music Game "Sync-in-Team"". In: *Proceedings of the International Conference on Computational Science and Engineering*. Ieee, 2009, pp. 837–842 (cited on page 31)

[Lem10] M. Leman. "An embodied approach to music semantics". In: *Musicae Scientiae* 5 (2010), pp. 43–67 (cited on pages 29, 31, 82)

[Ler12] A. Lerch. *An introduction to audio content analysis: Applications in signal processing and music informatics*. John Wiley & Sons, 2012 (cited on page 102)

[Leu+12] A. K.-y. Leung, S. Kim, E. Polman, L. S. Ong, L. Qiu, J. A. Goncalo and J. Sanchez-Burks. "Embodied metaphors and creative "acts"". In: *Psychological science* 23.5 (2012), pp. 502–509 (cited on page 31)

[Lev06] G. Levin. "The table is the score: An augmented-reality interface for real-time, tangible, spectrographic performance". In: (2006) (cited on page 11)

[Lin11] A. Linson. "Unnecessary constraints: A challenge to some assumptions of digital musical instrument design". In: *Proceedings of the International Conference Music Computing (ICMC)*. 2011 (cited on page 43)

[Lin15] J. Lindblom. "Meaning-Making as a Socially Distributed and Embodied Practice". In: *Aesthetics and the Embodied Mind: Beyond Art Theory and the Cartesian Mind-Body Dichotomy*. Springer, 2015, pp. 3–19 (cited on page 30)

[Lit+10] G. Little, L. B. Chilton, M. Goldman and R. C. Miller. "Exploring iterative and parallel human computation processes". In: *Proceedings of the ACM SIGKDD workshop on human computation*. ACM. 2010, pp. 68–76 (cited on page 39)

[Liv+10] S. R. Livingstone, R. Muhlberger, A. R. Brown and W. F. Thompson. "Changing musical emotion: A computational rule system for modifying score and performance". In: *Computer Music Journal* 34.1 (2010), pp. 41–64 (cited on pages 130, 157, 161, 162, 164, 165)

[LK13] A. Lindström and N. Klügel. *SoundDesigner*. Application, Distributed under GPL version 3. 2013. url: https://github.com/lodsb/Mutator (cited on pages XVI, 185)

[LRZ10] U. Laufs, C. Ruff and J. Zibuschka.
"Mt4j-a cross-platform multi-touch development framework".
In: *arXiv preprint arXiv:1012.0467* (2010) (cited on pages 59, 70, 71)

[LS67] D. F. Lott and R. Sommer. "Seating arrangements and status."
In: *Journal of Personality and Social Psychology* 7.1p1 (1967), p. 90 (cited on page 36)

[LTE08] O. Lartillot, P. Toiviainen and T. Eerola.
"A Matlab Toolbox for Music Information Retrieval".
In: *Data analysis, machine learning and applications*.
Ed. by C. Preisach, H. Burkhardt, L. Schmidt-Thieme and R. Decker. Vol. 35.
Studies in Classification, Data Analysis, and Knowledge Organization M. Springer, 2008,
pp. 261–268 (cited on pages 102, 116)

[LV08] S. Le Groux and P. F. Verschure.
"Perceptsynth: mapping perceptual musical features to sound synthesis parameters".
In: *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*. 1. Ieee, Mar. 2008, pp. 125–128 (cited on pages 109, 110)

[LV10] S. Le Groux and P. F. M. J. Verschure. "Emotional Responses to the Perceptual Dimensions of Timbre: A Pilot Study Using Physically Informed Sound Synthesis."
In: *Proceedings of the 7th International Symposium on Computer Music Modeling and Retrieval, CMMR*. 2010, pp. 1–15 (cited on page 138)

[Ma+15] J. Ma, L. Sindorf, I. Liao and J. Frazier. "Using a Tangible Versus a Multi-touch Graphical User Interface to Support Data Exploration at a Museum Exhibit".
In: *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM. 2015, pp. 33–40 (cited on page 38)

[Mac06] R. MacDonald. "Creativity and flow in musical composition: an empirical investigation".
In: *Psychology of Music* 34.3 (July 2006), pp. 292–306 (cited on page 32)

[Mac07] T. MacFarlane. *The Beatles' Abbey Road medley: extended forms in popular music*.
Scarecrow Press, 2007 (cited on pages 16, 17)

[Mac08] R. Maconie. "Sounds of Heavy Metal: Revisting Mikrophonie I".
In: *Proceedings of the International Conference of Music Educucation*. May. 2008
(cited on page 1)

[Mag+11] W. L. Magee, M. Bertolami, L. Kubicek, M. LaJoie, L. Martino, A. Sankowski, J. Townsend, A. M. Whitehead-Pleaux and J. B. Zigo.
"Using music technology in music therapy with populations across the life span in medical and educational programs". In: *Music and Medicine* 3.3 (2011), pp. 146–153
(cited on page 7)

[Mag06] T. Magnusson. "Screen-based musical interfaces as semiotic machines". In: *Proceedings of the international Conference on New interfaces for musical expression*.
IRCAM—Centre Pompidou. 2006, pp. 162–167 (cited on pages 5, 25, 31, 39, 47, 59)

[Mag10] T. Magnusson. "An epistemic dimension space for musical devices".
In: *Proceedings for New Interfaces for Musical Expression Conference*.
University of Technology, Sydney. 2010, pp. 43–46 (cited on page 4)

[MAI06] R. L. Mandryk, M. S. Atkins and K. M. Inkpen. "A continuous and objective evaluation of emotional experience with interactive play environments".
In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*.
ACM. 2006, pp. 1027–1036 (cited on pages 45, 46)

[Mai14] M. Mainsbridge. "Musician Experiences with Gestural Interaction".
In: *Practice-Based Research Workshop at the International Conference for New Interfaces for Musical Expression (NIME'14)*. Workshop organised by the Creativity and Cognition Studios, University of Technology, Sydney, 2014 (cited on page 46)

[Man03] P. Manning. “The Influence of Recording Technologies on the Early Development of Electroacoustic Music”. In: *Leonardo Music Journal* 13 (2003), pp. 5–10 (cited on page 16)

[Mar+03] J. Marozeau, A. de Cheveigne, S. McAdams and S. Winsberg. “The dependency of timbre on fundamental frequency”. In: *The Journal of the Acoustical Society of America* 114.5 (2003), pp. 2946–2957 (cited on pages 120, 205)

[Mar10] A. Marsden. “Schenkerian analysis by computer: A proof of concept”. In: *Journal of New Music Research* 39.3 (2010), pp. 269–289 (cited on page 206)

[Mat14a] K. Matheussen. “Radium: A Music Editor Inspired by the Music Tracker”. In: *Proceedings of the Linux Audio-Developer Conference*. 2014 (cited on page 65)

[Mat14b] Matrixsynth - Everything Synth. *Roland TB-303*. [Online; accessed 10-November-2015]. 2014. url: http://www.matrixsynth.com/2014/12/roland-tb-303.html (cited on page 55)

[Mat15] Matthias Schöttle. *MT4j - Multitouch for Java (new revision)*. [Online; accessed 10-November-2015]. 2015. url: https://github.com/mschoettle/mt4j (cited on page 72)

[MC07] J. Marozeau and A. de Cheveigné. “The effect of fundamental frequency on the brightness dimension of timbre”. In: *The Journal of the Acoustical Society of America* 121.1 (2007), pp. 383–387 (cited on page 104)

[McA+95] S. McAdams, S. Winsberg, S. Donnadieu, G. De Soete and J. Krimphoff. “Perceptual scaling of synthesized musical timbres: common dimensions, specificities, and latent subject classes.” In: *Psychological research* 58.3 (1995), pp. 177–92 (cited on page 205)

[McA99] S. McAdams. “Perspectives on the contribution of timbre to musical structure”. In: *Computer Music Journal* 23.3 (1999), pp. 85–102 (cited on pages 77, 105)

[McC02] J. McCartney. “Rethinking the computer music language: SuperCollider”. In: *Computer Music Journal* 26.4 (2002), pp. 61–68 (cited on pages 59, 64)

[McD+10] J. McDermott, J. Byrne, J. M. Swafford, M. O’Neill and A. Brabazon. “Higher-order functions in aesthetic EC encodings.” In: *IEEE Congress on Evolutionary Computation*. 2010, pp. 1–8 (cited on page 188)

[McD+13] J. McDermott, T. Giffort, A. Bouwer and M. Wagy. “Should Music Interaction Be Easy?” In: *Music and Human-Computer Interaction*. Springer, 2013, pp. 24–48 (cited on pages 5, 6, 42–44)

[McL+] A. McLean, D. della Casa, S. Knotts, S. Aaron, B. Andrew and J. A. Romero. *TOPLAP*. [Online; accessed 28-February-2015]. url: http://toplap.org/ (cited on page 64)

[McL01] K. McLeod. “Genres, subgenres, sub-subgenres and more: Musical and social differentiation within electronic/dance music communities”. In: *Journal of Popular Music Studies* 13.1 (2001), pp. 59–75 (cited on page 53)

[McL11] A. McLean. “Artist-Programmers and Programming Languages for the Arts”. PhD thesis. 2011 (cited on pages 57, 81)

[MDH12] A.-r. Mohamed, G. E. Dahl and G. Hinton. “Acoustic modeling using deep belief networks”. In: *Audio, Speech, and Language Processing, IEEE Transactions on* 20.1 (2012), pp. 14–22 (cited on page 144)

[MGB14] E. Morgan, H. Gunes and N. Bryan-Kinns. “Instrumenting the Interaction: Affective and Psychophysiological Features of Live Collaborative Musical Improvisation”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by B. Caramiaux, K. Tahiroglu, R. Fiebrink and A. Tanaka. Goldsmiths, University of London, June 2014, pp. 23–28 (cited on page 40)

[Mil13] K. Milburn. “Futurism and Musical Meaning in Synthesized Landscapes”. In: *Kaleidoscope* 5.1 (2013), pp. 109–116 (cited on page 17)

[Mil14] J.-T. Milde. “Using natural language to specify sound parameters”. In: *Proceedings of the Joint International Conference Music Computing (ICMC) and Sound and Music Computing (SMC)*. Ed. by A. Andreopoulou, G. Kosteletos, N. Georgiou, A. Kolkou and A. Charoumenidou. National and Kapodistrian University of Athens, Sept. 2014, pp. 286–293 (cited on page 108)

[Mit98] M. Mitchell. *An introduction to genetic algorithms*. MIT press, 1998 (cited on page 187)

[Mix14a] MixGenius Incorporated. *LANDR*. [Online; accessed 23-March-2015]. 2014. url: https://www.landr.com/ (cited on page 44)

[Mix14b] MixGenius Incorporated. *MixGenius*. [Online; accessed 23-March-2015]. 2014. url: https://www.mixgenius.com/ (cited on page 44)

[MKM12] R. MacDonald, G. Kreutz and L. Mitchell. *Music, health, and wellbeing*. Oxford University Press, 2012 (cited on page 7)

[MM07] T. Magnusson and E. H. Mendieta. “The acoustic, the digital and the body: A survey on musical instruments”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. ACM. 2007, pp. 94–99 (cited on pages 5, 25)

[MMM02] R. A. Macdonald, D. Miell and L. Mitchell. “An investigation of children’s musical collaborations: The effect of friendship and age”. In: *Psychology of Music* 30.2 (2002), pp. 148–163 (cited on page 42)

[MO11] J. McDermott and U.-M. O’Reilly. “An executable graph representation for evolutionary generative music”. In: *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM. 2011, pp. 403–410 (cited on page 188)

[Mon09a] MongoDB Incorporated. *MongoDB*. [Online; accessed 28-March-2015]. 2009. url: http://www.mongodb.org (cited on page 116)

[Mon09b] I. Monson. *Saying something: Jazz improvisation and interaction*. University of Chicago Press, 2009 (cited on page 165)

[Mor+06] M. R. Morris, A. Paepcke, T. Winograd and J. Stamberger. “TeamTag: exploring centralized versus replicated controls for co-located tabletop groupware”. In: *Proceedings of the SIGCHI Conference of Human Factors in Computing Systems*. ACM, 2006, pp. 1273–1282 (cited on page 38)

[Mor04] M. R. Morris. “Beyond “social protocols”: Multi-user coordination policies for co-located groupware”. In: *Proceedings of the International Conference on Computer Supported Cooperative Work*. ACM Press, 2004, pp. 262–265 (cited on page 38)

[Mor80] G. Morgan. “Paradigms, Metaphors, and Puzzle Solving in Organization Theory”. In: *Administrative Science Quarterly* 25.4 (1980), p. 605 (cited on pages 24, 39)

[Moz15] Mozilla Developer Network. *Browser Touch Events*. [Online; accessed 03-March-2015]. 2015. url: https://developer.mozilla.org/en-US/docs/Web/Guide/Events/Touch_events (cited on page 70)

[MRO10] I. Maier, T. Rompf and M. Odersky. *Deprecating the observer pattern*. Tech. rep. 2010 (cited on pages 62, 66)

[MS10] M. Meier and M. Schranner. “The Planets”. In: *Proc. Int. Conf. NIME* Nime (2010), pp. 501–504 (cited on page 172)

[MSO13] J. McDermott, D. Sherry and U.-M. O’Reilly. “Evolutionary and generative music informs music HCI—and vice versa”. In: *Music and human-computer interaction*. Springer, 2013, pp. 223–240 (cited on pages 186–188)

[MSR93] G. Marino, M.-H. Serra and J.-M. Raczinski.
"The UPIC system: Origins and innovations".
In: *Perspectives of New Music* (1993), pp. 258–269 (cited on page 88)

[Mül+02] E. Müller, T. Zvonko, L. Asnaghi and E. Jälevik. *Renoise*. 2002.
url: http://www.renoise.com (cited on pages 108, 111)

[Mul+11] M. Muller, D. P. Ellis, A. Klapuri and G. Richard. "Signal processing for music analysis".
In: *Selected Topics in Signal Processing, IEEE Journal of* 5.6 (2011), pp. 1088–1110
(cited on page 104)

[Mun07] R. Munday. "Music In Video Games".
In: *Music, Sound and Multimedia: From the Live to the Virtual* (2007), p. 51
(cited on page 129)

[Nat10] Native Instruments GmbH. *Maschine*. [Online; accessed 28-March-2015]. 2010.
url: http://www.native-instruments.com/en/products/maschine/ (cited on page 108)

[Nat96] Native Instruments GmbH. *Reaktor*. [Online; accessed 28-February-2015]. 1996.
url: http://www.native-instruments.com/en/products/komplete/synths/reaktor-5/
(cited on pages 64, 86)

[NB11] C. Nash and A. Blackwell. "Tracking virtuosity and flow in computer music".
In: *Proceedings of the International Computer Music Conference 2011*. University of
Huddersfield, UK. Ann Arbor, MI: MPublishing, University of Michigan Library, 2011
(cited on pages 31, 55)

[NBG04] C. Nicol, S. Brewster and P. Gray. "Designing Sound: Towards a System for Designing Audio Interfaces using Timbre Spaces." In: *Proceedings of the International Community for Auditory Display Conference* (2004), pp. 4–7 (cited on page 110)

[Nei02] B. Neill. "Pleasure beats: rhythm and the aesthetics of current electronic music".
In: *Leonardo Music Journal* 12 (2002), pp. 3–6 (cited on page 172)

[Nie09] G. Nierhaus. *Algorithmic composition: paradigms of automated music generation*.
Springer, 2009 (cited on pages 156, 187)

[Nie10] L. Niebur. *Special sound: the creation and legacy of the BBC Radiophonic Workshop*.
Oxford University Press, 2010 (cited on page 16)

[Nij09] L. Nijs. *The musical instrument of natural extension of the musician*.
Witwatersrand University Press, 2009 (cited on pages 3, 4, 31, 32)

[Nor88] D. A. Norman. *The design of everyday things*. Basic books, 1988 (cited on page 3)

[OC08] A. Oliveira and A. Cardoso. "Emotionally-controlled music synthesis".
In: *Encontro de Engenharia de Áudio da AES Portugal* (2008), pp. 10–14
(cited on page 138)

[OC10] A. P. Oliveira and A. Cardoso. "A musical system for emotional expression".
In: *Knowledge-Based Systems* 23.8 (2010), pp. 901–913 (cited on pages 138, 156, 157)

[Ode+04] M. Odersky, P. Altherr, V. Cremet, B. Emir, S. Maneth, S. Micheloud, N. Mihaylov,
M. Schinz, E. Stenman and M. Zenger.
An overview of the Scala programming language. Tech. rep. 2004 (cited on page 59)

[Ode06] M. Odersky. "Pimp my library".
In: *URL http://www. artima. com/weblogs/viewpost. jsp* (2006) (cited on page 67)

[Ohm11] Ohm Force SARL. *OhmStudio*. [Online; accessed 28-March-2015]. 2011.
url: https://www.ohmstudio.com (cited on page 19)

[Ora14] Oracle Corporation. *JavaFX 8*. [Online; accessed 03-March-2015]. 2014.
url: http://docs.oracle.com/javafx/ (cited on page 70)

[Ovi06] S. Oviatt. "Human-centered design meets cognitive load theory: designing interfaces that help people think".
In: *Proceedings of the 14th annual ACM international conference on Multimedia*.
ACM, 2006, pp. 871–880 (cited on page 30)

[Pad+03] A. Padova, L. Bianchini, M. Lupone and M. O. Belardinelli. “Influence of Specific Spectral Variations of Musical Timbre on Emotions in the Listeners”. In: *Proceedings of the 5th Triennial Conference of the European Society for the Cognitive Sciences of Music (ESCOM 05)*. Ed. by R. Kopiez, A. C. Lehmann, I. Wolther and C. Wolf. September. 2003, pp. 227–230 (cited on page 137)

[Pam04] E. Pampalk. “A Matlab Toolbox to Compute Music Similarity from Audio.” In: *Proceedings of the International Conference on Music Information Retrieval*. 2004 (cited on page 115)

[Par98] J. Pareles. “The Rhythm Century: The Unstoppable Beat”. In: *New York Times* (1998), pp. 2–3 (cited on page 172)

[Pat08] A. D. Patel. *Music, Language, and the Brain*. Vol. 66. 1. Oxford University Press, 2008, pp. 59–62 (cited on pages 77, 78)

[PB04] T. Pinch and K. Bijsterveld. “Sound studies: New technologies and music”. In: *Social Studies of Science* (2004), pp. 635–648 (cited on pages 17, 81)

[Pee+11] G. Peeters, B. L. Giordano, P. Susini, N. Misdariis and S. McAdams. “The Timbre Toolbox: Extracting audio descriptors from musical signals”. In: *The Journal of the Acoustical Society of America* 130.5 (2011), pp. 2902–2916 (cited on page 104)

[Per61] V. Persichetti. *Twentieth-Century Harmony: Creative Aspects And Practice*. WW Norton New York, NY, 1961 (cited on page 161)

[PH09] E. W. Pedersen and K. Hornbæk. “mixiTUI: a tangible sequencer for electronic live performances”. In: *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*. ACM, 2009, pp. 223–230 (cited on pages 2, 10, 11)

[Pho14] Phosfiend Systems Incorporated. *FRACT*. [Online; accessed 28-February-2015]. 2014. url: http://fractgame.com/ (cited on page 64)

[Pic95] R. W. Picard. *Affective computing*. MIT press, 1995 (cited on page 133)

[Poo14] I. Poole. *RF filter basics tutorial*. [Online; accessed 30-March-2015]. 2014. url: http://www.radio-electronics.com/info/rf-technology-design/rf-filters/rf-filter-basics-tutorial.php (cited on page 89)

[Por04] T. Porcello. “Speaking of Sound Language and the Professionalization of Sound-Recording Engineers”. In: *Social Studies of Science* 34.5 (2004), pp. 733–758 (cited on page 81)

[Pri+09] S. Price, G. Roussos, T. P. Falcao and J. G. Sheridan. “Technology and embodiment: Relationships and implications for knowledge, creativity and communication”. In: *Beyond Current Horizons* (2009) (cited on page 29)

[PRI02] J. Patten, B. Recht and H. Ishii. “Audiopad: A Tag-based Interface for Musical Performance”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by C. Casey, K. Schneider and E. Hammond. 2002, pp. 148–153 (cited on pages 2, 11)

[Puc+96] M. Puckette et al. “Pure Data: another integrated computer music environment”. In: *Proceedings of the Second Intercollege Computer Music Concerts* (1996), pp. 37–41 (cited on pages 64, 86, 193)

[PZC90] M. Puckette, D. Zicarelli and Cycling ’74. *Max*. [Online; accessed 28-February-2015]. 1990. url: https://cycling74.com/ (cited on page 64)

[Qt 14] Qt Company. *Qt 5.4*. [Online; accessed 03-March-2015]. 2014. url: http://qt.io/ (cited on page 70)

[Ram+09] P. Ramanahally, S. Gilbert, T. Niedzielski, D. Velázquez and C. Anagnost. “Sparsh UI: A Multi-Touch Framework for Collaboration and Modular Gesture Recognition”. In: *ASME-AFM 2009 World Conference on Innovative Virtual Reality*. American Society of Mechanical Engineers. 2009, pp. 137–142 (cited on page 70)

[Ram06] P. Ramshaw. “Is Music Production now a Composition Process?” In: *Proceedings of the First Annual Conference of the Art of Record Production* (2006) (cited on pages 17–19)

[Ras00] J. Raskin. *The humane interface: new directions for designing interactive systems*. ACM Press/Addison-Wesley Publishing Co., 2000 (cited on page 35)

[RD00a] R. M. Ryan and E. L. Deci. “Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being.” In: *American psychologist* 55.1 (Jan. 2000), pp. 68–78 (cited on page 7)

[RD00b] R. Ryan and E. Deci. “Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being”. In: (2000) (cited on pages 27–29)

[Res+05] M. Resnick, B. Myers, K. Nakakoji, B. Shneiderman, R. Pausch, T. Selker and M. Eisenberg. “Design Principles for Tools to Support Creative Thinking”. 2005 (cited on page 39)

[Res15] Residenties in Utrecht. *Residentie Richard Devine: Modular Madness*. [Online; accessed 10-November-2015]. 2015. url: http://www.residentiesinutrecht.nl/wp-content/uploads/2015/06/richard-devine-3.jpg (cited on page 53)

[Ret03] R. Rettie. “Connectedness, Awareness and Social Presence”. In: *6th Annual International Workshop on Presence*. 2003 (cited on page 37)

[Rey97] M. Reybrouck. *Gestalt concepts and music: Limitations and possibilities*. Springer, 1997 (cited on page 206)

[Ris03] J.-C. Risset. “The perception of musical sound”. In: *Computer music: Why?* (2003). [Online; accessed 21-November-2015]. url: http://www.utexas.edu/cola/insts/franceut/archives/risset.pdf (cited on page 108)

[RK12] C. Roberts and J. Kuchera-Morin. *Gibber: Live coding audio in the browser*. Ann Arbor, MI: MPublishing, University of Michigan Library, 2012 (cited on page 64)

[RM95] A. Reilly and D. McGrath. “Convolution processing for realistic reverberation”. In: *Audio Engineering Society Convention 98*. Audio Engineering Society. 1995 (cited on page 93)

[Roa96] C. Roads. *The computer music tutorial*. MIT press, 1996 (cited on pages 88, 98, 99)

[Röb10] A. Röbel. “Between Physics and Perception: Signal Models for High Level Audio Processing”. In: *Proceedings of the International Conference on Digital Audio Effects*. 2010 (cited on page 108)

[Rob97] T. Robertson. “Cooperative Work and Lived Cognition: A Taxonomy of Embodied Actions”. In: *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work*. Springer Netherlands, 1997, pp. 205–220 (cited on page 3)

[Rus04] M. Russ. *Sound synthesis and sampling*. Taylor & Francis, 2004 (cited on pages 98, 99)

[Rus67] N. Russo. “Connotation of seating arrangements”. In: *Cornell Journal of Social Relations* 2.1 (1967), pp. 37–44 (cited on page 36)

[Rus80] J. A. Russell. “A circumplex model of affect.” In: *Journal of personality and social psychology* 39.6 (1980), p. 1161 (cited on pages 130, 131)

[Rut10] H. H. Rutz. “Rethinking the SuperCollider Client…”. In: *Proceedings of the SuperCollider Symposium, Berlin*. Citeseer. 2010 (cited on pages 72, 73)

[Rut12] H. H. Rutz. “A Reactive, Confluently Persistent Framework for the Design of Computer Music Systems”. In: *Proceedings of the International Conference on Sound and Music Computing Conference*. 2012 (cited on page 204)

[Rut15] H. H. Rutz. *ScalaOSC library*. [Online; accessed 07-March-2015]. 2015. url: https://github.com/Sciss/ScalaOSC (cited on page 74)

[Saw06] R. K. Sawyer. “Group creativity: musical performance and collaboration”. In: *Psychology of Music* 34.2 (Apr. 2006), pp. 148–165 (cited on page 32)

[Saw11] R. K. Sawyer. *Explaining creativity: The science of human innovation*. Oxford University Press, 2011 (cited on pages 22, 24–26, 32, 53)

[SB09] F. Seddon and M. Biasutti. “A comparison of modes of communication between members of a string quartet and a jazz sextet”. In: *Psychology of Music* 37.4 (2009), pp. 395–415 (cited on page 82)

[SBD99] J. Stewart, B. B. Bederson and A. Druin. “Single display groupware: a model for co-present collaboration”. In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM. 1999, pp. 286–293 (cited on page 2)

[SC06] S. D. Scott and S. Carpendale. “Investigating Tabletop Territoriality in Digital Tabletop Workspaces”. In: *Technical Report*. Department of Computer Science, University of Calgary, 2006 (cited on page 37)

[Sch+11a] B. Schneider, P. Jermann, G. Zufferey and P. Dillenbourg. “Benefits of a tangible interface for collaborative learning and interaction”. In: *Learning Technologies, IEEE Transactions on* 4.3 (2011), pp. 222–232 (cited on page 203)

[Sch+11b] C. Scholliers, L. Hoste, B. Signer and W. De Meuter. “Midas: a declarative multi-touch interaction framework”. In: *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction*. ACM. 2011, pp. 49–56 (cited on pages 60, 70, 71)

[Sch+13] K. R. Scherer, E. Coutinho, T. Cochrane, B. Fantini and K. Scherer. “How music creates emotion: a multifactorial process approach”. In: *The Emotional Power of Music: Multidisciplinary perspectives on musical arousal, expression, and social control* (2013), p. 121 (cited on page 130)

[Sch12] D. Schwarz. “The Sound Space as Musical Instrument: Playing Corpus-Based Concatenative Synthesis”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by G. Essl, B. Gillespie, M. Gurevich and S. O’Modhrain. University of Michigan, May 2012 (cited on page 109)

[Sch62] M. R. Schroeder. “Natural sounding artificial reverberation”. In: *Journal of the Audio Engineering Society* 10.3 (1962), pp. 219–223 (cited on page 94)

[Sch75] A. Schoenberg. *Style and Idea: selected writings of Arnold Schoenberg*. University of California Press, 1975 (cited on page 4)

[Sco+04] S. D. Scott, M. Sheelagh, T. Carpendale and K. M. Inkpen. “Territoriality in collaborative tabletop workspaces”. In: *Proceedings of the 2004 ACM conference on Computer supported cooperative work*. ACM, 2004, pp. 294–303 (cited on page 11)

[Sco+12] J. Scott, E. Schmidt, M. Prockup, B. Morton and Y. Kim.
"Predicting Time-Varying Musical Emotion Distributions from Multi-Track Audio".
In: *Proceedings of the 9th International Symposium on Computer Music Modelling and Retrieval* June (2012), pp. 19–22 (cited on pages 131, 139)

[SDP12] Y. Song, S. Dixon and M. Pearce.
"Evaluation of Musical Features for Emotion Classification".
In: *Proceedings of the International Society for Music Information Retrieval Conference*. Oct. 2012, pp. 523–528 (cited on pages 137, 142)

[Sea13a] A. Seago. "A New Interaction Strategy for Musical Timbre Design".
In: *Music and Human-Computer Interaction*. Springer, 2013, pp. 1–23 (cited on page 108)

[Sea13b] A. Seago. "A new interaction strategy for musical timbre design".
In: *Music human-computer Interact.* Springer, 2013, pp. 153–169 (cited on pages 108, 109)

[SEL11] P. Saari, T. Eerola and O. Lartillot. "Generalizability and Simplicity as Criteria in Feature Selection: Application to Mood Classification in Music".
In: *IEEE Trans. Audio. Speech. Lang. Processing* 19.6 (Aug. 2011), pp. 1802–1812 (cited on pages 117, 118, 142)

[Ser97] X. Serra. "Musical sound modeling with sinusoids plus noise".
In: *Musical signal processing* (1997), pp. 91–122 (cited on pages 88, 109)

[SG00] U. Schimmack and A. Grob. "Dimensional models of core affect: a quantitative comparison by means of structural equation modeling".
In: *European Journal of Personality* 14.4 (2000), pp. 325–345 (cited on page 137)

[SGM03] S. D. Scott, K. D. Grant and R. L. Mandryk.
"System guidelines for co-located, collaborative work on a tabletop display".
In: *Proceedings of the 8th conference on European Conference on Computer Supported Cooperative Work*. Kluwer Academic Publishers, 2003, pp. 159–178 (cited on pages 37, 38)

[Sha+10] J. Shao, J. McDermott, M. O'Neill and A. Brabazon.
"Jive: A generative, interactive, virtual, evolutionary music system".
In: *Applications of Evolutionary Computation*. Springer, 2010, pp. 341–350 (cited on page 188)

[Shn83] B. Shneiderman. "Direct Manipulation: A Step Beyond Programming Languages".
In: *IEEE Computer Magazine* 16.8 (Aug. 1983), pp. 57–69 (cited on page 48)

[Sie+13] B. Sievers, L. Polansky, M. Casey and T. Wheatley. "Music and movement share a dynamic structure that supports universal expressions of emotion".
In: *Proceedings of the National Academy of Sciences* 110.1 (2013), pp. 70–75 (cited on page 130)

[Sik03] M. R. Sikonja. "Theoretical and Empirical Analysis of ReliefF and RReliefF".
In: *Machine Learning* 53.1-2 (2003), pp. 23–69 (cited on page 142)

[SL61] M. R. Schroeder and B. Logan. "Colorless artificial reverberation".
In: *IRE Transactions on Audio* 6 (1961), pp. 209–214 (cited on page 93)

[SM13] D. Stowell and A. McLean.
"Live music-making: A rich open task requires a rich open interface".
In: *Music and human-computer interaction*. Springer, 2013, pp. 139–152 (cited on pages 45, 50, 55)

[Smi02] J. O. Smith. *Mathematics of the discrete fourier transform (DFT), 2nd Edition*.
Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California, 2002 (cited on page 91)

[Smi06] J. O. Smith. *Physical audio signal processing: For virtual musical instruments and digital audio effects*. 2006. url: http://ccrma.stanford.edu/jos/pasp/ (cited on page 93)

[Son04] Sony Computer Entertainment. *SingStar*. [Online; accessed 28-March-2015]. 2004. url: http://www.singstar.com (cited on page 50)

[SPB08] D. Stowell, M. D. Plumbley and N. Bryan-Kinns. "Discourse analysis evaluation method for expressive musical interfaces". In: *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*. 2008, pp. 81–86 (cited on page 45)

[Spe+11] J. A. Speck, E. M. Schmidt, B. G. Morton and Y. E. Kim. "A Comparative Study of Collaborative vs. Traditional Musical Mood Annotation". In: (Oct. 2011), pp. 549–554 (cited on page 139)

[Spe01] M. Sperber. "Developing a stage lighting system from scratch". In: *ACM SIGPLAN Notices*. Vol. 36. 10. ACM. 2001, pp. 122–133 (cited on page 63)

[spl14] splice.com. *Splice*. [Online; accessed 28-March-2015]. 2014. url: https://splice.com (cited on page 19)

[SR04] B. Swift and A. Riddell. "Engagement Networks in Social Music-making". In: *Interfaces (Providence).* (2004) (cited on pages 3, 32)

[SS90] X. Serra and J. Smith. "Spectral Modeling Synthesis: A Sound Analysis/Synthesis System based on a Deterministic plus Stochastic Decomposition". In: *Comput. Music J.* 14.4 (1990), pp. 12–24 (cited on page 88)

[SSK12] E. Schmidt, J. Scott and Y. Kim. "Feature Learning In Dynamic Environments: Modeling The Acoustic Structure of Musical Emotion". In: *ismir2012.ismir.net* Ismir (2012), pp. 325–330 (cited on pages 131, 150, 153)

[Ste06] J. Stepánek. "Musical sound timbre: Verbal description and dimensions". In: *Proceedings of the 9th International Conference on Digital Audio Effects (DAFx-06)*. 2006, pp. 121–126 (cited on page 108)

[Ste82] R. A. Stebbins. "Serious leisure: A conceptual statement". In: *Pacific Sociological Review* (1982), pp. 251–272 (cited on page 28)

[Ste99] R. J. Sternberg. *Handbook of creativity*. Ed. by R. J. Sternberg. Perspectives on individual differences. Cambridge University Press, 1999, p. 490 (cited on pages 21, 22, 24)

[Sto+09] D. Stowell, a. Robertson, N. Bryan-Kinns and M. Plumbley. "Evaluation of live human–computer music-making: Quantitative and qualitative approaches". In: *International Journal of Human-Computer Studies* 67.11 (Nov. 2009), pp. 960–975 (cited on page 3)

[Sut66] W. R. Sutherland. *ON-LINE GRAPHICAL SPECIFICATION OF COMPUTER PROCEDURES.* Tech. rep. DTIC Document, 1966 (cited on page 63)

[SVS10] L. Savioja, V. Välimäki and J. O. Smith III. "Real-time additive synthesis with one million sinusoids using a GPU". In: *Audio Engineering Society Convention 128*. Audio Engineering Society. 2010 (cited on page 88)

[Swi11] B. Swift. "A Chasing After the Wind : Experience in Computer-supported Group Musicmaking". In: *When Words Fail: What Can Music Interaction Tell Us about HCI? Workshop at BCS HCI* (2011), pp. 1–6 (cited on pages 3, 32, 172)

[Swi13] B. Swift. "Chasing a feeling: Experience in computer supported jamming". In: *Music and human-computer interaction*. Springer, 2013, pp. 85–99 (cited on pages 32, 45)

[Sym90] Symbolic Sound Corporation. *Kyma*. [Online; accessed 28-February-2015]. 1990. url: http://kyma.symbolicsound.com/ (cited on pages 18, 64)

[SZ01] K. R. Scherer and M. R. Zentner. “Emotional effects of music: Production rules”. In: (2001) (cited on pages 130, 132)

[Szw14] G. Szwoch. *Picture of waveforms and their spectra*. [Online; accessed 30-March-2015]. 2014. url: http://sound.eti.pg.gda.pl/student/eim/synteza/gusmann/ (cited on page 87)

[Tan+06] A. Tang, M. Tory, B. Po, P. Neumann and S. Carpendale. “Collaborative coupling over tabletop displays”. In: *CHI ’06: Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 2006, pp. 1181–1190 (cited on page 37)

[Tar95] E. Tarasti. *Musical signification: essays in the semiotic theory and analysis of music*. Vol. 121. Walter de Gruyter, 1995 (cited on page 18)

[Thu08] E. Thul. “Measuring the Complexity of Musical Rhythm”. MA thesis. McGill University, 2008 (cited on pages 162, 179, 191)

[Tie08] T. Tieleman. “Training restricted Boltzmann machines using approximations to the likelihood gradient”. In: *Proceedings of the 25th international conference on Machine learning*. ACM. 2008, pp. 1064–1071 (cited on page 145)

[Tin04] P. Tingen. *Autechre: recording electronica*. Accessed: 10.01.2015. 2004. url: http://www.soundonsound.com/sos/apr04/articles/autechre.htm (cited on page 79)

[TKI10] P. Tuddenham, D. Kirk and S. Izadi. “Graspables revisited: multi-touch vs. tangible input for tabletop displays in acquisition and manipulation tasks”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2010, pp. 2223–2232 (cited on page 38)

[TL11] J. Timoney and V. Lazzarini. “Exponential frequency modulation bandwidth criterion for virtual analog applications”. In: *Proceedings of the 14th International Conference on Digital Audio Effects*. Sept. 2011 (cited on page 98)

[TM02] M. Terry and E. D. Mynatt. “Recognizing creative needs in user interface design”. In: *Proceedings of the 4th conference on Creativity & cognition*. ACM. 2002, pp. 38–44 (cited on pages 39, 44)

[TM13] P. Thompson and P. McIntyre. “Rethinking Creative Practice In Record Production And Studio Recording Education: Addressing The Field”. In: *Journal on the Art of Record Production* 8 (2013) (cited on pages 22, 24)

[TP13] M. Thorogood and P. Pasquier. “Impress: A Machine Learning Approach to Soundscape Affect Classification for a Music Performance Environment”. In: (2013) (cited on page 138)

[Tru77] B. Truax. “The POD system of interactive composition programs”. In: *Computer Music Journal* (1977), pp. 30–39 (cited on page 88)

[Tym11] D. Tymoczko. *A geometry of music: harmony and counterpoint in the extended common practice*. Oxford University Press, 2011 (cited on page 158)

[UI 99] U&I Software LLC. *MetaSynth*. [Online; accessed 28-March-2015]. 1999. url: http://www.uisoftware.com/MetaSynth/index.php (cited on page 88)

[US05] B. Uzzi and J. Spiro. “Collaboration and creativity: The small world problem1”. In: *American journal of sociology* 111.2 (2005), pp. 447–504 (cited on page 24)

[Vag01] H. Vaggione. “Some ontological remarks about music composition processes”. In: *Computer Music Journal* 25.1 (2001), pp. 54–61 (cited on pages 4, 17, 49)

[Var02] Various Authors. *General Inquirer Augmented Spreadsheet*. [Online; accessed 28-March-2015]. 2002. url: http://www.wjh.harvard.edu/~inquirer/ (cited on page 140)

[Var14a] Various Open Source Developers. *Bacon.js*. [Online; accessed 28-February-2015]. 2014. url: https://beaconjs.github.io/ (cited on page 62)

[Var14b] Various Open Source Developers. *Reactive Collections*. [Online; accessed 28-February-2015]. 2014. url: https://http://reactive-collections.com/ (cited on page 62)

[Var14c] Various Open Source Developers. *ReactiveX*. [Online; accessed 28-February-2015]. 2014. url: http://reactivex.io (cited on page 62)

[Vas07] E. Vass. "Exploring processes of collaborative creativity—The role of emotions in children's joint creative writing". In: *Thinking Skills and Creativity* 2.2 (2007), pp. 107–117 (cited on page 158)

[Vin+04] B. Vines, M. Wanderley, C. Krumhansl, R. Nuzzo and D. Levitin. "Performance gestures of musicians: What structural and emotional information do they convey?" In: *Gesture-based Commun. human-computer Interact.* (2004), pp. 3887–3887 (cited on pages 3, 82)

[VPB09] A. Vinciarelli, M. Pantic and H. Bourlard. "Social signal processing: Survey of an emerging domain". In: *Image and Vision Computing* 27.12 (2009), pp. 1743–1759 (cited on page 36)

[VW66] E. Varèse and C. Wen-Chung. "The liberation of sound". In: *Perspectives of new music* (1966), pp. 11–19 (cited on page 79)

[VZA06] V. Verfaille, U. Zolzer and D. Arfib. "Adaptive digital audio effects (A-DAFx): A new class of sound transformations". In: *Audio, Speech, and Language Processing, IEEE Transactions on* 14.5 (2006), pp. 1817–1831 (cited on pages 103, 104)

[W+03] G. Wang, P. R. Cook et al. "ChucK: A concurrent, on-the-fly audio programming language". In: *Proceedings of International Computer Music Conference*. 2003, pp. 219–226 (cited on page 72)

[WAJ02] G. Weinberg, R. Aimi and K. Jennings. "The Beatbug network: a rhythmic system for interdependent group collaboration". In: *Proceedings of the International Conference on New Interfaces for Musical Expression* (2002), pp. 1–6 (cited on pages 50, 172)

[Wal+11] I. Wallis, T. Ingalls, E. Campana and J. Goodman. "A rule-based generative music system controlled by desired valence and arousal". In: *Proceedings of 8th international sound and music computing conference (SMC)*. 2011 (cited on pages 156, 157)

[Wal+13] I. Wallis, T. Ingalls, E. Campana and C. Voung. "Amateur Musicians, Long-Term Engagement, and HCI". In: *Music and Human-Computer Interaction*. Springer, 2013, pp. 50–60 (cited on pages 42, 43)

[WCC11] S. Wilson, D. Cottle and N. Collins. *The SuperCollider Book*. The MIT Press, 2011 (cited on pages 72, 86)

[WD05] G. Weinberg and S. Driscoll. "iltur – Connecting Novices and Experts Through Collaborative Improvisation". In: *Proc. 2005 Int. Conf. New Interfaces Music. Expr.* (2005), pp. 17–22 (cited on pages 11, 12)

[Wha99] I. Whalley. "Beyond pitch/duration scoring: towards a system dynamics model of electroacoustic music". In: *Proceedings of the 17th International Conference of the System Dynamics Society, 5th Australian New Zealand Systems Conference*. System Dynamics Society. 1999 (cited on page 78)

[WIC08] I. Wallis, T. Ingalls and E. Campana.
"Computer-Generating emotional music: The design of an affective music algorithm".
In: *Proceedings of the 9th International Conference on Digital Audio Effects (DAFx-08)* (2008), pp. 7–12 (cited on pages 157, 161)

[Wik07] Wikimedia Commons. *Picture of the reacTable*. [Online; accessed 28-February-2015]. 2007.
url: https://en.wikipedia.org/wiki/Reactable#/media/File:Reactable_Multitouch.jpg
(cited on page 9)

[Wik08] Wikimedia Commons. *Picture of Bessel Functions*.
[Online; accessed 28-February-2015]. 2008.
url: https://commons.wikimedia.org/wiki/File:Bessel_Functions_(1st_Kind,_n=0,1,2).svg
(cited on page 96)

[Wik11] Wikimedia Commons. *File:Roland TR-808 (large).png*.
[Online; accessed 10-November-2015]. 2011.
url: https://commons.wikimedia.org/wiki/File:Roland_TR-808_(large).png
(cited on page 53)

[Wil+14] D. Williams, A. Kirke, E. R. Miranda, E. Roesch, I. Daly and S. Nasuto.
"Investigating affect in algorithmic composition systems".
In: *Psychology of Music* (2014), p. 0305735614543282 (cited on page 133)

[WM91] P. C. Wright and A. F. Monk. "The use of think-aloud evaluation methods in design".
In: *ACM SIGCHI Bulletin* 23.1 (1991), pp. 55–57 (cited on page 45)

[Wol+10] B. W"oldecke, C. Geiger, H. Reckter and F. Schulz. "ANTracks 2.0 - Generative Music on Multiple Multitouch Devices Categories and Subject Descriptors". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*.
Ed. by K. Beilharz, B. Bongers, A. Johnston and S. Ferguson. 2010, pp. 348–351
(cited on page 11)

[Wor09] R. Worby. *Robert Worby's history of electroacoustic music*. 2009. url: http://www.londonsinfonietta.org.uk/content/robert-worbys-history-electroacoustic-music/
(cited on page 15)

[WW02] D. Wessel and M. Wright.
"Problems and Prospects for Intimate Musical Control of Computers".
In: *Computer Music Journal* 26.3 (Sept. 2002), pp. 11–22 (cited on pages 1, 4, 61)

[WW11] D. Wigdor and D. Wixon.
Brave NUI World: Designing Natural User Interfaces for Touch and Gesture.
Morgan Kaufmann, 2011, p. 253 (cited on pages 2, 38)

[Xam+11] A. Xambó, R. Laney, C. Dobbyn and S. Jordà. "Multi-touch interaction principles for collaborative real-time music activities: towards a pattern language".
In: *Proceedings of the International Computer Music Conference*. July 2011
(cited on pages 38, 39, 58)

[Xam+13] A. Xambó, R. Laney, C. Dobbyn and S. Jordà.
"Video analysis for evaluating music interaction: musical tabletops".
In: *Music and Human-Computer Interaction*. Springer, 2013, pp. 241–258
(cited on pages 8, 45, 55)

[Xam+14] A. Xambó, G. Roma, R. Laney, C. Dobbyn and S. Jordà. "SoundXY4: supporting tabletop collaboration and awareness with ambisonics spatialisation". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*.
Ed. by B. Caramiaux, K. Tahiroglu, R. Fiebrink and A. Tanaka.
Goldsmiths, University of London, June 2014, pp. 50–53
(cited on pages 10, 58, 124, 125, 204)

[Xen92] I. Xenakis. *Formalized music: thought and mathematics in composition*. 6.
Pendragon Press, 1992 (cited on page 88)

[XFe14] XFer Records. *Serum*. [Online; accessed 28-March-2015]. 2014.
url: http://www.xferrecords.com/products/serum/ (cited on page 108)

[XJP14] A. Xambó, C. Jewitt and S. Price. "Towards an integrated methodological framework for understanding embodiment in HCI".
In: *CHI'14 Extended Abstracts on Human Factors in Computing Systems*. ACM. 2014, pp. 1411–1416 (cited on page 30)

[XLD11] A. Xambó, R. Laney and C. Dobbyn. "TOUCHtr4Ck: Democratic Collaborative Music".
In: *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*. TEI '11. ACM, 2011, pp. 309–312 (cited on pages 2, 10, 19)

[ZA02] U. Zölzer and X. Amatriain. *DAFX: Digital Audio Effects*. Vol. 1.
Wiley Online Library, 2002 (cited on pages 85, 91–93)

[Zac+12] A. Zacharakis, K. Pastiadis, J. D. Reiss and G. Papadelis. "Analysis of musical timbre semantics through metric and non-metric data reduction techniques".
In: *Proceedings of the 12th International Conference on Music Perception and Cognition (ICMPC12) and the 8th Triennial Conference of the European Society for the Cognitive Sciences of Music (ESCOM 08)*. 2012, pp. 1177–1182 (cited on page 80)

[Zat05] L. Zattra. "Analysis and analyses of electroacoustic music".
In: *Proceedings of the International Conference on Sound and Music Computing*. Vol. 36. 2005 (cited on page 80)

[ZPR14] A. Zacharakis, K. Pastiadis and J. D. Reiss. "An Interlanguage Study of Musical Timbre Semantic Dimensions and Their Acoustic Correlates".
In: *Music Perception: An Interdisciplinary Journal* 31.4 (2014), pp. 339–358
(cited on page 80)