

An alternative design framework for the generation of sound particles

by

Pere Villez

Supervisor: Dr Janet Delve, Co-supervisor: Professor David Anderson

Submitted in partial fulfilment of the requirements for the award of the degree of
PhD of the University of Portsmouth

November 2014

Copyright

Copyright © 2014 Pere Villez. All rights reserved.

The copyright of this thesis rests with the Author. Copies (by any means) either in full, or of extracts, may not be made without prior written consent from the Author.

For Anne, Alba, Helen, Caroline, Natalia and Beatriz, for being so loving,
uniquely core and instrumental in getting me to this place.

Preface

I would like to thank Dr Janet Delve and Professor David Anderson for their continued support and assistance during the various stages of this thesis. I am very grateful to both for their extensive supervision, encouragement and attention to detail. I am also indebted to all of those who have had faith in this work including many of my colleagues at the University of Portsmouth.

A special thank you to all my friends and colleagues from around the world who have intentionally or unwittingly given me so much encouragement, inspiration and kept this mind on fire and engaged with the many different facets of electronic music and sound art.

Declaration

Whilst registered as a candidate for the above degree, I have not been registered for any other research award. The results and conclusions embodied in this thesis are the work of the named candidate and have not been submitted for any other academic award.

Word count: 48408 words excluding appendices, references, bibliography and figures.

Relevant Publications

Byrne Villez, P. (2000). Processing Samples with Csound's FOF Opcode, chapter 15, pages 307-320. In *The Csound Book* MIT Press, p653-656, Publisher: MIT Press; Pap/Cdr edition (10 April 2000), Language English, ISBN-10: 0262522616, ISBN-13: 978-0262522618.

Clark, J.J. (2010). Villez, P. FOF Synthesis. Available at: http://www.cim.mcgill.ca/~clark/nordmodularbook/nm_oscillator.html#FOF retrieved on the 2010-07-31 at 16:24:30.

Villez, P. (2008) Composing with Waveholes and Microsounds. Proceedings from *Reproduced Sound 24*, Vol 30. Pt6, Immersive Audio, 20-21 November 2008. ISBN 1 901656 95 0, ISBN 1478-6095.

Villez, P. (2009) Elementary Signal Engine, Microsound, Vague Terrains.
Available: <http://vagueterrain.net/journal15/pere-villez/02> retrieved 21rst April
2013 at 13.54.

Table of contents

Chapter 1	Introduction	18
1.1	Research introduction	18
1.2	Thesis structure	19
1.2.1	Reviews and critical assessment of microsound literature	19
1.2.2	Critical work and formulation of research questions	20
1.2.3	Structuring the research.....	21
1.2.4	Results	22
1.2.5	Using the results to Inform new implementations	22
1.2.6	Contributions	22
Chapter 2	Theoretical origins and aesthetic considerations	23
2.1	Quantifying time and movement	23
2.1.1	Counting drops of water	24
2.1.2	Microsound	25
2.2	The Fourier theorem	27
2.2.1	The technology of the Fourier transform	27
2.2.2	Denis Gabor and elementary signals	29
2.3	Science and representations in the music of microsound	31
2.3.1	Sound grains and the modernist formalism	32
2.3.2	Xenakis' influence on microsound.....	32
2.4	The physiology and perception of microsound	34
2.4.1	Microtemporal perception	34
2.4.2	Microtemporal fusion and fission	35
2.4.3	Composing outside the boundaries of perception.....	35
2.4.4	Discrete versus continuous perception	36
2.4.5	Non-concurrent masking and pitch perception	38
Chapter 3	Ontological issues in parametric thinking, composition and performance across multiple time levels.....	42
3.1	The influence of Horacio Vaggione.....	42

3.2	Automatic parameter weakness	44
3.2.1	Automatic control using computer programs	47
3.3	Determinism, indeterminism and monotony	48
3.3.1	Monotony	48
3.3.2	Intervention	50
3.3.3	Notations and extreme formalisms	51
3.4	Working across multiple time scales.....	52
3.4.1	Cybernetic versus generative composition	52
3.4.2	The specification of materials for the musical thesis.....	53
3.4.3	The need for interventionist tools.....	54
Chapter 4	Review of particle synthesis. Implementations and limitations	55
4.1	Digital Implementations of microsonic synthesis	55
4.1.1	Particle synthesis. Granular	56
4.1.1.1	The effect of particle shapes on timbre	56
4.1.2	The issue of strong and weak parameters	57
4.1.2.1	The effect of parameters on the synthesis	57
4.1.3	Composing and organising particles of sound in time	60
4.1.3.1	The Xenakis screen	60
4.1.3.2	Temporal grid interfaces	62
4.1.4	Formant Wave Functions (FOF)	65
4.1.4.1	The FOF envelope	66
4.1.4.2	FOF layers	68
4.1.4.3	The different faces of FOF	69
4.1.4.4	Octavation and its limitations.....	70
4.1.5	VoSim	71
4.1.5.1	Hard Synchronisation.....	72
4.1.6	Pulsar synthesis.....	73
4.1.6.1	Pulsar masking.....	74
4.1.7	PAF	76

4.1.8 Other methods	77
Chapter 5 Research focus. Overcoming existing PS limitations	79
5.1 Differentiation of particle organising features	79
5.2 The originating conceptual model translates into different mental models to different developers	80
5.3 Idiosyncratic limitations	81
5.4 Feature parity. Differentiation	83
5.5 Concept and context of difference	84
5.6 Too many features	85
5.7 Standard glue parameters and variations	85
5.8 Missing core features	86
5.9 Lack of standardisation across glue parameters	86
5.10 Developers	86
5.11 Question 1. Structuring the framework	89
5.12 Sound synthesis design criteria and universal principles of design	89
5.13 Question 2. Proto-frameworks	90
Chapter 6 Identifying MS specific design criteria	92
6.1 Jaffe's ten criteria	92
6.1.1 Jaffe's classification of criteria into domains	93
6.1.1.1 1 to 4 Usability. " <i>concerns the usability of parameters...</i> " (Jaffe, p76)...	93
6.1.1.2 5 to 7 Sound generation "Other criteria deal with the sounds produced" (Jaffe, p76)	94
6.1.1.3 8 to 10 Implementation. "The remaining criteria focus on efficiency and implementation" (Jaffe, p76)	94
6.1.1.4 The informal nature of Jaffe's Ten Criteria	95
6.1.2 Re-evaluating the classification of Jaffe's criteria	96
6.1.2.1 J5. Identity	96
6.1.2.2 J4. "How well-behaved are the parameters?" Jaffe, p77	97
6.1.2.3 J6. Sound classes	98
6.1.3 The framework analysis criteria classifications	99

6.1.4	Creating individual criterion attributes	100
6.1.5	Jaffe's attributes as cognitive and functional dimensions. The synthesiser as an Information artefact	102
Chapter 7	Cognitive Dimensions. Measuring cognitive complexity	103
7.1	Cognitive Dimensions and Jaffe's criteria	103
7.1.1	The Initial mapping of Jaffe's criteria to CDs	105
7.1.2	J2 Metaparameters as rules	107
7.1.3	Microsound cognitive complexity	108
7.2	Specific MS CD criteria and attributes analyses approaches	108
7.2.1	Synthesis artefact notations	109
7.2.2	The context of CDs and usability engineering heuristics in this research	110
7.2.3	Independent developers need more than discursive frameworks. Operationalisation.	111
Chapter 8	Functional Dimensions. Measuring functional complexity.....	113
8.1	Introduction	113
8.1.1	Functional capability and limitless behaviour	113
8.2	Converting Jaffe's functional criteria and attributes into MS measurable functional dimensions.....	116
8.2.1	J3. How physical are parameters in microsound?	116
8.2.2	Algorithmic interpretation of <i>physicality</i>	117
8.2.3	J4. " <i>How well behaved are parameters?</i> ". Defining functional behaviour in a synthesiser context	118
8.2.4	Behaviour and reliability	120
8.2.5	Consistent external behaviour and the code layer	120
8.2.6	The myth of completeness in testing	120
8.2.7	The testing ethos	121
8.2.8	External behaviour and hardware dimensions	121
8.2.9	Consistency, robustness in response	122
8.2.10	Latency and viscosity (Jaffe, p83-84. Green, 1989, p6)	122
8.2.11	Error frequency	122
8.2.12	Parameter dependency effects	123

8.2.13	Initialisation	124
8.2.14	Hardware behaviour analysis.....	124
8.2.15	CPU	125
8.2.16	Memory	126
8.2.17	Algorithm efficiency	126
8.2.18	Behavioural noise. Discontinuities, aliasing, distortion and other types of non-functional noise (consistency, reliability).....	127
8.2.19	Microsound behavioural analyses.....	127
8.3	J5 How robust is aural identity?	128
8.3.1	J5_A1. Identity in the context of variation	128
8.3.2	J5_A2. Can expression be synthesised in this technique?	129
8.3.3	What defines a tool?	130
8.4	J8 What classes of sound can be represented?. Phenomenology	132
8.4.1	J8_A1. Some techniques can produce any type of sound given enough control data	132
8.5	J10 Do analysis techniques exist?	134
8.5.1	The Chant model of production.....	134
8.5.2	Synthesis by rule.....	135
8.5.3	Data acquisition and driving.....	136
8.5.4	Accelerometers	138
Chapter 9	Design analysis process applied to MS specific criteria.....	140
9.1	The catalogued MS design criteria	140
9.2	Method for applying the MS criteria to empirical design analysis	142
9.2.1	Analysis methods.....	142
9.3	Navigating the analyses reports in appendices B-O.....	144
9.4	Summarising the analyses report's.....	144
Chapter 10	Summary of the design analysis framework results.....	146
10.1	Distilling the CD and FD analyses results.....	146
10.1.1	Visualising and reporting the analyses outcomes.....	147
10.2	Validating informal observations in chapter 5	149

10.2.1	Multitemporal composition (5.1).....	149
10.2.2	Interpretation of implicit specifications and multiple mental models (5.2, 5.4, 5.5) 151	
10.2.3	Idiosyncratic limitations (5.3).....	152
10.2.4	Too many features (5.6).....	153
10.2.5	Missing / redundant core features (5.8)	154
10.2.6	Glue parameters and standardisation (5.9, 5.10)	155
10.3	Interpreting and distilling, reducing the results for implementation	155
Chapter 11 Informed implementation.....		158
11.1	Particle synthesis using wave-orientated synthesis engines	158
11.2	The Elementary Signal Engine	158
11.3	ESE architecture	160
11.4	The Particle Wave Stream. PWS introduction	161
11.4.1	Emission	162
11.4.2	The formant component.....	164
11.4.3	Noise classes.....	165
11.4.4	Doubling the number of formant layers	165
11.4.5	Glissons	166
11.4.6	Particle Shapes	166
11.4.7	Teethlets	166
11.4.8	Overlaps.....	168
11.5	Waveholes	168
11.5.1	Bypassing the limitations of FOF octavation. Introducing multiviation	170
11.6	Libraries and metaparameters.....	172
11.6.1	The difference between the ESE and granular technique including other PS systems 174	
Chapter 12 Summary of the evaluation of the design analysis and development framework 176		
12.1	Introduction	176
12.2	Evaluating the effectiveness of the design strategy.....	176

12.2.1	Table 21 design goal 1. Minimise major misbehaviour dysfunctions. Dependency management, Catastrophic side-effects, behavioural noise	177
12.2.1.1	M1.A4 Dependencies. CD.....	177
12.2.2	Table 21 design goal 2. Feature disparity and integration of missing parameters	178
12.2.2.1	Noise classes	178
12.2.2.2	Formant harmonicity / inharmonicity	178
12.2.2.3	Overlaps.....	178
12.2.2.4	Real-time analysis / re-synthesis	179
12.2.2.5	Eliminate behavioural noise	179
12.2.3	Table 21 design goal 3. Comprehensive multitemporal composition features 179	
12.2.3.1	Scroller and Wavehole design issues to consider.....	179
12.2.4	Table 21 design goal 4. Metaparameter control in order to A. control dependency and for B. modelling physical properties.....	180
12.2.5	Table 21 design goal 5. a, Expertise levels, b. Premature commitment, c. reduction in HMO and d. libraries.....	181
12.2.5.1	Expertise levels	181
12.2.5.2	Premature commitment, HMO	181
12.2.5.3	Libraries	182
Chapter 13	Summary and conclusions	184
13.1	Answering research question 1	184
13.1.1	Efficiency of the research process	185
13.2	Answering question 2	186
13.2.1	Identifying and balancing the lack of standardisation	187
13.2.2	Revealing parametric parallels.....	187
13.2.3	Unifying modes of synthesis	188
13.2.4	Missing core features.....	188
13.2.5	Too many features	189
13.3	Beyond microsound.....	189
13.4	Final thought	191

Index of figures

Figure 1. Gabor matrix. (from Gabor 1946, p435)	30
Figure 2. Probability pulse as described by Gabor (from Gabor, D. 1946, p436).	30
Figure 3. Discrete to continuous.	37
Figure 4. Forward masking and microtemporal perception.	39
Figure 5. Xenakis <i>Metastasis</i> score (1992). Elements of musical scores are auxiliary.	40
Figure 6. 3D lattice representation of sound depicting time, timbre and pitch (Wishart, 1999, p26).	41
Figure 7. Audiomulch range sliders permit a random parameter range (screenshot).	46
Figure 8. Buzz Machines Pulsar Generator stochastic pan sliders (screenshot).	47
Figure 9. Audiomulch (screenshot).	58
Figure 10. Xenakis Screen.	61
Figure 11. Screens in a book and granular clouds.	62
Figure 12. Drum machine grid and corresponding musical notation grid.	63
Figure 13. AudioTool <i>Rhythm Composer</i> emulation of Roland TR909 (screenshot).	63
Figure 14. AudioTool <i>Rhythm Composer</i> emulation of Roland TR808 (screenshot).	64
Figure 15. Step programming switches as a function of time Δt	64
Figure 16. <i>Logic Studio</i> screen shot of the <i>Matrix</i> editor.	65
Figure 17. Elementary FOF particle (Rodet, 1984, p11).	67
Figure 18. Summation of FOF streams and resulting spectrum (Rodet, 1984, p13).	68
Figure 19. FOF layers.	70
Figure 20. Consecutive VoSim \sin^2 pulses shaped by a staircase envelope.	71
Figure 21. Hard synchronisation.	72
Figure 22. Example of Pulse Width Modulation and duty cycles of 50, 75 and 90%.	73
Figure 23. Pulsar synthesis implementation of Jeskola Buzz Machines.	75
Figure 24. Additive and subtractive microsound.	77
Figure 25. Some core parameter differences between systems.	83
Figure 26. PS Design Framework summary.	143
Figure 27. Appendix P table format.	146
Figure 28. Detail of Appx P OSTM showing the "Binary States" node.	148
Figure 29. Multi-paragraph branches.	149
Figure 30. OSTM Multitemporal Composition / synthesis node.	150
Figure 31. M5 excel screenshot detailing PS particle organisation scope.	151
Figure 32. Implicit specifications and mental models branches.	151
Figure 33. CPU Functional analyses.	152
Figure 34. Functional behaviour branches.	153
Figure 35. Missing and redundant paramters.	154
Figure 36. Reduced OSTM as a result of similar analyses outcomes.	156
Figure 37. The Elementary Signal Engine component diagram.	159
Figure 38. Formant module library interface and parametric data.	161
Figure 39. The three main PWS components.	162
Figure 40. PWS emission component structure.	162
Figure 41. ESE emission interface. Orange values are real-time display units only.	163
Figure 42. PWS formant component structure.	164
Figure 43. PWS formant component structure.	165
Figure 44. Particle Teeth. Two periods per original period.	167
Figure 45. The main ESE Wavehole interface.	168
Figure 46. The main ESE Wavehole interface.	169
Figure 47. The Wavehole procedural programmer.	171
Figure 48. Formant library window showing the tenor belcanto vowel "ah".	172
Figure 49. ESE composition sequence of components libraries.	173
Figure 50. Scroller parameter and composition states.	174
Figure 51. Composition / synthesis and multitemporal relationships.	174
Figure 52. OSTM of ESE design analyses.	183

Index of tables

Table 1. Musical Time Scales (Roads, 2001a, p3)	26
Table 2. Microsound synthesis techniques.....	55
Table 3. Granular synthesis parameters.....	56
Table 4. Jaffe's ten criteria.....	59
Table 5. FOF synthesis parameters.	68
Table 6. Usable PS Implementations.	80
Table 7. Particle formant <i>glue</i> parameters.....	85
Table 8. Jaffe's ten criteria.....	93
Table 9. Jaffe's criteria and attributes.....	101
Table 10. M1 Cognitive complexity. Notations and notation control.....	108
Table 11. M2. Cognitive complexity. Perceptibility of parameter changes.	108
Table 12. Jaffe's criteria reclassified as functional dimensions.	116
Table 13. M3 Metaparameters and physical behaviour.....	118
Table 14. "Desirable" functional characteristics.....	119
Table 15. M4 Microsound external behaviour.....	128
Table 16a. M5 Identity. Particle synthesis identity dimensions.....	131
Table 16b. M5 Identity. Organization of particles.	131
Table 17. M6 Sound classes and phenomenology.....	134
Table 18. Singing synthesis rules system (Sundberg, J. 2009).....	137
Table 19. M7. Data acquisition and rule based systems.	138
Table 20a. Taxonomy of the CDs for the analysis of formant based PS.	140
Table 20b. Taxonomy of the FDs for the analysis of formant based PS.	141
Table 21. An example PS implementation strategy.....	157
Table 22. Multiviation pitch intervals in the ESE.....	171
Table 23. ESE differences from other PS systems.....	175
Table 24. Possible new ESE design and development strategy.	182
Table 25. Chapter 5 summary of superficial limitations in particle formant synthesis.	186

Abstract

This research contributes to an empirical design analysis framework for informing the implementation of a type of microsound synthesis called Particle Synthesis (PS).

Microsounds are sound particles with a duration typically lasting from the threshold of perception up to ~100ms (Roads, 2001a).

Microsound synthesis software is popular in music production and is used for processing audio. PS, however, still remains relatively unknown to composers and only recently have independent developers made PS available on popular computer platforms. Variations in implementation, however, disguise similar underlying techniques and produce inconsistencies. Perhaps this is because independent developers (often individual researchers/hobbyists) lack the development formalities of large software corporations.

Although different sound classes are generated by PS systems, they share very similar parametric features. However, parameters, which are key in one system, are commonly implemented partially or omitted in another. In order to obtain a wider more flexible range of microsound sound classes, users have to operate several of the systems simultaneously across incompatible operating systems and environments.

To address this situation, this thesis examined the questions as to what specific functional and usability criteria might inform new PS designs and whether this criteria and consequent design framework would be successful in informing new PS artefacts.

Key theoretical foundations of microsound techniques are identified, and their use in 20th century music composition examined. PS techniques lack a specific design framework. However, Jaffe (1995) proposed criteria to assess general synthesiser technologies, and design analysis principles called Cognitive Dimensions (CDs) were revealed which help evaluate the design of information artefacts including software synthesisers.

Many elements of Jaffe's criteria may be viewed as industrial design principles, which may be generalised into microsound synthesis criteria.

By combining the CDs framework with the new microsound artefact criteria and operationalising them as variables, an empirical analysis framework for studying usability factors of PS software was made possible. Furthermore, these can be applied to the functional elements of design.

Subsequently, empirical studies were conducted in which seven identified PS implementations were quantitatively assessed against the new microsound criteria in order to ascertain the weight of individual usability and functional characteristics across the systems.

The collated results from the studies confirmed that many common usability and functional features exist. They further revealed uneven and incomplete implementation of features for composition of sound particles over multiple time scales, and seamless manipulation of continuous and discrete particle sound streams. The results were collated in order to establish specific design goals and used to implement and adapt a prototype PS compositional system called the Elementary Signal Engine (ESE), which includes the common and disparate features of the systems studied. This is the ancillary contribution to this work.

The success of the new MS design analysis framework is subsequently evaluated by studying the ESE artefact using the same analysis framework against the design goals established from the original seven PS artefact analyses.

Chapter 1 Introduction

1.1 Research introduction

According to Roads (2001, p21), microsounds are:

"...a broad class of sounds that extend from the threshold of timbre perception (several hundred microseconds) up the duration of short sound objects (~100ms)"

In the past few years, synthesis software for the generation of microsounds has become popular in many types of music production. Specifically, a type of microsound synthesis (MS) called "granular" synthesis, has become a common technique and is often used to process recorded audio samples. Other microsound techniques however, such as particle synthesis (PS), still remain relatively unknown to sound designers and composers and only recently have independent developers fashioned these tools on popular computer platforms. Despite this recent emergence, some implementations of the same synthesis technique vary enough to seem like different synthesis products.

It is determined that this is because the software development intents of independent developers are different from large-scale industrial software companies, which have tended to put economics at the fore of the development process. Industrial development is usually guided by strict software requirements and specifications (Dubberly, 2001). Independent developers are often individual researchers who are motivated to create software for personal interest/research and who often share the resulting artefacts with a select group of similarly interested users and thus do not necessarily follow a formal criteria

design framework to effectively interpret the original synthesis technique specifications. A consequence of this is that two artefacts purporting to implement the same particle synthesis technique produce inconsistent phenomenology.

In order to remedy this situation, this thesis examines the questions as to what specific functional and usability criteria could inform an effective design and implementation framework for improving existing and new PS artefacts. Furthermore, it examines whether the consequent design framework is successful in informing new and improved PS artefacts. Formally rationalised in paragraphs 5.11 and 5.13:

“What are the functional and usability criteria which could establish a design and implementation analysis framework for the effective, unified and robust development of particle formant synthesis / composition environments?”

and,

“Is the consequent microsound specific design analysis framework successful in informing new particle synthesis artefacts?”

1.2 Thesis structure

1.2.1 Reviews and critical assessment of microsound literature

In order to further understand the scope of the research questions, a review of the domain literature of MS was conducted to pinpoint its theoretical origins (chapter 2), its main theoretical musical composition perspectives and their relevance to MS artefact implementation (chapter 3). These are followed by a software review that distinguishes the techniques and implementations of PS (chapter 4). The latter identifies four main particle synthesis techniques and

seven implementations. Furthermore, it reveals the general architecture of synthesis and compositional features afforded in these artefacts.

1.2.2 Critical work and formulation of research questions

The original work of this thesis commences in chapter 5 with a critical assessment of the background information obtained from the review process. It examines the PS software review presented in chapter 4 in order to assess the design characteristics of the PS techniques, the current implementations and their impact on the general usefulness of PS in creativity. The chapter 5 critiques discuss specific core parametric inconsistencies identified in the seven PS implementations. The chapter 4 survey reveals that key parameters in one system are only partially implemented or altogether omitted in another, resulting in limitations in each artefact's sound classes. It is determined in chapter 5 that in order to obtain a wider and more flexible range of microsound sound classes, potential users would have to operate several of the systems simultaneously or in tandem and across incompatible operating systems and environments. Consequently, the role of independent developers in this type of artefact is examined and the part played by design criteria driven frameworks on the implementation of PS techniques is discussed.

It is hypothesised that because of the relative newness of PS, a design framework for the implementation of this type of artefact does not exist. This is an important factor in the absence of standardisation in the implementation of common core parameters in PS. Consequently, it is conjectured whether incorporating existing generalised synthesis criteria with established general design analysis principles, helps create specific design analysis criteria for PS design purposes. This premise is developed and formally presented as the research questions.

1.2.3 Structuring the research

Chapter 6 proposes an approach to identifying design criteria in order to answer the research questions. The focus of the research begins in this chapter.

A key article (Jaffe, 1995) is identified in the literature review in which informal criteria are presented which are used for the discussion of usability and functionality in synthesiser technology and which could be employed to informally assess the effectiveness of general synthesiser technology.

Furthermore, Blackwell (2001b) had considered that an analytical framework called the Cognitive Dimensions of Notations, first proposed by Thomas Green (Green, 1989), could be used for the evaluative discussion of cognitive technologies or information artefacts such as synthesisers.

Consequently, a design framework for the usable and functional design of PS is formulated in this chapter and is constructed from the translation of Jaffe's general criteria into microsound specific criteria using a mapping of Thomas Green's Cognitive Dimensions (CDs) (Green, 1989).

By combining the CDs framework with the new microsound artefact criteria and operationalising the results as variables, an empirical analysis framework for studying usability factors of particle synthesiser software is made possible. Furthermore, it is determined that these can be also expanded to consider the functional elements of PS design.

Subsequently and, in order to evaluate the effectiveness of this approach, further original work in the form of empirical studies are conducted (appendices B to O) in which each of the seven previously identified PS implementations from chapter 4 are analysed using the new analytical design framework to ascertain any individual usability and functional strengths and weaknesses.

1.2.4 Results

The collated results from the empirical studies are summarised in chapter 10 with a detailed report in appendix P. Succinctly, the results reveal that there are a significant number of common usability and functional issues shared between the studied systems. In particular, they confirm and quantify the problems found in the software review in that there is an uneven and incomplete implementation of features for the general effective composition and organisation of synthesised particle in particular over multiple time scales and the seamless manipulation of continuous and discrete particle streams.

1.2.5 Using the results to Inform new implementations

Consequently, the results are used to inform, improve and extend an experimental particle formant synthesis and composition environment called the Elementary Signal Engine (ESE). Chapter 11 summarises this implementation and is presented in full technical detail in appendices Q and R.

In order to evaluate the effectiveness of the design analysis framework resulting from the investigation, the ESE is subjected to the same framework's empirical analyses in order to determine whether the new artefact design has been successful in addressing the design disparities and deficiencies observed in the original artefacts. Furthermore, the evaluation is used to pinpoint possible further areas of design improvement. The results of these are summarised in chapter 12 and presented in detail in appendix S.

1.2.6 Contributions

The PS design analysis framework developed in this investigation is presented as the principle original contribution to the field of microsound synthesiser design together with an auxiliary original contribution; the ESE particle synthesis environment.

Chapter 2 Theoretical origins and aesthetic considerations

2.1 Quantifying time and movement

We are used to the experience of people and objects passing us by; people walking, cars travelling in the street, balls bouncing by in the park. It is rare that our everyday lives are devoid of any such motion. Everything in the universe, in fact, is in motion in some sense or other. Generally speaking, we are able to describe how these movements occur. We have developed special descriptions, such as "up" or "down", in order to describe vertical motion or "fast" or "slow" to describe speed and "straight" or "curved" for the shape of a trajectory. If we wish, we can, up to a certain point, quantify some of the movement; e.g. "...the ball bounced 5 times when the child threw it on to the ground." or "...it bounced three times when it hit the wet grass".

When things get too small or move too fast however, we begin to have difficulty in quantifying an object's movement and place. Instead, we quickly rely on qualitative descriptions; e.g. "...it was a blur...".

One such thing which generally permeates our everyday world, is sound. We are mostly capable of hearing and feeling it and have built a varied and complex language to describe its myriad of manifestations yet no matter how sophisticated this language has become, we cannot see the movement of the atoms which make up the sound waves let alone count how many times these waves repeat in the course of one second. It would seem that the smaller the object and the faster it moves the less we can use direct observation and our

capacity to measure becomes uncertain as it gets smaller and faster. We find it difficult to quantify at this scale and rely uniquely on qualitative descriptions.

2.1.1 Counting drops of water

The reader could imagine the sound of something familiar such as the short sounding of a drop of water. Most of us are familiar with rain and how it starts. It would not be difficult for us to imagine how a particular instance of this frequently experienced phenomenon unfolds over time. Apart from the slow random falling of individual drops of water and their dampening effect, we are also used to the sound they make when they impact on other objects. We would not find it difficult to imagine such a sound. Most of us are familiar with the *rain* game played in kindergarten in which a child gently taps the tips of their fingers slowly and as other children join in, they gradually create the sound of a continuous gentle downpour.

When a drop of water collides with the hard ground, perceptually the sound lasts for a brief moment. It is significant enough however, that as more and more of these short sounds fall in parallel and become multitudinous, we begin to lose the perception of them being the separate discrete acoustic events we called "drops" and a moment arrives in which we hear a continuous sound; a torrent of rain. There is a moment in which our perception seems to blur and loose accuracy. The perception of the continuity of the sound of rain, like many other phenomena, may be a limitation of our neuropathology and perceptual relativity. Nevertheless, this perceptual continuity can be shown to be an illusion and that the acoustic phenomenon of the torrent is in fact made from layers of discrete acoustical components.

These short discrete sound events are technically known as *microsounds* (Xenakis, 1991; Roads, 2001a). According to Roads (2001a, p21) microsounds are:

"...a broad class of sounds that extend from the threshold of timbre perception (several hundred microseconds) up to the duration of short sound objects (~100ms)."

When microsounds are grouped into hundreds and thousands, they meld into a continuous acoustic event. The microsonic representation of sound can be seen as a 20th century reaction to the established model of sound in classical physics, where sound is represented as a continuous phenomenon (Gabor, D. 1946). The emergence of a representation of sound waves being composed of discrete quantities would inspire new methods with which to create and organise musical elements. These new techniques employ a formalised aesthetic rooted in mathematics and science (Xenakis, 1992. pxii) instead of the traditional European musical compositional ontologies, which had evolved for some 400 years.

2.1.2 Microsound

In terms of this thesis, the term microsound is defined using Road's definition as stated above. This classifies microsounds on a temporal scale (Table 1) and occupies a place in between the sound object and the scale of a digital audio sample (Roads, 2001a, p3).

1. Infinite. The ideal time span of mathematical durations such as the infinite sine waves of classical Fourier analysis.
2. Supra. A time scale beyond that of an individual composition and extending into months, years, decades, and centuries.
3. Macro. The time scale of overall musical architecture or form, measured in minutes or hours, or in extreme cases, days.
4. Meso. Divisions of form. Groupings of sound objects into hierarchies of phrase structures of various sizes, measured in minutes or seconds.
5. Sound object. A basic unit of musical structure, generalising the traditional concept of note to include complex and mutating sound events on a time scale ranging from a fraction of a second to several seconds.
6. Microsound. Particles on a time scale that extends down to the threshold of auditory perception (measured in thousandths of a second or milliseconds).
7. Sample. The atomic level of digital audio systems: individual binary samples or numerical amplitude values, one following another at a fixed time interval. The period between samples is measured in millionths of a second (microseconds).
8. Subsample. Fluctuations on a time scale too brief to be properly recorded or perceived, measured in billionths of a second (nanoseconds) or less.
9. Infinitesimal. The ideal time span of mathematical durations such as the infinitely brief delta functions.

Table 1. Musical Time Scales (Roads, 2001a, p3)

Microsounds *en masse* produce a continuous stream of events that would seem to be beyond the limits of our ordinary sensory processing (Roads, 2001a, p22). There is a threshold beyond which we cannot use our innate counting capacity and instead require external methods involving quantifying technology in order to understand and explain what happens in the blur of the moments that make up these events.

The representation of sound as being continuous is an idea, which has influenced the development of music making technologies for the past three

centuries. The conception is a notion dating back to the early 19th century and one that governed the analyses methods and musical instrument making technologies of the time. In the early 20th century however, new scientific models of sound emerged which were based on new radical physics and mathematical concepts and which challenged classical models of the world. For the past forty years, these scientific ideas have influenced and inspired the way in which we design computer musical instruments and how we compose with them.

2.2 The Fourier theorem

In the 18th century, the French scientist Joseph Fourier gave us a method by which we would eventually analyse and quantify many types of periodic movement including sound waves. The Fourier theorem (Fourier, 1807) presented a convenient but paradoxical way of quantifying the frequency of these phenomena. The method is still used today. The digital representation of the theorem, which is referred to as the Fourier Transform, is used to analyse wave motion. The method represents any signal as a sum of simpler constituents called harmonics.

Fourier created this world-changing theorem as a way to solve a heat transmission equation. He found that he could model complex heat sources by superimposing sine functions to solve the equation that described how heat moves through metal. It was found that the same principle could be applied to other forms of periodicity such as the modelling of complex sound waves travelling through a medium.

2.2.1 The technology of the Fourier transform

Much of the slow movement of the theoretical background and development of the Fourier analysis was due to the technological limitations of the time. Several mechanical technologies were employed including "singing flames" and

"singing water jets" (Roads, 1996, p545). The limitations of mechanical analysers lay in that they could not interpret continuous data such as transient waveforms. It was not until the development of the electronic filter that time-variant sound waves became easier to analyse (Roads, 1996, p1087).

It was the invention of the oscilloscope in the early 1940s and the digital computer in the 1950s that generated a completely new field of research and development into analysis and re-synthesis methods. One of the first new approaches was the use of sampling data based on Norbert Wiener's interpretation which expands the Fourier analysis interpretation from a static harmonic analysis to one which also includes time (Roads, 2001a, p63). This was adopted by Blackmann and Tukey and became the principle method for analysing signals after 1950s (Roads, 1996). The digital model of the Fourier theorem is termed the Fourier transform (FT).

Parallel to the development of these analysis technologies, advances in sound synthesis permitted the creation of primitive sounds by using AS. However, this was a financially costly technique implemented using analogue hardware. To create a reasonably complex spectrum, many individual hardware sine-wave oscillators were needed. An alternative method called *subtractive* synthesis (SS) (Moore, 1990, p263) used the relatively cheaper technologies found in filters (combinations of low-pass and high-pass) to exclude unwanted spectral components.

It is worth noting that much of the sound synthesis developments (including those discussed) were adopted from technological developments which took place in communication laboratories and broadcast centres such as radio stations (Ernst, 1977, p41). Filters and calibration oscillators are but two examples of the technological borrowings adopted in additive and subtractive synthesis. This trend has continued throughout the 20th century with other developments such as *frequency modulation* synthesis (FM) (Chowning, 1977) which originated in radio broadcasting in the research of sound spatialisation. *Granular* synthesis (GS) (Roads, 1978b, Truax, 1988) emerged from communications (Gabor, 1946). Other synthesis technologies are informed by physics models and include physical modelling (Karplus & Strong, 1983) and

the mathematical modelling of physical phenomena such as the representation of the singing voice (Sundberg, 1978).

2.2.2 Denis Gabor and elementary signals

In his "Theory of Communication" (1946), Gabor presents the idea of *quanta of information* (Gabor, 1946, p429) containing both frequency and time information as representations of information signals (radio, television, telephony). The term *sonic quanta* was later used by Iannis Xenakis and inspired by Gabor's paper. He used the term in relation to his mathematical theory of musical composition (Xenakis, 1992). Gabor's quanta are representational parallels to *photons* or packets of electro-magnetic energy found in quantum physics. Gabor's sound quanta refer to a context in which continuity is represented by the discrete product of a probability function (Gaussian function). The quanta are represented as individual pairs of frequency and time data. Gabor refers to these as an *elementary signal*. The information containing the amplitudes and loci in each elementary signal are referred to as *logons*. The logons are graphically represented as rectangles on a matrix (Fig.1).

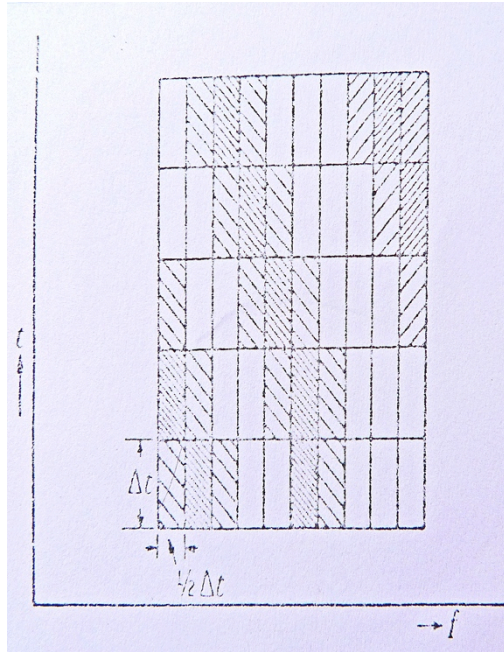


Figure 1. Gabor matrix. (from Gabor 1946, p435)

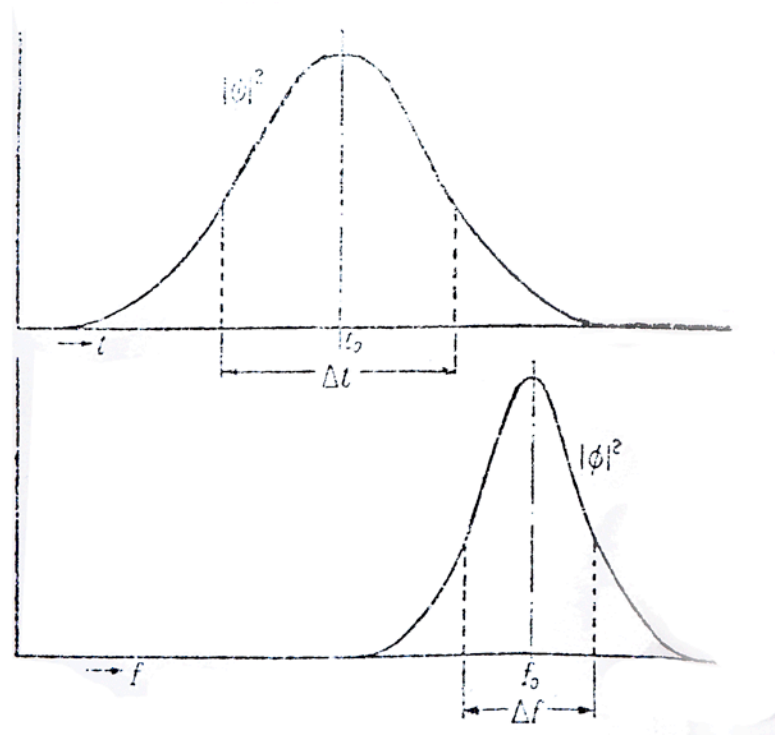


Figure 2. Probability pulse as described by Gabor (from Gabor, D. 1946, p436).

The *Gabor expansion*, as it is often referred to, is the expansion of an arbitrary real-world signal into elementary signals composed of sines and cosines (Gabor, 1946, p436). The Gabor transform (GT) is a Fourier transform (FT) that in addition to frequency analysis adds the discrete notion of time similarly to the DFT. The basis is a series of local (windowed) FFTs on a complex signal (Roads. 2001a, p296). The function of the probability pulse in Gabor's theorem is similar in function to that of the window in a DFT. In essence, Gabor's representation of complex information signals can be viewed as granular.

In terms of representations of sound and specifically musical signals, Gabor's work has inspired important areas in the artistic and technology domains. Central to this research is sound synthesis, the composition of music and design of sound for film, games and animation.

2.3 Science and representations in the music of microsound

The granular view of sound or music and specifically the term *grain* is attributed to the composer Iannis Xenakis who originally attributed his theory of *grains of sounds* to Gabor's information:

"All sound is an integration of grains, of elementary sonic particles, of sound quanta. Each of these elementary grains has a threefold nature: duration, frequency, and intensity.¹ All sound even all continuous sonic variation, is conceived as an assemblage of a larger number of elementary grains adequately disposed of in time" (Xenakis, 1992, p43).

The same scientific and mathematical developments of the 20th century, which influenced technology, also had a major impact on the composition techniques of the European classical music of that time.

2.3.1 Sound grains and the modernist formalism

Iannis Xenakis' compositional and theoretical legacy is of primary importance with regards to granular synthesis. Xenakis (1992) and Stockhausen (1959) worked within the domain of microsound using analogue electronics. Xenakis, however, offers a detailed theory of sound grains and their organization (Xenakis, 1992). After the early 20th century and as electronic musical hardware and culture emerged, it seemed that the traditional established ideas of musical composition were becoming less relevant. The flexibility of the new electronic musical instruments meant that the traditional five musical attributes of pitch, harmony, pulse, dynamics and timbre could be made more “plastic” and further extended into new realms.

Serialism (Schoenberg) and *Total* Serialism (Boulez, 1971) emerged to *freshen* the old order of *tonal* composition (Schoenberg, 1951; Boulez, 1971). The musical structures of the *tonal* language had evolved over some 400 years from the renaissance and modernism. In the 20th century, mathematics, science and particularly physics would provide new avenues as sources of inspiration for a new musical ontology (Roads, 2001a, p64). The literature exploring mathematical processes in the generation, processing and organisation of sound has expanded considerably during the past two decades. Testimony of this is available in publications such as MIT's Journal of Computer Music, Leonardo, AES Journal and others.

2.3.2 Xenakis' influence on microsound

Iannis Xenakis viewed composition as a very different process from the traditional *tonal* craft. Xenakis saw artistic profit from the formalisation of musical composition through probability, set theories, calculus and the influence of quantum mechanics (Xenakis, 1992, p255). The stochastic and set theory organisation of sound material is central in much of Xenakis' work.

Xenakis first experimented with microsounds in *Metastasis* (1954) using the traditional western orchestra and later in an electro-acoustic piece for magnetic tape *Concret PH* (1958) (Roads, 2001a, p64). *Metastasis* is particularly relevant in that it uses naturally occurring microsounds such as the crackling of fire as the spectral basis of the work. Karlheinz Stockhausen experimented with filtered impulse generators (Chamberlain, 1980) in his piece *Kontakte* (1960). Like Xenakis, Stockhausen developed a theory of microtemporality (Stockhausen 1959).

Both of these composers are today regarded as pivotal in the development of an aesthetic, culture and theory of microtemporal music composition techniques. With Xenakis as an influence, Curtis Roads (1978) developed the first digital generator of microsounds, specifically, a granular generator. It was during the same period that the formulaic organisation of technogenic microsound and its ontology was created (Roads, 2001a, p86). Curtis Roads has dedicated much of his artistic and technical life to the development, study and dissemination of this field. Because of Roads' early research, a variety of microsound techniques has emerged. These are generally categorised into granular or particle generators (Roads, 2001a, p118). A detailed taxonomy expands this to a wider variety of techniques, the implementations of which are discussed in chapter 4. The term particle synthesis refers to a specific set of techniques that are somewhat different in their application to sound synthesis (Roads, 2001a, p121). The phenomenologies are often distinct from granular synthesis. Henceforth, when referring to the term "grain", granular synthesis is implied. Furthermore, when referring to particle synthesis the term "particle" will be used.

The formalised approach to composition can be found in many genres of electronic music today and includes genres such as *glitch*, *post digital*, *laptop*, *circuit bending* and *the aesthetics of failure* (Cascone, 2000). Microsound is a wide aesthetic that influences many genres of music and sonic art (Wishart, 1996) and is embraced by major pop artists such as Bjork. Many of the quantum mechanics inspired granular representations of musical signals (wavelets, grains, particles) are similar and could, in essence, be seen as

variations on a theme, that is, the convolution of some exponential (Kaegi, Werner, Tempelaars, 1978) (Rodet, 1984) or probability function (Roads, 1978) by a sine/cosine function. Later this would be expanded to include recorded sounds. The only difference would seem to be the implementation details and context (This is explored in more detail in chapter 4).

2.4 The physiology and perception of microsound

2.4.1 Microtemporal perception

To demonstrate the underlying principles behind his "information quanta", Gabor demonstrated the practical implementation of sound expansion and contraction. By modifying a 16mm film projector through the addition of a slotted drum, Gabor demonstrated a crude implementation of the equivalent of a probability pulse. He was able to contract recorded signals (Gabor, 1946, p452). Gabor introduces the notion of frequency conversion (without time distortion) in an ingenious implementation using magnetic recorders (Gabor, 1946, p453). Variants of this technology can be found later in the Tempophone (Anon. 2013. Tempophone). It is the separation of the representations of time and frequency as independent elements of the sonic grain which has attracted composers from the 1950s onwards. It is the manipulation of this relationship and its resulting phenomenology which is central to the ontology of microsound music. Roads identifies seven areas of microtemporal perception. (Roads, 2001a, p21). Of particular interest to this research is microtemporal *fusion* and *fission*.

- Microtemporal Intensity Perception
- Microtemporal Fusion and fission
- Microtemporal Silence Perception
- Microtemporal Pitch Perception
- Microtemporal Auditory Acuity

- Microtemporal Preattentive Perception
- Microtemporal Subliminal Perception

2.4.2 Microtemporal fusion and fission

Stockhausen's *Kontakte* is perhaps one of the first pieces of electroacoustic music which uses analogue impulse generators (Ernst, 1977, p34) and which explores this perceptual phenomenon. Of particular interest is the *glissando* previously described at the beginning of this thesis in 1.1. There is, to quote John Dack, a *teleological imperative* (Dack, 1999) in terms of the plasticity of passing of time at this point in the composition. The crossing of this perceptual boundary is approximately 30 Hz (Villez, 2009) and the distinction between continuous pitch and discrete sonic events, becomes blurred. Fusion / disintegration is unique to all microsonic synthesis techniques. All microsonic synthesis methods share this attribute of identity and approach the manipulation of this feature in different ways. Unlike *linear* wave based synthesis, microsound techniques exhibit a unique characteristic in that at sub-audio fundamental frequencies the listener is able to perceive the primary acoustic signature of the technique, that is, its sonic unit; the *grain*, *particle*, microsound.

2.4.3 Composing outside the boundaries of perception

As particles of sound are generated periodically and at up to between 20-40 times per second, they are perceived as discrete events, however, at faster rates the listener begins to hear a *fusion of particles* and perceives continuity of tone. The exact threshold of discrimination rate depends on the duration and timbre of the sound but somewhere in between lays a subjective transitional boundary which will be referred to in this research as the *flutter* region (as in bird flutter). The boundary between the perception of discrete acoustic events and continuous sound waves is an area that has been documented variedly since Gabor's work. Yet the psycho-acoustic use in musical composition and

sound design would seem to need further research (Miranda, 2001, p7, pp2). Many sound classes such as vocal *gurgling* and *rippling* sounds would seem to fall into this category and have been used in compositions such as *Mälarsång* (Clarke, M. 1987), *Chreode* (Barrière, 1983) and much of Roads' own work (Roads, 2001a). There is, however, controversy as to the extent composers have creatively engaged with this region of audio perception and it raises the question as to how far we can shake off the legacy and constraints of the traditional elements in music making when applied to electronic music (Miranda, 2001, p7, pp2).

Mälarsång is a composition that often crosses the flutter region. Realistically synthesised singing voices using FOF (Rodet, 1984) generators, repeatedly evaporate from solid belcanto vocalisations into wave like chimes. The listener's perceptual reference is shifted seamlessly from one sound object; the *voice* to another; the *chime* which seems to have emerged from the depths of the former and creating a dramatic sonic narrative. It is evident then that dynamic manipulation of microtemporal events either by *fission* or *fusion* (Roads, 2001a, p22) is one of the main compositional interests in microsound. In microsound, the traditional five elements of music are extended and intentionally spread along the *meso* and *micro* time levels (Vaggione, 2001). These elements can be manipulated and blurred in increasingly creative ways, creating sonic dimensions and aural paradoxes, which challenge the listener's aural and psychological expectations.

2.4.4 Discrete versus continuous perception

The detailed study of the neuroanatomy and psychoacoustics of *fusion/fission* are beyond the scope of this research. Nevertheless, it can be thought of in terms of the density of perceived concurrent events. At higher fundamental frequencies, one event masks the transient onset of the next (Roads, 2001a, p24). In granular synthesis, it is thought of parametrically in terms of *density*. In formant particle synthesis (Roads, 2001a), it is characterised by rhythm (sparse) or tonality (dense). However, the term *density of stream* would seem

an appropriate description for all types of microsonic synthesis. By *stream*, it is meant, the sequence of microsonic events.

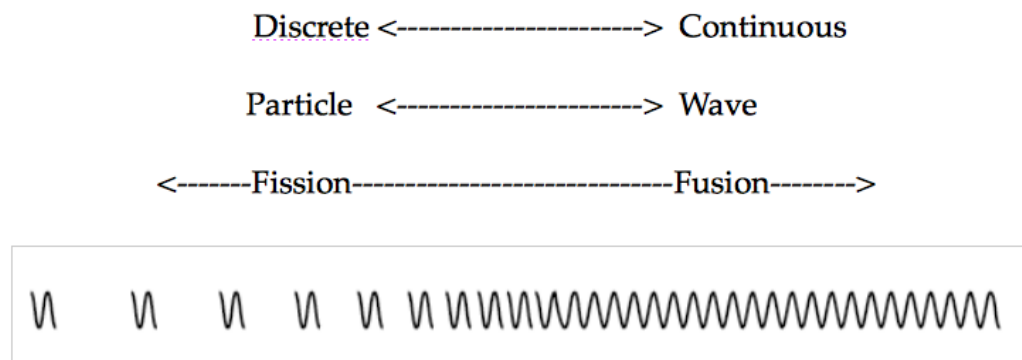


Figure 3. Discrete to continuous.

In *Kontakte*, the drama produced by this boundary crossing can be thought of as an inherent narrative (Dack, 1999), which is also embedded into the process of much microsonic music. This drama is in reference to the event itself. To borrow Dack's terminology, the pulse or microsound is the *smallest unit of signification*. It reveals itself as it is resolved from behind the audio-rate perceptual masking. Microsound as a sound generation technique is unique in its ability to switch fluidly and with ease between the time scales of sound.

Indeed, the musical exploration of the psycho-acoustic boundaries between tone and rhythm began with those early compositions and resulted in several key writings by the composers mentioned earlier (Stockhausen 1959; Xenakis 1992). FOF synthesis (Rodet, 1984) as used in the compositions *Mälarsång* and *Chreode* effectively use the movement between the continuous tone and discrete particles to present a dramatic psycho-acoustic illusion to the listener. Dramatic examples of shifts between the sound object and micro time levels examples can be found on an LP collection published by *IRCAM Un Portrait* (IRCAM, 1983) in which a re-synthesised human voice 'disintegrates', 'evaporate' into high frequency chimes as the fundamental frequency glides into the *infrasonic* range. The digital synthesis of microsound differs in behaviour from frequency domain synthesis techniques such as AS and SS in that at low fundamental frequencies, microsounds maintain their spectral composition. In

terms of pitch research and from a specifically scientific point of view the term *infratones* is used when referring to pitch below the considered threshold of hearing, ~30Hz

"...we will name such periodic sounds as infratones or infratone stimuli and their corresponding sensory attribute infrapitch" (Warren, 2008, p65)

2.4.5 Non-concurrent masking and pitch perception

With sounds lasting less than 200ms it would seem that the effect of *non-simultaneous (forward) masking* has an effect on the way we perceive pitch because one sound makes another inaudible. This is possibly an explanation as to why a fundamental frequency of 15Hz~ is difficult to resolve as either pitch or rhythm (Fig.4).

"Forward masking may correspond in part to the time required for the receptors to regain their sensitivity and/or the persistence of activity"
(Warren R. M, 2008, p71)

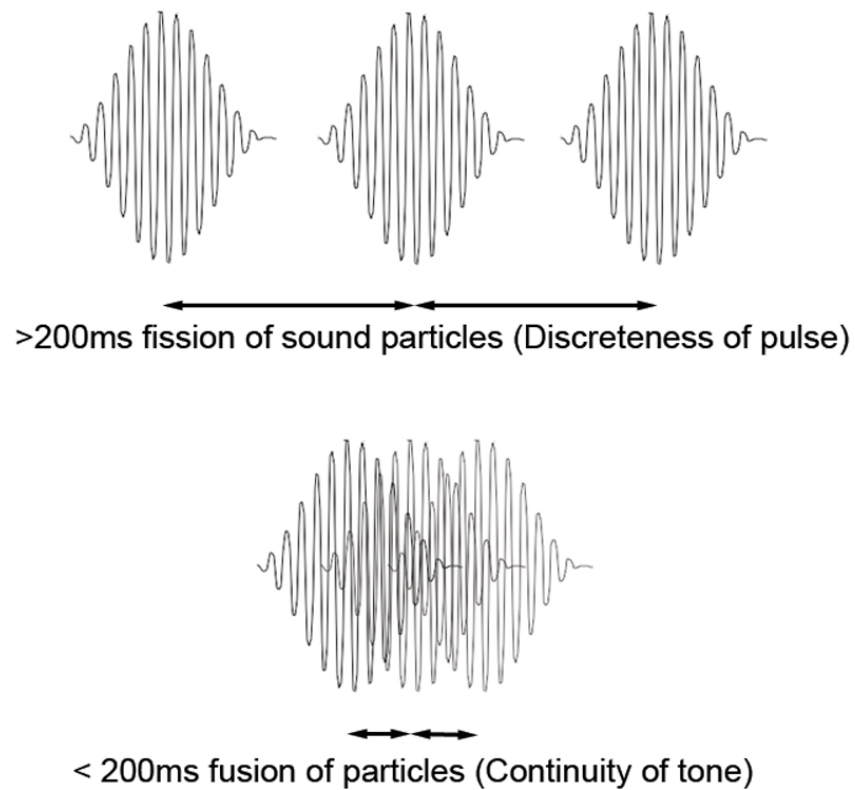


Figure 4. Forward masking and microtemporal perception.

The neuropathology of hearing, together with the composition of microsound is an area of study which could greatly benefit the understanding of the compositional process in the time domain. Because this is such a relatively new area, one could imagine the need for the creation of a formal musical vocabulary or organisational grammar.

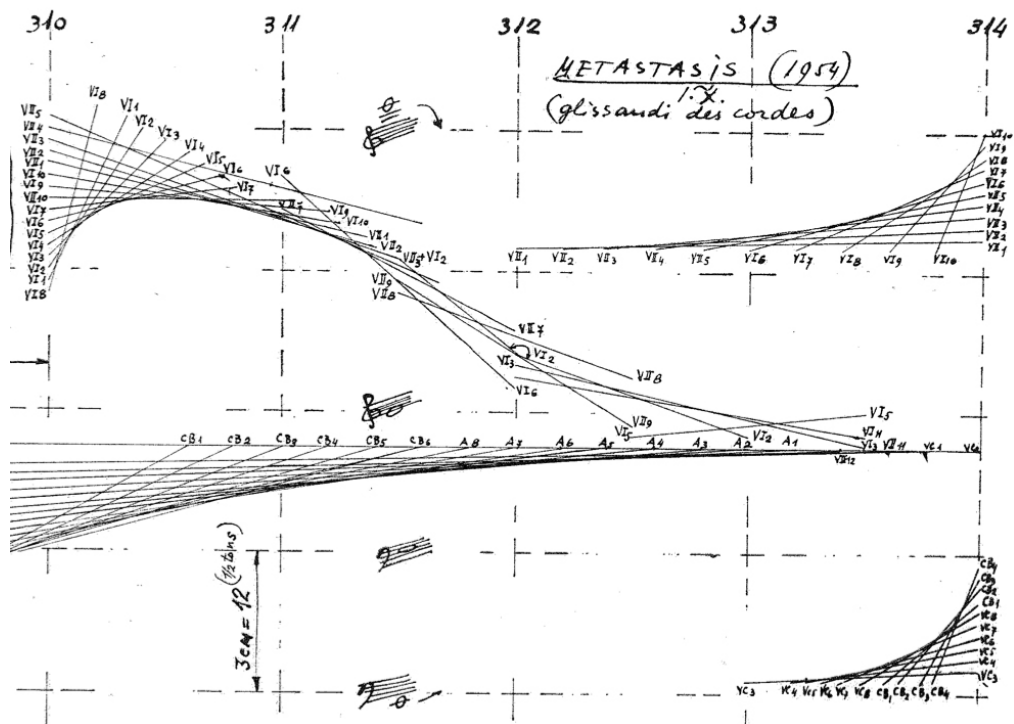


Figure 5. Xenakis *Metastasis* score (1992). Elements of musical scores are auxiliary.

Such a vocabulary, including visual representation, would aid the creative exploration of the relationships between different microsound processes and the effect these have on boundary percepts. Playing with the audience's expectations is a fundamental dramatic element in the composition of electronic music. The writings of Trevor Wishart (1996) explore many of the psycho-acoustic effects of electro-acoustic composition. Figures 5 and 6 illustrate two distinct visual representations of such formalist approaches. Traditional music scoring elements become ancillary rather than primary components.

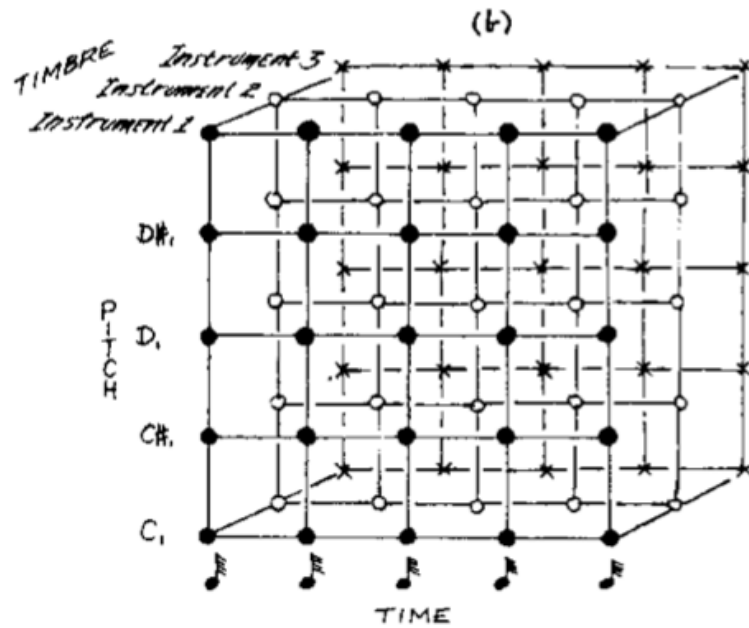


Figure 6. 3D lattice representation of sound depicting time, timbre and pitch (Wishart, 1999, p26).

Finally, the multitemporal uniqueness of the phenomenological identity of microsound synthesis and composition leaves much room for the expansion of new formalisms directly based on perceptual cues. Inevitably, this also impacts on the technologies used to create microsound compositions. With this in mind, the following chapter introduces two key microsound compositional approaches and their relevance in the design of microsound instruments.

Chapter 3 Ontological issues in parametric thinking, composition and performance across multiple time levels

3.1 The influence of Horacio Vaggione

Horacio Vaggione is an Argentinian composer and a contemporary in the field of computer music who has written many influential essays on compositional methods using computers. His influence is particularly relevant to the field of microsound and the manner in which composers work across multiple time domains. Sound synthesis is the basic material in electroacoustic composition and is part of its ontological hierarchy. It is part of the specification of what he terms the musical thesis (Vaggione, 2001, p57). In computer music these two domains are interchangeable. Vaggione sees the composition principle in computer music as an integration of the creation of computable functions, perception, and composer/user feedback (Vaggione, 2001, p57):

“Vaggione was one of the first to introduce fractals into music. Only too conscious of music’s richness and specificity, he was not one to be taken in by the musical limits of the important idea of self-similarity. In his theoretical texts, we find very pertinent remarks on the notion of complexity, on the relationship between synthesis and notation” (Risset 2005, p290, pp2)

Vaggione would seem to be one of the main driving forces of a post-reductionist aesthetic in computer music art. At the heart of Vaggione’s creative process lies

a classic Wienerian cybernetic (Wiener, 1948) interaction in which human intervention is at the heart of the creative principle. This drive would seem to be supported by luminaries such as the composer Claude Risset (Risset, 1995) and Curtis Roads (Cornicello, 2010) amongst others. Vaggione's perception is that the musical thought in the mid to late 20th century was occupied by a totalitarian formalism (Boulez, 1971) which had often been locked by a prescribed set of aesthetics embedded with the classical dogmas of proportion and philosophical universals. Together with reductionist formalism, these controlled and validated almost all aspects of the musical art form (Vaggione, 2001, p55 pp6). The accompanying reductionism often framed and re-circulated this formalism as dogma and tried to represent musical thought in a way not always intended by the composer.

"Debussy's saying, 'The work makes its own rules', summarizes well the situation of the composer's constraints..." (Vaggione)

Vaggione considers a view of electroacoustic composition in which interaction and perceptual feedback are key components of the process. This is not inconceivable in the ontology of microsonic synthesis in which action and perception are principal determinants of the synthesis model parameterisation and not just a cybernetic relationship which focuses specifically on the machine (Vaggione, 2001, p57, pp4). In essence, this is a contradiction to the reliance on generative and automated tendencies.

There is an elegant logic to Vaggione's conception and one that would seem to practically translate to real-time microsound synthesiser implementation. Not only does it not rely on some external strict formalism but also considers the concept of working across multiple time scales as essential (Vaggione, 2001, p60). This is a challenge in terms of designing sound and music creation tools to work with micro-events in the meso and macro compositional spheres.

It is not only a matter of creating events on the microsound scale but also of organising these across the sound object and meso time scales with an encompassing real-time perception of them by the composer. It would seem desirable that this is done without too many *weak* parameters (Jaffe, 1995). It is

well documented that using strict compositional techniques, based purely on mathematical formalisms such as Boulez's 1950s "total" serialism in *Structures* for Two Pianos (Ruch, 2004), resulted in criticism from his peers,

"...projected a static quality, a musical equivalent of alphabet soup..."
(Ligeti 1965 via Loy 2006, p332)

"Boulez tightened the reins on his music, pursuing increasing levels of serialization: not only would pitch be regulated to mathematical formulae, but other dimensions such as rhythm, duration, intensity, and so on would also be subject to control. Despite Boulez's claims to the contrary, this was not a completely original idea -- Messiaen had tried and abandoned it, and Milton Babbitt had been independently working on a similar concept in America." (Ruch, 2004)

3.2 Automatic parameter weakness

"This formalist approach in controlling composition has often been applied to synthesis too." (Roads, 1978, p61).

One of the cornerstones of Wienerian cybernetics is the idea of the parameter. According to Edward Miranda (Miranda, 2001, p12), formalised compositions such as Boulez's *Structures* and others are the result of parametric thinking. This approach to composition is particularly common in most classic computer music. This is because the range of variables used in making music has expanded from the limit of the traditional five musical elements (pitch, rhythm, dynamics, harmony and melody), to working with the parameters of the synthesis / processing method and the compositional approach, governed by interface technology. The meaning of parametric in this instance however, is a contradiction to the idea of the *parameter* being an actioning element for intervening in a *system*. In the Wienerian sense, the parameter is pure interventionism; it involves human *feedback* (Wiener, 1965). In Boulez's *Structures* sense however, the parameter is formalised and a result of

premature constraints. David Jaffe (1995) proposes that there is a relationship between the size of the combinatoriality of elements in a composition and the weak parameter relationship in synthesis. This is also echoed by Miranda with reference to computer music composition:

“The fewer the combinatorial possibilities, the easier it is for a composer to handle the material” (Miranda, 2001, p13, ppp2)

This is similar to; "the greater [*the number of*] parameters in a system, the weaker their effect on the system" (Jaffe, 1995). In appendix J, this parameter strength relationship is discussed in more detail. Here it is explained succinctly. Jaffe (1995) proposes that the more parameters present in a synthesis technique, the weaker their perceptual effect when changed. Jaffe uses an additive synthesis example to illustrate this point. Tweaking the amplitude of one single partial amongst a hundred, for example, will not necessarily be perceived at the output. It will depend on a number of factors such as frequency range. In this investigation, it is proposed that the perception of phenomenological change is dependent on whether weak parameters are controlled individually or in relation to other parameters or using metaparameters (see appendix J). This parametric relationship can be found in industrial testing strategies for parameter dependency / coupling testing (Kaner, et al. 2001).

“The range of possibilities can of course be augmented by increasing the inventory, but the larger the inventory the greater the chances of producing pieces beyond the human threshold of comprehension”(Miranda, 2001, p3, pp2)

The use of automated methods for organising possibilities across time levels becomes an attractive proposition when faced with such a large quantity of control data. Trends in granular synthesis composition have adopted automatic or semi-automatic approaches to composition because of the need for the mass generation of grains per second. In some genres of contemporary music, the grain cloud has become a sort of stochastic cliché. The number of grains per

second is so great that direct real-time user control of individual events is impractical.

“Granular synthesis requires a massive amount of control data.” (Roads, C. 2001, p87)

An example of this integration can be found in “Swarm Granulator” (Blackwell, T; Young, M. 2004), where bird flocking and swarm (Reynolds, 1987) algorithms are used to impose order and chaos on micro- and meso-time scale events. Another approach is to simplify the automated process in a way that gives the user some control by affording meta-parameters such as range sliders or a generative function such as stochastic randomness controlled by a single parameter. Two examples are shown in Fig.13, 14. Audiomulch implements random range sliders for each parameter in a granular generator with corresponding envelope time-line control to enhance added user control (Fig.7).

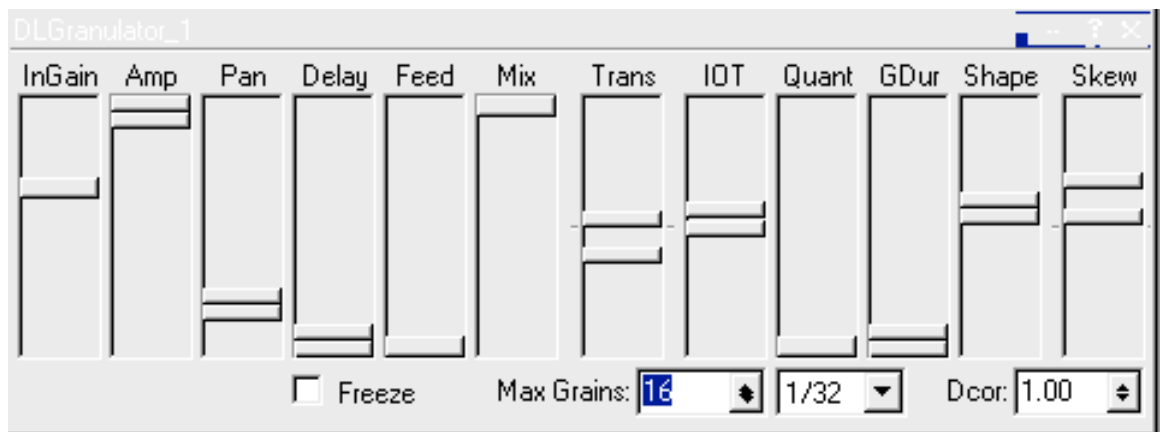


Figure 7. Audiomulch range sliders permit a random parameter range (screenshot).

Another example can be found in *Buzz Machines* in which the *burst* masking parameter is modulated by a stochastic function (Fig.8).



Figure 8. Buzz Machines Pulsar Generator stochastic pan sliders (screenshot).

The literal translation of scientific processes to sound synthesis and computer music can be found in many systems and encompass anything from unpredictable non-linear functions (Miranda, 2001, p83) to *artificial intelligence*, *artificial life* (Miranda, 2001, p108), particle physics (Sturm, 2001) and specific formulae as in Schrödinger's equation (Fischman, 2003). The contents of Eduardo Miranda's *Composing Music with Computers* (Miranda, 2001, pv)) lists discrete mathematics, algebraic modelling, set operations, set algebra, combinatorics, logic, formal grammars, probability, Markov chains, chaos, fractals, artificial intelligence and evolutionary computing. In order to simplify further discussions about the above processes, these will be referred to as *automatic processes*.

3.2.1 Automatic control using computer programs

Until capable desktop computers and flexible real-time processing appeared, interaction at this scale could only occur in none real-time computing environments via rigid computational methods and limited human feedback. Today, it is relatively easy to generate masses of events on the microsound level without little interaction from the user:

"By specifying several parameters at a higher level, a Composer can call for the automatic of thousands of grains" (Roads, 1978b)

Models of GS and particle synthesis implemented in *Music V* or *Csound*, rely on an understanding of the *machine tongue* (Krasner, 1980) in order to implement the particular model and require a deep mental model of the synthesis technique, which can often span tens of weak parameters. There is an experimental futility (Cornicello, 2010, p4) and as such it necessitates a precise understanding of the syntactical rules, together with a clear conception of the desired results. It is understood that in the early days of computer music, very few users without the required DSP or audio engineering knowledge, were able to do this. Unfortunately, the results were often a mismatch due to the poor literal translation of a scientific formalism using compositional common denominators such as frequency or amplitude.

“...trying to do it all with 1980s computers with the Music 11 programming language-good luck! “One more perf” was always the mantra at MIT. One is typing numbers to adjust envelopes, and it will probably be stilted...”(Cornicello, 2010, p4)

3.3 Determinism, indeterminism and monotony

3.3.1 Monotony

There would seem to be a deep incompatibility between phonography and strict automatic compositional processes. Recording such types of work is a contradiction to the works' conceptual basis. Listening to a passage in which pitch, rhythm, dynamics or timbre have been serialised will often require repetitive listening. If the listener is fortunate to maintain curiosity long enough in order to motivate them to explore the work further, the compositional process gradually becomes transparent. This might take several listenings. Once this process is undertaken however, the listener will have likely become familiar with the details of the work and possess a degree of perceptual consonance with it.

The listener could predict, “what comes next” at the expense of seeding possible monotony.

Yet, if a composition is totally generated by a grammar, it can be dull. If it breaks rules on the other hand, it is likely to be unintelligible (Lehrdahl & Jackendoff, 1983, via Reybrouck, 1989, p82)

Vaggione sees this in other systems including connectionist models of automated composition:

“The connectionist approach looks to embrace a whole thing in a none-analytical manner (in a “brute force” strategy), prescribing to the machine (to the neuronal computer) to “learn”, that is, to clone a given musical form and to reproduce it later with the desired variants. But, as Laske’s polemic shows, there is at least an argument running against this procedure, an argument that stresses the lack of invention allowed by the paradigm itself” (Vaggione, H. 1993)

In terms of applying automatic processes to weak parameters, it could be argued that perceptually this continual differentiation (Deleuze, 1998) of elements, becomes the very opposite; “*a monotonous repetition resulting from the processes own reductionist trap*” (Vaggione, 2001, p58). It is ironic that the use of automation and other GUI based operation control of parameters, especially those controlling stochastic density and pitch, create a perceptual determinism in which the percept is the grain cloud cliché. Granular *mush* or to use another analogy, smoothing the *striated space* (Deleuze, 1998, p71) (Cascone, 2009) (Villez, 2009) which is perceived as having diffused character rather than possessing a singular identity. This greatly reduces premeditated or composed salience and logically reduces individuality in the resulting work. The reader can imagine the analogy of the shapes and forms which emerge by staring into random visual patterns. It can take time and eventually something concrete emerges.

Vaggione’s stated that,

“...there is no musical composition process (instrumental, electroacoustic, or otherwise) without representational systems at work...” (Vaggione, 2001, p60, ppp2)

3.3.2 Intervention

Part of the problem is that weak parameters require more personal knowledge and intervention from the composer in order to have any effect. In an automatically controlled system, this translates to a convoluted knowledge representation. For example, an additive synthesis system requires the accurate control of many parameters in order to imitate the sound of a tenor church bell. It can do it (Harvey, 1981) but requires mass event control at the expense of computing resources. Clearly, an additive synthesis knowledge representation for modelling a variety of church bells would consist of some complexity. This narrow concentration of mass control data is at the heart of an effective implementation of real-time synthesis tools (Jaffe, 1995, p83) especially using a contemporary control protocols such as MIDI (Anon. 2013. MIDI) or wireless OSC (Anon. 2013. OSC). A strong parameter, on the other hand (the cut-off frequency in a low-pass filter), greatly reduces the above problem, however, it is less accurate and is perceptually simplistic. The result, which is immediately apparent to the listener, is that of a spectral and perceptual reduction offering anything between dullness and brightness. Consequently, this requires a simpler representation when compared to the real acoustic bell.

This proposes a dilemma in terms of implementation. How does one include real-time control of weak parameterisation in a sparse control stream, whilst maintaining spectral accuracy (Jaffe, 1995, p81)? Without a strong representational system, algorithmic approaches to composition or synthesis would seem too literal in the Vaggione aesthetic.

“It is crucial to Vaggione that the local context, singularities and the carrying out of figural work remain important. The objective is to produce,

coordinate and contrast single events, and not to govern the successive by global laws. This excludes the stochastic perspective; mass criteria are not relevant. Transformations modify singularities without losing them.” (Risset, 2005, p289)

At the heart of his technique is a craftsman’s ethos; an integrated interventionist approach to the labour of composition.

He composes the sound material itself (Risset, 2005, p289, ppp2)

“...The satisfaction of a specification as something that is not formally granted, but must be reached through action: consistency "performed" by the composer...” (Vaggione, 2001, p57)

The use of computer functions are integrated with the user and in that a homogeneity of algorithm, perception, action and feedback coexist as a unitary cybernetic system. The composer or sound designer is part of the system. They are also listeners as they perform the production bit by bit. Action as well as reaction would seem an imperative in the ontology. This stance could be seen as a direct critique of previous totalitarian approaches mentioned.

3.3.3 Notations and extreme formalisms

Vaggione’s validation is a discourse on the inconsistency of musical notation. (Vaggione, 2001, p58) Here, musical notation cannot be compared to scientific notations that refer to exacting units of measurement. Musical notation is relative to interpretation, which by default renders any strict formalism in musical composition fuzzy. Notation is therefore highly contextual and pertinent to the design of synthesis engines. The fuzziness of traditional musical notations is diametrically opposed to the use of units of measurement such as hertz. We can specify musical notes such as "do, re, mi" but the notation doesn't specify their relative place within a tuning temperament. A frequency of 100Hz is an absolute quantity. Interpreting exacting formalised rules is a cognitive contradiction in terms. It is no surprise that after some experimentation

both Boulez and Xenakis distanced themselves from such rigid compositional approaches (Schiff, D. 1995). Xenakis does present an interesting approach to the organisation of sound grains using what he calls *screens* (Xenakis, 1992, p50). These are technically explored in the following chapter. Vaggione was indirectly critical of post-Schoenberg modernist formalism in musical composition by stating that,

“Irreducibility is perhaps a key word in this context, as we are dealing with music's categories and ends. Music is not dependent on logical constructs unverified by physical experience. Composers, especially those using computers, have learned sometimes painfully that the formal rigour of a generative function does not guarantee by itself the musical coherence of a result. Music cannot be confused with (or reduced to) a formalized discipline: even if music actually uses knowledge and tools coming from formalized disciplines, formalization does not play a foundational role in regard to musical processes.” (Vaggione. H. 2001. p54-61)

3.4 Working across multiple time scales

3.4.1 Cybernetic versus generative composition

Although Vaggione's compositional approach acknowledges complexity and multitemporal intervention, he would seem to insist in retaining control of the *singularity* through continual multitemporal perceptual feedback (Vaggione, 2001, p60) rather than tendering the process out through the literal adoption of automatic compositional processes. It is not enough to have a rigid approach to crafting sound, there has to be a strong perceptual feedback and the immediate presence of the composer in controlling what he terms computable functions (Vaggione, 2001, p57). When Vaggione talks about performance, he is talking about performing direct actions which influence the final composition in the process of being recorded or produced. Vaggione objects to generative

processes without intervention because they deny the ontology of its identity and, in turn, they obfuscate its creation principle (Vaggione, 2001. p56 ppp1). Vaggione prefers to work with the *singularity* by hand, literally by placing a multiplicity of these single events manually onto the *ProTools* interface (Cornicello, 2010, p6). In order to do this, Vaggione creates a unique multitemporal syntax of sound (Risset, 2005) and composition. The sequential process of using a matrix-based timeline such as the one provided by *ProTools* and other sequencers is one familiar in popular music production and in which user interaction is fundamental to the creative process.

3.4.2 The specification of materials for the musical thesis

As discussed previously, the synthesis of sound, the “creation of the material” for the composition is part of the compositional process. Real-time sound synthesis by its very nature demands this perceptual feedback and interaction in order to alter or adjust the computable functions (parameters) that Vaggione describes. It is a mode of composition that most computer music composers have experienced and used. It is a mode which ironically would seem closer to popular music production than the classical formalist approach. Using computers in order to compose microsound music involves a different approach. The creative process is not focused on pitch, dynamic, pulse, harmony or a static model of timbre as in traditional acoustic instruments and orchestration. As Dufourt (via Risset, 2005) points out, “Music has changed scale” (Dufourt, 1991, pp.332-333).

“a continuum between microstructure and macrostructure”

The synthesis of the sound is therefore threaded within the act of composition. It is part of the *specification* of the musical *thesis*. In microsound synthesis and composition, for example, the composer works across many time scales often simultaneously encompassing the sample time, micro, sound object, meso and macro (length of a musical compositions) levels. Whilst doing this they also have to consider the real-time control of general parameters in synthesis such

as transients, spectral content, spatial locations and intensities. Apart from these there are the microsound idiomatic parameters, such as *density* (GS) or *emission* of particles per second, spatial distribution and their spectral composition. In terms of the design of parameters in multitemporal synthesis systems, the ability to respond to the shaping of timbre, through immediate perceptual feedback, would seem to be a necessary component in order to provide a strong alternative to systems which are governed by a reduced set of formal constraints.

3.4.3 The need for interventionist tools

In terms of an implementation, it would seem to suggest that what is needed are tools which allow for the seamless direct operation on inter-temporal states and maintain neutrality to any specific formal or temporal syntax.

“Operations realized at some of these levels may of course not be perceived when working directly: in order to perceive (and therefore validate) the musical results, the composer should temporarily leave micro-time, “taking the elevator” to macro-time. As a painter who works directly on a canvas must step back some distance to perceive the result of his or her action, validating it in a variety of spatial perspectives, so must the composer dealing with different time scales.” (Vaggione, 2001, p60)

Here, Vaggione is referring to the manual depositing of microsound elements within the context of his work, that is, except for the production process, it does not happen in real-time. As Vaggione points out, there are zooming tools in audio editors, however, in a real-time synthesis system it is not sufficient to be able to zoom in and out, one must be able to account for the other time levels in real-time to permit effective interaction. To approach this goal there would have to be microsound synthesis systems designed in such a way that affords simultaneous user control of the micro, sound object, meso and macro temporal levels.

Chapter 4 Review of particle synthesis. Implementations and limitations

4.1 Digital Implementations of microsonic synthesis

The methods for generating digital microsonic events has greatly expanded over the past 20 years. Since the initial pioneering experiments in GS (Roads 1978b; Truax, 1978), other techniques have emerged providing various uses for sound design and composition. Table 2 is a summary of techniques including those listed by Roads in his book *Microsound* (2001a).

Granular synthesis (GS)
Pitch-Synchronous Granular synthesis (PSGS)
Synchronous and Quasi-Synchronous Granular synthesis (SGS)
Asynchronous Granular synthesis (AGS)
Glisson synthesis (granular)
Grainlet synthesis (granular and wavelet (Morlet and Grossman 1984) combination)
Trainlet Synthesis (microsonic impulse generator)
Pulsar synthesis (PS) (granular and variable shape pulse trains)
Sonographic granular synthesis (more of a method of organizing grains)
FOF synthesis (particle formant synthesis)
FOG synthesis (particle formant synthesis using recordings of sound)
VoSim synthesis (wave-orientated formant synthesis)
Window Function synthesis (pulse streams)
Transient Drawing synthesis (drawing transients for use as particles)
Particle cloning synthesis (general particle system)
Abstract particle synthesis (mathematical and physical modelling)
Phase Alignment Synthesis (PAF) Formant particle synthesis

Table 2. Microsound synthesis techniques.

Of interest in this study are the particle formant techniques PS, FOF, VoSim and PAF. These use *pulse trains* of varying shapes to generate formant rich

timbres. GS is also mentioned in this text because of its pioneering impact on the microsound domain and serves as an introduction and illustration to some of the issues latent in microsound implementations.

4.1.1 Particle synthesis. Granular

Roads' first experiments in GS were directed towards the automation of granular synthesis generation (Roads, 1978). Granular synthesis, by its very nature, is difficult to control manually. The sheer number of parameters (Table 3) present in the simplest implementation of GS is too large (Roads, 1988).

Beginning time
Duration
Initial waveform
Waveform slope (the transition rate from a sine to a band-limited pulse wave)
Initial centre frequency
Frequency slope.
Bandwidth
Bandwidth slope
Initial grain density
Grain density slope
Initial amplitude
Amplitude slope

Table 3. Granular synthesis parameters.

4.1.1.1 The effect of particle shapes on timbre

The implementation of grains in Roads' "Introduction to Granular Synthesis" article (1988), is basic by contemporary standards. Roads uses linear particle shapes instead of the Gaussian envelopes proposed by Gabor. The reason for this is based on the practical realities of available technologies. In 1978 to 1988 the processing power required for the generation of non-linear functions, such as the Gaussian function, would have been high due to the requirement of floating point arithmetic. Mathematically, the problem is that the minima and maxima in the probability pulse never reach zero. Instead, Roads and concurrently Truax (1988), employed linear ramps to create quantised versions of the function.

It has been the general view that the angular discontinuities inherent in ramps and line-segment envelopes produce a brighter timbre from curved functions such as the Gaussian function.

There is some logic in this. A ramp unit function (Lynn and Fuerst, 1999) widely used in digital signal processes exhibits a different timbre from a sine wave. It is brighter. Angular waveforms generate brighter timbres. Conversely, it can be demonstrated that the fewer partials contained in a wave function, the smoother its shape. Theoretically, the sidebands would be more numerous in particles using linear geometries. Damián Keller and Chris Rolfe (1998) however, demonstrated that in terms of granular synthesis involving numerous grains (100's to 1000's) per second, the resulting spectrum is not very different either by the use of linear envelopes or Gaussian functions. Keller and Rolfe's study suggests that in spectral terms, there are no major perceptual differences between particle shapes created from probability, linear, sinusoid or exponential functions. This is true in asynchronous overlapped GS. In pitch-synchronous techniques however, it can be demonstrated that without overlaps, the angularity of the particle shape makes a dramatic contribution to the resulting timbre. There is a consistent relationship between the angularity of the particle shape and the brightness of the timbre.

4.1.2 The issue of strong and weak parameters

4.1.2.1 The effect of parameters on the synthesis

The real-time production of grains can be in between 1 and 5000 (if not more) per second. The load on computing processing power can be intense. However, most contemporary implementations of granular generators (and other microsonic generators), such as Audiomulch (Becnina, 2010), *Crusher X* (Stelkens, 2010) or RTGS (Wierckx, 2010), restrict the number of grains per second to nominal processing power, although with each software update and processor speed advances, this limit expands. Pulsaret, a recent (2011) implementation of PS and GS can generate up to 512. All of these systems offer relatively simple real-time control of the granular processes. Audiomulch

(Fig.9) splits the screen into three areas in which the upper half is dedicated to the generation of the sound and the lower half provides control of the generator of time.

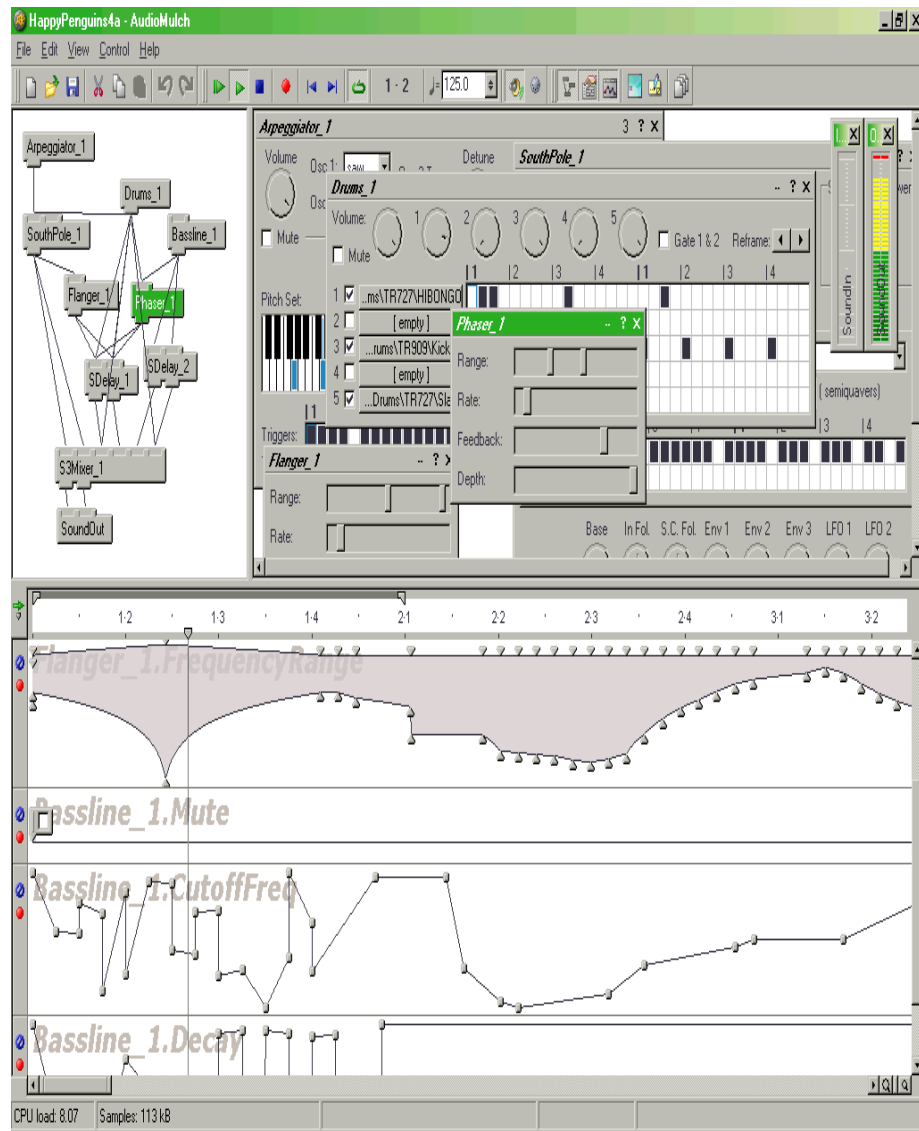


Figure 9. Audiomulch (screenshot).

The upper right half shows the unit generators (oscillators, processors) and the data flow connections between them (upper left half). The lower part of the screen shows the breakpoint function graphs associated with each parameter of the unit generators. The breakpoints can be set over time and then played as a timeline so that movement of the parameters create motion in the resulting sound. As discussed in 3.2, one of the most fundamental challenges with implementing synthesis methods with large quantities of parameters is that the

effect each individual parameter has on the resulting timbre can be weak. Jaffe (1995) suggested there is an inverse proportional relationship between the quantity of parameters and the strength of the immediate perceptual effect they have on the perceived change of timbre.

4.6.2.2 Metaparameters in granular synthesis

This parametric / strength relationship is part of the criteria proposed by David Jaffe (1995) to determine the design and implementation strength of any synthesis method. Jaffe's ten criteria are:

1 How intuitive are the parameters?
2 How perceptible are parameter changes?
3 How physical are the parameters?
4 How well behaved are the parameters?
5 How robust is the sound's Identity?
6 How efficient is the algorithm?
7 How sparse is the control stream?
8 What classes of sounds can be reproduced?
9 What is the smallest latency possible?
10 Do analysis tools exist?

Table 4. Jaffe's ten criteria.

Since his proposition was first offered, researchers have used Jaffe's criteria for the design specification of synthesis methods including Physical Modelling (Castagne and Cadoz, 2003), (Erkut, 2006), (Smith, 2004) and synthesis using cellular automata (Pearson, 2000).

Criterion 2 as seen in section 4.3 illustrates how additive synthesis parameters are in essence, weak. Envelopes or break point functions reduce this weakness by including the time domain. They act as meta-parameters affording variations simultaneously across the spectrum. The use of breakpoint function graphs in Audiomulch and others, are commonly known as *envelope generators* and are familiar tools used by many classes of electronic musicians and composers of computer music. The upper "dual" breakpoint (just below the centre of Fig.9) is of interest in that it allows a breakpoint between two limits, for example, the control of random values set between a minimum and maximum.

The image in Fig.9 shows a very basic synthesis system consisting of two sound generators. Once the user starts adding more than two or three granular streams, the control of tens of breakpoints could become unwieldy. An added issue is that including too many control events in the time domain, has an impact on criterion number seven; sparseness of the control stream. This in turn affects criterion number six; the efficiency of the system. According to Jaffe's viewpoint, adding more breakpoints is adding more parameters and weakening the system. This is especially an issue for interface communications protocols such as the Musical Instrument Digital Interface (MIDI) (Moore, 1988). MIDI has a relatively low communication bandwidth.

4.1.3 Composing and organising particles of sound in time

4.1.3.1 The Xenakis screen

In *Formalized Music*, Xenakis discusses various methods with which to organise microsounds. Of particular interest is his notion of screens (Xenakis, 1992, p50). A Xenakis screen could be seen as a special form of a Gabor Matrix, as seen in chapter two. The representation is musical and interpreted differently from the theoretical signals as represented in communications literature. The Xenakis screen is a multi-dimensional dimensional grid (ΔF frequency, ΔD density and ΔG amplitude) which exists in one moment of Δt (Fig.10)

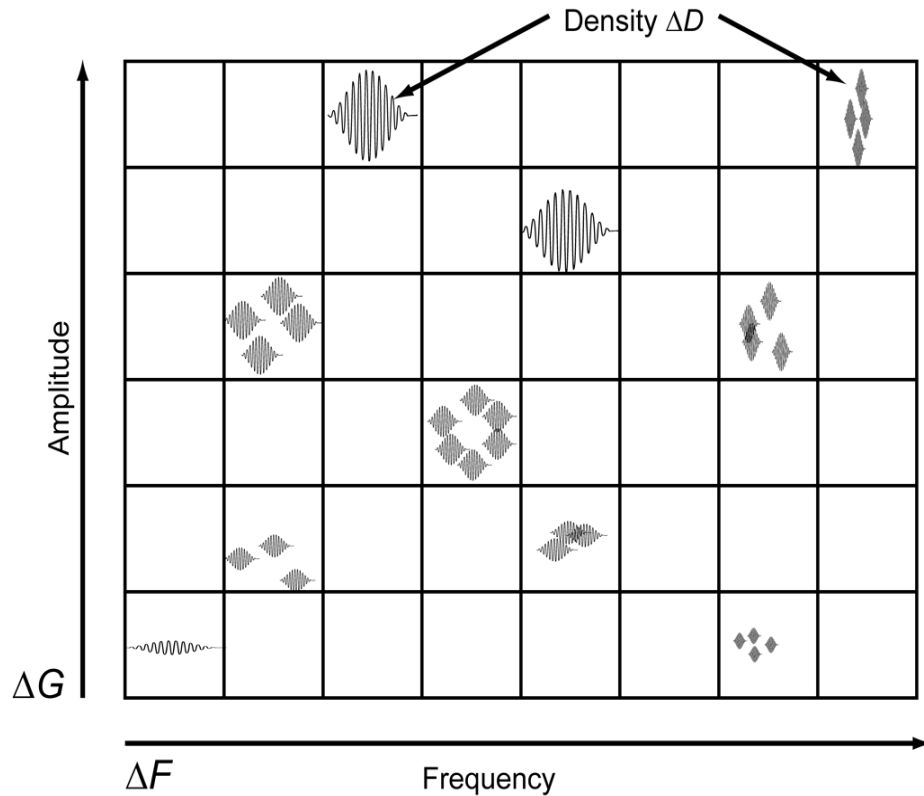


Figure 10. Xenakis Screen.

Each cell of the grid can house a silence or microsound event. Thus, each screen can have, depending on its resolution, a sparse or dense population of grains. These are referred to as *clouds*. Each occupied cell has a frequency ΔF and gain ΔG parameter. Xenakis clouds are the grouped grain topology in one screen at a given moment in Δt as in Fig.10. The use of the term granular clouds has a slightly different meaning today. It commonly refers to dense groups of particles as a function of time rather than at a given slice of time. The meaning is a lexicalisation of microsound phenomena rather than belonging to mathematical topology. Xenakis treats parameters as dimensions so that together with Δt *screens* consist of four dimensions ($\Delta t, \Delta f, \Delta g, \Delta d$). *Screens* sequenced over Δt form a *book* (Fig.11).

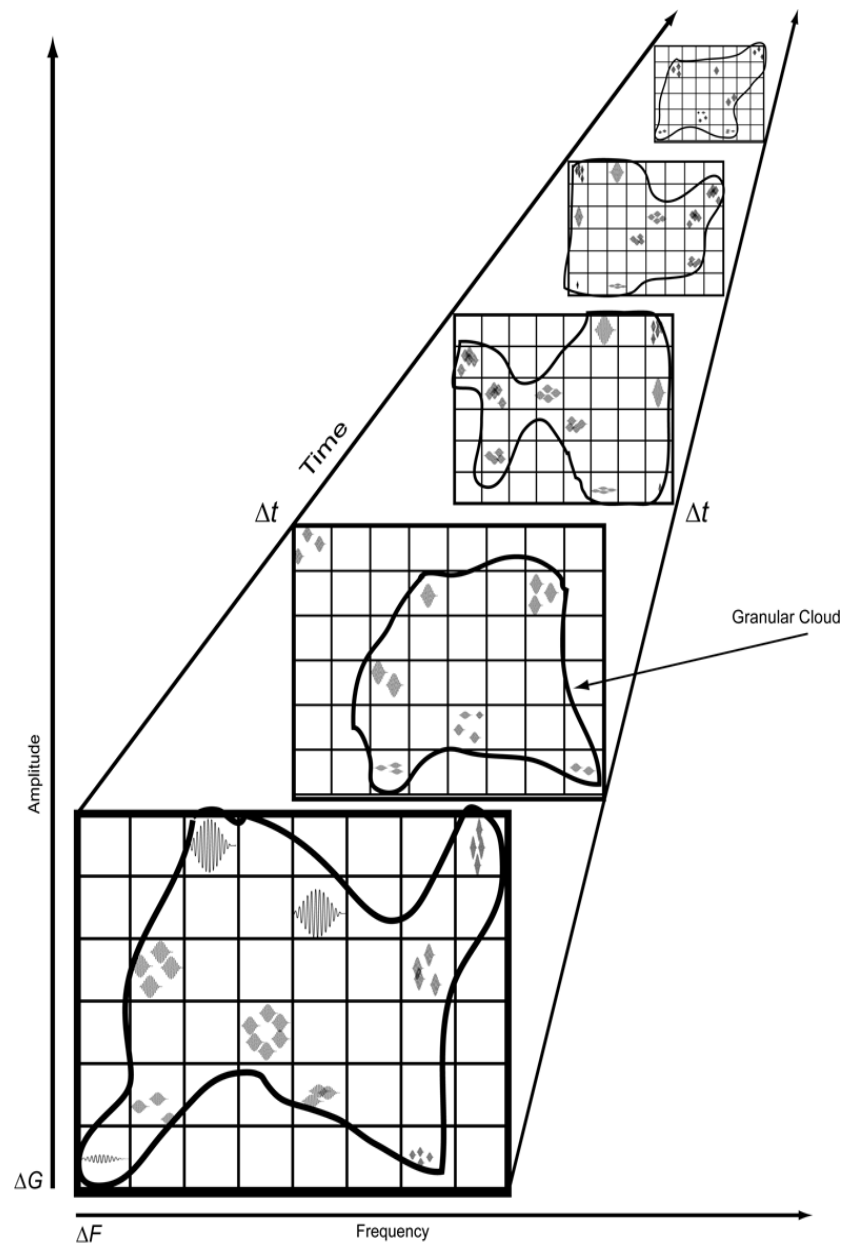


Figure 11. Screens in a book and granular clouds.

Screens and the contents within them are controlled and organised by Xenakis using a combination of set theory placed in Markovian stochastic theory (Xenakis 1992, p43) together with elements of perceptual theory (Xenakis, 1992, p47)

4.1.3.2 Temporal grid interfaces

Gridding musical elements for composition purposes is not unique to any particular musical ontology, in fact, it is a standard historic method for

organising musical time. It is a convenient way to represent the temporalisation (Deleuze, 1998) of sound events and their relationships. The use of such a representation spans from the invention of musical notation by Guido d'Arezzo (Goodall, 2001, p8) in the 11th century to contemporary digital music production. The early history of digital production of popular music uses a similar mechanism to Xenakis *screens* in which control events. These are entered in step or real-time to create musical patterns. (Fig.12)

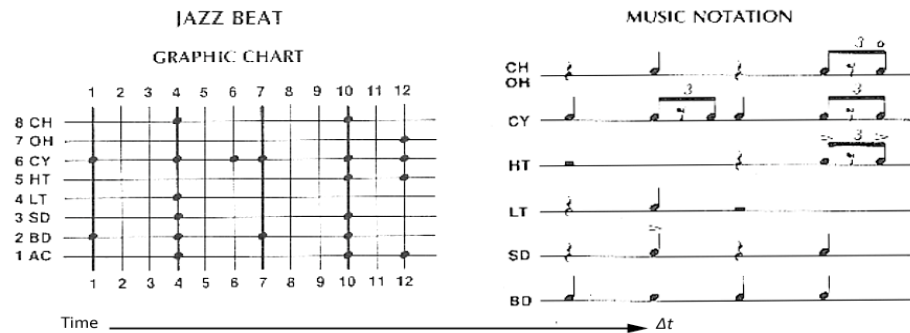


Figure 12. Drum machine grid and corresponding musical notation grid.

Patterns are then sequenced in some temporal order to produce musical events in the meso and macro domains (Feldstein, 2001)

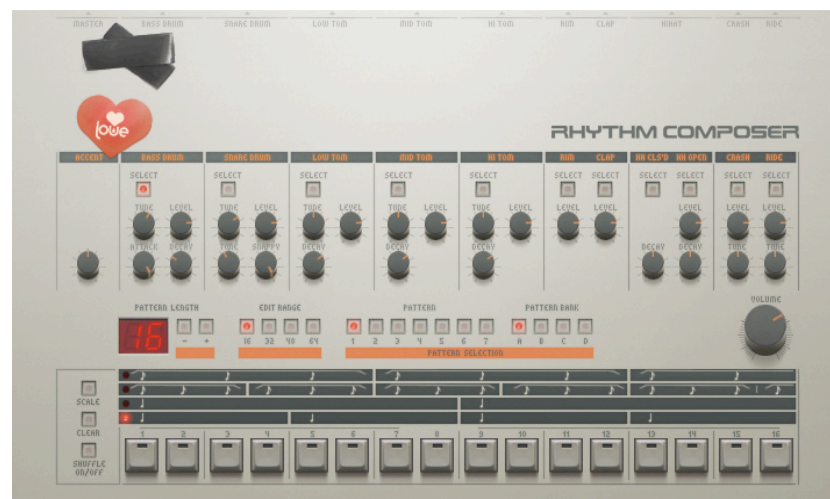


Figure 13. AudioTool *Rhythm Composer* emulation of Roland TR909 (screenshot).

Figures 13 and 14 show screen shots of the *AudioTool* (<http://www.audiotool.com/>) replicas of the Roland *TR-808* and *TR-909* drum

machines. The buttons at the bottom of the units emulate the single row time grid of the *TR-808-909* for entering rhythm events.



Figure 14. AudioTool *Rhythm Composer* emulation of Roland TR808 (screenshot).

Like many other drum programming units of this time, the interface is simple and within the limitations of its resolution (typically 16 switches, Fig.15) providing up to 64 divisions per musical measure). This permits the arbitrary programming of several layers of beats.



Figure 15. Step programming switches as a function of time Δt

Music production sequencers, such as Apple's *Logic Studio* or *Avid Protools*, extend this idea by increasing the resolution and span of the grid thus enabling very complex positioning of microtemporal events (Fig.16). Horacio Vaggione uses such an approach by placing micro events manually onto a ProTools timeline. (Cornicello, 2010)

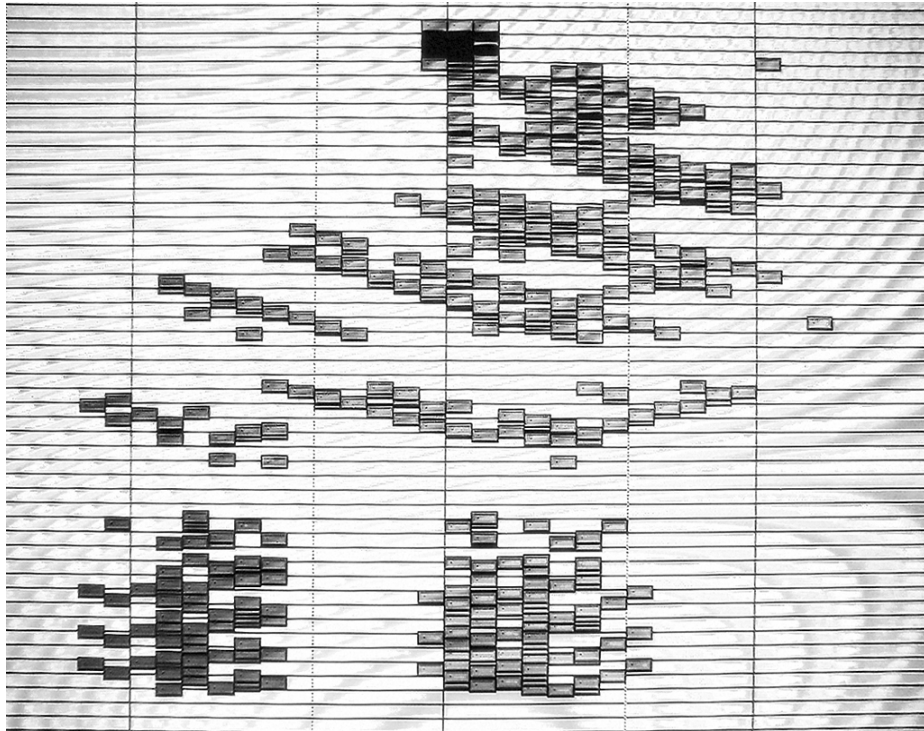


Figure 16. *Logic Studio* screen shot of the *Matrix* editor.

This method however, is limited to the grid resolution of the host sequencer, thus limiting the resolution of the grid to the maximum control event resolution. Furthermore, it limits composition to a specific mode of production not specifically tailored to the microsound idiom. It clearly does have the resolution of the micro-time scale but the focus of these environments is on the division of the beat, the sound object level and upwards; e.g. drum samples, bass lines, and guitar chords. Despite the complexity of controls present in granular synthesis however, the resultant sounds controlled by an experienced user are rich and vary from individual droplets of sine waves to textures consisting of thousands of grains with indefinite pitch or resonance. The further provision for the limiting of stochastic functions within parameters (Xenakis, 1960; Truax, 1988) incorporates ecological textures.

4.1.4 Formant Wave Functions (FOF)

Formes d'Onde Formantiques (FOF) or *Formant Wave Functions* (FWF) is a method developed by Xavier Rodet (Rodet, 1984) in the mid 1980's. It belongs

to a set of microsound techniques called Particle synthesis (Roads, 2001a, p119). There are similarities between GS and FOF synthesis. A major difference is in terms of the FOF particle shape. It is asymmetrical and unlike the characteristic GS grain shapes which tend to be canonically symmetrical (Roads, 2001a, p88-89). This single feature has much to do with the harmonic spectrum it produces. The GS method is effective in generating asynchronous sound classes such as swarms, blurred textures, and smudged timbres. It is effective at creating pseudo-chaotic timbres or the expansion and contraction of sampled materials. FOF differs in that it generates clean and precise timbres, offering an alternative sound design approach. FOF has been used to successfully model the human voice. An important characteristic of FOF synthesis is the production of formant rich spectra. Traditionally, electronic formant production had been produced using a source filter combination. The source being an impulse excitation that is then filtered using a bandpass filter with its centre frequency tuned to the desired formant (Rodet, 1984, p9). To vary all the parameters at a fundamental audio rate requires intense processing using the filter method (Rodet, Potard, Barrière, 1984, p24). FOF synthesis elegantly produces a formant using a unique and relatively simple formula with a limited number of functions.

4.1.4.1 The FOF envelope

In its simplest form, each FOF particle envelope is the product of a sine wave multiplied by an exponential decay (Fig.17). This produces an asymmetrical shape not dissimilar to that of a Poisson distribution function. Each FOF produces energy around a centre frequency, creating a spectral peak. The contents of the envelope are a sine wave whose phase is re-aligned on each period to eliminate A.M. inharmonic artefacts.

Envelope $Ae^{-\alpha[(t-c)^2/t]}$

where $\alpha/\pi = 50 \text{ Hz}$

and $c = 1.8 \text{ msec.}$

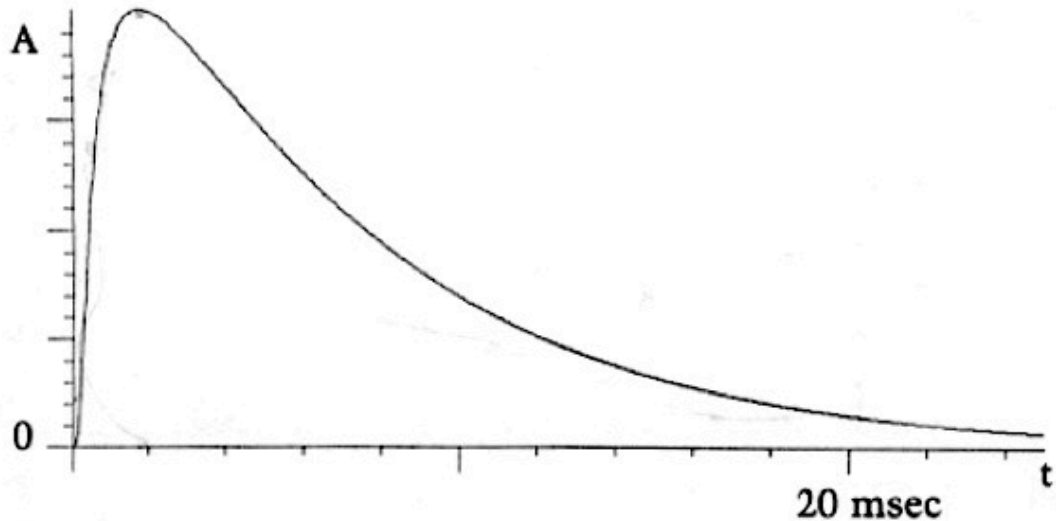


Figure 17. Elementary FOF particle (Rodet, 1984, p11).

For reasons of legibility the equation in Fig.17 reads as follows:

$$\text{Envelope } Ae^{-\alpha} \left[(t-c)^2 / t \right]$$

Rodet sees the advantages of using varying functions with which to model different sounds.

“For modelling arbitrary spectra one can use other types of envelopes $A(t)$ chosen accordingly to their spectra” (Rodet, 1984, p11)

The shape of each FOF particle determines the content and shape of the formant region being synthesised. Rodet chose a variation of the above function in order to enhance the flexibility of the original formula. In this case the dampened exponential is controlled by an arbitrary number of samples thus enabling control of the rise (attack) and decay of the particle (Rodet, 1984, p11), enabling the control of the width of the particle and the resulting formant spectrum.

4.1.4.2 FOF layers

By summing several FOF layers, Rodet, Potard and Barrière demonstrated the possibility of synthesising very realistic acoustic emulations of both voices and instruments (Rodet, Potard, Barrière, 1984, p21).

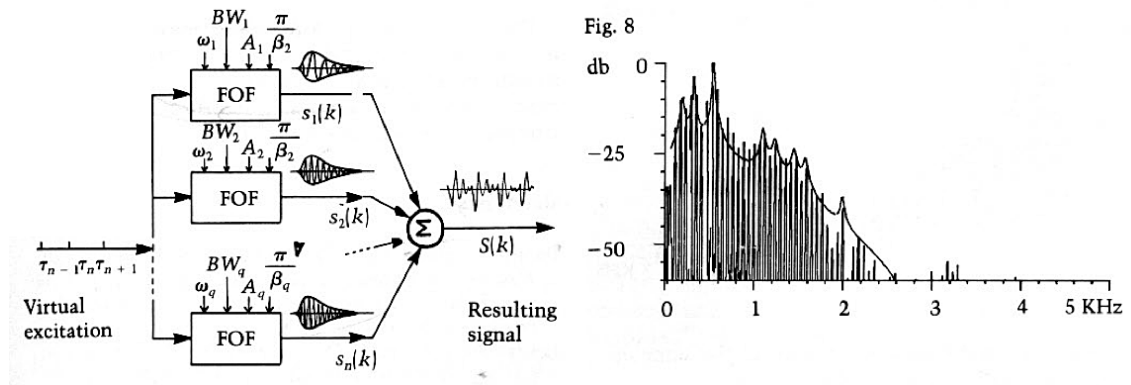


Figure 18. Summation of FOF streams and resulting spectrum (Rodet, 1984, p13).

The parameters (Table 5) required to control a FOF stream are fairly concise with regards generating each individual particle (Roads, Potard and Barrière, 1984, p20).

FOF particle Parameters
Fundamental Frequency
FOF attack
FOF decay
Centre frequency of Formant (resonant peak)
Bandwidth of Particle (control of width of particle)
Amplitude
Skirt width
Initial Phase

Table 5. FOF synthesis parameters.

Collective FOF layers, however, can present as many parameters as GS.

During the first implementation of FOF synthesis at IRCAM in Paris, a special program called *Chant* was designed to control the FOF stream for the production of complex synthesis of the singing voice (Rodet, Potard and Barrière, 1984). One of the pragmatic functions of *Chant* is to provide “strong” (Jaffe, 1995) parameter control. It is possible however, to customise the level of

control by the user using programming languages to create rules or algorithms which *Chant* stores as parameter files.

4.1.4.3 The different faces of FOF

Like GS, FOF synthesis can be cast into multiple categories of synthesis techniques. Often, algorithmic techniques such as FOF and GS can be modelled using simpler generalised algorithms. If analysed in enough detail this can be said of most techniques when deconstructed into simpler DSP states:

Physical Modelling

The combined control of fundamental frequency, using random jitter, together with the FOF particle shape, which is similar to the physical glottal shape, is an elementary type of physical model (Sundberg, 1978).

“it is notable that the parameters of the model are particularly representative from a perceptual point of view” (Rodet, 1984, p10)

Additive synthesis

The summation of several FOF streams containing strong formant regions is a special case of partial addition in order to build a complex spectrum.

Subtractive

“According to this model, speech signal is the response of a time-varying filter (the vocal tract) to an excitation function (the vocal cords' movement). If the excitation function is assumed to be a pulse train, the waveform resulting from speech can be regarded as a concatenation of filter impulse responses. Furthermore, if the filter's impulse response is assumed to be finite and shorter than the period of excitation (the pitch period), then cutting a sound source's waveform at the instances of excitation effectively isolates each response within a single excerpt.” (Behles et al, 1998)

Waveshaping and amplitude modulation

The multiplication of the exponentially dampening function by a sine wave (giving the formant centre frequency and local envelope) could make FOF a type of waveshaping (Le Brun, 1978) or AM synthesis technique.

4.1.4.4 Octavation and its limitations

In the FOF stream, it is possible to fade in and out every other particle to create the illusion that the fundamental frequency is changing in octaves. This is achieved by pre-calculating the fundamental frequency required (Rodet, 1984). The fundamental is then built from consecutive layers of sparser streams of FOF particles and summed into one stream. Density is dependent on how many layers are present. The more layers, the higher the fundamental frequency.

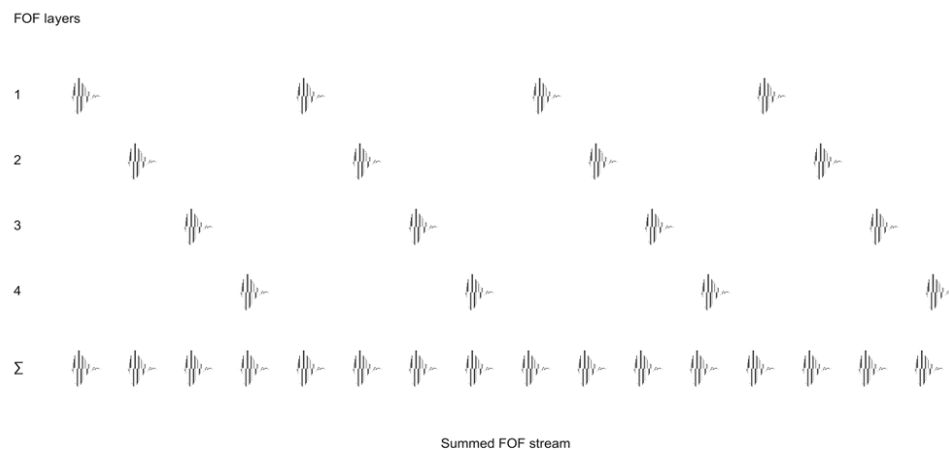


Figure 19. FOF layers.

The advantage of this technique is that one can fade layers in and out of the stream (Fig.19) creating the illusion that the fundamental frequency is morphing between octaves and without the intervening musical intervals. Because the formants in the spectrum are preserved, it affords the production of dramatic gender morphing vocal timbres and the dynamic resizing of perceived acoustic objects. Octavation is the mechanism by which Michael Clarke creates the evaporation and coalition of FOF events in Mälarsång. The dynamic movement between temporal boundaries is a key identity (Jaffe, 1995) in the sound world created using FOF generators.

Fig.19 is in essence a time matrix illustrating how the separate FOF layers are distributed in time. The limitation however, is that the user does not have control of the individual placement of each FOF particle in time. It is limited to the fading in and out of consecutive events.

4.1.5 VoSim

VoSim stands for VOice SIMulation (Kaegi, 1973 via Tempelaars, 1977) and after analogue impulse generators, is one of the earliest examples of microsonic synthesis. As is the case with FOF, VoSim produces strong formant content and was conceived for modelling the human voice. Very early implementations were developed using analogue wave based oscillators. The VoSim particle is characterised by a \sin^2 (Hanning windows) oscillator shaped by staircase shaped decaying envelope. (Tempelaars and Scherpenisse, 1976)

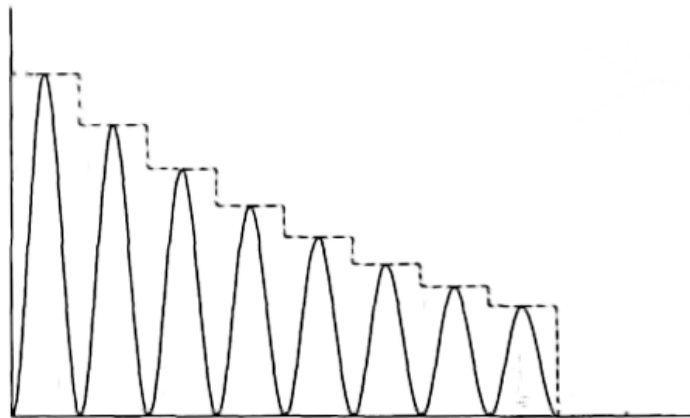


Figure 20. Consecutive VoSim \sin^2 pulses shaped by a staircase envelope.

The staircase shape of the VoSim particle (Fig.20) came about through an interesting approach to efficiently implementing an exponential decay. This is an attribute exhibited by many sounds and employed in FOF synthesis. A major issue posed by oscillator-based pitch synchronous microsound systems is the disconnection of the static particle decay length. The lower the fundamental frequency, the longer the decay. This is of limited use in comprehensively modelling certain sound sources. As demonstrated in a contemporary

implementation by Rob Hordjyck (2010) however, VoSim is very able to produce smooth rich formant based spectrums thanks to the hard synchronisation technique (Brandt, 2001).

4.1.5.1 Hard Synchronisation

Hard synchronisation involves the use of two oscillators in which one, the master, resets to zero the phase of a slave at the fundamental period of the master. It is a technique that has been used extensively and expressively in analogue subtractive synthesis (Fig.21).

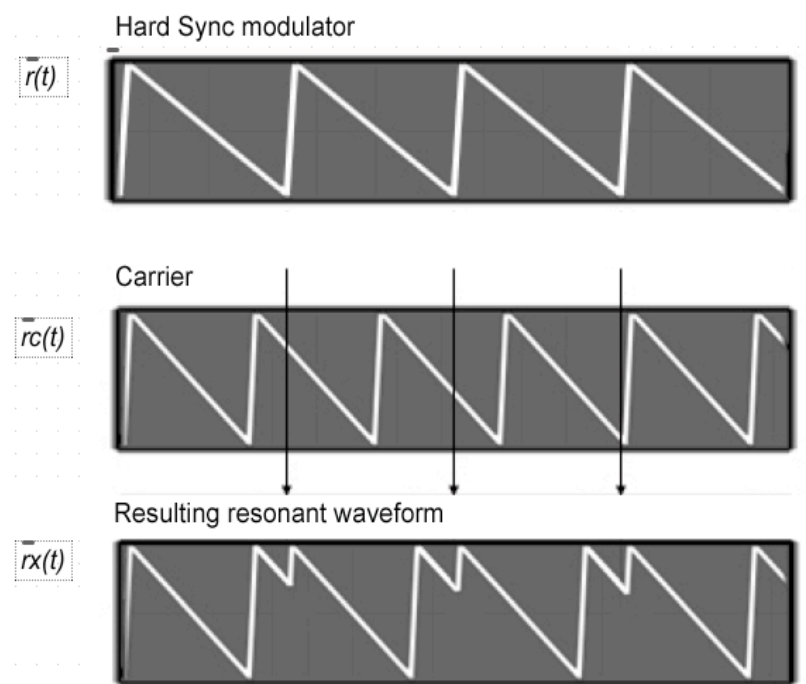


Figure 21. Hard synchronisation.

One of the primary uses of *HS* is the production of strong spectral sidebands (frequency components above and below the carriers fundamental) which are harmonically related as there is a simple integer relationship between the fundamental of the carrier and modulator spectral content. In order to control these sidebands hard-sync is a simple and CPU cost-effective solution. Even if the carrier oscillator's frequency is continuously varying, it will relate harmonically to the modulators fundamental frequency. The result is a timbre containing a strong resonant peak, an essential feature of microsonic particle formant synthesis techniques. The impractical side is that it excludes the

generation of strong inharmonic spectra and is prone to aliasing (Brandt, E. 2001).

4.1.6 Pulsar synthesis

Pulsar synthesis (PS) like VoSim is a hybrid wave-microsound method for synthesising tones. PS, a digital technique, is based on analogue impulse generators such as those used by Stockhausen and Koenig (Roads, 1978a). According to Roads, the technique originated from his fascination and preference for an old vintage electronic sound world (Cornicello, 2010). At the heart of the pulsar generator is a wave based microsound generator similar to the early electro-acoustic music impulse generators described earlier (Roads, 2001b, p134). At the centre of the implementation is pulse-width modulation (PWM), which like hard-sync, is another technique borrowed from early synthesis methods (Roads, 2001b, p135). PWM is a technique in which the symmetry of a periodic waveform can be changed arbitrarily (Fig.22).

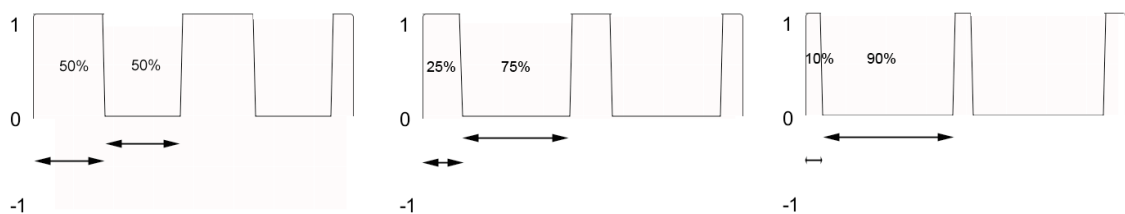


Figure 22. Example of Pulse Width Modulation and duty cycles of 50, 75 and 90%.

The result of this is the strong phase shifting of the spectral content. In the microsonic context it permits a change of duration of a particle independently of the fundamental frequency. PS offers the advantages of a great number of waveforms called *pulsaret* shapes, which is a departure from classic impulse generators. Like VoSim and FOF (Roads, 2001b, p135), the technique offers particle shapes with an exponential decay. Rodet's initial FOF implementation however, is somewhat more complex in that the attack and decay can be

independently adjusted with the addition of an independent formant bandwidth parameter (Rodet, 1984, p11). This makes it more flexible in a sound modelling capacity (Jaffe, 1995, p79)

4.1.6.1 Pulsar masking

In the pulsar generator, synchronised pulse masks (Roads, 2001b) can be generated to break up the pulsaret stream. Unlike octaviation, this allows the generation of different types of regular microsonic patterns, albeit with some caveats. *Burst masking* generates periodic bursts of pulsars similar to those emitted by early impulse generators (Roads, 2001b, p138). *Channel masking* produces concurrent channels of pulsar streams. *Stochastic-masking* produces stochastic masks in the stream. Increasing the stochastic mask parameter breaks up the regularity of the pulsar pattern. Periods of activity and rest can be programmed in whole number of pulses; e.g. users can program 4 on, 3 off or 8 on, 8 off. Depending on the emission rate (fundamental frequency), users can create interesting rhythms or amplitude modulation effects (Fig.23).

Unfortunately, there is no provision for arbitrary configurations in which one could single out specific particles or groups of them in a stream. Like FOF, it cannot produce an arbitrary pattern of particles or the placement of individual particles in time, thus limiting its use to repetitive patterns mixed with the stochastic inclusion or exclusion of particles.

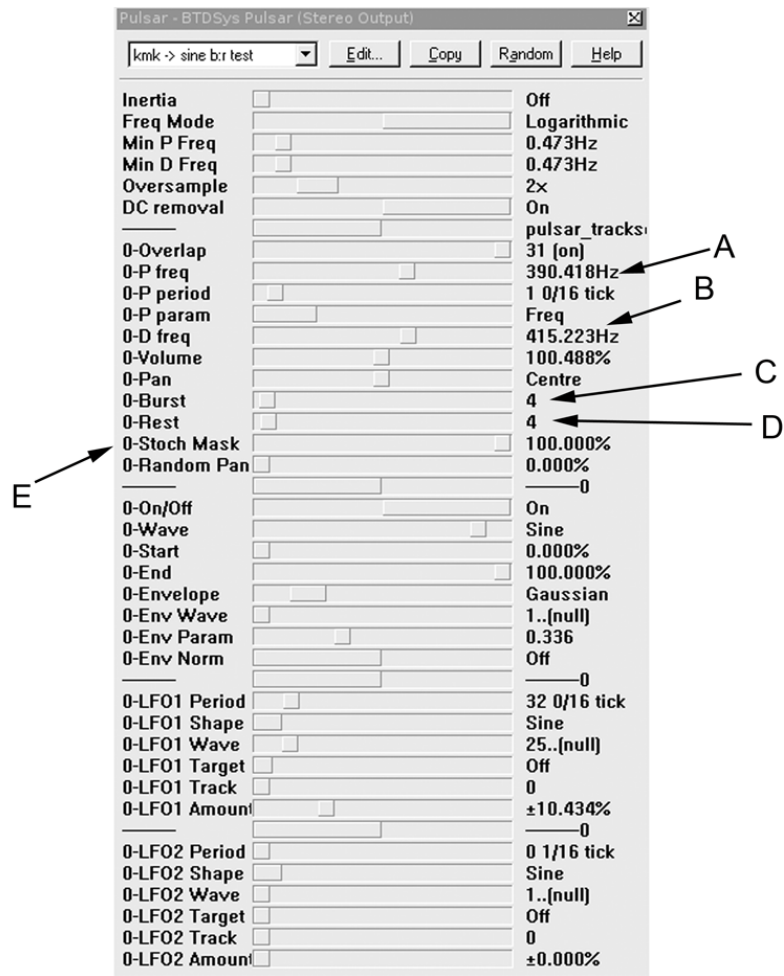


Figure 23. Pulsar synthesis implementation of Jeskola Buzz Machines.

The major limitation of Pulsar synthesis is the relatively few existing implementations. The *Pulsar Generator* as described in the original PS paper (Roads, 2001b), is implemented on an Apple MAC OS 9, limiting its use to very few users. Initially, it was employed in this research for basic experimentation however, it was found to be functionally unreliable. This is not necessarily the fault of the implementation. It could have been machine-specific issues. However, without a wider platform implementation, the system remains esoteric and personal to its creator. Only one other faithful implementation has been found in the modular synthesis environment; *Buzz Machines*. This implementation has since become a central apparatus in this investigation (Anon, 2014. BTDSys). A more recent program called *EmmisionControl* created by Roads in conjunction with David Thall, though promising, also suffers from

the limited platform restriction in that it is only available for non-Intel CPU Macintoshes (Cornicello, 2010, p5).

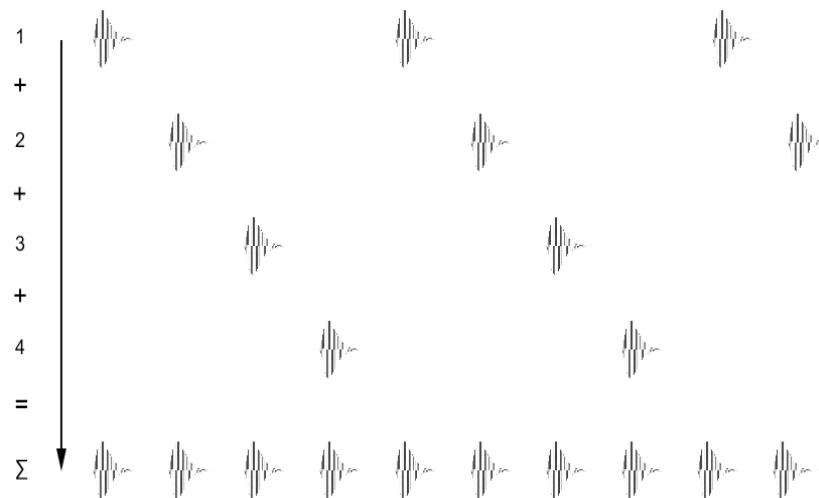
4.1.7 PAF

PAF stands for *Phase Aligned Formant synthesis*. PAF is not conceived as a microsonic technique per-se. PAF like VoSim, lacks the essential property of independent particle duration in respect to the fundamental frequency. Like FOF synthesis however, it is used effectively in order to physically model the formants of the singing voice and therefore, contains several characteristics which are not dissimilar to the other microsound techniques discussed.

All microsound techniques use special elaborations of amplitude modulation (AM). PAF uses Gaussian shaped pulse trains to create specific formant spreads (Puckette, 2003, p153). Like VoSim and PS, hard and soft sync (see below) is used to force pitch synchronisation to overcome the inharmonic sidebands present in elementary AM. Hard synchronisation is the *Achilles heel* of these techniques. Vocal formants can be modelled fairly accurately in terms of frequency, amplitude phase and bandwidth but they exclude the possibility of inharmonic resonances such as those present in some voice types (Sundberg, 1978). FOF avoids this by using overlap-sync (Behles 1998, p46). In this investigation, this type of particle generation is referred to as *additive microsound* (Villez, 2009) (Fig.24); additive because generated layers are summed in order to create a composite stream. It uses overlapping particles to give smooth frequency gradients even at high fundamentals.

Additive Microsound. FOF layers added to create a formant stream

4 particle layers



Subtractive Microsound. A pulsaret train is subtracted from using pulse masks to highlight specific pattern of particles.

Pulse train from a periodic function

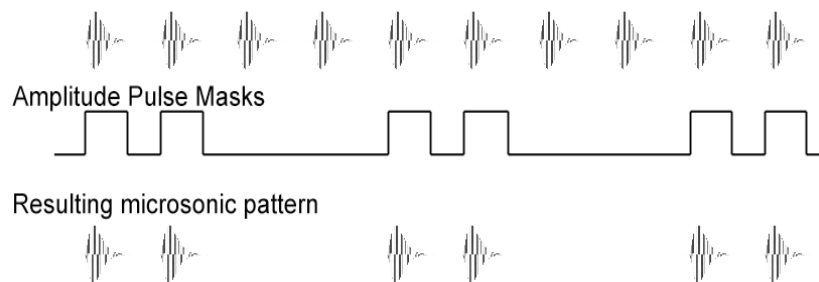


Figure 24. Additive and subtractive microsound.

PAF has no inherent provision for the organisation of particles in the sound object or meso-time levels.

4.1.8 Other methods

Other methods repackaging many of the techniques described above. An example is the Partikkel opcode for Csound (Brandtsegg, 2011), which is a "catch-all" microsound generator with over 48 parameters per stream. *Glisson* Synthesis (Roads, 2001a, p121) is similar to GS in which the local signal frequency of a particle can glide independently to the fundamental frequency.

Grainlet synthesis (Roads, 2001a, p121) combines GS with Wavelet theory. This offers wave synthesis-like properties in which the duration of particles is linked to the emission rate. Many of these methods are hybrids, cross-references or derivations from one or another of the techniques described above. Many of the algorithms presented in this study contain similar standard generalised synthesis or processing techniques such as AM combined with hard synchronisation. An area that is crossing into the microsound arena is concatenative Sound Synthesis (CSS). This data driven technique is used mainly for matching existing sounds. Using a database of source recordings the CSS algorithm tries to find units (segments) in the database which closely match the sound to be generated (Schwartz, 2006, p3).

Chapter 5 Research focus. Overcoming existing PS limitations

5.1 Differentiation of particle organising features

In the previous chapter, it was rationalised that one of the challenges faced in the generation and composition of particles is the ease with which they can be manually organised into sound objects or meso-time structures etc. (4.1.3). The potential problem with these designs is that the connection between synthesis and multitemporal composition (Risset, 2005) seems to not be implicit in the original specifications describing the technique.

It is concluded that the reason for this omission is that microsound synthesis is still relatively young when compared to wave-orientated synthesisers, which use continuous waveforms for subtractive and additive synthesis. Composition is treated as a separate creative process from the process of synthesis (instruments) and is limited to those two domains.

Microsound synthesis integrates those two domains with a third, the time domain. In compositional terms and as seen in chapter 2, this domain is pluralised into multiple scales ranging from the microsound to the *macro-time* levels (see Table 1). As observed in 3.2 and 4.1.3, the organising principle in GS has generally been based on Xenakian formalisms and focused predominantly on stochastic techniques. FOF implementations afford some specific control but are limited to the consecutive inclusion or exclusion of particles rather than individual or groupings of micro events. FOF was not conceived for composition in the time domain even though the technique is easily interpreted in the time domain (Clarke, 1996). Few systems exist for

automatic microsound composition; e.g. Csound, *Diphone* and *FTM for Max*. Some however (except for generalised audio compositional software like Avid's *Pro-tools* or *Logic Audio*), allow for the Vaggione style hand crafted composition in which microsound events are placed point by point across multiple time-scales and with microsound specific tools. Csound, as well as other text based computer music environments facilitate the Vaggione style of composition but its lack of the cybernetic aspect discussed in 3.1 makes the process slow and daunting to many users. The user is forced to *prematurely commit* (Green, 1989) some amount of value settings or calculations prior to run-time.

5.2 The originating conceptual model translates into different mental models to different developers

Another significant disparity exists in the interpretation of the implicit specifications, that is, the original papers that describe the synthesis process. Pulsar synthesis, for example, exists in at least four different usable interpretations (see Table 6).

Synthesiser	Host	Platform
PulsarBTD	Buzz	Win
Pulsaret	Live	Win/Mac
iPulsaret		iOs
Nuklear	VST	Win/Mac

Table 6. Usable PS Implementations.

Roads' conceptual model of *pulse trains* is quite specific (Roads, 2001b). It is an extended pulse width modulation technique applied in the amplitude and time domains as opposed to the spectral domains. PulsarBTD is implemented with most of the originating specification ideas. Its channel masking is stochastic unlike the burst masking in which some intervention is possible. Pulsaret is a different interpretation. So different in fact, as to seem like an extended GS environment with a very flexible set of particle shapes. iPulsaret, a recent addition adds *Trainlet* and *Glisson* synthesis but no *D-frequency* from Roads' original conception. Glisson synthesis extends a modulation method already

employed in VoSim; i.e. the technique of formant gliding is an elementary form of Glisson. It could be easily argued that FOF and *FOG* (Clarke, M. 1999) have both used this technique much earlier. VoSim's *pulseCount* parameter could be interpreted as an early *Trainlet* prototype. The original FOF paper (Rodet, 1984) does not refer whatsoever to VoSim even though it precedes it by a decade. VoSim is in effect a true formant function generator (FOF). It is as if some of the subtleties of the original papers, which first describe these particle synthesis algorithms, have not been understood, detected or simply ignored. It is reasoned that there is an explanation for this significant discrepancy that is elaborated below.

5.3 Idiosyncratic limitations

The limitations observed in chapter 4 are not common to all the particle systems. Instead, they are spread out as design idiosyncrasies in each system. Most of them are very powerful features but exclusive to one system or another that creates a significant distance between their design and functional identities. It is possible that part of the problem lies in the individuation of plugin and platform standards (or lack of them) used by the synthesis systems and host environments; VST, AU, M4L, OsX and Windows. This make it difficult to use them together in real-time so that users can easily create microsound music.

It is clear from the available literature (Harvey et al, 1985; Sundberg, 1989; Clarke, 1987) that they all produce complex “compelling” and differing phenomenologies. It is important for the composer and sound designer to have a variety of such tools. The learning curve, however, means that in spite of the sound design potential, the user is faced with learning many radically different mental models of interfaces together with a lot of disparate hardware idiosyncrasies (Johnson, J. Henderson, 2012, p9) and notations (Blackwell et al, 2001). This is an irony because one of the original promises of the general-purpose desktop computer was to liberate the user from all the differing incompatible hardware and standards. No longer would the user have to "paint the walls through a letter box" as was commonly said about the Yamaha *DX7*'s

LCD display. Though it is possible to conceive of situations in which such restrictions can lead to creative explorations, in commercial music and sound design environments the opposite is true because the production of media assets are constrained by time and economical production practices. In these situations, Incompatible hardware and standards can be a hindrance.

FOF for example, is freely available in Csound but not, at the time of writing, as a plugin for *Intel* based *Ableton Live* or *Logic*. The implementation of PS is hosted in *Buzz Machines* exclusively on Microsoft *Windows* or alternatively a version for Ableton. Csound's FOF can be used with the *Open Music Chant* libraries or using *Csound4Live*. However, it is cumbersome and is the technical equivalent of a *Russian doll*, in other words, a virtual machine (Csound) sitting in a virtual machine (Max4Live) in the host. Its CPU and memory usage would need to be measured to ascertain its practical role in real-time operation.

This situation means that cutting edge synthesis tools such as the ones presented in the previous chapter, are difficult to use together in a project and require not only several pieces of software but also need different and often incompatible hardware. With the relatively recent arrival of tablet computer platforms this situation has worsened. In order to use a standard MIDI hardware for backwards compatibility, the user has to buy system specific hardware to get a five pin MIDI connector to talk to the tablet. The result of this general critique is one which points to a lack of immediacy and lack of standardisation in the way in which design information is interpreted by the developer and presented to the user. The developer creates a conceptual model based on his or her own mental model of the original specs. In turn, the developer presents a unique mental model to the user. This generalisation above can be divided into specific types of issues.

5.4 Feature parity. Differentiation

This sea of interpretations creates a wide set of modalities of particle formant synthesis in which some core features are present in one system but “half baked” or missing in others. The core parameters are almost identical. It is not that they cannot be present but rather that the original design concept did not include them; e.g. the multitemporal compositional aspect has not been considered. The differentiating distance is wide enough because of the conceptual model. There is no reason, for example, except for narrowing a sound phenomenology, why FOF could not have as many particle shapes as Pulsar. Viewed in this way, one could argue that Rodet might have inadvertently limited its capabilities.

	Pulsaret	Pulsarbtd	Nuklear	FOF	VoSim	Partikkel	PAF	Max	Min	Parameter Difference
Formant Freq	1	1	1	1	1	1	1			
Particle Shape	22	7	4	1	1	1	1	22	1	21
Form Wave	22	22	26	1	1	4	1	26	1	25
Particle Duration	1	1	0	1	1	1	0	1	0	1
Overlaps	1	1	1	1	0	1	0	1	1	0
Vibrato	1	1	1	0	0	1	1	1	0	1
Jitter	1	1	1	0	0	1	0	1	0	1
Trainlets	0	0	0	0	1	1	0	0	0	0
Gliss	1	1	1	1	1	1	1	1	1	0
Organise P	Stats - Arb	Mask - stat	Mask	Oct	non	Stat - Arb		0	0	0
Freq/Time	1	1	1	1	1	1	1	1	1	0
Scope of masks	64 Arb		8	1	0 Arb		0	64	0	64

Figure 25. Some core parameter differences between systems.

A quick analysis can reveal salient differences between core parameters. If the reader views Fig.25 they will notice that Hamburg Nuklear has no less than 26 formant waves from which to choose from. The highest number out of all the systems. However, it only has 4 particle shapes compared to 22 in Pulsaret. FOF's particle shape was conceived in order to model the vocal glottal stop and thus is limited to one. This is a uniquely identifying attribute of FOF synthesis. Its limitation however, is purely conceptual. If we ignore this limitation and the differing ways in which FOF and PS organise particles, FOF can synthesise timbres similar to those which Pulsar synthesis can and vice-versa. This is something which Clarke demonstrated and which transformed the concept of

FOF synthesis from a frequency domain generator (Rodet, 1984) into a time domain synthesiser (Clarke, 1996).

5.5 Concept and context of difference

Measuring parameter difference by the quantity of its elements might be useful for getting an idea of superficial capability, however, it does not tell us if the difference of particle shapes implemented are actually contextually useful. The context in the difference is important. It does not mean we cannot have more particle shapes, it just means we have to be measured in deciding which ones are crucial to the phenomenological purpose of the tool. An example is to afford the user with a feature in which they can upload pre-analysed shapes from other musical instruments.

To exasperate this situation, developers conceive of modalities of features which distance them even further; i.e. adding Glisson techniques to Pulsar synthesis. A good feature in one system (*octaviation* in FOF) is not available with the same flexibility of interpolation in any of the other systems. There might exist a similar feature but it is differentiated enough as to make it difficult to re-create the same effect. In some cases such as VoSim, there is no implementation that permits the composition or organisation of particles at an individual level. Microsound textures possible in Pulsar synthesis are not possible in VoSim synthesis. The conclusion is drawn that this design and implementation plurality keeps the techniques at the fringe of production as they do not sit comfortably within aesthetics of modern “real-time” production host environments. It would seem that Pulsaret and *Hadron* (Partikkel) and to a limited extent, Nuklear, are the exception. It is not surprising that this state of play has made the advancement of composition in the flutter region somewhat slow to emerge (Miranda, 2002).

5.6 Too many features

In chapter 3, it was discussed that a synthesis system with too many parameters tends to weaken the effectiveness of the synthesis system (Jaffe, 1995). At first glance, it could be argued that the sheer number of parameters in the Csound opcode Partikkel creates an environment that is cumbersome to operate because it offers every microsound feature possible but has the opposite effect on productivity because it is too cumbersome to operate manually in the interventionist manner, especially when many of its parameters have dependencies on other parameters; e.g. loading tables which require starting and stopping the Csound DSP engine. Using automatic compositional strategies however, does offer some possibilities using neural networks or genetic algorithms to breed new parameter combinations but in the case of the current implementation of Partikkel and for the reasons discussed above it would need a more flexible real-time engine which excludes stopping and starting the audio engine in order to change table elements and a flexible system to manage the large quantity of parameter dependencies in the current version.

5.7 Standard glue parameters and variations

The essence of the observation is that the particle formant synthesisers referred to, have common core synthesis elements that are more or less standard (Table 7):

particle shape
formant wave
fundamental frequency
particle duration
formant bandwidth
overlaps
organisation

Table 7. Particle formant *glue* parameters.

These however, are either slightly different in capability or have elementary features missing (number of particle shapes, organising principles etc.). For future ease of discussion, this investigation refers to these parameters as particle formant *glue* parameters. The main differentiation noted is in the particle shapes and the multitemporal techniques for particle organisation in the meso-time. Differences can also be detected in systems using the same organising principles; e.g. pulse masking pattern sizes in Nuklear (8), PulsarBTD (arbitrary) and Pulsaret (64).

5.8 Missing core features

Some of the systems like Pulsaret, Nuklear, PulsarBTD offer vibrato and jitter. Others like Csound's FOF, which ironically conceptually models singing, lack vibrato and pitch jitter as core parameters.

5.9 Lack of standardisation across glue parameters

It is proposed that this lack of standardisation, in the centre part of the compositional process in microsound, ties the user into learning disparate idiosyncrasies in each system; the burst mask interface in PulsarBTD is different in Nuklear and radically different (snap sequencer) in Pulsaret.

5.10 Developers

It could be rationalised that the general unevenness observed in the glue parameters and aspects of the compositional functionality, has a lot to do with the developmental framework from which each system emerged.

Generally speaking, the synthesis artefacts described in this research are created by small teams of developers, researchers, hobbyists etc. More often than not, they work alone as individual developers. Some like Curtis Roads are researchers associated with academic institutions who have developed the techniques and tools for research or creative use and who have published the original specification (implicit specification) from which others, Alessandro Petrolati of the Pulsaret software (Petrolati, 2013) or Ed Powley from *BTD Sys*, have created their own interpretations (explicit specification) of techniques like Pulsar synthesis. The interesting outcome from this is that the interpretation of the implicit specifications has lead to a variant of the original technique rather than an exact copy. This raises many of the issues discussed above but it can also be positive as variation enables a wider set of uses and users. As the reader will observe in subsequent chapters, Pulsaret and PulsarBTD are quite different synthesis engines though they share common interface notations, core parameters and features.

It is hypothesised that the reason for this is because independent developers and researchers do not necessarily follow a strict, commercial design framework, which is informed by a set of analysed requirements. Furthermore, the commercial products design process weighs more towards either functionality or usability. Engineering departments generally follow rigorous engineering methods that insure the feasibility and quality of the technology. Similarly, marketing, sales, and other business units, follow their own well-established methods for ensuring the commercial viability of new products. What is missing is a repeatable, analytical process for transforming an understanding of users, into products that both meet users needs and excite their imaginations (Cooper, et al, 2003 p8).

Independent developers instead, tend to apply *opportunistic strategies* (Green and Blackwell, 1998) to a process, often through experimentation.

It might be that a researcher needs a piece of software for the purpose of testing a hypothesis. Alternatively, it might be the result of a patch created to highlight a question on an online forum. In the case of FOF (Clarke, 1996) and other opcodes in Csound, the design is the result of an individual academic

researcher who created the opcode as a secondary support to illustrate or validate aspects of their research. There will be endless variations on this theme.

It is concluded that it is one of the principle reasons why the interpretations are so different. It is hypothesised that an added reason for the variation of interpretation of core synthesis parameters in PS, is that the design has been informed heuristically (through ‘well known results’), rather than through any empirically driven design framework based on functional and usability requirements. Unfortunately, frameworks with the kind of structure, which might inform the unique nuances in particle synthesiser design, do not exist at the time of writing. The possible reason they do not exist is because the subject domain is comparatively young and still quite niche. It is *boutique research*. It is speculated from this that this electronic music sub-domain has the capability to mature aspects of existing artefacts into new and more powerful sound design tools. There are many papers on the theoretical framework of the synthesis engines, however, none are about implementing them or creating usable environments for this specific domain. Those textbooks and papers, which do approach design, do not cover the microsound multitemporal compositional aspect that is so unique to this class of sound design tools.

“Some software developers start designing an application by designing the implementation of its functionality. They define the application’s architecture — platform, functional modules...in the early days of software development, when engineers and scientists were still the main users of software applications. The design attitude was: “Get the functionality right, then slap a UI onto it...” (Johnson and Henderson, 2012, p9-10)

Csound opcode API specifications (Vercoe, 2013a) do exist for developers of Csound opcodes in terms of the C or C++ language structures and notations. The same goes for Max/MSP and Pure Data API.

5.11 Question 1. Structuring the framework

It is concluded from the above that in order to repair this lop-sided approach to design and implementation, a design analysis framework would need to consider a close interpretation of the implicit specifications together with a strong set of key criteria. This would have to specifically serve the domain in order to establish the functional and usability parts of the design. Furthermore, because as previously stated, its purpose is to inform independent design (experimental or commercial) and thus would need to be scalable and extensible. Scalable, so that it can suit the size of any artefact purpose, the detail of the project and potentially portable to other synthesis domains. The first key research question asked at this juncture is:

“What are the functional and usability criteria which could establish a design and implementation analysis framework for the effective, unified and robust development of particle formant synthesis / composition environments?”

5.12 Sound synthesis design criteria and universal principles of design

Susan Weinschenck's “Designing Effective Speech Interfaces” dedicates a whole chapter to design approaches to speech recognition and synthesis interfaces. Though the text refers to another type of sound analysis and synthesis environment, it harvests 20 universally accepted principals of design (Lidwell, et al, 2010) gleaned from a range of related design disciplines which include; cognitive psychology, HCI and internet interface design. (Weinschenck and Barker, 2000). The “laws” as the above authors refer to them, range from human factors to the *heuristic* design and evaluation principles of Jakob Nielsen (Nielsen, 2012). Many of the design principles, referred to by Weinschenck and Nielsen, have equivalents in Thomas Green's *Cognitive Dimensions of Notations* system which are a set of tools that aid structuring opportunistic

design through the lexicalisation of design heuristics (Green, T. 1989). Universal design principles such as those described above, form an essential component and criteria for informing the design process in hardware and software information artefacts (Lidwell, et al, 2010), including operating systems, websites, software applications and computing interfaces.

It is speculated that because of their organisational and discursive nature, design principles might be transposed to other information based artefacts, such as PS artefacts. It is reasoned that once these “design principles” are put into the specific PS context, they could perform the function of design and implementation analysis criteria.

5.13 Question 2. Proto-frameworks

The nearest thing to a sound synthesis specific design analysis scheme that discusses functionality and usability can be found in Jaffe's paper "Ten Criteria for Evaluating Synthesis techniques" (Jaffe, 1995). Jaffe's original purpose was to help composers, musicians and sound designers evaluate synthesis tools on their effective design merits. In doing so, he identified a number of specific elements in each criterion which refer to effective design. Jaffe's evaluative guide is an informal and loose proto-framework that hints rather than names the type of discursive tools such as universal design principles. Jaffe's article is a general guide applied to all types of sound synthesis artefacts. Castagne and Cadoz (2003) identified the potential of Jaffe's approach and transposed his evaluation criteria to a specific type of sound synthesis, physical modelling and consequently created their own specific criteria suitable to this unique domain.

Castagne and Cadoz see the potential use of Jaffe's criteria as:

- A synthetic representation of the possible uses of a technique
- A condensed summary of the main features that may be expected for an hypothetical optimal technique and,

- A multidimensional evaluation of the existing techniques (Castagne and Cadoz, 2003, p1).

Consequently, this research also examines whether by making Jaffe's criteria microsound specific and amalgamating these with established design principles, can better inform PS artefact implementation.

“Is the consequent microsound specific criteria based design analysis framework successful in informing new particle synthesis artefacts?”

The following chapters examine these questions in the context of existing approaches to the design of generalised information artefacts. It presents a novel approach to consolidating design criteria for the purposes of PS design analysis and implementation.

Chapter 6 Identifying MS specific design criteria

The method chosen in this research for creating the PS design and implementation framework consists of two main steps.

1. The design criteria are identified using pertinent domain literature and are subsequently mapped and converted to microsound specific criteria in order to form the design analysis scheme. The methods by which this is accomplished are documented in this chapter and in chapters 7, 8.
2. The resulting collated and classified criteria (Table 20) are then used as the basis for a series of design analyses studies from which to extract the usability and functional design characteristics of the seven existing PS systems identified in chapter 5.

6.1 Jaffe's ten criteria

A reduced table of Jaffe's criteria is presented in Table 8.

J1 How intuitive are parameters?
J2 How perceptible are parameter changes?
J3 How physical are the parameters?
J4 How well behaved are the parameters?
J5 How robust is the sound's identity?
J6 How efficient is the algorithm?
J7 How sparse is the control stream?
J8 What classes of sounds can be reproduced?
J9 What is the smallest latency possible?
J10 Do analysis tools exist?

Table 8. Jaffe's ten criteria.

6.1.1 Jaffe's classification of criteria into domains

Jaffe originally classifies his criteria into three areas, usability, sound production and implementation (Jaffe, 1995, p76) thus:

6.1.1.1 1 to 4 Usability. *“concerns the usability of parameters...”* (Jaffe, p76)

J1 *“How Intuitive are they?”* (Jaffe, p76).

J2 *“Do parameter changes have a perceptible effect?”* (Jaffe, p77).

“Changing a parameter by a significant amount should have an obvious audible effect...We call such parameters strong or powerful, in contrast to weak parameters, the effect of which is barely audible” (Jaffe, p77).

J3 *“Do the parameters map to physical attributes of musical instruments or to other physical sound producing mechanisms”* (Jaffe, p79).

J4 *“Are they well behaved or wildly non-linear?”* (Jaffe, p79-80).

6.1.1.2 5 to 7 Sound generation "Other criteria deal with the sounds produced" (Jaffe, p76)

J5 *"Do they retain their identity in the context of variation?" (Jaffe, p80).*

J6 *"Can all classes of sound be produced" (Can a violin produce all kinds of sounds? This is a misnomer" (Jaffe, p84).*

J7 *"Are there analysis techniques for deriving parameters for real-world models?" (Jaffe, p85).*

6.1.1.3 8 to 10 Implementation."The remaining criteria focus on efficiency and implementation" (Jaffe, p76)

J8 *"How efficient is the technique?" (Jaffe, p81).*

J9 *"Does it have unavoidable latency?" (Jaffe, p84).*

J10 *"How sparse is the control stream?" (Jaffe, p83).*

Intentionally or not, this classification is not dissimilar to that used in standard engineering criterion measures (Salvendy, 2012):

- System descriptive Criteria (functionality)
- Task Performance Criteria (feature set)
- Human Criteria (usability)

In order to illustrate each criterion, Jaffe presents the reader with several synthesis and processing scenarios,

"Changing the FM index of a cascade modulator slightly can cause a drastic and difficult-to-predict change in tonal quality..." (Jaffe, p77).

Each of these contexts presents a usability or functional consequence arising from one the scenarios.

"This situation may be merely annoying to a composer, who can take the time to find the proper value for his or her application, but it can drive a

performer crazy if he or she is trying to control such a parameter” (Jaffe, p77).

The cause and effect relationship presented in Jaffe’s criteria illustrates the creative use and functionality of synthesis and processing techniques. In the approach presented in this research, they are adapted, expanded and presented from the perspective of explicitly assessing the worth of a particular class of microsound synthesis techniques and how these, achieve a unique goal; the affordance of usable and functional systems for the effective synthesis and composition of formant rich particles.

6.1.1.4 The informal nature of Jaffe's Ten Criteria

Jaffe’s scenarios are essentially *user-centric*. Jaffe presents these as a generalised guide and which leans towards tacit inference rather than empirical analysis. No attempt is made to quantify the criteria as this is not its purpose. One could treat them as ‘personal’ to the composer, inferred from extensive creative experience and used to inform other composers/sound designers in choosing an effective synthesis tool. Jaffe’s criteria also facilitate *informal discussion* amongst composers, music researchers, synthesiser designers and audio programmers as to what makes a “good” synthesiser design. Because digital synthesisers have moved away from hardware to software together with the availability of tools for designing synthesisers, this is quite relevant today.

“The question may arise as to which technique is best. As it turns out, there is no simple answer. The best technique depends on the priorities of the user and the problem to be solved” (Jaffe, p76)

“The intention is not to survey all known techniques but to outline the criteria...” (Jaffe, p76)

“The choice was biased by the techniques with which I am most familiar, having implemented them myself or used them in musical compositions” (Jaffe, p86).

6.1.2 Re-evaluating the classification of Jaffe's criteria

6.1.2.1 J5. Identity

As previously stated, Jaffe's classifies his criteria into three areas. The scenarios provided by Jaffe however, could be reduced to two; usability and functionality. It could be argued that though J5 is classified as sound generation, the arguments presented are somewhat short and reducible to usability and functionality. One expects that for practical reasons (the length of the original article), the issue of the *phenomenological identity* based on the functional affordance in synthesisers was not considered in any length. The research presented here however, views the *identity* criterion as fundamental if not primary to the design of the synthesiser. This stance is derived from the following approach. In adapting Jaffe's criteria for physical modelling synthesis, Castagne and Cadoz (2003) argue that:

“two motivations for designing a physically-based model should be distinguished: one which aims for a better understanding of real objects and musical instruments (as in musical acoustics) and another which is more oriented to sound and music creation.”

The quote, “*a better understanding*” is read as ‘understand’ in relation to “how a musical object functions and what it does”. In the methods presented here, the term phenomenology is used very much in the context Castagne and Cadoz (2003) use it; a classification of adjectives and processes to describe perceptual characteristics associated with the sonic output of a synthesis technique. In the microsound context, some examples have already been previously discussed in this thesis; granular, particulate, fusion, fission, bursts, evaporation and disintegration.

In this evaluation, the issue of phenomenological identity based on functional characteristics is presented instead of Jaffe's J5 criterion and is subsequently labelled Microsound-5 (M5) “Identity”. This is presented below after the identification of specific attributes in each of Jaffe's criteria. The above premise assumes that in order to successfully generate a wide variety of microsound

timbre classes, the synthesiser must be designed with a set of functions which differentiate it from other synthesiser types or at least and must tap hidden and unused functionality in existing ones. These particular functions are what would endow it with an aural identity permitting the generation of the phenomenology associated with this type of sound. M5, however, does not measure the degree of effort needed to operate the synthesiser in order to achieve this phenomenology.

The research presented here, evaluates the base functional affordance and identity of each system in measuring the capability of pitch-synchronous microsound synthesis. This includes:

Does the synthesiser have the functionality to synthesise sound events in which the duration of the sound particles is independent from the fundamental frequency?

Can it create a variety of useful particle lengths?

Can it generate a variety of particle shapes that affect the perceived physical material of the microsounds; e.g. metallic-like, wooden-like, plastic-like, water (bright, hard, dull, fluid)?

Can it move dynamically between dense groupings (continuous) or sparse (discrete) streams of particles to facilitate fusion and fission? Or vocal gender changes when modelling the singing voice?

This analytical approach permits a practical comparison between systems to evaluate which microsound or synthesis strengths may be harvested for the design or re-design of other systems which exhibit a stronger and wider phenomenology.

6.1.2.2 J4. "How well-behaved are the parameters?" Jaffe, p77

J4 could be classified in either the usability or the functional domains. The behaviour of a system such as a synthesiser results from the implementation of the amount of *forgiveness* (Lidwell, et al, 2010) designed into the algorithm implementation when handling parametric errors. It is proposed that with judicious implementation, this usability issue could be minimised through

functional constraints, value boundaries and parameter intelligence (parametric rule systems). An example of such dysfunctional behaviour is described in Jaffe's article (1995, p78),

"Chaotic behaviour can arise in non-linear feedback techniques...which are often used in physical modelling of wind instruments..."

It would therefore seem more pragmatic to classify it in the functional domain. There are already many testing and analysis techniques (Kaner, et al, 2001) from which to harvest rich design metrics, many of which include functional behaviour testing. Many of the techniques, commonly used by manufacturers, can analyse system or parameter boundaries such as stresses, dependencies, user scenarios, internal or external behaviours etc (Kaner, Bach and Pettichord, 2001). The methods developed and presented in this and the following chapters, form a framework for analysing and testing functionality. One of the methods (M4) includes individual synthesis artefact behaviour testing in order to facilitate a systematic functional analysis contextualised to microsound synthesis. It is hypothesised that with further research and development the framework could also be adopted and used to analyse other types of audio software.

The intention in this thesis is to identify and discuss these methods as potential testing tools for independent developers/designers. Independent developers are also known as double experts; they often act as both software engineer and usability designer. Often they lean more towards one than the other. Therefore, it is not the objective of this research to provide a corporate industrial strength functionality analysis of the functional behaviour of synthesisers. The analysis framework is provided as a first step to inform further research.

6.1.2.3 J6. Sound classes

J6 may also be reclassified in the context of a system's analysis.

"Can all classes of sound be produced" (Can a violin produce all kinds of sounds? This is a misnomer" (Jaffe, p77)

The potential capability of a synthesiser to produce a wide variety of timbres has more to do with the artefact algorithm design than with the user's ability to generate them. This is in effect, similar to M5. It is not difficult to deduce that no matter how experienced a *Mini-Moog* user may be, they will be extremely challenged when attempting to produce a timbre as spectrally rich as that of a *waveguide* based Gong. The Moog's synthesis engine design is just not capable of generating such non-linearities; it does not possess the necessary functionality. Its capabilities could be expanded to reach that goal using its inputs and outputs plus numerous external modules. However, as a solo artefact it does not have that potential.

6.1.3 The framework analysis criteria classifications

Consequently, the original three part (usability, sound generation and implementation) classification of Jaffe's criteria is reduced to two (usability and functionality).

Usability

J1 *"How Intuitive are they?"* (Jaffe, p76).

J2 *"Do parameter changes have a perceptible effect?"* (Jaffe, p77).

J3 *"Do the parameters map to physical attributes of musical instruments or to other physical sound producing mechanisms?"* (Jaffe, p79).

Functional

J4 *"Are they well behaved or wildly non-linear?"* (Jaffe, p79-80).

J5 *"How Robust is the Sound Identity?"* (Jaffe, p80).

J6 *"Can all classes of sound be produced?"* (Jaffe, p84).

J7 *"Are there analysis techniques for deriving parameters for real-world models?"* (Jaffe, p85).

J8 *"How efficient is the technique?"* (Jaffe, p81).

J9 *"Does it have unavoidable latency?"* (Jaffe, p84).

J10 “How sparse is the control stream...?” (Jaffe, p83).

Minimizing the criteria classification permits a far more elementary approach to adapting Jaffe’s framework for analysing and design feature mining. This reduction is essential in small-scale development where cost and time are of the essence. The usability domain can be analysed combining elements harvested from “goal” orientated design (Cooper, et al, 2003), usability evaluative methods such as Nielsen’s *Heuristics* (Nielsen, 1994b) and Green’s *Cognitive Dimensions* (Green, 2000). Likewise, identifying the pertinent attributes of the functional domain, as done in the following pages, systematically facilitates the analysis of the synthesis environments, by using established specification based software testing techniques similar to those used in commercial and industrial artefacts (Kaner, et al. 2001). However, as will be discussed below, a full technical analysis of the functional domain is beyond the scope of this thesis. All that can be given in this respect is a general summary of the approach needed to guide further research.

6.1.4 Creating individual criterion attributes

Using the causal relationships described by Jaffe and in order to create a measurable framework, it is proposed to atomise his criteria into *attributes* grouped under each criterion. By identifying individual usability and functional elements in the criterion, we can come closer to identifying variables which may be operationalised. This operationalisation allows the discursive nature of Jaffe’s criteria to be turned into a framework for quantifying the instruments qualities mentioned earlier. The intention is that this process should afford a wider and empirical basis for design. Each attribute is labelled “A” plus an ordinal so that Jaffe criterion “1” and attribute “1” would be labelled thus *J1_A1*, attribute 2 *J1_A2* and so on (Table 9).

J1 "How intuitive are parameters" p76
J1_A1. Notation is/are musical? Mathematical variables? Or other? (Jaffe, p76)
J1_A2. Linear or wildly non-linear? Jaffe refers to the way values display a uniform correlation between what the user's input value measures and the output it produces. (Jaffe, p79-80)
J2 "How perceptible are parameters" p77
J2_A1. Strong/weak parameters. (Jaffe, p77)
J2_A2. Does it have a meta-parameters mode for weak parameters? (Jaffe, p78)
J3 "How physical are parameters" p79
J3_A1. Do parameters mimic physical behaviour? (Jaffe, p79)
J3_A2. Do meta-parameters exist or are possible to mimic pseudo physical attributes. (Jaffe, p79)
J4 "How well behaved are the parameters?" p80
J4_A1. Proportional change of parameter and effect. (Jaffe, p80)
J4_A2. Is the technique linear or non-linear? (Jaffe, p79-80)
J4_A3. Sensitivity to initial conditions. (Jaffe, p80)
J5/M5 "How robust is the sounds identity?" p80
J5_A1. <i>"How well a sound retains its identity in the context of variation..."</i> (Jaffe, p80)
J5_A2. Can expression be synthesised in the technique? (Jaffe, p80)
J5_A3. Does the technique require extraneous techniques to give identity? (Jaffe, p80)
J6 "How efficient is the algorithm?" p84
J6_A1. Does it work in real-time without hiccups? (Jaffe, p84)
J6_A2. What is the relationship between sampling rate/bit depth and polyphony? (Jaffe, p82)
J7 "How sparse is the control stream?" p83
J7_A1. Is it the synthesis technique which produces the sound identity or is it the control stream (Jaffe, p83)
J7_A2. Are there any meta-parameters which could clog up the control stream? (Jaffe, p83)
J7_A3. Is the control stream at audio rates or control rates? (Jaffe, p83)
J8 "What classes of sound can be represented?" p84
J8_A1. Some techniques can produce any type of sound given enough control data (Jaffe, p84)
J9 "What is the smallest possible latency?" p84-85
J9_A1. <i>"Some techniques have an inherent and unavoidable latency that may cause problems in an interactive situation..."</i> (Jaffe, p84-85)
J9_A2. How complex does a patch have to be before the system struggles? (Jaffe, p85)
J10 "Do analysis techniques exist?" p85
J10_A1. "You need the tools to derive the proper parameter values from a specification of a desired result" (Jaffe, p85)

Table 9. Jaffe's criteria and attributes.

6.1.5 Jaffe's attributes as cognitive and functional dimensions.

The synthesiser as an Information artefact

Synthesisers are contemporary music instruments and sound design tools. They are prime tools for communicating live or recorded music, sound effects for film or animation and scientific research (sonification of data). Sound synthesisers have also been described as '*interactive information artefacts*' (Green, 1989). As this research has progressed and the criteria have been analysed, there has been a natural reshaping and re-titling of the attributes as dimensions. This has been true for both cognitive and the functional dimensions discussed here. Perhaps the most awkward of the criteria titles discussed is found in J1 "*How intuitive are parameters?*". The term intuitive is too general and diffuse when ascertaining a measure of a parameter or system. Some process needed to be found to operationalise each attribute/dimension as a variable. Logically, the variables have to be named or labelled. As Jaffe's criterion attributes have gradually been re-classified, so has the meaning and titles of the original criterion which have metamorphosised into more appropriate titles. Therefore, terms such as "intuitive" have been discarded in favour of terms which have operational and quantifiable potential. This is presented in the following chapter.

Chapter 7 Cognitive Dimensions.

Measuring cognitive complexity

7.1 Cognitive Dimensions and Jaffe's criteria

Green and Blackwell (1998) propose that information artefacts may also be referred to as *cognitive technologies*. In earlier research, Green proposed a framework for the evaluative discussion of the design of such artefacts. Green referred to this framework as the *Cognitive Dimensions of Notations* (Green, 1989). Green and Blackwell (1998) subsequently published a tutorial to help designers and developers evaluate the cognitive dimensions (CDs) of an information artefact.

It is proposed that Green's original 11 CDs (Green, 1998) have something in common with Jaffe's 10 criteria. The latter are intended, like CDs, to facilitate the discussion of synthesisers between practitioners and instrument designers. They are usability factors, design patterns ("universal principles of design") and functional guidelines. Whereas Jaffe's criteria are domain specific (evaluating the design of synthesisers), Green's CDs are generalised. However, an important difference is that Green's dimensions like Nielsen's usability *Heuristics* (Nielsen 1994b) are lexicalised and offer a general qualitative design vocabulary (Green, 1989). Nielsen's Heuristics and Green's CDs are in effect small scale, "cost cutting" design guidelines and evaluation tools based on known successful design outcomes (Nielsen, 1995a)

There have been a couple of research projects which have touched upon the analysis of music information artefacts using CDs. Collins and Blackwell (2005) discussed programming languages as musical instruments, revealing many

common traits between both domain's notations. Their brief CD analysis of Ableton *Live* has some relevance to this research. Duignan, Noble and Biddle (2010), discussed another approach to the use of CDs in the analysis of music production artefacts. However, to this date there has not been a study of the usability and functional potential of microsound synthesis artefacts using a combination of Jaffe's criteria or CDs. Below are four hypothetical reasons why the CD approach could offer many potential benefits for design comparison.

- They provide a powerful set of tools to evaluate the usability constraints inherent in the artefact, which may potentially impose on the user.
- It does not require a specific design expertise and is adaptable to double expertise and end-users.
- It provides a means for comparing and evaluating design approaches within a unique domain.
- The context of approach to the analysis is done as an actual example of a scenario as and End-User/Independent developer. This is the final context and purpose of the research outcome.

They are meant both to make the evaluation concrete and to provide a basis for comparison between designs or design choices. "(Blackwell, 2001, p331).

Through detailed translation, Jaffe's criteria can be reduced to a *heuristic* and *CD* framework. The methods used in this research borrow and combine both Green and Nielsen's design patterns in order to lexicalise, illustrate and evaluate the attributes implied in Jaffe's criteria. Here they will be referred to as *dimensions* too. It is a pragmatic first term for contextually connecting numerical data to categorical data.

The notion of adapting design patterns and terminology arose out of an initial background investigation into the measurement of parameter dependencies, a term which is widely used in both software testing (Kaner, et al, 2001) and design usability research (Green, 2000). It seemed logical to avoid re-inventing the wheel and expand this lexicon with 'shadow' terminologies. This only encourages the terminological equivalent of *Chinese whispers*.

An essential difference between the approaches taken by Jaffe, Nielsen and Green is that the research presented in this thesis widens the purely discursive nature of the original CD framework into a framework of empirical design measurement. A very simple premise explains this approach in favour of the purely discursive; operationalising qualitative variables enable a greater in-depth conceptual model of design. Green et al (2000) discuss normalisation and operationalisation but this investigation could not find any in-depth study in relation to the design of synthesis software development. Therefore, this work concentrates on the operationalisation of cognitive dimensions and heuristics. It uses ordinal variables in order to provide useful evaluative and comparative design data.

“The value in this approach is its immediacy; the usage is pragmatic and accessible, making a cognitive dimensions analysis a low-cost tool to add to a design repertoire. Putting CDs readily into use is the best way to demonstrate their relevance to practice. But the process of operationalization itself is informative...giving perspective on definitions and concepts, exposing interrelationships among design choices, reflecting on the impact of tasks and environments...” (Green, et al, 2000)

7.1.1 The Initial mapping of Jaffe’s criteria to CDs

Analysing the scenarios presented in Jaffe’s text suggests the following list of CDs and heuristics.

In J1 we obtain the following:

J1_A1 Flexibility of notations (Green, 1989)

*“Musical attributes...dynamics, articulation...mathematical variables...”
(Jaffe, 1995, p77)*

J1_A2 Notation constraints (tables) (Jaffe, p77), (Lidwell, et al, 2010, p50)

J1_A3 Notation constraints (units) (Jaffe, p77), (Lidwell, et al, 2010, p50)

“Changing the FM index of a cascade modulator slightly can cause a drastic and difficult-to-predict change in tone quality” (Jaffe, p77)

J1_A4 Dependencies (Jaffe, p77, Green 1989 p5)

“Thus decoupling distance... from dynamics...” (Jaffe, p77)

“Changing the FM index of a cascade modulator...” (Jaffe, p77),

“Especially dangerous are filter structures that can become unstable during transitions from one set of coefficients to another” (Jaffe, p80)

J1_A5 Dependency rules (Jaffe, p78)

“Still, with sufficient care, such a model can be made to operate in the regions of its space in which it behaves predictably and effectively” (Jaffe, p80).

Further dimensions can be derived using the same process. Relevant to this evaluation are:

J1_A6 Hard mental operations / interference effect (Jaffe, p78), (Green, 1989 p9), (Lidwell, et al, 2010, p113))

“The user-supplied partial amplitudes and frequencies are defined as arrays $Amps[i]$ and $Freqs[i]$, respectively, where i is the partial index...” (Jaffe, p78)

J1_A7 Premature commitment (Jaffe p83, Green,T.1989, p5)

“The best technique is one which everything is pre-determined, a situation that is, by definition, non-interactive” (Jaffe, p83)

In J2 we obtain the following heuristics.

J2 Consistency/expectation effect

“If a tiny change causes a huge effect. A performer may have difficulty controlling the technique...If you change the amplitude envelope of a less-important harmonic, the change can be completely inaudible” (Jaffe, p78), “ideally, a change in a parameter produces a proportional change in the sound” (Jaffe p79, Nielsen, 1995a, p1)

J2_A2 Dependency rules. This is logically interpreted to mean, "in order to control dependencies" (Jaffe, p78 (J2)).

7.1.2 J2 Metaparameters as rules

The second attribute in J2 *meta-parameter control* has been changed to *parameter rules system*. This is reasoned from Jaffe's question as to whether meta-parameters are afforded in a system in order to manage weak or strong parameters,

“An additive brightness parameter can be defined that behaves similarly to the low pass filter brightness parameter” (Jaffe, p78)

This implies rules. Consequently, it is thought more appropriate to rename this attribute as “If the parameter is either too weak or too strong, are there any rule based parameters from which to control them?” It is interesting to note that there are several mentions of meta-parameters in Jaffe’s text. This suggests that there might be just one dimension in which meta-parameters are considered. However, because of the strong contextual meaning of the term it was decided to keep the affordance of meta-parameters for dependencies and parameter strength separate. This is a trend followed through the rest of the research.

7.1.3 Microsound cognitive complexity

Using the mapping from Jaffe's attributes to CDs we obtain Tables 10 and 11. In order to distinguish the criteria system from Jaffe's criteria, they are labelled "M" for Microsound instead of "J" for Jaffe and each attribute is given a CD or Heuristic term where possible. If not, they are made up to represent as closely as possibly what it is they do.

M1.A1. Flexibility of notations afforded – Musical, numerical (hertz, cps), scaled (8 bit 0-127)). Was J1_A1 value representation, musical attributes etc., Jaffe, p77)
M1.A2. Notation table boundaries (can the value ranges be bound, constrained?) (Jaffe, p78)
M1.A3. Notation units boundaries (is there a constraint so as to limit the value range?) (Jaffe, p78)
M1.A4. Dependencies/influence (The more dependencies, the greater the cognitive complexity (Jaffe, p77)
M1.A5. Rules to control dependencies (if there are dependencies are there any rules for keeping them in order?)
M1.A6 Meta-parameters (to control dependency/influence relationship)
M1.A7. HMO. Hard mental operations
M1.A8. Premature commitment

Table 10. M1 Cognitive complexity. Notations and notation control

M2.A1. Parameter strength
M2.A2. Rule based parameters

Table 11. M2. Cognitive complexity. Perceptibility of parameter changes.

7.2 Specific MS CD criteria and attributes analyses approaches

In the analyses presented in this research, the M1 and M2 criteria and attributes are measured using a simple arbitrary scale grading system adopted from GAP analyses (Anon. 2013. Gap Analysis). This is a process development analysis technique which compares a specific function or process against an idealised goal metric. The difference is the numerical gap. There is very little literature on GAP analysis and this has a wide interpretation. This ranges from feature

omission measurements in architecture, to business practice potentials such as SWOT analysis. In the case of M1_A4 (Appendix J), statistical descriptive analyses are employed in order to derive comparisons between systems and parameters. The exact method detail, purpose and apparatus are discussed at length in the individual analysis appendices (appendices B-O). Each analysis appendix comprises a data description / report detailing the findings and inferences and an accompanying spreadsheet (on the pen drive). In the case of the CD analyses, these are relatively simple. The research methods proposed develops this approach further in order to include the *functional complexity* inherent in Jaffe's text. This will be referred later in this text as the set of *functional dimensions*. Importantly and in relation to this research is that these variables are treated as domain specific rather than general. This contextualisation exists for two reasons. One, in order to cohere the domain specific language for expert discussion and design and two, so that there is a strong inter-relationship between the explicit specifications (manufacturer, developer) and implicit specifications (originating papers or authoritative research literature) (Kaner, et al, 2001).

7.2.1 Synthesis artefact notations

The following clarification is offered in order to elucidate the relationship between Green's concept of notations and environment in relation to the synthesiser. *Notations* in this thesis refer to the synthesis and processing parameters within each interface window or sub-window. These could be composed of *path*, *axial* or *barrier* constraints (fader, rotary, switches) (Lidwell, W et al. 2010) or/and numerical, equalisation or waveform displays.

They may also refer to the individual objects in Max/MSP, *Pure Data*, *Reaktor* or the opcodes and statements, variables in Csound. It would seem logical to classify these notations as a separate category. The *environment* refers to the windows and sub-windows and may contain apart from notations other sub-notations such as the "file", "view", "Window" found in pull down menus (Blackwell, et al, 2001). A *system* is the host sequencer such as Ableton, Logic,

Cubase or a host plugin environment such as standalone virtual plugin rack such as Native Instruments *Kore*. Audio editing systems such as Ableton Live or Apple's Logic are meta-environments that nest other complex environments. The timeline for editing multitrack events is a complex and self-contained system, which talks to other equally complex systems for managing synthesis or processing plugins.

7.2.2 The context of CDs and usability engineering heuristics in this research

CDs were developed as a light and broad “pallet” of discursive tools for non-expert designers. It is assumed *non-expert* includes independent/lone developer researchers. In essence, they provide a nomenclature for lexicalising existing design factors and patterns. From this point of view, they offer a practical starting point for this type of developer to identify and discuss processes for use in the design of software. The artefacts in this research are examples of this type of development and therefore CDs have been adopted, combined, re-synthesised alongside usability heuristics and ideas from conceptual modelling of software development in order to determine a process framework which is developer *rich* for the creation of software solutions in microsound synthesis. The reality of using such design tools, as created by Green, Nielsen and others, doesn't come without problems. The main issue is contextual to the synthesiser domain and to empiricism.

Heuristic evaluation and CDs often cross paths using differing terminologies to mean the same thing. However, the context is different. Whereas CDs are created as discursive tools, heuristics are mainly presented as information mining tools (Nielsen, 1994b). One example of this can be found in *heuristic evaluation*. Weinschenck and Barker (2000) use the heuristic “Linguistic clarity”. Gerhardt-Powels (1996) on the other hand use names that are conceptually related to function.

The other problem is that CDs are published copiously, by their creators, as being somewhat broad,

“...generalised broad-brush approach...” (Green, 2000, p21)

“...they consciously aim for broad-brush treatment rather than lengthy, detailed analysis” (Green, Blackwell 1998, p5)

“The framework therefore avoids any kind of detailed cognitive analysis, although it has a cognitive underpinning.” (Green, 2000, p22)

“The framework emphasizes the design choices available to such designers, including characterization of the users activity, and the inevitable trade-offs that will occur between potential design options.” (Blackwell, Britton, Cox, et al, 2001, p325)

“The Cognitive Dimensions approach should be seen as complementary to other approaches” (Green, Blackwell, 1998, p6)

7.2.3 Independent developers need more than discursive frameworks. Operationalisation.

A strong premise for this approach is presented by Green (1998). Independent designers of synthesiser software need strong conceptual models (Johnson, Henderson, 2012, p10) of the originating papers or implicit specifications (Kaner, Bach, Pettichord, 2001). The simple rationalisation for this can be found in the double expert situation described earlier.

Whilst implicit specifications aid the forming of conceptual model depth (Henderson 2012), explicit specifications (manufacturer instructional manuals) create mental models. Cognitive complexity cannot be presented as *broad brush* if it is to forge a deep conceptual model in the mind of a developer. If anything it could be argued that it may do the opposite. The equations presented in the explicit specifications need to be understood with this deep conceptual basis so that the developer can offer a simpler, efficient mental model to the end user. Understanding and communicating the numerical flow between variables in a digital signal processing equation requires that kind of depth. A *broad-brush* approach would exclude this.

This is particularly true in visual programming in which GUI elements can be developed simultaneously as part of the coding notation. A floating-point

variable initialised in Pure Data or Reaktor is a GUI element. It is reasoned in this thesis that good functional and usable design, in this domain, cannot be produced without the base conceptual model as a starting point. Logically, this implies a deep understanding of the implicit specifications. As the analyses descriptions and reports presented in appendices B to O will demonstrate, the explicit specifications can often leave users with differing mental models of the implicit specifications. Briefly, the BDYSYS Pulsar synthesiser is a very different implementation from the Pulsaret plug-in implementation. The former favours pitch-synchronicity and the latter asynchronicity; two very distinct phenomena. Both originate from the same conceptual model, Roads' implicit specification but they engender very different mental models in users.

For this reason, it is not possible to imagine how a *broad-brush* treatment using un-operationalised CDs can offer any discussion of deep feature design in PS artefacts. The operationalisation is a prerequisite of this research because it moves it away from the purely discursive to an empirical design development process. By refining Jaffe's criteria the research can then determine the 'what' and 'detail' that is being analysed.

Chapter 8 Functional Dimensions.

Measuring functional complexity

8.1 Introduction

In the same way that the term *cognitive complexity* becomes central to the usability side of the design analysis framework presented here, the same approach has been taken with the treatment of evaluations based on functional complexity. In the functional domain, attributes similar to those seen in the usability domain are extracted and expanded from the text. Together they form a basic software requirements document for the design of a particle synthesis software.

8.1.1 Functional capability and limitless behaviour

A departure from Jaffe's descriptive scenarios is that in this research the functional performance is not measured extensively in terms of hardware performance; e.g. CPU efficiency, Latency and Control stream. There is a rationale behind this and one which is very relevant to independent development. Measuring all functional behaviour is an impossible expectation:

“Now enter the computer, the first machine created by humans that is capable of almost limit-less behaviour when properly coded into software.” (Cooper, Reinmann and Dubberly, 2003, p6).

1. Independent researchers/developers in music, by the very nature of what they do, rarely consider computer systems outside the ones they are

developing on. Generally, these computers tend to be “domestic” computers (Desktops, laptops, Tablets) rather than super-computers.

2. Computer hardware gets faster and faster and processors are made which deal with various different parts of the processing; e.g. separate processors for generating audio and controlling data. Though one can imagine a synthesis scenario in microsound generation that might require premature commitment, this is not considered an issue. Most of the processes described here perform reasonably well in real-time and are well documented in the appropriate implicit and explicit specifications. *Pulsaret*, *Audiomulch*, *Metasynth*, *Delay Lama* are a few of the similar synthesisers which perform competently in real-time even with complex patches on average hardware. The caveat in this is the way in which Csound handles initialisation.
3. Microsound does not require the complexity of mathematical processing demanded by physical modelling or convolution.
4. It is beyond the scope of this research to measure each microsound technique for an accurate set of performance benchmarks for each system available. This is potentially of interest for future research. Processors get faster and memory gets bigger, more polyphony is available etc. There are contextual reasons for this as there are too many scenarios in which these artefacts can be used. This feeds into point 1 above.
5. The use of specifications. As an independent developer, it is difficult to equal the standards required by industrial specifications, however, specifications of some kind do help in achieving better standards of design. This would be a good study for a further PhD in which masters' students carry out experiments in order to determine if they are able to improve design using some kind of formal design requirements.

What is intended in the framework offered here is to provide a formal tool which helps overcome many of the limitations and inconsistencies observed in the implementations surveyed in chapter 4. The framework in this research, takes a measured approach which is based on empirical analysis with the aim of

producing artefacts with reliable usable and functional behaviour. As the functional analyses provided in appendices K to O demonstrate that functionality can be measured with a degree of precision useful to inform robust and efficient design. As Kaner, et al (2001) point out however, completeness is impossible and an unrealistic expectation when applying testing methods to software no matter how strict the originating specifications and requirements are.

Using the criteria/attribute translation and mapping process adopted earlier for CDs we can obtain Table 12 which is a similar set of attributes but contextualised to functionality.

J3 “How physical are parameters” Jaffe, p79
J3_A1. Do parameters mimic physical behaviour? (Jaffe, p79)
J3_A2. Do meta-parameters exist or are possible in order to mimic pseudo physical attributes. (Jaffe, p79)
J4 “How well behaved are the parameters?” p80
J4_A1. Proportional change of parameter and effect. (Jaffe, p80)
J4_A2. Is the technique linear (such as additive or Subtractive synthesis) or non-linear? (Jaffe, p79-80)
J4_A3. “...sensitivity to initial conditions...” (Jaffe, p80)
J5 How robust is the sounds identity? p80
J5_A1. "...how well a sound retains its identity in the context of variation..." (Jaffe, p80)
J5_A2. Can expression be synthesised in the technique? (Jaffe, p80)
J5_A3. Does the technique require extraneous techniques to give identity? (Jaffe, p80)
J8 What classes of sound can be represented? p84
J8_A1. “Some techniques can produce any type of sound, given enough control data” (Jaffe, p84)
J10 Do analysis techniques exist? p85
J10_A1. “You need the tools to derive the proper parameter values from a specification of a desired result” (Jaffe, p85)

Table 12. Jaffe's criteria reclassified as functional dimensions.

8.2 Converting Jaffe’s functional criteria and attributes into MS measurable functional dimensions.

8.2.1 J3. How physical are parameters in microsound?

J3 and its two attributes do not completely fit comfortably inside the particle synthesis environments.

J3_A1 would seem to mean a matter of characteristic identity rather than raise a question about usability and control. The relationship between the microsound instrument and its phenomenology are not the same as that of a waveguide based physical modelling environment. (Castagne, Cadoz, 2003, p4).

As we saw in chapter 2, particle synthesisers exhibit a unique characteristic; multitemporality. Not only can you synthesise acoustic particles but you can also organise them in time using a number of strategies. The multitemporal

compositional aspects discussed in chapter 3 are central to microsound and distinct from any other synthesis techniques. For this reason, J3_A1 is seen as irrelevant in this new set of dimensions.

8.2.2 Algorithmic interpretation of *physicality*

J3_A2 is defined in terms of the existence of meta-parameters. The physicality versus virtual sound environment, however, is different in particle synthesis. Physical waveguides model the physical apparatus rather than the effect.

“...both through ear and signal analysis. PM2 is obviously important when the aim is to reproduce the sounds of a real instrument (see section I). However, it is of a lesser importance when the user is mainly seeking a convincing sound plausibility but does not want to model a specific sound object.” (Castagne, Cadoz, 2003, p3)

Microsound instruments do not offer the same plausibility relationship between the synthesiser, the way it is played and the resulting perceived phenomena. One could argue that on many occasions it does the opposite, as in the case of the fission of a human voice into droplets. Instead of plausible, it is surreal. Surrealism is at the heart of particle synthesis, especially techniques like FOF synthesis.

“We prefer the notion of physical plausibility of a sound. The important feature for a musical sound is not to cause the listener to infer its physical cause, but to present a set of subtle dynamic variations among perceptual parameters that lead the listener to think it was produced in some physical manner. (Castagne, Cadoz, 2003, p3)

Instead an alternative and more relevant question is asked in this evaluation:

Do parameters or meta-parameters exist with which users can model simulated physical aspects of the particles such as physically plausible particle materials (metal, wood and water) or do rule based or procedural parameters exist which

permit the morphing between vowel spaces in a similar manner to the one offered in *AudioNerdz Delay Lama* or the x/y pad controller in *Pulsaret*?

J3 then is presented in Table 13 as M3 and with only one attribute:

M3 "Are there means to imitate physical and environmental acoustic phenomena?"
M3_A1. Do meta-parameters exist or is it possible to mimic physical phenomena such as materials or environments.

Table 13. M3 Metaparameters and physical behaviour.

8.2.3 J4. "How well behaved are parameters?". Defining functional behaviour in a synthesiser context

Jaffe proposes that a well behaved parameter is one which,

"...produces a proportional change in the sound" (Jaffe, 1985, p80)

His discussion proceeds that there is a relationship between the synthesis technique's linearity and behaved parameters.

Achieving the level of detail usually contained in an industrial formal requirements specification is a complex and demanding process, especially with newer technologies and concepts. Some technology practitioners identify a need for the simplification of the process.

"In practice, technology influenced design may lead the way, in some cases, to a simplification of the design process depicted in Table 1, as little room is given to new concept generation, since the concept is determined by the application of technology to enable a particular functionality and as such, circumvents the search for new ideas, and promotes the continuation of a particular product archetype..." (Coelho, 2011, p4)

Independent developers and researchers rarely apply industrial formality to the software artefact they create. As discussed earlier, the motivation for developing is not necessarily guided by corporate economics but usually to

support individual research or bespoke “boutique” projects. Developing synthesis artefacts as complex as physical modelling or microsound synthesisers is a complex process. It is often an ad-hoc development without much formalism except for the mathematics necessary to understand the algorithms and programming knowledge to translate them into code. Jaffe’s approach is central in this research because it can be considered as one of the first pieces of literature that inadvertently hints at the importance of the role of implicit specifications for use as synthesiser design guides. It could be treated as a set of heuristics much in the same way as Nielsen’s heuristics. However, from a measurement point of view some kind of formal requirement is viewed as essential in developing a functionally efficient synthesis artefact. What we can do is to expect a certain essential behaviour from software. By creating a *functional requirements* document we can say “*what it is*” that we expect from the microsound synthesis software. Above, we have seen that creating the cognitive complexity framework has shaped the non-functional software requirements. In contrast to the CDs described earlier and from a pragmatic point of view we could interpret the other elements of Jaffe’s paper as a *choosing guide* to identifying ‘desirable’ functional characteristics as presented in Table 14.

Reliability
<i>“...that may cause problems in an interactive situation in which the sound must be computed and played immediately in response to an asynchronous event.” (Jaffe, p84).</i>
Efficiency
<i>"The third efficiency consideration is the heaviness of the required control stream." (Jaffe, p81-84).</i>
Consistency
<i>“Changing a parameter by a significant amount should have an obvious audible effect.” (Jaffe, p77- 79).</i>
Robustness
<i>"You can vary such parameters as pick position...to a great degree...whilst never leaving the realm of string like identity." (Jaffe, p80).</i>
Functional performance (p83-84)
<i>“For example, a violinist can make many changes to his or her sound, but the sound is still clearly a violin.” (Jaffe, p80).</i>
Functional affordance
<i>“Is there a class of sounds that it is impossible to produce...?” (Jaffe, p84-85).</i>

Table 14. "Desirable" functional characteristics.

8.2.4 Behaviour and reliability

Traditionally, these characteristics are considered non-functional requirements in the development of commercial and industrial software (Ashish, 2010). However, the term has been adopted here to differentiate the measurable behaviour of the functioning systems as opposed to the user's potential cognitive input and response to functionality. It could be argued that this is the manner in which Jaffe treats them and often the text uses some manner of reference to a system's *behaviour*. In this sense, one can think of them as *soft-functional* requirements. However, for the sake of brevity the term 'functional' is used instead. What is presented is a set of methods for establishing, through quantifiable means, some degree of functional and behavioural reliability.

8.2.5 Consistent external behaviour and the code layer

More precisely, the testing and analysis scope presented in these pages is one of *black-box testing*. It is an analysis of *external behaviours* (Kaner, et al, 2001). The analysis or the tester is not concerned or informed by the *code internals*. The testing is concerned with how the artefact responds as a result of its use. Therefore, it is not proposed to measure every critical condition in order to repair the artefact but instead to look for consistency of external behaviour. All of the functional analyses are informed by both the explicit and implicit specifications in the same way dependency testing is informed in the CD chapters.

8.2.6 The myth of completeness in testing

Testing the external behaviour of each parameter in a synthesis system, especially one afforded as Partikkel opcode in Csound, presents a major challenge. It is difficult to test every possible parameter in every possible scenario (Kaner, et al, 2001, p32). The context for completeness is that the analysis compares *external behaviour* in order to identify characteristic

inconsistencies across particle formant synthesisers in order to arrive at a more satisfying design approach.

Be aware that the definition of “complete” is not the kind of thing that can be settled conclusively at the start of the project. You have to reconsider it as the test project evolves and as new test tasks crop up.” (Kaner, et al, 2001, p32).

8.2.7 The testing ethos

The approach for evaluating the functional dimensions is through “live” parameter testing and statistically derived inference. Each system is tested for a range of *External Behavioural* FDs in order to glean possible pitfalls in the design of this type of synthesiser. The intention is not to find faults in the system in order to label such system as “weak”. Kaner et al, describe this approach rather elegantly:

“Testers focus on failure because it improves their chances of finding it. Look for key problems in the product with all your creativity and skill. If you don’t find them, they can’t be fixed, and then the users may find them for you. By finding what’s there to find in the product, you help the project team learn more about their own skills and the product’s risks, and you help them make the product better, more supportable, and probably more successful in the marketplace.” (Kaner, et al, 2001, p31)

Failure infers substantial reason for design improvement.

8.2.8 External behaviour and hardware dimensions

Below are presented the list of Behavioural FDs suggested in Jaffe’s text and expanded from inference using the implicit specifications. Three types of test are proposed: general, side effects and hardware response. Each one falls into the functional characteristics (in parenthesis) presented above.

8.2.9 Consistency, robustness in response

This includes detecting effects from linearity and chaotic behaviour.

(Consistency, robustness)

“If the filter bandwidth is adjusted on a note by note basis so that a requested fundamental amplitude is obtained regardless of frequency...”
(Jaffe, p77)

“Ideally, a change in a parameter produces a proportional change in sound”
(Jaffe, p79)

“Changing the FM index of a cascade modulator slightly can cause a drastic and difficult to predict change in tone quality” *(Jaffe, p77)*

8.2.10 Latency and viscosity (Jaffe, p83-84. Green, 1989, p6)

Does the system become slow when moving variables or GUI components?
(Good functional performance. Efficiency).

“...the control stream may consist of many megabytes of data, more than can fit in random-access memory” *(Jaffe, p83)*

“Some techniques have an inherent and unavoidable latency that may cause problems in an interactive situation in which a sound must be computed and played immediately in response to an asynchronized event.”
(Jaffe, p84)

8.2.11 Error frequency

This attribute is adopted from Green’s dimension ‘error proneness’ but interpreted as a functional dimension in order to measure how error prone a particular synthesis system might be. This is in contrast to its original use which is to measure whether the system causes the user to commit errors. We can

measure how much a system "breaks" during 'normal' use. (*Reliability, consistency, good functional performance*). By measuring error frequency, the developer can identify specific problematic parameters in order to identify problem parameter patterns.

"complex behaviours that arise during unstable moments in a tones evolution..."(Jaffe, p79)

When parameters change rapidly, energy is injected into the system, allowing the possibility of unpredictable or unexpected results (Jaffe, p80)

8.2.12 Parameter dependency effects

Sometimes the error pattern can be created from a parameter dependency. Whereas dependencies are harvested from the CD analysis, here they are revisited in terms of cause/effect relationships between those parameters. The rationale behind this is that if these dependencies do cause major effects, as suggested in Jaffe's text, it would seem logical to find exactly the effects of these dependencies, across the parameters involved, with the view of determining design strategies to mitigate those effects. Such strategies might involve value boundaries or algorithmic rules. Whichever way it is considered vital development material. (*Reliability, functional affordance, performance*)

"Changing the FM index of a cascade modulator..." (Jaffe, p77)

Whilst referring to particle overlaps "...Some techniques have a processing requirements that change with the parameter values..." (Jaffe, p81)

Jaffe discusses the case of overlapping particles in FOF synthesis. The larger the quantity of overlapping particles, the bigger the amplitude of the final signal. It would be a logical outcome that some rule-based system is used to control the amplitude disparity in order to avoid clipping the output. This has implications for the functionality of all the systems because of the similarity of the techniques used in the sound generation algorithm.

8.2.13 Initialisation

An issue related to the overlaps function, and which is found in the explicit documentation in the Csound opcodes, is the errors caused by the pre-runtime initialisation parameters. (*Functional affordance*)

A5. Does the parameter stop functioning without user input?

A6. Does the parameter need rules to set up a well-behaved control environment so that dependencies interact within limits?; e.g. *iolaps*, *xband*, *kdur*, *xfund* in FOF.

8.2.14 Hardware behaviour analysis

Though some measure of functional behaviour is presented in this study, it does not propose the depth of detail perhaps as needed to test other types of software such as dedicated embedded systems. Independent developers of software synthesisers tend to develop using generalised computing systems. Evidence of this can be found in the software synthesis environments such as Ableton *Live* and Native instruments *Kore*. The term “Live” clearly proposes the hardware scenario on which such software are run, “live performance”. Ableton *Live* hosts software synthesiser plugins in which the purpose of the environment is mainly live performance. Therefore, the hardware chosen to run such software is generally portable. The front page presents Ableton Live as:

"Ableton Live is about making music; for composition, song writing, recording, production, remixing and live performance. Live's nonlinear, intuitive flow, alongside powerful real-time editing and flexible performance options, make it a unique studio tool and a favorite with live performers." (Anon. 2012. Live)

The minimal system requirements, recommended by Ableton to run is advertised as:

"Mac: 1.8 GHz G4/G5 or faster (Intel® Mac recommended), 2 GB RAM (4 GB recommended, if supported by your computer), Mac OS X 10.4.11 (10.5 or later recommended), DVD-ROM drive

Windows: 2 GHz Pentium® 4 or Celeron® compatible CPU or faster (multicore CPU recommended), 2 GB RAM (4 GB recommended on Windows Vista and Windows 7), Windows XP (home or Pro), Windows Vista or Windows 7, sound card (ASIO driver support recommended), DVD-ROM drive, QuickTime recommended

The installation size of the Essential Instrument Collection 2 is 15 GB." (Anon. 2012. Live)

As can be seen, these are modest computer hardware requirements met by most laptop computers in manufacture at the time of writing. (World, 2012)

The recent addition of *smart-phone* and *tablet* computing devices has proliferated the quantity and quality of computer hardware for sound synthesis. Apple's iTunes *app store* already furnishes a large catalogue of microsound synthesisers with complex functionality. *Boulanger Labs* has already released *CsGrain* an iOS application for the generation of granular synthesis using a Csound framework (Boulanger, 2012)

8.2.15 CPU

There are a number of dependent factors involved in measuring CPU in an analysis of this kind. One of the problems is the individuation of users systems because they will have different software and hardware settings. These are not strict laboratory test apparatus. This study however, goes someway to test these on a representative system such as might be employed in the sound studio or live performance. A back-up system was also employed in which to verify and repeat analyses which exhibited odd functional behaviour.

"FOF...by adding up overlapping vocal tract impulse responses, becomes more expensive as the frequency rises. There are more pitch

periods per second, and thus more additions and table lookups per output sample” (Jaffe, 1995, p82)

8.2.16 Memory

Memory is required to store overlaps, particle shapes, tables, etc. In Csound’s FOF this can be pre-allocated at little cost. It protects the system from ceasing to operate if you choose more overlaps than you have stated at initialization. (Efficiency, functional performance, functional affordance). This method for measuring is similar to that employed measuring CPU statistics.

8.2.17 Algorithm efficiency

The efficiency of algorithm is discussed in Jaffe’s text (Jaffe, 1995, p81-84) It is difficult to measure the effect of an algorithm on a system without having the developers specification and even then, the variability between machines and configurations make this an impractical task. For these pragmatic reasons, this has been taken as CPU percentage against a set of normalised synthesis processes (see Appendix J). With this consideration in mind, several algorithm efficiency scenarios are presented in appendices P and Q in order to minimise computer resources during real-time synthesis. These are derived from analogue computing techniques used in programming modular synthesisers. A technique which is explored in this research is for the proliferation of particle shapes and subsequent transformations using linear distortion. This is a core technique for generating many types of waveforms in other synthesis techniques. Here it is explored in considerable depth and always with algorithmic efficiency in mind. (Efficiency, functional performance, functional affordance)

8.2.18 Behavioural noise. Discontinuities, aliasing, distortion and other types of non-functional noise (consistency, reliability)

“Some synthesis techniques require such a bulk of control data that it actually exceeds the number of samples synthesized. (Jaffe, 1995, p83)

A related issue can be found in noise created by the discontinuities between control changes (Puckette, 2003, p96). This phenomenon is also known as *zipper* noise. On some systems, changes to a parameter might cause a specific type of audio interference. This usually happens in parameter control systems with low bit depths (7 bit or less). This was more common in older MIDI-based hardware systems and usually occurred whilst the user made faster than normal parameter changes. Other behavioural noises analysed for are aliasing, pops, crackle, and amplitude distortion.

8.2.19 Microsound behavioural analyses

Table 15 summarises the characteristics mined from Jaffe’s J4 text. These have been expanded and transformed for the objectives proposed in this research. Instead of Jaffe’s question *“How well behaved are parameters?”* the criteria are formally presented as *External Behaviour* dimensions and *Hardware Behaviour* dimensions (Table 15).

M4 External behaviour analysis
M4_A1 Consistency, robustness in response. (Jaffe, p77-79)
M4_A2 Latency and Viscosity (Jaffe, p83-84; Green, 1989, p6)
M4_A3 Error Frequency (Jaffe, p79-80)
M4_A4 Parameter Dependency Effects (Jaffe, p77, p81)
M4_A5 Behavioural Noise (zipper noise, aliasing, pops, crackle, distortion)
M4 Hardware behaviour analysis. CPU/Memory/ Efficiency. (Jaffe, p81, 82)
M4_A6 CPU Scope
M4_A7 Memory Scope

Table 15. M4 Microsound external behaviour.

8.3 J5 How robust is aural identity?

8.3.1 J5_A1. Identity in the context of variation

One could imagine that any system with a specific aural identity might lose that identity when operated beyond the boundaries in which it was designed to operate in. It could also be argued that in a complex systems such as a particle synthesiser, it is not possible to measure all the possible aural phenomena it is capable of because of the inherent plasticity of these systems. This is of prime interest to a flexible particle synthesis environment because by using very different physical and sound analysis data, one can change dramatically the character of identical microsound streams. Simulating the natural jitter in the vocal stream makes an enormous difference between a plausible and implausible FOF soprano. This attribute is therefore presented as a functional dimension in J10, “do analysis techniques exist?”

8.3.2 J5_A2. Can expression be synthesised in this technique?

Applying an external technique to a synthesiser furnished with generous modulation routing may affect the identity of a synthesiser's sound which has a wide spectral plasticity. This is the continued appeal of the analogue modular synthesiser and all of its digital representations. The context of application is central to this question. Rule based synthesis (Berndtsson, 2010), especially in particle synthesis is a point in question. Mälarsång by Michael Clarke demonstrates how the application of rules derived from the physiological analysis of belcanto singers (Sundberg, 1989) can create very plausible and expressive imitations of a soprano singer.

Two simple rules can illustrate this. Pitch jitter modulating the fundamental frequency, creates subtle irregularities in the fundamental frequency stream which help generate a more life-like quality to the final timbre. The rule determines *how much* is applied and this is kept within strict parameter values in order for it to create a plausible effect. The 2nd formant rule is another instance. If the 2nd formant frequency approaches the same frequency of the fundamental frequency a spike in amplitude will occur because of the summation of amplitudes in the same frequency (Sundberg, 1989 and Berndtsson, 2010). Thus the rule specifies that if the 2nd formant approaches the same frequency as the fundamental this is slightly shifted to avoid the resulting amplitude summation. Because these rules are data acquired from extensive physiological and physical mechanisms the J5_A2 attribute is treated as an attribute in the Analysis criterion J10.

The question of essential identity then is presented here as a question related to the most basic attributes which make up a synthesiser. The duration of microsonic particles span from the threshold of aural perception (Roads, 2001a). The frequency and duration parameters are independent etc. The question of phenomenology is treated independently in M6. As will be proposed, microsound phenomenology needs more than just the identity parameters. It also needs additional control of the basic functions through rules harvested from external data analysis. A further relevant question is “are rules built in and invisible to the user at the identity parameter level?”.

8.3.3 What defines a tool?

This research treats the question of identity as central to the design/development strategy of the functional dimensions presented in the investigation. It proposes that it should be the first question about the overall design. All other considerations and analyses follow.

It simultaneously identifies the tools primary functional purpose whilst simultaneously acknowledging the users effectiveness in achieving the sound design goal. In essence it could be seen as goal orientated design (Cooper, Reinmann and Dubberly, 2003, p6). The question of phenomenology is harder to measure and is secondary because depending on the combination, strength and versatility of the identity parameters chosen it is difficult to predict all the phenomenological capabilities of a system because like dependencies, the numerical combinations of parameters can endow a potentiality not predicted in the original design.

The derived set of identity dimensions (labelled M5) are presented in Tables 16a and 16b.

M5 Particle synthesis identity dimensions. Microsound time scale
M_A1 Particle duration. Particles of sound lasting between a few milliseconds and approximately 100 milliseconds (Roads, 2001)
M5_A2 Particle shape. Particles can have different shapes, linear, curved etc., giving rise to different timbres. These affect the spectrum of the particle. (Kaegi, Werner, Templaars, 1978) (Rodet, 1984), (Roads, 1978)
M5_A3 Formant Content. Because of the synthesis algorithms employed (hard sync paper), particles exhibit strong resonant (formant) spectrums. This enables it to model vowels and other strong resonant spectrums. (Kaegi, Werner, Templaars, 1978) (Rodet, 1984)

Table 16a. M5 Identity. Particle synthesis identity dimensions.

M5 Organization of particles. Sound object and Meso time scales
M5_A4 Time versus frequency independence. The rate at which particles can be emitted is independent from the duration of the particles. " (Gabor, 1946, p431, para 3)
M5_A5 Discreteness. Particles emitted below approx. 20Hz can be perceived as discrete sonic entities, which together form particulate or rhythmic structures. (Roads, 2001a, p296)
M5_A6 Continuity. Above 20Hz the particles fuse together as continuous tones because of the phenomena of forward masking (Roads, 2001a, p21)
M5_A7 Fusion/Fission. Groups of particles can be made to travel from a state of continuity to discreteness and vice versa. Also known as fusion to fission. This permits unique spectral and phenomenological transitions in which a continuous timbre such as the singing voice can be seamless transformed into discrete chimes yet still maintain feature of the previous timbre. (Roads, 2001a, p22) and (Ernst 1977, p34), (Miranda 2001, p7, pp2)
M5_A8 Pitch Perception. Continuous groups of particles can be made to have a strong uniformity in their continuous generation giving rise to a strong perceived sense of pitch (Pitch synchronous behaviour) or not asynchronicity. (Roads, 2001a, p24)
M5_A9 Particle Distribution. Particles can be distributed in time, acoustic space, or processing environments using any number of processes.
M5_A10 Particle Distribution Method. Particles are distributed in the time-honoured methods proposed by Xenakis using mathematical formalisms or Vaggione's manual compositional approach.
M5_A11 Particle overlaps. Because particles can maintain their duration according to the fundamental frequency it is necessary to overlaps particle in order to fit them in the present pitch cycle. This can give a sense of density in the particle stream (Roads, 2001a).
M5_A12 Multiple Time Level Composition. Particles can be composed across multiple time-levels (Vaggione, 2001, p60).

Table 16b. M5 Identity. Organization of particles.

8.4 J8 What classes of sound can be represented?.

Phenomenology

8.4.1 J8_A1. Some techniques can produce any type of sound given enough control data

There can be little doubt that synthesisers employing particle formant synthesis techniques are capable of creating a wide palate of timbres each of which is dependent on the strength and variety of the identity dimensions it is afforded with. However there are a class of sounds which are difficult to produce using some of the techniques discussed. VoSim, PulsarBTD and FOF cannot easily produce inharmonic formant spectra. Not at least in the present implementations of these systems. The reason is quite simple: they all use *hard-sync* (Brandt, 2001) in order to produce strong formant spectra. Hard-sync prevents inharmonic spectra from being formed because it forces the phase of the formant wave to zero on each start of the cycle of the particle shape. This results in a strong harmonic relationship dominated by the harmonic content of the particle shape.

“The plethora of hybrids now commonly used shows that various techniques can be combined to maximize strengths and minimize weaknesses from several areas.” Jaffe, 1995, p86)

Jaffe’s criterion question is therefore very pertinent to particle synthesis. Jaffe also considers the question in relation to identity. The greater the timbre diversity of a synthesis technique the weaker its phenomenological identity. An inverse relationship is proposed.

Two fallacies are identified:

1. Determining all the sound classes a synthesiser can represent is nearing the impossible. All we can do is review what sound classes have been created in various published works. The idea of a sound designer sitting down for the next several years trying to extract an unlimited number of timbres and

then categorising them is by any means somewhat impractical if not impossible.

“Is there a class of sounds that is impossible to produce with a given synthesis technique?” (Jaffe, 1995, p84)

2. Having a single tool from which to create every single type of microsound timbre is an enormous task and one which has already been attempted by the developers of one of the PS artefacts (Partikkel) reviewed. It proposes its own set of problems and as will be seen in some of the analysis and not without significant usability and functionality problems. Simply put, you would rarely use a hammer to extract a half driven screw in a piece of wood. A screwdriver is by far a superior designed tool for that purpose.

The problem presented in this analysis is that it is very difficult to state exactly *what* sound classes are possible from any apparatus with such variations of implementation and identity. What can be stated is that it usually takes considerable time before practitioners, using new technologies, publish critically assessed musical works. Whether the works have a narrative or are purely abstract, the criticism gained from them can inform whether the sound artefacts produced belong to a specific type of sound class. Particle synthesis has existed for some 30 years. First in academic institutions such as IRCAM and CCRMA and recently as generally available computer software. Since then, a substantial body of work has been composed and reviewed from which to mine sound class characteristics. Therefore, the method employed here looks at the implicit specifications in which the synthesis engine is created with a particular sound class purpose, for example, FOF synthesis as described by Rodet (1976) is created as a solution to the singing voice. Simultaneously it identifies literature that reviews a published musical or sound-art work in relation to a particular synthesis technique; e.g. Clarke’s 1987 *Mälarsång* concentrates on modelling Belcanto singing voices using synthesis by rule (Sundberg, 1989). Using this method, J8 is translated into microsound M6 in Table 17.

M6 Sound classes and phenomenology.
M6_A1. What is the sound class (phenomenology) strength in relation to its identity? (Jaffe, p80)

Table 17. M6 Sound classes and phenomenology.

8.5 J10 Do analysis techniques exist?

An interesting observation made by Jaffe is that by using a sine wave and enough control data, one can create extremely complex timbres which belie the simplicity of the original sine wave. Using data to drive appropriate parameters can make a substantial difference to the phenomenology offered by an otherwise simple synthesis technique.

"In fact many synthesis techniques can be made near equivalent when fed enough data. A single sine wave that is frequency modulated very quickly with a complex signal can produce intelligible speech" (Jaffe, 1995, p83)

Useful synthesis data is often derived from the analysis of recorded sounds. Jaffe focuses his text on frequency and spectral analysis (Jaffe, 1995 p85) a common approach to frequency domain synthesis.

8.5.1 The Chant model of production

FOF synthesis, was conceived for the synthesis of the human singing voice. It leaned more towards the frequency domain.

"CHANT was developed at IRCAM by a team led by Xavier Rodet (Rodet, Potard and Barrière, 1984). As the name of the program suggests, imitation of the human singing voice was one of the main aims of the original project. It employed a new synthesis algorithm developed by Xavier Rodet, FOF (Fonction d'Onde Formantique)

synthesis or, in English, Formant Wave Function synthesis. Each FOF generator provides detailed control over the spectral shape of a single formant region. “ (Clarke, 1996)

8.5.2 Synthesis by rule

There are substantial benefits in deriving formant frequency data using frequency analysis so that plausible vowels may be synthesised. The reader will recall that this approach has been successfully applied to FOF synthesis, using the *Chant* system and more recently *Diphone*. In essence it is a *synthesis by rule* system. It is by no means the only rule-based system for synthesising the human voice. Other systems have been used to drive speech rather than singing interfaces, using alternative rule systems (Weinschenck, 2000, p106). Synthesis rules are algorithms of varying complexity that direct certain parameters to behave in such a way as to imitate established complex sound phenomena. An example of such a rule is called the *second formant rule* (Sundberg, J. 1978). When a fundamental frequency approaches the frequency of the second formant, there is a summation of amplitudes, which can result in amplitude distortion. The rule detects this coincidence and shifts the second formant frequency enough to avoid distortion. Clarke (1996), driven by compositional needs, investigated alternatives to frequency domain analysis. After having worked with IMPAC, a time domain based granular synthesis distribution based system, he decided that a hybrid frequency/time system would potentially benefit the time domain plasticity offered by FOF, in terms of generating microsound

“Like most granular synthesis programs its primary concern was the distribution of grains of sound in time. It was not especially concerned with the frequency domain.” (Clarke, 1996, p108)

Clarke achieves this approach in his work *Mälarsång*.

Sundberg (1978), dedicated substantial part of his research to analysing and classifying vocal data in order to study the relationship between anatomy and

vocal behaviour. This work has been essential in the design of synthesis rules and to inform vocal synthesis systems such as FOF. *Chant* is heavily informed by Sundberg's work (Rodet, et al, 1984)

8.5.3 Data acquisition and driving

It is often the case that when the term *sound analysis* is discussed amongst synthesis practitioners, it invariably defaults to frequency domain analysis. The term however, could be broadened to general data acquisition with many contexts including, particle synthesis systems. A whole host of data can be derived which includes the time and frequency domains. It could also be expanded in this particular context to encompass multiple time levels. It is proposed therefore that complex rule based systems are a primary component of the synthesis package. As the reader will recall (3.4), the synthesis of sound particle extends beyond the microsound level to include composition (Risset, 2005). To be used effectively, rules cannot be confined just to drive the level of the note or sound object but also to musical phrases or sequences of particles in the meso-time level. This area intersects with music informatics. To illustrate the kind of rules being discussed the following summary is presented below. It is derived from acoustic and anatomical studies of singers, which have been successfully used for singing synthesis. Table 18 contains examples of analysis informed rules (Sundberg, 2009).

Spectral tilt
<i>"In the human voice, as well as in most musical instruments, an increase in sound level is associated with a decrease in spectral tilt..."</i>
Harmonic charge
<i>"The remarkableness of chords in a given harmonic context is quantified in terms of the harmonic charge (CH). It is computed from the melodic charge of the chord tones as related to the root of the tonic"</i>
Phrasing
<i>"Rules for marking sub-phrases. and the final tone of the melody operate on signs, which can be added in the score file."</i>
Vibrato tail
<i>"The vibrato frequency (VF) is speeded up toward the end of a tone. This is referred to as the vibrato tail (Prame 1984). The vibrato tail is modelled after measurements of human singers."</i>
Diphthongs
<i>"The first vowel receives 65 percent, and the second vowel 35 percent, of the duration of the diphthong. These relations may be style dependent."</i>
Tracking formants In high tones
<i>"F0 may surpass the first formant frequency (F1). By varying appropriate articulatory parameters (such as increasing the jaw opening), F1 can be raised to a frequency slightly higher than FO (Sundberg, 1975)."</i>
Overtone singing (optional)
<i>" Overtone singing is a special singing technique used in some parts of Asia (especially Mongolia...A particular single overtone can be made clearly audible by tuning the 2nd and 3rd formants to a frequency very close to that of the overtone. In the synthesis, a useful distance between these formants has been found to be about 100 Hz"</i>

Table 18. Singing synthesis rules system (Sundberg, J. 2009).

8.5.4 Accelerometers

Sound analysis for the purposes of feeding data to particle synthesisers, is referred to in this investigation as *data acquisition and driving*. An example of a method of data acquisition is provided as an illustration of the possibilities offered by this approach. We saw above how micro-fluctuations can disturb the fundamental frequency of the voice. Accelerometers have become common data gathering systems and are often built into smartphones and tablet computers. Analysing the disturbances of an accelerometer in a moving vehicle, offers many possibilities for data driving particle synthesis parameters including an alternative to the anatomical source of jitter in the fundamental frequency. We have all experienced, at some time in our lives, the fundamental frequency jitter modulating our voice as we try and talk or sing in a moving car on a bumpy road. Many other types of modulating data could be acquired in order to simulate realistic sound effects, such as climbing stairs, gaits systems from the animal kingdom, mechanical vibrating systems, such as motors. Furthermore, it would provide a system which facilitates hybrid sounds providing an extension to the work proposed by Clarke and others for composing between the time and frequency domains. With this rationale the criterion analysed in this section expands from the frequency domain analysis of sounds to also include time and amplitude domain data analysis systems. This encompasses rule-based systems for controlling the particle synthesis (Table 19).

M7 Do data acquisition and rule based systems exist?
M7_A1. <i>"You need the tools to derive the proper parameter values from a specification of a desired result" (Jaffe, p85)</i>
M7_A2. <i>"Are rules built in and invisible to the user at the identity parameter level or can an external one be used to drive and control the synthesis? " (Jaffe, p85)</i>

Table 19. M7. Data acquisition and rule based systems.

This concludes the re-mapping of Jaffe's criteria for use as microsound specific criteria driven analysis scheme in order to evaluate current cognitive and functional dimensions in PS systems. The above classification could be expanded and transformed further. It has been purposely limited in order to

make it practical to develop this primary investigation and create the proposed development framework. To this purpose, it is speculative.

Chapter 9 Design analysis process applied to MS specific criteria

9.1 The catalogued MS design criteria

Collating the CD and FD microsound criteria and attributes from chapters 7 and 8 respectively, produces Table 20 of microsound criteria and attributes. Each criterion (in bold), is followed by its attribute(s). Note that the new classification has resulted in seven criteria instead of Jaffe's original ten. The criteria is split between CDs (Table 20a) and FDs (Table 20b)

M1. Cognitive complexity 1. Notations and notation control
M1_A1. Flexibility of Notations
M1_A2. Notation Mapping
M1_A3. Notation Constraints
M1_A4. Dependencies/influence
M1_A5. Rules to control dependencies
M1_A6 Meta-parameters
M1_A7. Premature commitment
M1_A8: HMO. Hard mental operations
M2. Cognitive Complexity 2. Perceptibility of parameter changes.
M2_A1. Parameter strength
M2_A2. Rule based parameters

Table 20a. Taxonomy of the CDs for the analysis of formant based PS.

M3 Are there means to imitate physical and environmental acoustic phenomena?
M3_A1. Is it possible to mimic physical phenomena such as materials or environments?
M4 External Behaviour analysis
M4_A1 Consistency, robustness in response.
M4_A2 Latency and viscosity
M4_A3 Error frequency
M4_A4 Parameter dependency effects
M4_A5 Behavioural noise zipper noise.
M4 Hardware Behaviour Analysis
CPU/Memory/ Efficiency.
M4_A6 CPU scope
M4_A7 Memory scope
M5 Particle Synthesis Identity Dimensions. Microsound Time scale
M5_A1 Particle duration.
M5_A2 Particle shape
M5_A3 Formant waves.
M5_A4 Formant harmonicity
M5 Organization of particles. Sound object and Meso Time Scales
M5_A5 Time versus frequency independence.
M5_A6 Discreteness/continuity
M5_A7 Fusion/fission
M5_A8 Pitch perception
M5_A9 Particle distribution
M5_A10 Particle distribution method
M5_A11 Particle overlaps
M5_A12 Multiple time level composition
M5_A13 Exotic microsound features
M6 What is the sound class strengths in relation to identity?
M6_A1. "What is the sound class (phenomenology) strength in relation to its identity?"
M7 Do data acquisition and rule-based systems exist?
M7_A1. "You need the tools to derive the actual parameter values from a specification of a desired result"
M7_A2. Is the analysis and rule based system built into the package or can an external library be used to do so?

Table 20b. Taxonomy of the FDs for the analysis of formant based PS.

9.2 Method for applying the MS criteria to empirical design analysis

The design and implementation analysis process follows in such a manner that the criterion attributes form the basis of individual studies. Each of the seven PS artefact's CD and FD criteria and attributes are analysed and recorded onto the appropriately labelled spreadsheets on the accompanying pen-drive. The data from the analyses spreadsheets is described and reported in appendices B to O. Each spreadsheet is clearly referred to at the beginning of each individual appendix report. Each MS criterion attribute is operationalised according to a set of metrics depending on the type of study. This is described at the beginning of each appendix report.

9.2.1 Analysis methods

The analyses methods vary depending on specific attributes. For example, in M1 the attributes M1_A1 to M1_A3 are analysed using simple heuristic evaluations, whereas M1_A4 is a set of more elaborate descriptive statistical analyses in which the frequency and averages of dependencies are mapped out to ascertain operational characteristics. These analyses spreadsheets are detailed over four grouped artefact worksheets and seven individual sheets, which help focus on individual PS artefacts. One important characteristic of the spreadsheets is that each one is duplicated so that the reader can have text notes and comments about the data in the worksheet cells and another identical spreadsheet with just the pure numerical data. This was necessary, in order that Excel calculations can be processed (all are clearly marked). This method is implemented for all the criteria and attribute analyses. The text comments in the cells accompanying the numerical data help explain some finer detail about the numerical results, which in turn inform the analyses reports. In some cases they are handy breadcrumb tracks to return and further study parameter analysis details. In the case of the M1.A4 group of analyses, these are used to inform the FD analysis of parameter dependencies in "M4_A4.xlsx". Each analysis

appendix (B-O) report, together with its appropriate spreadsheet(s), describes the nature of the data harvested and its usefulness in informing a successful design of this class of software synthesiser. Once the analyses were completed, the results were collated into a detailed descriptive MS design criteria summary (appendix P) of which a compact version is presented in the next chapter (chapter 10). These form the basis for informing new or existing expanded PS system implementation. The following diagram (Fig.26) gives a pictorial representation of the design analysis framework process.

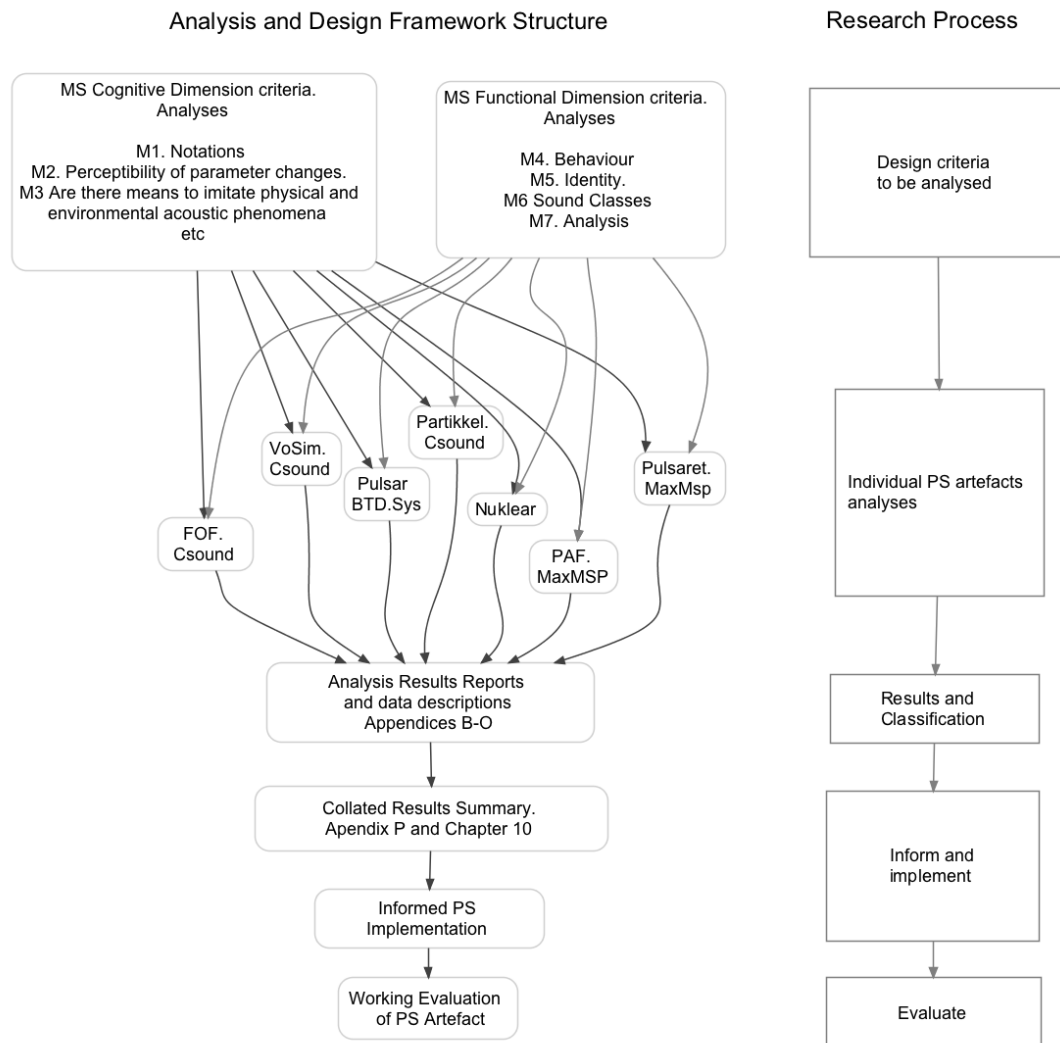


Figure 26. PS Design Framework summary.

9.3 Navigating the analyses reports in appendices B-O

At the beginning of each appendix, the title is followed by the location of its appropriate data sheet association in order to aid navigation. Each appendix report, describes from the outset the associated spreadsheets, documents and metrics as thus:

Associated analysis

M1_A5_CD.xlsx

M1_A4_CD.xlsx

Metrics

0.0 It is not possible to route rule data

0.5 The parameter can be routed using the

*Csound or Max language or through an external
event control protocol such as OSC or MIDI*

1.0 There are rules embedded into the system"

9.4 Summarising the analyses report's

As indicated in 9.2.1, appendix P is a collated summary of all the MS design criteria analyses reports presented in the appendices. It was reasoned that because of the amount of analyses data and outcomes, summarising them in one place would provide a "simple to navigate" textual map of significant conclusions with which to explore key aspects of the investigation results. Furthermore, this process prepares a means for representing all the descriptive outcomes in graphic form for the purpose of visualising the main outcomes. These two approaches create a succinct evaluation "geography", which is easily cross-referenced. Being able to access outcomes in this way has one other specific advantage. It brings into focus the original discursive power of Green's cognitive dimensions, with the distinct advantage of being inferred by empirical

methods. It is concluded that this approach greatly helps in visualising key design relationships and consequently applying them to new artefact implementations.

Chapter 10 then, presents a compact textual and visual summary of the results in relation to appendix P and specifically the results which are used to inform a new PS implementation.

Chapter 10 Summary of the design analysis framework results

Associated documents

Appendices P, Q, R

Appx P Outcomes Summary Tree Map.pdf

Appx P Reduced OSTM.pdf

10.1 Distilling the CD and FD analyses results

As discussed in the previous chapter, appendix P presents the detailed collated outcome excerpts from the microsound criteria analyses results reports in appendices B to O.

P.2.5 Consistency of menu elements

Avoid leaving "information holes" in unexpected locations in a data set, i.e., empty presets at the beginning of a preset array (from C.1.2. bullet point 5)

reference to originating analyses report

P.2.6 Narrowing the scope of identity parameters

Avoid narrowing value boundaries in core *identity* functions (see M5) for example, particle organisation parameters, which limit the transposition from one time, scale to another. Instead, afford existing auxiliary core functions such as LFO, with similar functions and scope (from C.3.1)

yellow highlight indicates paragraph used to inform new design

P.2.7 Rules for dependencies

Any parameters, which are dependent on a core identity parameter, such as particle *duration*, should not be permitted to run beyond the absolute limits of the parent parameter (from C.4.2).

Figure 27. Appendix P table format

At the end of each appendix P outcome paragraph, the original appendix B to O paragraph number is annotated using the standard format of the thesis contents, for example "from C.3.1" (see mid paragraph in Fig.27). Some of the entries are not ad-verbatim copies but suggestions, inferences or conclusions. In these cases, the paragraph number is noted next to the paragraph in question. Finally, those entries which have been identified as being useful for applying the analyses results to a new PS artefact implementation, are highlighted in yellow (Fig.87). This is a quick-hand referencing mechanism in order to make it easy to collate specific outcomes to inform design. These have informed many of the ideas presented in the detailed implementation appendices Q and R. This is discussed further below in 10.3

10.1.1 Visualising and reporting the analyses outcomes

During the collation of the results, it became apparent that there were common themes which appeared repeatedly in many of the analyses reports. For example, the paragraphs P.1.2, P.3.3, P.6.1 and P.14.4 discuss similar and related analyses outcomes. These are related to binary switching parameters, morphing of parameters and composition. The first two paragraphs discuss the negative side effects of parameters with only binary on/off states and the consequences of these on the smooth morphing of values. P.6.1 also discusses morphing and the overall compositional process. Consequently, it was decided to map all appendix P paragraphs in such a way as to create a visual tree map of common outcome design areas. This was realised in graphical form and is abbreviated for convenience as the OSTM (Outcomes Summary Tree Map). The resulting map is titled *Appx P Outcomes Summary Tree Map.pdf* on the accompanying pen drive. It gives a broad categorised visualisation of the reporting process detailed above. A detail of the map is shown in Fig.28

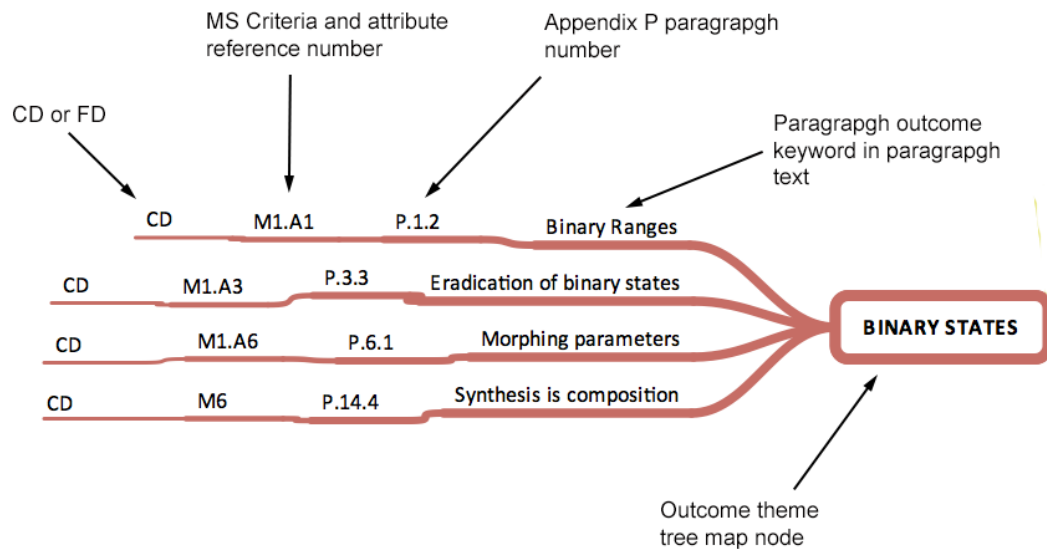


Figure 28. Detail of Appx P OSTM showing the "Binary States" node.

Aside from providing a clear categorical view, the OSTM also provides a further function in facilitating the cross referencing of other branches from different nodes, in which further design relationships may be explored and considered for alternative implementations. The simultaneous classification and visualisation of repeating outcomes resulted in the following list of 11 design and implementation themes:

1. Binary States and morphing issues
2. Multitemporal composition issues
3. Interface and usability issues
4. Metaparameters
5. Analysis tools
6. Sound classes
7. Dependencies
8. Identity
9. Premature Commitment
10. Functional Behaviour
11. Missing and redundant parameters

Out of the 11 design analyses outcome themes, the following merit special attention in terms of the design issues revealed in the analyses outcomes:

Interface (24 branches), dependencies (13), functional behaviour (9), multitemporal composition (7) and identity (7).

Each branch in the OSTM may have more than one outcome paragraph as in the case of the "analysis tools" node which has one branch covering two outcome paragraphs P.15.1, P.15.7 (Fig.29). As will be seen in 10.3, this approach offers a way to reduce the branch count in the case of remodelling the map for informing implementations.

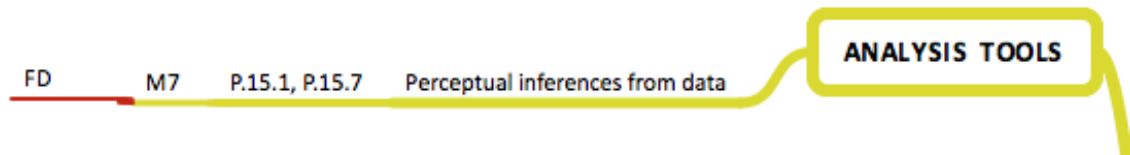


Figure 29. Multi-paragraph branches.

10.2 Validating informal observations in chapter 5

Studying the OSTM in detail and referencing the branches to the appendix P summaries, it becomes clear that there are many design implementation outcomes. These confirmed the issues first found in the informal software review in chapter 4 and subsequently discussed in chapter 5. They are presented in the following paragraphs in the original order in which they were reviewed in chapter 5.

10.2.1 Multitemporal composition (5.1)

In 5.1, it was noted that there was a disparity of multitemporal particle organisation techniques across the current implementations. Amalgamating the

results from the M1.A1, M4 and M5 analyses, it is determined that the informal observations had merit and that there is ample scope for developing this area with a more unified approach to particle composition (homogenising differing particle organisation techniques) and unifying the composition and synthesis tasks into a single process (see Fig.30) This is also echoed in the "Binary States" P.14.4 node branch. A significant group of outcomes (P.12.11, P.14.2 and P.14.3) point at the lack of mechanisms for manually or procedurally organising particles in the meso and macro time scales.

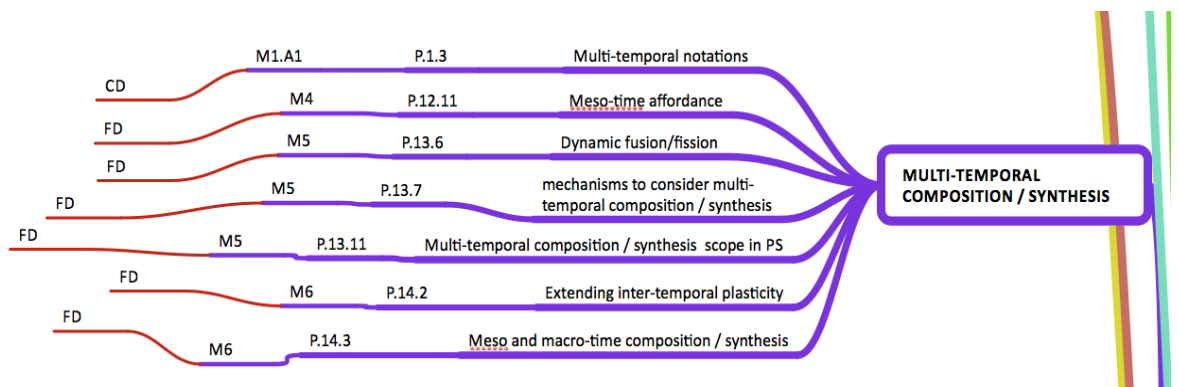


Figure 30 OSTM Multitemporal Composition / synthesis node.

The M1.A1 analyses deals with notation issues with regards particle organisation. In M4 and M5 however, the reader can see the extent of the particle organisation parameter scope for facilitating multitemporal composition. (Fig.31).

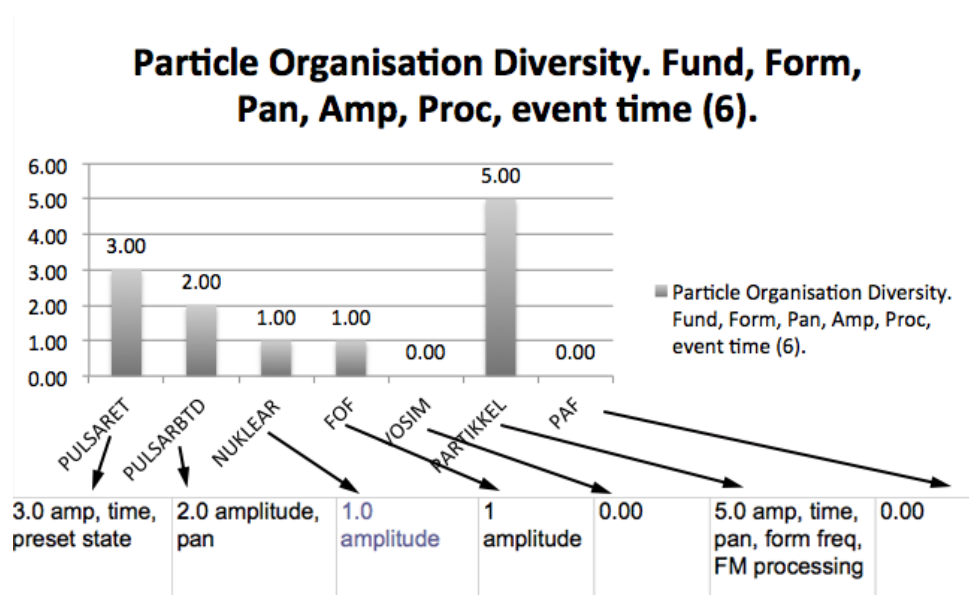


Figure 31. M5 excel screenshot detailing PS particle organisation scope

10.2.2 Interpretation of implicit specifications and multiple mental models (5.2, 5.4, 5.5)

The issue of separate mental models observed in PulsarBTD and Pulsaret (5.2) is also confirmed in the analyses results. It is shown in branches P.1.4, P.1.8 and P.13.5 in the Interface node (Fig.32).

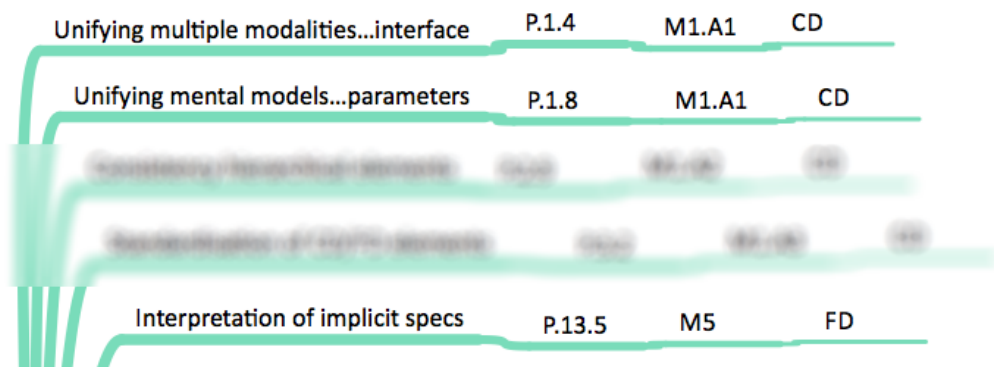


Figure 32. Implicit specifications and mental models branches.

One observation made in the original software review, with regards to differing interpretations of the implicit specifications, is that users have no choice but to learn different mental models (5.3), in order to bring together the tools for benefiting from the full scope of the sound design of PS. The MS criteria analyses confirm this in the spreadsheet analyses (M1.A1, M5) and can be visualised across a variety of the tree nodes including branches P.4.4 in the Missing / Redundant Parameter node, P.2.9, P.13.5, P.7.4 in the interface / usability node, 2.6 in the identity node.

This set of outcomes also confirms the observations reported in 5.4 and 5.5, with regards to the distilling of disparate features and multiple modalities of single features.

10.2.3 Idiosyncratic limitations (5.3)

It was noted in 5.3 that one of the issues raised was the problematic combination of hardware and CPU constraints because of the need of different and often incompatible hardware and OS configurations in current PS implementations. The combined empirical studies of the CPU, memory and functional behaviour clearly demonstrated a major limitation in working with complex environments such as Csound. The Csound opcodes FOF, VoSim and Partikkel FD analyses (M4_FD.xlsx) confirmed, through measurement (Fig.33), the serious functional limitations observed in 5.3. The branches of the functional behaviour node are almost exclusively a direct result of those systems (Fig.34).

Memory Readings	Host CPU unloaded	Synthesis Only. CPU% Low	Synthesis Only. CPU% High	SYNTHESIS CPU Low	SYNTHESIS CPU High	REAL MEMORY HOST	REAL MEMORY SYNTHESIS
PULSARET	7.45	8.4	111.65	15.85	119.1	175.5	363.9
PULSARBTD	0.365	0.135	6.525	0.5	6.89	82.1	82.2
NUKLEAR	7.785	3.88	72.28	11.665	80.065	174.5	277
FOF	2.445	97.155	114.585	99.6	117.03	55.1	89.1
VOSIM	2.241	98.954	104.249	101.195	106.49	54.5	89.8
PARTIKKEL	2.2495	98.8605	116.7005	101.11	118.95	54.6	93.7
PAF	12.774	0.03711	0.03711	12.81111	12.81111	73.4	84.3

Figure 33. CPU Functional analyses.

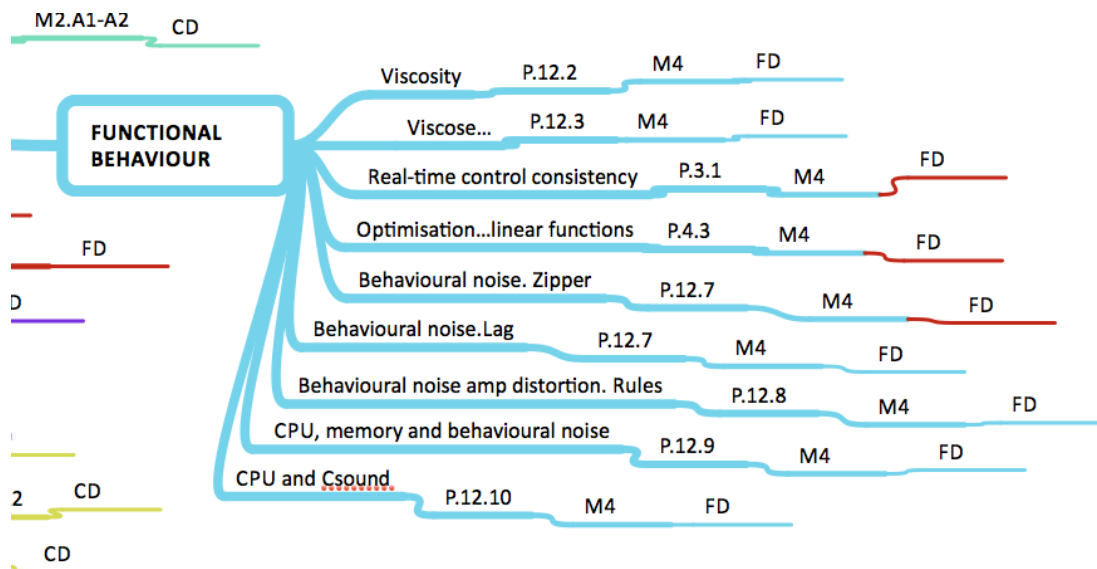


Figure 34. Functional behaviour branches.

Significantly, the M4 studies results revealed new information not detected in the informal software reviews; the systems have many related functional problems (some of them severe) in other MS criteria analyses including dependencies (P.4.1, P.4.2 and P.10.3) and interface/usability (P.2.11 and P.7.2). Such a set of analyses outcomes could be decisive in the approach to any new implementation.

10.2.4 Too many features (5.6)

The software review noted that Partikkel is furnished with a comprehensive set of parameters in order to cater for a wide variety of PS techniques. It was also noted (5.6) that in doing so it requires that users commit parameters prior to DSP execution. The formal analyses (M1.A7, M1.A4 and M1.A2) confirmed this in a variety of ways. P.7.4, for example, reports that the M1.A7 study clearly shows that Partikkel requires that many parameters are pre-calculated before runtime in order to control glue parameters effectively. Furthermore, it can be argued that the combination of PC (P.4.5, P.8.1 and P.8.2) and HMO (P.7.2 and P.7.4) outcomes infer a more streamlined approach to feature sets that do not

rely on software designs which introduce audio interruptions. The secondary functional side-effects of designs with high HMO, PC requirements show a clear impact in many of the M4 analyses summarised in the branches shown in Fig.34.

10.2.5 Missing / redundant core features (5.8)

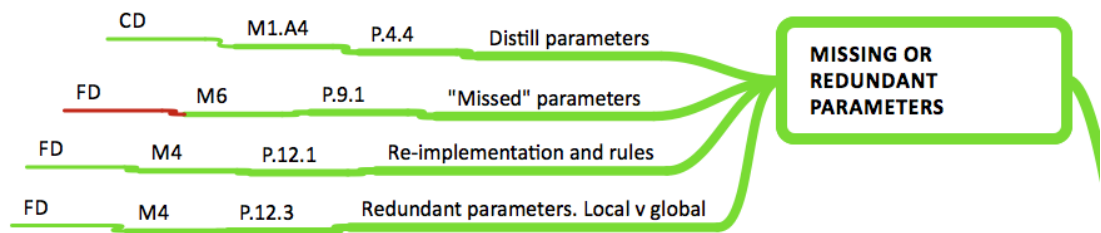


Figure 35. Missing and redundant paramters

The observation in 5.8 with regards to missing core parameters / features, was confirmed in studies M1.A4 and generally in M6 and M4 (Fig.35). Whereas M1.A4 discusses parameter redundancy, more intriguing is the omission of vibrato and jitter as core parameters in the engines such as FOF, and VoSim. Intriguing because the original specifications (Rodet, 1984; Tempelaars 1977) present these techniques as singing voice models. Results of this kind suggest a strong case for implementations which improve the original interpretation of the synthesis technique.

From a more functional perspective, the M4 (P.12.1, P.12.3) analyses demonstrate there is a case for unifying some of the differing interpretations of glue parameters (specifically, particle overlap implementations) for the purpose of minimising many of the observed dysfunctions. Furthermore, the studies reveal the importance of minimising duplicate parameters which are present simultaneously in local and global interface locations (PulsarBTD).

10.2.6 Glue parameters and standardisation (5.9, 5.10)

The software review concluded, in 5.9 and 5.10, that the lack of standardisation amongst the PS artefact's core and compositional feature was one of the main issues inherent in current PS systems. The empirical analyses documented in this research confirm that much of the detail in those inconsistencies resides within the differing implementation of core parameters and the divisive approach to compositional affordance in the systems.

10.3 Interpreting and distilling, reducing the results for implementation

The appendix P summary together with the OSTM, serve as a broad map of outcomes from the results recorded in the analyses. In order to use this information to determine the scope of a new design it must be distilled and reduced to a practical size in order to address specific implementation issues. The functional and usability saliences observed in chapter 5 and confirmed in the analyses serve as a good starting point in determining which outcomes are used to directly inform a new implementation. With this purpose, outcomes considered in this light have been highlighted in yellow in the main appendix P summary. Here they are presented visually as a reduction of the original OSTM. This offers a practical working document (Fig.36) with which to reference the original Appendix P outcome paragraph and a pointer to the data in the analyses spreadsheets. The reduction is accomplished by grouping common outcome paragraph themes on one branch.

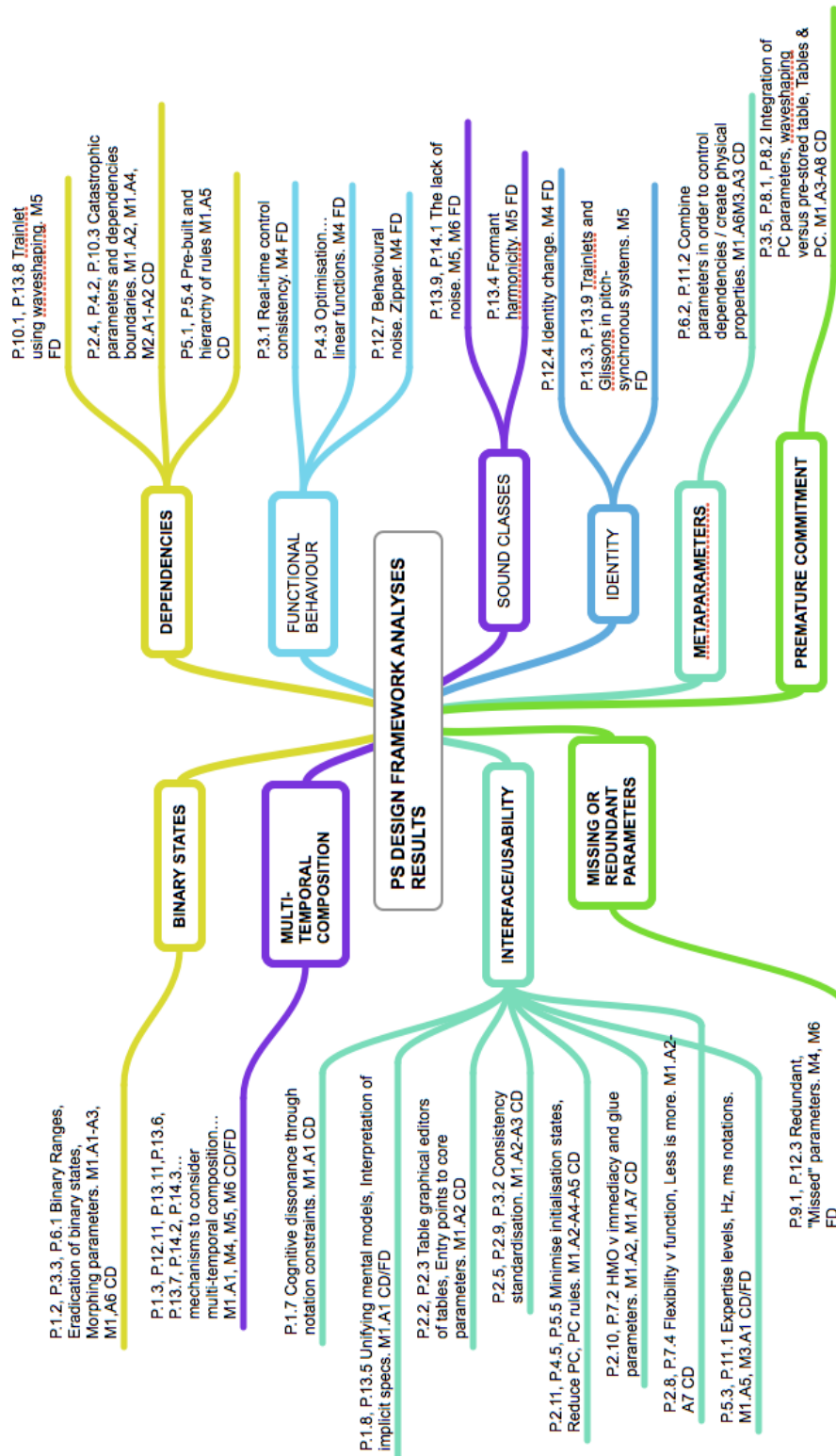


Figure 36. Reduced OSTM as a result of similar analyses outcomes

The reduced OSTM entries make it easier to list an implementation strategy such as the example below in Table 21.

Minimise major behavioural dysfunctions such as parameter dependency management and catastrophic parameter side effects, behavioural noise. P.12.7,P.2.4, P.4.2, P.10.3 (5.3 Idiosyncratic limitations).
Feature disparity integration and missing parameters. Trainlets and Glissons, formant inharmonicity, vibrato, jitter, noise sound classes, P.8.1, P.10.1, P.13.8, P.13.4, P.13.9, P.14.1, P.13.3, P.9.1, P.12.3 (5.8 Missing / Redundant features).
Comprehensive PS multitemporal composition features which reduce the limitations observed in P1.3, P12.11, P13.6, P.13.7, P.14.2, P14.3 and consequently reducing binary states P.1.2, P.3.3, P.6.1 (5.1 Multitemporal composition).
Metaparameters for controlling dependency and for creating physical properties P6.2, P.11.2. (5.6 Too many features, 5.9, 5.10 , Glue Parameters and Standardisation).
A. Expertise levels, B. PC, C. HMO and D. libraries P.55.3, P.11.1, P.2.11, P.4.5, P.5.5. (5.2, 5.4, 5.5 Interpretation of implicit specifications and multiple mental models).

Table 21. An example PS implementation strategy

In order to evaluate the PS design analyses framework, the following chapter uses the outcomes from the reduced OSTM arrived at in Fig 36 in order to inform an alternative PS implementation. This addresses the outcomes described in appendix P and visualised in the reduced OSTM. For convenience and in order to align with the original issues, the chapter 5 software review paragraph numbers (*italics*) are attached at the end of each entry of Table 21. Each individual developer will interpret their own artefact analyses in a manner that suits their design goals.

Chapter 11 Informed implementation

11.1 Particle synthesis using wave-orientated synthesis engines

The full detailed account of the PS artefact implementation together with appendix P paragraph markers can be found in appendices Q and R.

11.2 The Elementary Signal Engine

The Elementary Signal Engine (ESE) is the resulting artefact informed from the design analysis framework. The ESE is a standalone sound particle synthesis / composition environment. It consists of a wave-orientated PS and composition feature set which are organised by independent parametric libraries and which are driven by a metaparameter system. This controls parameters in real-time in order to create complex morphing microsound textures. It is able to do so across multiple time scales.

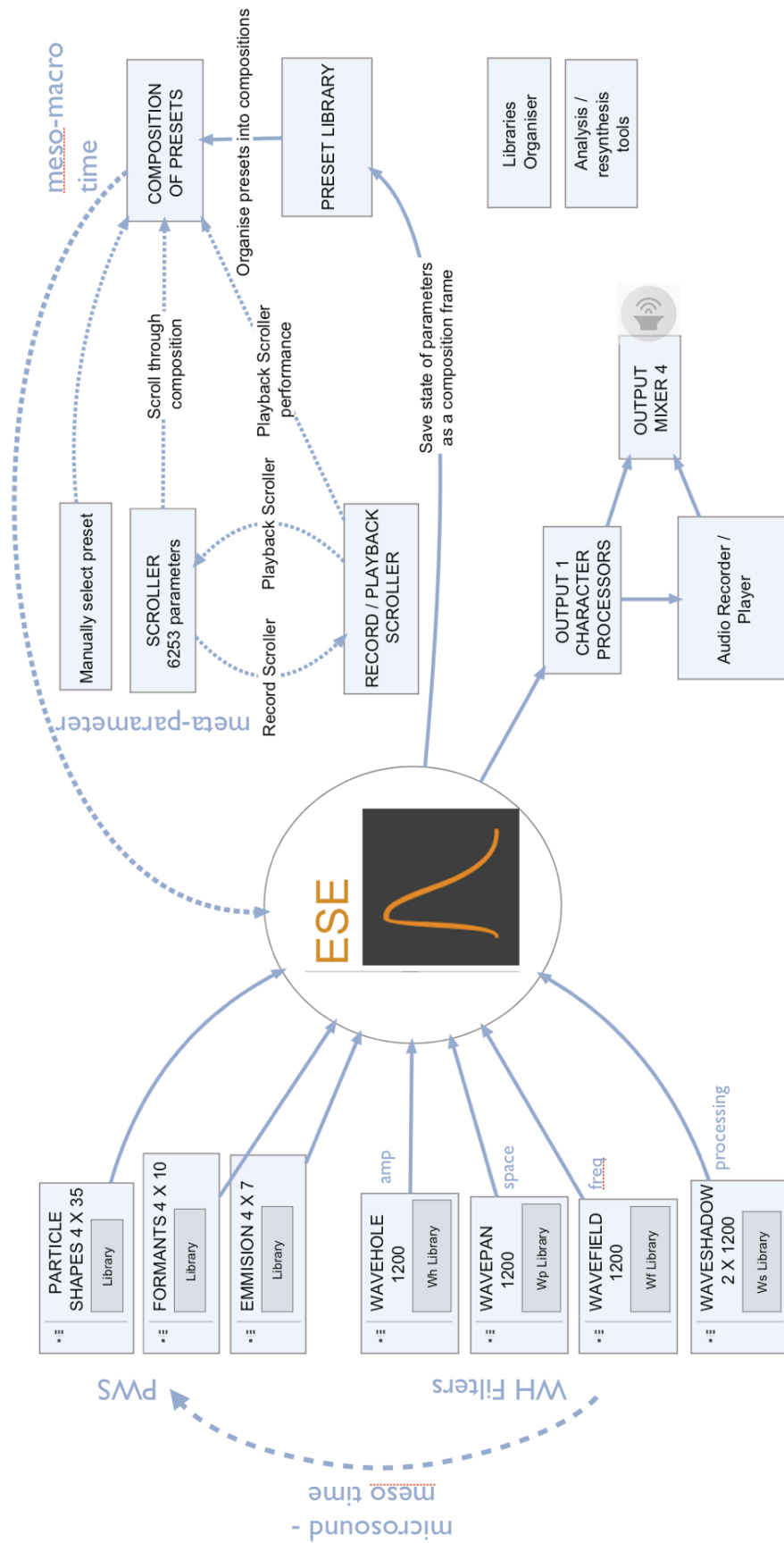


Figure 37. The Elementary Signal Engine component diagram

11.3 ESE architecture

The ESE consists of four main modules termed the Particle-Wave Stream (PWS) synthesis engine, the Wavehole filter (WH), the *Scroller* (metaparameter) and individual *component* libraries (Fig.37). An example of a component is the formant engine parameters in the PWS module or the wavepan, wavefield and waveshadow parameters in the Wavehole filter (WH).

The PWS deals with the main sound particle synthesis functions, such as generating formant rich particles at a variable emission rate. It also generates an audio rate master phase timer from which the WH functions derive timing information. The WH system is a set of phase-synchronised pulse masking filters which afford the control of individual particle amplitude (wavehole), spatial location (wavepan), frequency (wavefield) and processing (reverb, echo, etc.).

"Wavehole" with a capital "W" refers to all the various processes, such as a "wavehole", "wavepan" etc. Individually, they are presented in lower case. An important disambiguation of the term filter is order. The term "filter" in the Wavehole does not refer to spectral filters; rather it is a convenient term because the main function of a Wavehole filter is to filter out some audio information be it frequency, amplitude, process or spatial location.

Each component of a module has its own library of data which are either the result of a real-world spectral analysis as might be in the case of the formant engine module or data acquired through some procedural, formalist method as in the case of the Wavehole. The data can be changed manually at anytime and added as a library preset. Several experimental data analysis modules have already been implemented but not yet added to the current version of the ESE. These include particle shape extraction, jitter extraction from accelerometers and Wavehole envelope extraction.

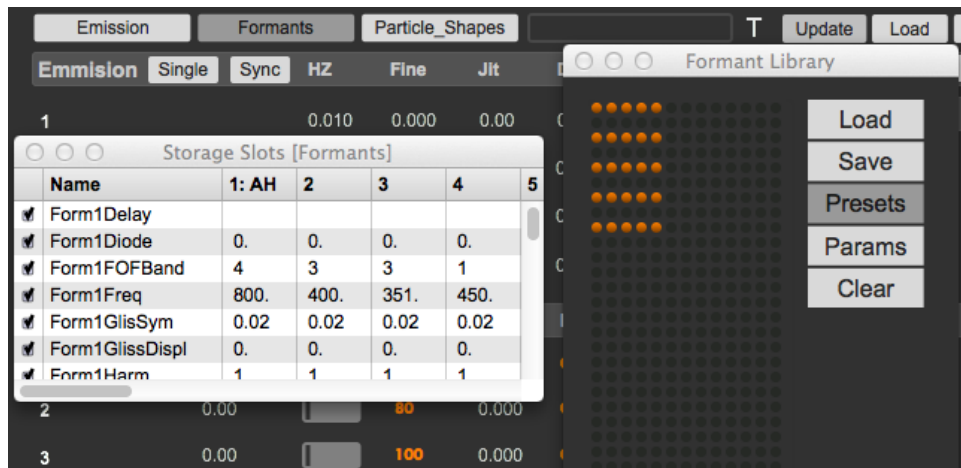


Figure 38. Formant module library interface and parametric data.

11.4 The Particle Wave Stream. PWS introduction

The ESE has four PWS generators. This number was chosen in order to afford the PWS with the capability of generating relatively realistic vowel timbres. Rodet (1984) suggests that five formants are the minimum needed. The four PWS generators have a novel feature which doubles the number to eight. This is discussed further below in 11.4.4. The PWS consists of three main components: particle shapers, formant engine and an emission generator. These work together to create periodic unipolar wave functions which act as amplitude envelopes (particle shapes) in order to modulate a carrier wave (Q.1.2). The rate of emission is the fundamental frequency. Despite being wave-orientated, the particle duration can act, albeit with a caveat, independently from the fundamental frequency (Q.5.5). The *stream* is perceived as discrete sound particles at sub-audio rates and continuous tones at audio rates.

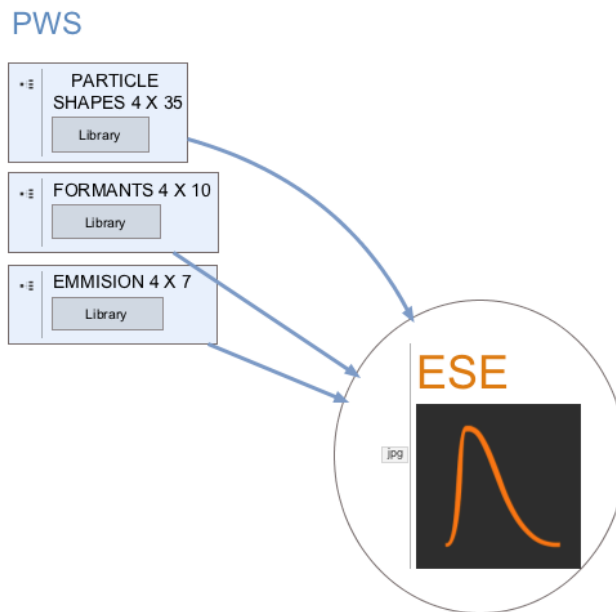


Figure 39. The three main PWS components.

11.4.1 Emission

The emission component (Fig.40) determines the emission rate in Hz with the added option of a fine frequency adjustment. The implementation approach in the PWS is that the fine frequency adjustment is offered in cents so that users with musical training can explore fundamental frequency intervals (between the 4 layers) using other musical temperaments than just equal temperament.

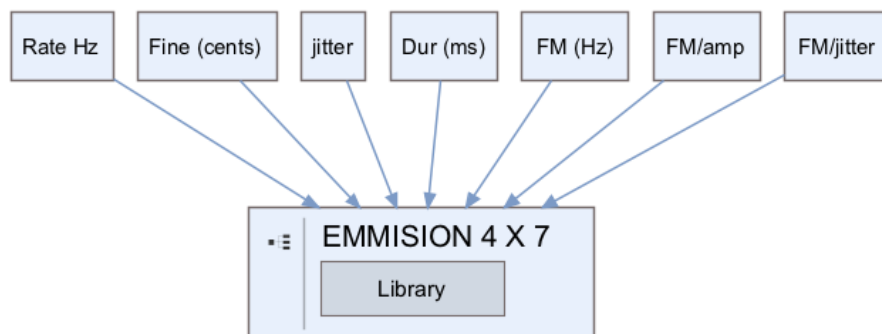


Figure 40. PWS emission component structure.

The Jitter and FM parameters add two essential components to the emission component. These have been overlooked in the canonical Csound FOF and VoSim generators (P.9.1). They are included for the effective synthesis of vocal sound classes. The FM can be used to add vibrato at low settings or audio FM sound classes at extreme values. The jitter circuit allows the user to imitate the slight vibrato frequency variations which are common in real voices. These are essential parameters in modelling vocal sound classes.

Emmision	Single	Sync	HZ	Fine	Jit	Dur	A-Dur	FM	FM Amp	FM Jitt
1			1.140	-0.060	0.10	1006.	877.2	1.61	0.71	0.15
2			1.254	-0.050	0.08	813.0	797.4	1.61	0.71	0.15
3			1.480	-0.050	0.05	735.0	675.7	1.61	0.71	0.15
4			1.500	-0.070	0.04	737.0	666.7	1.61	0.71	0.15

Figure 41. ESE emission interface. Orange values are real-time display units only.

The duration (*Dur*) parameter can be used to control the length of the particle as long as the emission rate is below the maximum wavelength. For example, a PWS emission rate of 10Hz gives a particle duration of 100ms. As long as the emission rate is below 10Hz, the duration of 100ms can be frozen. It automatically changes when it reaches the 10Hz threshold again and retains this until the users changes the duration threshold. This is the caveat with regards to using wave-orientated synthesis for the purpose of generating sound particles. It could be argued however, that at emission rates higher than 10Hz and towards 20 upwards, the ear perceives any changes of duration as high pass filter effects in the timbre (similar to PWM). In granular synthesis, this is not a problem because the grains are generated as independent sound entities and thus, the duration can be much longer than the limits of its emission rate (density). This is an essential difference between the ESE and traditional particle synthesis methods. Two other features have been added in order to improve usability and reduce interface viscosity. The first is the "Single" button which allows the user to change the values of layers 2, 3 and 4 simultaneous by

adjusting layer 1 thus saving users repetitive work when working with similar layer parameter values. The "sync" button is discussed below in relation to the Wavehole layers.

11.4.2 The formant component

The formant generation (Fig.42) in the PWS is created using the classic phase synchronising technique called "hard-sync" (Strange, 1984). The technique is not new to the PWS and can be found as the basis of many other sound generators, including formant generators. What is unique about the PWS formant implementation (see Q.7.1) is that it allows the user to alter the degree of phase synchronisation in order to generate inharmonics into the timbre.

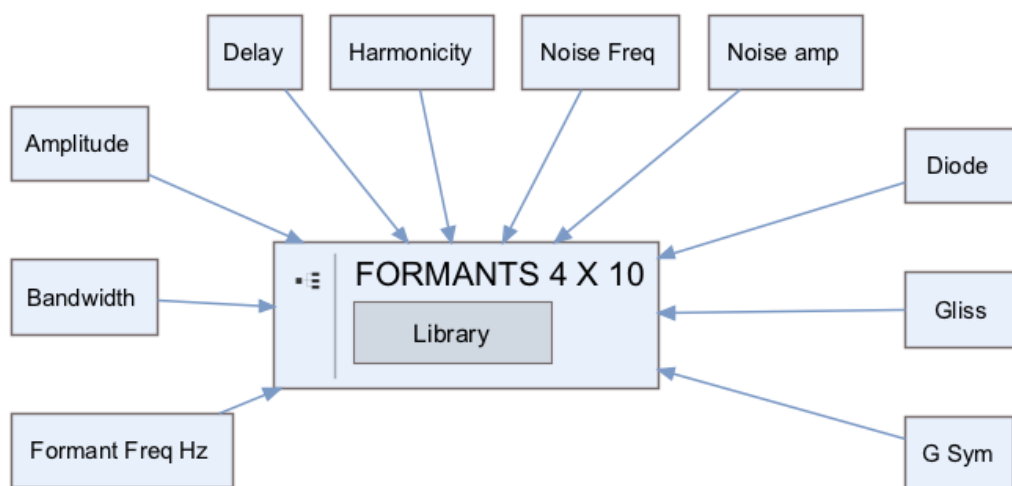


Figure 42. PWS formant component structure.

11.4.3 Noise classes

An audio rate random number generator is implemented with control over the frequency of random changes and the amount of amplitude in modulating the formant output. This noise generator allows two important microsound features: spectral noise classes at audio frequencies and stochastic particle modulation (organisation) at sub-audio frequencies.

Formants	F Hz	Band	B Hz	F Out	Lin	dB	Delay	Harm	Noise F	Noise A	Diode	Gliss	G Sym
1	2001.0		200	0.550	0.008	-41.5	0.000	1.00	144.0	0.99	0.00	1467.0	0.98
2	1344.		80	0.330	0.44	-7.22	0.000	1.00	144.0	0.98	1.00	-808.0	0.98
3	1289.		100	1.000	0.47	-6.57	0.000	1.00	144.0	0.85	0.40	390.00	0.98
4	1261.		120	1.000	0.08	-22.3	0.000	0.00	144.00	0.90	0.00	-858.0	0.98

Figure 43. PWS formant component structure.

11.4.4 Doubling the number of formant layers

The PWS formant component adds a feature which literally doubles the number of ESE formant layers to 8. This feature is borrowed from a simple analogue electronics technique (Hordjick, 2010) and its operational characteristics are detailed in Q.8.2. In essence, it is a digital version of a full wave rectifier which uses diodes and which "folds" the original signal in order to generate a copy of the signal one octave above the original. The amplitude of this parallel signal can be controlled independently from the formant output. This means that any formant layer can simultaneously generate a second formant an octave above. This is a feature not present on any currently available PS or microsound system to date. Consequently, this design feature diminishes the reliance on five formants in order to create realistic vocal textures because often vowel formants contain string octave elements. Importantly, it addresses one of the

most important outcome inferences of the FD analyses; economy of CPU synthesis cycles.

11.4.5 Glissons

By modulating the formant frequency at slow rates one can integrate the basic functionality of glissons or formant sweeps. The current implementation is limited to unidirectional sweeps.

11.4.6 Particle Shapes

The PWS incorporates the principle particle shapes found in FOF, VoSim and PAF whilst limiting the number of shapes observed in Pulsaret, Nuklear and PulsarBTD. The PWS engine offers ten shapes including a user-defined shape. The native PWS shapes include variable ramp, sine, square, 2 x wavelets, FOF, Gaussian, Cauchy, Hann, user determined shape. These were chosen for the distinctive differentiated timbre they produce in PS, rather than for the sake of adding complexity. It is surprising that the non-linear classes of shapes, such as sine, FOF, Gaussian, Cauchy and Hann, offer such distinct timbres.

The FOF particle shape is worthy of note because it uses an unconventional wavetable approach for creating variable FOF bandwidths (see Q.5.3). These were created using Csound. A single FOF cycle was rendered using a zero Hz formant frequency. This was done for all bandwidths (durations) from 20 to 320 in 10 Hz steps. Each one was then sequentially concatenated to form a two dimensional wavetable which the user scans with a single parameter. At the time of writing, this technique is singular to the PWS and ESE.

11.4.7 Teethlets

A distinctive feature of the PWS system was designed in order to extend the inharmonic sound class capability. The technique fuses the concept of trainlets

and the *pulsecount* parameter in VoSim. Both techniques can produce rich spectra, however, the conceptual model in the PWS is different and is inspired from the Japanese art of origami. See Q.6.1 for a detailed description of the technique.

By applying linear wave-distortion techniques to the PWS phase driver, we can generate multiple copies of the original shape per each period of the phasor (Fig.44). The result is then used to read the individual particle shape functions thus allowing different teethlets count to be simultaneously mixed. The term teethlets has been adopted because of the visual likeness of the teeth of a linear gear bar (rack).

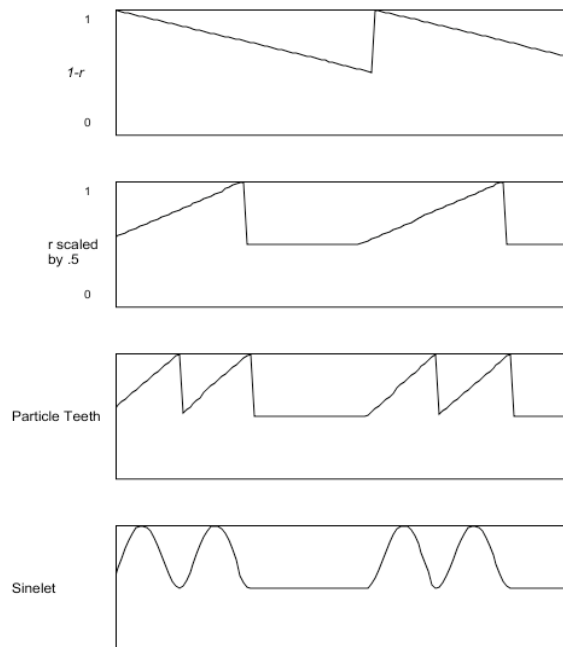


Figure 44. Particle Teeth. Two periods per original period.

Fig.44 shows two teethlets per phase cycle. The PWS affords up to 100 teeth per original phase cycle. The resulting timbre differs depending on the emission rate; formant frequency and particle duration can vary from ring modulator type sounds at audio rates, to complex tuplets patterns at sub-audio frequencies. It also exhibits the high-pass filter sound classes observed in Partikkel but with operational predictability.

11.4.8 Overlaps

The approach to creating overlapped particles in the PWS is distinct from granular techniques because the PWS relies on wave-orientated synthesis techniques. The overlap technique developed for the ESE uses delay techniques, the detail of which can be explored in Q.9.1. This will be incorporated at some point in to future updates of the ESE.

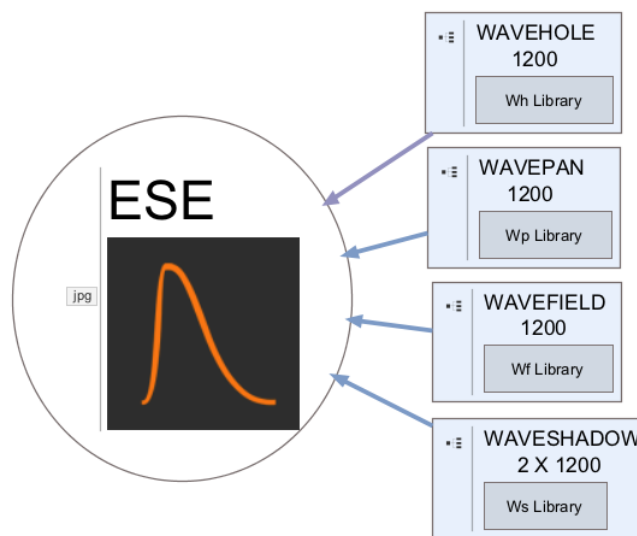


Figure 45. The main ESE Wavehole interface

11.5 Waveholes

The Wavehole concept affords the PWS synthesis system with a multitemporal particle organisation mechanism based on the wave-orientated mechanisms (phase synchronisation) described in the PWS section. In essence, it is a pulse-masking engine in which every mask can have a variable state. In practice, it is based on classic poke and peek techniques for inputting and reading data to a sample buffer (currently with 1200 indexes). The PWS master phase counter is used to read the buffer indexes which in turn modulate individual amplitude, panning, processing and formant frequency of each particle layer. At an emission rate of 1Hz, it takes 1200 seconds to read all values of the buffer

before it loops. This size gives ample scope for meso time scale composition though it can be imagined that in future updates this size will be user controlled in order to meet their specific needs. Currently, it is a first approximation. Like audio buffers in general, a Wavehole can be retriggered from any index point, looped, read forwards, backwards. The user can create Wavehole patterns from analysis data, procedural techniques or manually by drawing values directly into a Wavehole interface. Furthermore, each particle layer has an independent wavehole system, thus different emission rates will read the wavehole buffer at different rates. Each Wavehole state can be stored and retrieved for later use. States can be morphed continuously thus permitting endless particle collections and textures.

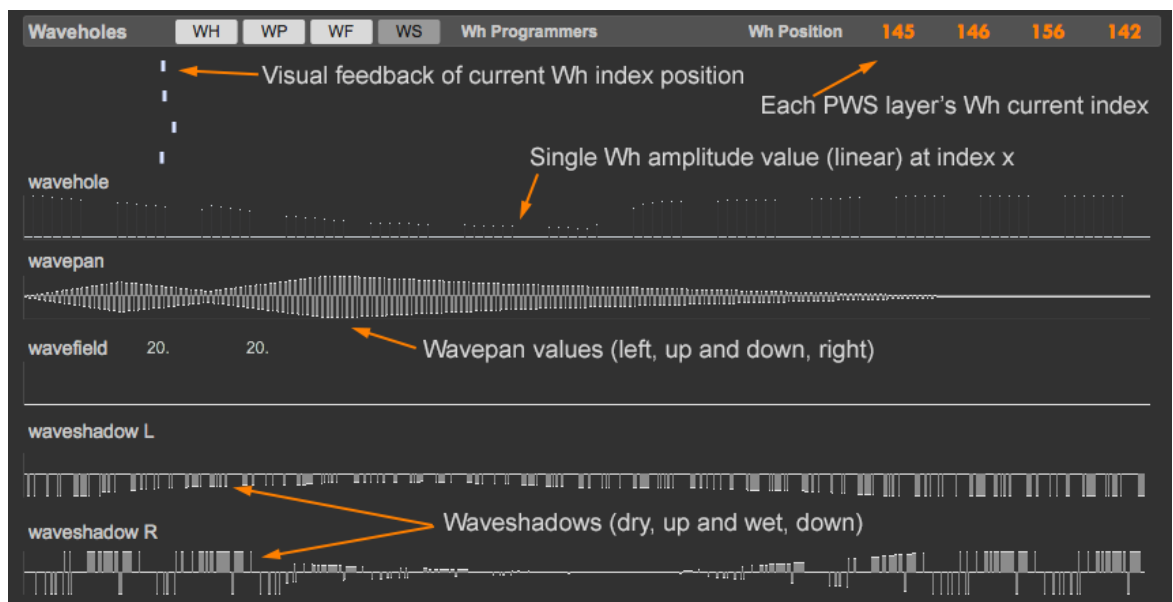


Figure 46. The main ESE Wavehole interface

Each Wavehole has a specific programming interface (Fig.47) called the *Procedural Programmer*. At present it is limited to a simple repeating patterning procedure in which 120 on/off values (the number is explained below in the next paragraph)) can be set in order to create a longer Wavehole pattern. Other techniques have been implemented using chaos theory, stochastic techniques and genetic algorithms. These however, have not been finalised and it is expected they will appear in ESE updates in the near future.

11.5.1 Bypassing the limitations of FOF octavation.

Introducing multiviation

In order to extend FOF microtemporal fission and fusion techniques beyond the limitations of octavation (see 4.1.4.4), which is limited to fading in and out consecutive particles, a more capable system is needed in order to consider other patterns that can afford multiple intervals such as fifths, fourths etc. A convenient term *multiviation* has been used here in order to represent all interval states.

One of the first considerations in the design of a multiviation programmer was whether it would be comfortable to program manually. Programming several tens of switches by hand, might be comfortable but programming hundreds could be time consuming and a burden to the user.

Furthermore, the number chosen should have the greatest number of divisors in order to afford the largest number of multiviation states for that number. After trial and error, the number 120 was decided upon as it is marginally comfortable and has 16 divisors: 1, 2, 3, 4, 5, 6, 8, 10, 12, 15, 20, 24, 30, 40, 60, 120.

Together with its inversions this gives 32 programmable patterns.

2s 1010101010101... Inversion 01010101010101...

3s 100100100100... Inversion 011011011011011...

...

12s 100000000000100000000000... Inversion 01111111111101111111111...

The main patterns and their inversions sound quite different. The former slowly disintegrate the sense of fundamental frequency into a particulate texture (Villez, 2014a).

The inversion on the other hand retains the fundamental frequency and creates subharmonics under it (Villez, 2014b).

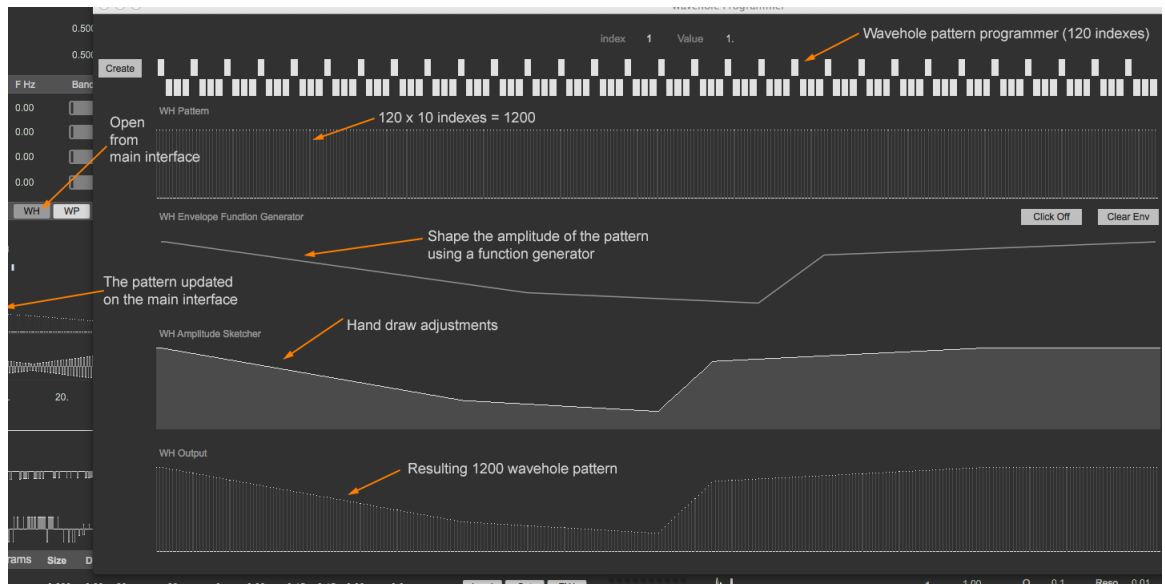


Figure 47. The Wavehole procedural programmer

The multiviation pitch interval sequence is presented in Table 22.

Preset	Divisor	Interval	Note
1	1	Root	C 0
2	2	-12	C -1
3	3	-19	F -2
4	4	-24	C -2
5	5	-28	G+ -2
6	6	-31	F -2
7	8	-36	C-3
8	10	-40	G+ -3
9	12	-43	F -3
10	15	-47	C+ -3
11	20	-52	G+ -4
12	24	-55	F -4
13	30	-59	C+ -4
14	40	-64	G+ -5
15	60	-71	C+ -5
16	120	-83	C+ -6

Table 22. Multiviation pitch intervals in the ESE.

By moving between these intervals the user can create interesting harmonic shifts. By moving in between multiviation states and finding middle point values, pitch triads can be created from one single formant layer. The particle shape determines the final effect and differences can be quite dramatic. The effect is not one in which there is a perception of glides between one harmonic state and

another. Instead, the intervals fade in and out and depending on the emission rate, can be heard as harmonies or definite particle textures with varying roughness. Importantly, this extends to panning, formant frequency and processing. This is a new area of study and one which deserves further research, especially extending the size of patterns beyond 120 in order to create any type of multivibration interval.

Finally, the noise generator in the formant component also adds to this a way of creating stochastic amplitude interferences into the stream for each layer. This combination affords both a formalised and interventionist approach to composing with particles of sound.

11.6 Libraries and metaparameters

The principle formant parameters in the four formant layers in the ESE can be fed numerical values acquired from previous spectral analysis. The parameters of formant frequency (Hz), amplitude (linear and dB) and bandwidth (Hz) use notations which can be directly gleaned from analysis values. Because of this, the ESE can create stylistic vocal imitations. The vocal formant data is stored in its own library (Fig.48).

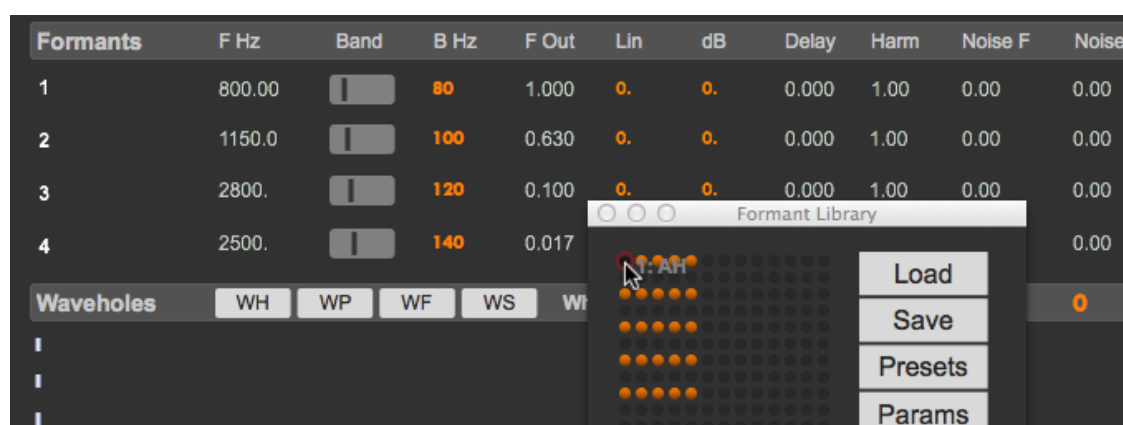


Figure 48. Formant library window showing the tenor belcanto vowel "ah".

Currently, two vocal styles come with the ESE; Belcanto vowels and Xhoomei overtone vowels. The former is data taken from analyses published in The

Csound Book (Boulanger, 2000) and the latter is from analyses undertaken in this research, from recordings of overtone throat singing. All the components use values which permit external analysis data to be directly translated.

Libraries are not limited to the formant component alone. The emission, particle shape and all the Wavehole components, have individual libraries allowing for the mixing of data from different unrelated data analysis. The resulting mix is stored as an ESE preset which is a record of the current state of the ESE. The benefit of this is that a user can mix the amplitude jitter data taken from an accelerometer of a train ride, for example and feed the values to indexes of the wavehole. Mixing these library presets together with other ESE component analysis such as vowel data opens up many interesting sound exploration avenues.

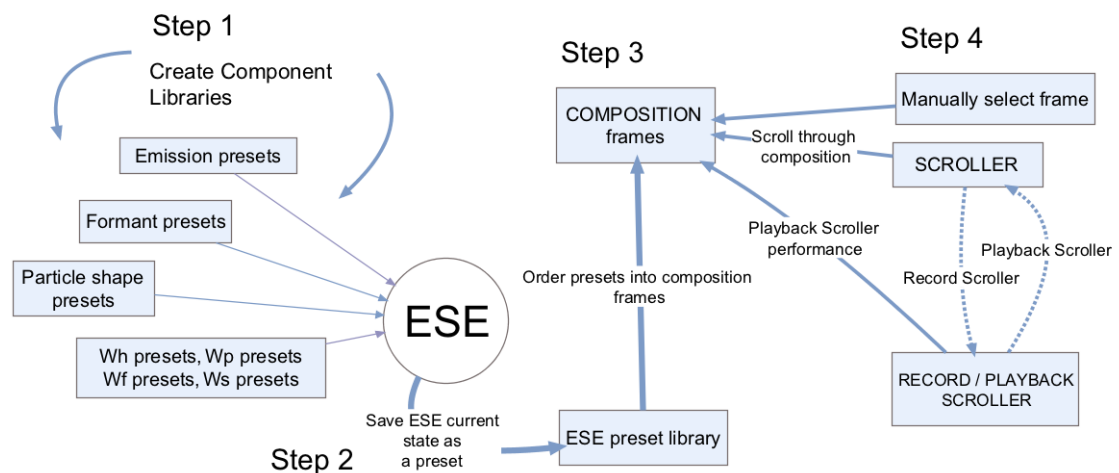


Figure 49. ESE composition sequence of components libraries.

Fig.49 illustrates the process. Step 1 involves the synthesis and storage of component presets. Step 2 stores all currently selected active presets as a global ESE state. Step 3 involves composing presets into a linear time sequence of frames. The user has to predetermine which order the ESE frames will be composed in. Step 4 is the performance of the composition which involves the interpolation between each successive state. This involves one main parameter called the "Scroller" (Fig.50). The Scroller literally scrolls

through the states morphing from one to the next. The states could be thought of as the individual frames in a movie. The performance of this scroll can be recorded via its own sequencer or using a function envelope generator.

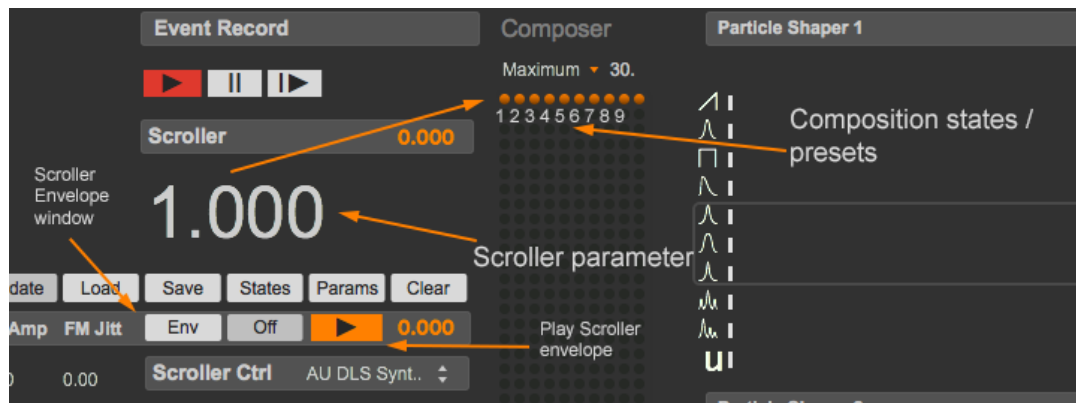


Figure 50. Scroller parameter and composition states.

The result of this process is complex phenomenologies in which microsounds, sound objects and larger musical time structures coexist. Fig.51 shows how the ESE synthesis / composition process falls into different time scales. Because all these processes can be performed in real-time, it is possible to adjust parametric processes which affect a particular time-scale in real-time.

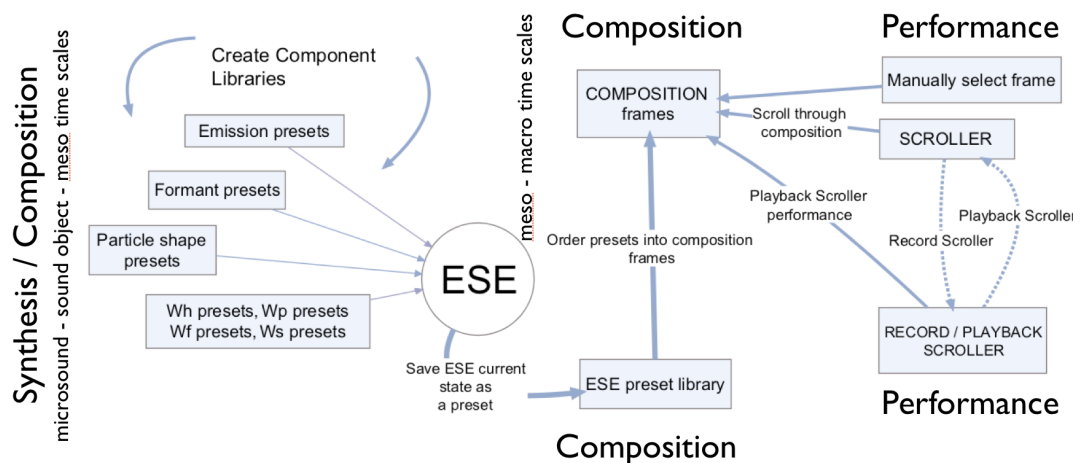


Figure 51. Composition / synthesis and multitemporal relationships.

11.6.1 The difference between the ESE and granular

technique including other PS systems

Besides affording the missing features from current PS systems discussed previously, the ESE differs from other traditional granular methods in that it produces strong formant content with a defined periodicity. Furthermore, it integrates sophisticated multitemporal composition functions which permit both a formalistic and interventionist approach to composition. Despite its scope of controlling thousands of parameters, it also affords the user with a relatively simple and effective operational interface. The ESE can produce a wide variety of sound classes which simultaneously encompass many granular synthesis phenomenologies and pitch-synchronous particle synthesisers.

To reiterate the unique features discussed above, they are presented in Table 23:

Dynamic wavelet particle shapes
Accurate FOF particle shapes and bandwidths
Reduction of formant layers needed for vowel synthesis
Metaparameter control over composition
Multiviation
Formant octave doubling
Inharmonic spectrums
Noise classes
External data library driven components

Table 23. ESE differences from other PS systems

Despite the effective functionality described above, it remains to evaluate how the design goals set in Table 21 have been met and whether the framework is successful. The next chapter summarises the outcomes from these subsequent analyses with the full version being available in appendix S.

Chapter 12 Summary of the evaluation of the design analysis and development framework

Associated document

Appendix S

Pen drive Folder 05 ESE Framework Eval Workbooks

ESE Analyses OSTM.pdf

12.1 Introduction

The design and implementation goals which were identified and derived from the design analyses framework and listed in Table 21 are used in this summary of ESE analyses outcomes as a point of comparison in order to evaluate whether those goals were met and whether any design and implementation issues (consequential and unknown) have arisen which could be used for further correctional or innovative implementation. A summary of the MS analyses results is presented below.

12.2 Evaluating the effectiveness of the design strategy

Each of the pursued design goals presented in Table 21 are examined individually in relation to the ESE analyses outcomes to evaluate whether those

goals have been met or not and to estimate the success of the analysis framework. The same method of paragraph referencing used in appendix P is employed for reference purposes.

12.2.1 Table 21 design goal 1. Minimise major misbehaviour dysfunctions. Dependency management, Catastrophic side-effects, behavioural noise

12.2.1.1 M1.A4 Dependencies. CD

The ESE was found to have the greatest number of dependencies of any of the PS systems studied. The dependencies observed correlated with the core glue parameters revealed in the analyses of the other systems. It is noted, however, that in the case of the ESE they have little functional impact because of the nature of the ESE meta-parametric system design. The Scroller / composition-states memory combination, mitigate dependency effects as a result of perceptual feedback; e.g. the user adjusts settings and stores the desired results (from S.2.4.1).

Apart from this, the ESE system showed few functional side-effects such as noise or interruptions. Simple audio rate Boolean conditionals are used extensively throughout the DSP chain of the ESE in order to detect NaNs which had been found to generate consistent catastrophic failures. Unfortunately, this added some CPU cycles to the system.

The NaN filtering system can be considered as a step in assuring that erratic and functional behaviour of the system is kept to a minimum (from S.4.1).

Despite the CPU and memory results, the ESEs performance results show the system to operate as intended with very few problems of latency or CPU clicks (from S.4.1, S.4.2, S.4.4 and S.4.5).

12.2.2 Table 21 design goal 2. Feature disparity and integration of missing parameters

12.2.2.1 Noise classes

The available formant noise engine introduces an expanded set of sound classes not possible otherwise. This, however, needs expanding to more sophisticated noise generation including different spectral filtering techniques in order to "sculpt" noise characteristics such as pink noise, etc. (from S.6.2).

12.2.2.2 Formant harmonicity / inharmonicity

This has been achieved successfully. However, the effect relies on the de-synchronisation of the "hard sync" engine in the main PWS synthesis section. This limits the inharmonicity to classic ring modulation sound classes. A more extensive model could use FM instead (or as well as) in which the four layers are setup as two parallel FM pairs. This has been experimented with some success but requires an extensive re-structuring of the ESE engine (from S.6.1).

12.2.2.3 Overlaps

The overlaps issue is certainly a problem in order to afford users with cloud and swarm sound classes without resorting to convolution. This could be included in the next design strategy. It could be argued that like VoSim and PAF, the lack of overlaps makes these systems more suited to pitched sound classes. It is a technical challenge however as to how overlap technology might enhance the ESE and afford it with cloud, swarm and smeared sound textures.

The overlap issue also has repercussions on the effectiveness of the teethlet parameter because of the duration to emission rate independence. This has to be considered carefully because it could too easily make the ESE synthesis a granular synthesis engine. It is possible that in a future version both synthesis techniques could be employed side by side, thus expanding the sound design scope of the ESE (from S.5.2).

12.2.2.4 Real-time analysis / re-synthesis

A future version should incorporate real-time analysis capabilities into the ESE which are able to cope with the CPU and memory requirements and deliver both effective emission rate and formant parameters from external audio sources such as live or recorded singing (inferred from S.7.2.5).

12.2.2.5 Eliminate behavioural noise

A future update would need to address the detected zipper noise in the formant bandwidth parameter (S.4.5).

12.2.3 Table 21 design goal 3. Comprehensive multitemporal composition features

In N.2.4, it was discussed how Partikkel is the only system which goes some way to offer manual organisation of particles. The tendency amongst the other systems is to afford short repetitive patterns of particles.

The combination of the Wavehole filtering system together with the Scroller metaparameters and Wavehole procedural programmers afford the ESE with extensive multitemporal compositional features (from S.5).

Extending scope and apparatus for creating Multiviation states. Affording the system with the capability of diving the wavehole procedural programmer with other division than just the 120 in order to expand the scope of multiviation intervals (inferred from S.6.3 and S.6.5).

12.2.3.1 Scroller and Wavehole design issues to consider

The management of presets and composition states are limited in the current version:

- There is no provision for inserting states in between adjacent states.
- There is no provision for selecting a group of states in one area and moving or copying them together to another location.

- A group of states from one composition cannot be loaded and inserted into another. The result is that doing non-linear editing a composition is hard work and can be easily interpreted as requiring HMO from the user.

The Scroller sequencer can be clunky and slow to operate. It needs careful revision including the points about the organisation of compositional states. This scenario also applies to the Scroller function editor page.

Updates should consider extending the procedural methods of generating Wavehole patterns using other mathematical and generative process such as AL and AI techniques (general inference).

12.2.4 Table 21 design goal 4. Metaparameter control in order to A. control dependency and for B. modelling physical properties.

The Scroller metaparameter offers a single parameter with which to create rich sound particle phenomenology. It has been shown in the analyses that it is an effective control mechanism for controlling dependencies and functional misbehaviours, through the use of interactive PC (from S.2.4).

An expansion of this technique might be a Scroller like mechanism which is local to the Wavehole procedural programmer pages and which affords a metaparameter for experimenting in local morphs rather than global morphs. This might be useful in testing quick local settings and for creating the morph of physical modelling process like formants to material resonances (general inference).

12.2.5 Table 21 design goal 5. a, Expertise levels, b. Premature commitment, c. reduction in HMO and d. libraries

12.2.5.1 Expertise levels

It may be possible in future versions to consider other notations besides units such as Hz via mapping. This was a design goal resulting from the PS analysis framework. It was determined, however, that the benefits afforded by the capability of using raw spectral and data analysis, benefitted the general implementation at the expense of requiring users to learn how to use linear amplitude values rather than the 7 bit range of 1–127 commonly found on many commercial synthesisers.

12.2.5.2 Premature commitment, HMO

Though PC has been largely eliminated in the ESE, the preset and composition state library together with the Scroller metaparameter and the Wavehole procedural programmers, require a great deal of PC because the knowledge in these libraries has to be constructed by the user through analytical or manual means. This process, however, can be controlled in real-time starting from the most elementary settings. The user never has to interrupt the audio engine. The same can be said for HMO. An example can be found in the belcanto formant library included in the standard ESE download. A set of libraries was necessary in order to give new users some initial material so that they could operate and become familiar with the ESE. In future versions, however, a real-time analysis engine would afford users with the means to analyse and store formant or other library component collections in real-time. Even with the best current analysis technologies, however, this requires some perceptual adjustment and consequently may require some HMO (S.2.6, S.2.7, S.2.8 and general inference).

12.2.5.3 Libraries

The inclusion of functionally independent formant, wavehole, wavepan, waveshadow and emission libraries endows extensible sound design scope to the ESE system. The independence of these libraries, however, does demand that the user organise these independently. It is a large quantity of information to deal with. This is an important outcome and an area for further design development in which a dedicated library manager is included (from S.2.6, S.3, S.5 and S.6.3). The ESE analyses OSTM presented in Fig.52 illustrates the outcome summary. A larger version is available on the accompanying pen drive and is titled *ESE Analyses OSTM.pdf*

In the same way a design strategy for implementing the ESE were derived from the design analysis framework and detailed in Table 21, a list of examples evaluation issues, which arise from the same design analyses framework applied to the ESE, are listed in Table 24 as a continuation of the developmental process. This process can serve as a design continuum for either honing the same implementation or creating new ones.

Implementation of real-time analysis / resynthesis
Overlap implementation
Zipper noise in formant bandwidth
Extend multiviation and consequently the size of Wavehole buffers and divisors
Extend wavehole procedural programmers
Better component library management and organisation
Scroller sequencer and function generator extension
Extend noise classes

Table 24. Possible new ESE design and development strategy.

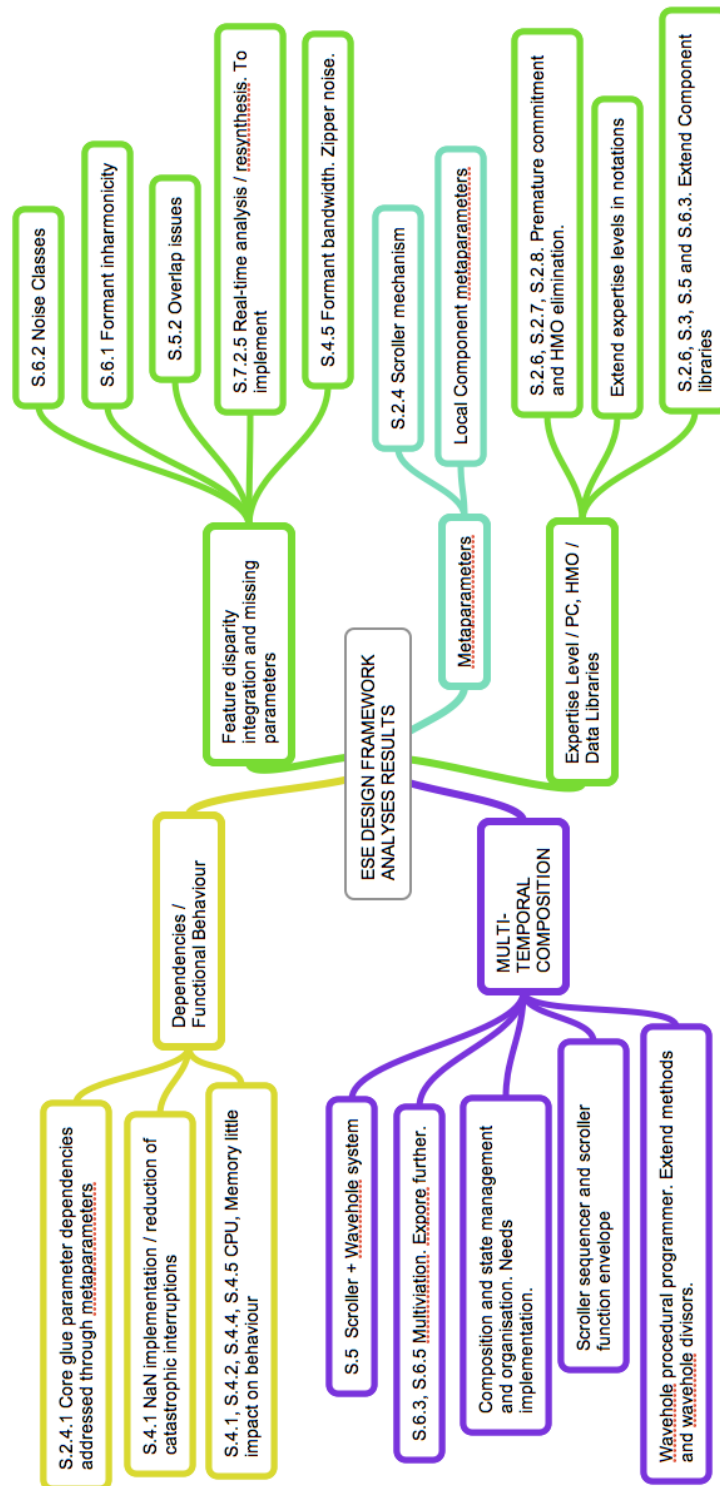


Figure 52. OSTM of ESE design analyses.

Chapter 13 Summary and conclusions

Associated documents

Appendix P

Chapter 5

13.1 Answering research question 1

“What are the functional and usability criteria which could establish a design and implementation analysis framework for the effective, unified and robust development of particle formant synthesis / composition environments?”

The implementation of an extensive PS artefact as presented in chapter 11 together with its evaluation using the same design analysis framework, illustrate the effectiveness of the approach of structured design and implementation in independent development.

Through the systematic and critical analysis of Jaffe's ten criteria in chapter 6, the research was able to narrow the four criteria classifications to just two; usability and functionality (6.3.3). Through a further process of informed critique, it was possible to systematically translate Jaffe's criteria framework from a purely informal and discursive document into an empirical and measurable framework for design analysis. This was effectuated by identifying key attributes in Jaffe's text which could be operationalised as variables. In a parallel process,

the attributes were translated (6.2.1) from purely lexical mechanisms into discursive tools by means of Green's Cognitive Dimensions (Green, 1989). This approach simultaneously afforded an intermediate stage in the translation from the qualitative nature of Jaffe's critique into measurable attributes. Each attribute becomes a numerical analysis. Consequently, Jaffe's ten discursive criteria were reduced to seven (6.13). The criteria along with their attributes were identified and used to inform the design analyses from which numerical data was recorded. The analyses, in particular those in which both CD and FD behaviours were measured (M1.A4 and M4), give no doubt in respect as to the parameter interactions observed (appendices E, L).

In answering research question 1, not only does the research clearly and expansively identify the pertinent usability and functional criteria but goes further in that it identifies the associated qualitative attributes and operationalises these into measurable variables and consequently the methods for investigating design features. The framework becomes a systematic design analysis - measuring tool.

13.1.1 Efficiency of the research process

The framework's efficacy is further enhanced by the way in which the collected data is hierarchically and contextually collated in appendix P and summarised graphically in chapter 10. Hierarchical because the framework is designed so that the CD and FD analyses outcomes are numerically paragraphed and thus are a set of ungrouped proto-lists driven by the order and classification of the criteria.

Each appendix (B to O) report lists a set of paragraphed outcomes or design inferences which are gathered from the analyses results and presented graphically in the *Appx P Outcomes Summary Tree Map.pdf* on the accompanying pen drive. On completion of all of the criteria analyses, these elements constitute not only an overall design profile which identifies and highlights the potentials and limitations as discussed above but also the basis for a design specification or requirement document which can be used to inform

new design approaches. This is key because it allows the developer to highlight specific outcomes and inference paragraphs depending on their design requirements. In the case of this investigation and in order to test the analyses framework process, it was decided to concentrate on issues first detected in the chapter 5 software review and later confirmed by the analyses (see Table 21). This is what is meant above by contextually collated.

An important observation is that during the process of evaluating the effectiveness of MS design analysis framework in the previous chapter, it was revealed that in practical terms, the framework is actually not that complex or burdensome to navigate when examining a single specific artefact (ESE). In this case, the analyses were accomplished over a period of two days. This portability and efficiency is vital in rationalising artefact design features.

13.2 Answering question 2

“Is the consequent microsound specific criteria based design analysis framework successful in informing new particle synthesis artefacts?”

The limitations identified in chapter 5 resulted from superficial experiments with the artefacts under review. They are summarised below in Table 25 .

Paragraph	Limitation
5.1	Particle organisation features
5.3	Idiosyncratic limitations
5.4	Feature parity
5.6	Too many features
5.7	Standard Glue parameters and variations
5.8	Missing core features.
5.9	Lack of standardisation.
5.10	Developers

Table 25. Chapter 5 summary of superficial limitations in particle formant synthesis.

In order to illustrate how the framework has been effective in informing design, a few rationales are provided below. These expand on the observations first noted in Table 25.

13.2.1 Identifying and balancing the lack of standardisation

The results from the analyses of the seven artefacts effectively show that this class of synthesiser is far from a unified model. The artefacts share many of the topical features of microsound generators but also have features which are quite individual. There is no standard of design or notations between them. Parameters which are common to more than one artefact are presented in different ways or with disparate notations. As an example, there is a lack of amplitude normalisation features amongst all the artefacts studied which permit the control of particle amplitude, a core parameter.

13.2.2 Revealing parametric parallels

FOF, PAF and VoSim, specifically model acoustic instruments and consequently, share many parametric similarities, for example, formant frequency, formant amplitude and formant bandwidth. All exhibit coherent physical parameter notations (Jaffe, 1995, p79). Consequently, they can operate directly with analysis data for the purpose of modelling known acoustic phenomena such as vowels. This consideration was a central one when implementing the notational system in the *ESE*. The reason for this was not just the modelling aspect but the imagined promise of employing real-world analysis data to inform other key aspects of the synthesis and compositional stages such as particle envelopes, the shape and intricacies of meso and macro time events, the shape and distribution of the Wavehole system. It is concluded that this approach can fuse the synthesis of timbres with the composition of musical events. This inference emerged from studying and reflecting on the data reported in P.14.4; Synthesis is composition.

13.2.3 Unifying modes of synthesis

Thanks to efforts in the computer music community, it is possible to use FOF, PAF and VoSim simultaneously under one environment such as Csound or Max/MSP; however, the cross parameter syntax and differences in variable implementation indicate some difficulties with regards their simultaneous operation. In terms of this research, a unified system is favoured which affords, to some degree or another, the advantages of some of the similarities of implementation between the different techniques such as formant parameter controls.

13.2.4 Missing core features

FOF has octavation, Nuklear has limited pulse masking and PulsarBTD uses stochastic masks. None, however, have a long macro compositional strategy which combines the automatic and the cybernetic approaches discussed previously. Furthermore, the analyses clearly reveal the lack of processes which are used to control other aspects of the local occurrence of individual or group of particles; e.g. the control of spatial location, filtering, reverberation and formant frequency (see P.13.6 and P.13.7). This is compounded by the findings in P.1.2 that binary states exclude the seamless movement between time scales.

All this information collated in appendix P, becomes a central cross-reference of potentials and limitations. It is a document in which items can be dismissed or used depending on the context of the information needed.

It was revealed in P.13.1 that this class of sound generator generally lacks noise generating mechanisms. This became a guided design directive in order to find ways of generating broadband noise types that would be perceived beyond the particle level. The same situation was noted with the lack of overlaps in VoSim and PAF.

Other data results which emerged from the behavioural analyses, suggested a need to redefine the purpose of overlaps. It suggested that many of the sound

classes afforded by this feature might also be created in wave-orientated synthesis types. The overlap issue emerged because of the issues of memory pre-allocation, together with the problems of permanent interruptions observed in FOF and glitches in Pulsaret's automatic overlap allocations (M4.A4). Thus, a real-time variable overlap approach was sought by simply turning the overlap conception into a variant of the technology; digital echo. Overlaps are in essence an overlapped copy of the previous event. Modifying the technique and using dynamic parameter modulation, allows wave-orientated engines to create smeared and dense choral-like sound classes often associated with granular synthesis but with the advantage of retaining pitch-synchronicity.

13.2.5 Too many features

In both the CD and FD behavioural analyses, it became clear that Partikkel exhibited multiple issues of functionality and usability. Too many tables, too many generator types, too many dependencies and too many mental models. These create a diffused identity which requires special attention from the user. In terms of design, the issues identified in Partikkel suggest the need for an alternative modality of existing particle formant generation practices which acknowledges the strengths of existing methods, whilst maintaining flexible functionality while still offering strong phenomenological identity. A reduction of features in order to strengthen usability, functionality and consequently, sound class identity.

13.3 Beyond microsound

Functional priority has been a primary consideration in order to improve the original PWS design. Originally, the implementation emerged from the consideration that VoSim and FOF possessed an original and strong sound class identities. The PWS was initially conceived in order to imitate certain sound class aspects of both techniques (mainly vowels) at a time when real-

time systems for this type of sound generator did not exist in the general computing market. The application of the analyses framework to the PWS original design has dramatically altered and expanded the sound class potential.

The ESE, as it exists now, was implemented alongside the development of the methods research stage presented in this thesis. This was a fruitful approach because as data emerged and was recorded, it would remain immediate and current in order to suggest implementation approaches and the possible directions to avoid a limitation or strengthen a potential; e.g. the case of echo-overlaps and teethlets respectively. It is hypothesised that this immediacy is crucial to the success of this type of development. It returns us to Green's opportunistic planning. The resulting outcomes demonstrate that the order in which the analyses were effectuated was successful and that the data and conclusions being arrived at were being assimilated effectively into the resulting design work.

The most pragmatic conclusion arrived at, because of this methodology, is that the ESE is not a purely microsonic generator. It is a subtractive synthesiser and compositional environment with a capable and strong particle synthesis behaviour. It could be termed a super-class of subtractive synthesis because it goes beyond the simple source-filter model. Instead, individual phase-synchronised wave cycles are subtracted in real-time from the stream and re-directed to alternative processing channels that act in a number of domains outside the original one in order to particulate the sound output. The by-product is that it behaves strongly and effectively as a microsound particle formant generator. Its advantage is in that its temporal organisational features are more comprehensive because its design is directly informed by the deficiencies revealed in the analyses of the other PS engines.

The implications of this result had not been considered fully until the base difference between overlapped generators and non-overlapped generators had been understood as a result of the analyses data and translated into design actions. Fundamentally, the research demonstrates that in practical terms wave-orientated synthesis generators *can* synthesise convincing microsound

textures and operate in multiple-time levels as can be heard on the ESE Soundcloud examples (Villez, 2014e). If this is the case then the term microsound needs to be expanded to include this type of approach.

The design analysis framework has strongly informed the research of the nature of this class of generators and sufficiently so to also inform the design of an alternative and improved system for generating and organising particles of sound.

One of the most important objects of information derived from the framework is the idea of differentiation and uniqueness. By revealing this superclass of *subtractive particle synthesis* in chapter 11 and appendices Q and R, the research qualifies that wave-orientated synthesis systems can operate comprehensively and efficiently as time domain synthesisers.

13.4 Final thought

The irony of the resulting explicit document is that when the resulting design is created and realised into a new technique, the explicit becomes an implicit specification. The framework encompasses not only the seven criteria and associated measurable attributes but also the data collection process and the way the data is reported and collated into the final design requirement.

Finally, evaluating the analysis framework itself through the analysis of the resulting ESE implementation, demonstrated the scope and effectiveness of structured design in independent development. The ESE could not have been implemented in its current format without the detailed observations harvested from the empirical analyses of the seven original PS artefacts.

Chapter 14 Further research

14.1 Categories for further research

Four broad categories are recognised for expanding the primary research in these pages:

(1) Expansion of the main criteria driven design framework, (2) effective methodologies for identifying new classes of synthesis artefact criterion and attributes, (3) further implementation of the ESE artefact and (4) the development of a compositional language in order to profit from the multiviation techniques and new phenomenologies created using the ESE.

14.2 Framework and methodologies

14.2.1 Expansion of the main criteria driven design framework

It has been established in this research that creating a design framework using strong criterion driven analysis, is an effective way to discuss and understand key usability and functional elements of music artefacts in order to improve and proliferate existing designs. The implementation analysis of synthesiser design profits the creativity of the musical communities which use them. Furthermore, it also enables variant artefacts to be created for use in other fields such as data sonification, haptics, medical research etc.

Because the development of these types of artefacts is becoming more popular, it is concluded that streamlining the framework, in order to create a more portable and compact version of the analysis framework, would benefit the field considerably. This is discussed in 6.1.2.2, in which the scale of a full functional analysis of synthesis artefacts is recognised as an extensive undertaking. Subsequently, it was demonstrated that applying the analyses to one specific artefact (ESE) streamlined this process considerable. Consequently, it is concluded that reducing the size of this process would be a further positive step.

A simultaneous approach is to also identify a general criterion generator in order to customise the prime design framework for use in other audio and creative artefacts. Admittedly, the success of applying the framework to the ESE's design is a result of there existing a personal degree of closeness between the development of the criteria analyses methodology and the artefact structure. Communicating this to other researchers and designers using academic and practitioners publications is a necessary next step. With this in mind, a blog has been published (Villez, 2014c) in which the artefact and work is discussed and its knowledge base disseminated. The intention is to follow this with other publications.

A limitation identified in criterion M1_A4 is that only a handful of implementations of particle formant synthesis exist. It would be interesting to expand this study to other synthesis techniques in order to see if there are any attribute data correlations.

As illustrated in the last paragraph of 6.1.2.2, the work in these pages is a first approximation and therefore needs further research in order to make its scale more practical. This has been identified already in evaluating the design analysis framework using the data harvested from the first seven PS analyses.

14.2.1.1 User profiles

In this research, the usability profiles in chapter 7 (notations specificity) were kept deliberately reduced in order to develop the methodological process. It represents the two general sets of users likely to use the evaluated artefacts;

(1) musicians, composers, sound designers and (2) the acoustics engineer, sound researcher or scientist. Most sound design and synthesis artefacts use one or both of the notations in these profiles. It is likely that other profiles exist which would expand attributes such as notation specificity and notation expertise types.

14.2.1.2 Expanding the scope of Premature Commitment (PC) measurements

Identified in I.1.1, the premature commitment analysis measures the percentage amount of PC required by any of the systems before real-time synthesis. It measures this in terms of how many parameters require PC in one specific system. It does not measure whether the parameter requiring PC is core or key to particle synthesis. This would be a further avenue of research and it is possible to conceive it would involve a link with the dependency and HMO analyses because HMO are assumed to be performed in advance of run-time.

14.2.1.3 The behaviour of hardware in the analyses

In the functional dimensions behaviour analysis (L.6.3), it was noted that a further study would be needed in order to identify the specific activity measurements in dedicated physical hardware suited to native Microsoft *Windows*. This is certainly of use in future artefact analyses because of the disparity in CPU measurement observed during the recording process.

14.3 Implementation

14.3.1 Nested similarity in teethlets

The Teethlet concept presented in the *PWS* implementation (11.4.7 and Q.6) is based on *self-similarity*. It is inspired from the folding of paper into shapes as in

the Japanese art of *origami*. The implementation is elementary but with further research it is envisaged that the feature would have significant spectral influence on the resulting product between formant carrier and the particle shape. An area of interest is to find a model which allows more levels of nested similarities. It is envisaged that this process might have some effect on the harmonicity of the particle-wave stream.

14.3.2 The Shannon wavelet

During the implementation of the PWS, it was observed that the *Shannon-wavelet* particle shape (Q.2.1) exhibits some interesting formant generating properties. The wavelet shape was implemented in the late stages of the development presented in appendix Q. This was as a direct result of the lack of this specific shape in other pitch-synchronous generators. It deserves further attention because of the ease with which it can create multi-formant peaks using just one single layer of PWS.

14.3.3 Predictive normalisation

During the process of analyses reported in C.1.2, it was observed that many of the problems with the amplitude dependencies could be resolved by the use of some kind of predictive normalisation factor or adaptive look-ahead limiting. This area needs more study.

14.3.4 Particle shape and wavehole sequence envelope extraction

In O.1.3.2 it was noted that except for envelope extraction techniques, there are no current analysis systems that are capable of directly extracting acoustical data of particle shapes or their attack and decay properties into the synthesis environments. These descriptors can be best acquired using perceptual

estimation and observing the visual representations of the analysis. This helps estimate which particle shape is suitable for a particular context. It is possible that by using a measure of spectral brightness, one could determine the relative particle shape sharpness. This is an on-going part of the development of the ESE, however, it needs integrating into a UI elements which form part of the synthesis and compositional environments. The challenge lies in incorporating the various analysis techniques into one environment. Fundamental pitch estimation, formant frequency derivations, particle amplitude extraction and Wavehole envelope extraction.

14.4 Composition

14.4.1 Using the ESE to compose in the *flutter* region

Discussed previously in 2.4.3 and 2.4.5, there are two exciting compositional areas of exploration for systems such as the ESE. One is the trans-phenomenological capability of the system; morphing from one state to another and seamlessly moving between one sound class and another. The movement between time-scales is an important apparatus for this and as thus, the mechanisms (Wavehole durations, resolution and scope) are being investigated further. The other area for further research is the scope of composition in the *flutter* region. This is the area which lies between the temporal perception boundaries (Miranda, 2001, p7, pp2) which allow us to physically perceive whether a stream is continuous or discrete. Some experiments with the ESE have already resulted in fruitful progress and will be published after the completion of this part of the research.

14.4.2 Synthesis is composition

The facility to be able to procedurally compose long periods of sound particle activity proposes a challenge in that it requires a different compositional approach. Particle envelopes can be acquired for use as particles at the microsound level but they can also be used at the sound object and meso time levels. A particle stream could have a piano strike as a particle shape. The stream could be emitted in order to decay over several seconds with the same piano strike shape. Several of these sound objects could be emitted slower, faster or with less density. A compositional language needs to be developed in order to profit maximally from the ESE's potential. This is a step which has already started and is being documented in parallel to the continued development of the ESE system.

14.5 Publications

On termination of this thesis, writings and media will be collated in order to publish literature and media. This has already been accomplished, to a certain extent, in the ESE blog (Villez, 2014c). The objective is to disseminate the ESE concept and its compositional scope and techniques. Format and mediums of publication are currently being explored with this purpose in mind.

Chapter 15 Detail of original contributions

15.1 Contribution 1 Design analysis framework (principle)

15.1.1 Anatomy of the framework.

The specific anatomy of the MS design analysis framework presented in this thesis is unique in several ways. Jaffe's ten criteria (1995), Green's cognitive dimension of notations together with the heuristic principles of Nielsen (1994b) and Weinschenck (2000) were identified during the early stages of the research as being a flexible and an extensible generalised set of resources from which to discuss and inform specific aspects of the devices presented in this thesis. Adapting this set of generalisations into a unique set of formal empirical tools with which to analyse a very specific domain of artefacts is a unique alternative to the design and development process based on standard formal specifications and requirements found in manufacturing. It is also a formalising process which addresses many of the opportunistic planning pitfalls encountered in independent development; e.g. the lack of standardisation of software components (see 14.4.1).

15.1.2 Agnostic framework template

Now that the processes of creating the framework has been completed and a template formed (see chapter 6), it should not be too cumbersome for other researchers/developers to adapt the template in order to apply the methodological technique to other types of synthesiser. The development of this microsound synthesis template eases the burden of creating modalities of the framework. This is a key element of its portability. Its size means that it is easily managed and relatively economical (compared to commercial industrial process) to run. It is reasoned that this is a unique approach and a contribution to the community of independent developers wishing to create new and differentiating musical tools.

It is recognised that the methods reports presented in appendices B to O would seem to present issues with regards the size and time needed for the design analyses of future artefacts however, as discussed in 13.1.1, the time required for the analysis of one single artefact is greatly reduced even with an artefact with the complexity of the ESE.

15.1.3 The seven criteria for analysing particle formant synthesisers.

The translation of Jaffe's general synthesis criteria into seven specific synthesis criteria together with its own specific set of attributes, has afforded this thesis with the tools to gain a deeper understanding of this unique and relatively new type of synthesis. It contributes to a substantial body of knowledge of the functional anatomy of this type of artefact. It establishes a functionality and usability lexicon in order that practitioners are able to discuss technical problem areas, such as but not exclusive to the behaviour of *glue* parameters, their behaviour and their dependencies etc.

15.2 Contribution 2 (ancillary)

15.2.1 The ESE

The combination of the Particle Wave Stream synthesis engine together with the Wavehole filter system, establishes a new approach to synthesising and composing with particles of sound.

15.2.1.1 The PWS engine

It has been established that a unique attribute of the PWS system is that it is a standard model synthesis engine adapted to behave like a particle formant system (see 13.3). The combination of controlled amplitude modulation using short amplitude envelopes (particle shapes) together with the ability to control the duration of these envelopes independently from the fundamental (wave distortion applied to amplitude), afford the system with a different functionality compared with other PS systems. It is also able to create particle sound classes and affords a wide multitemporal flexibility compared to other established sound particle generators.

Key features of the PWS are:

- Integration of general microsound particle shapes in one environment including, FOF, *Gaussian*, wavelet, shapes from the external sound analyses, *Scanlet* using *band indexing* (for the FOF shape).
- A new behaviour controlled alternative to *Trainlet* synthesis. *Teethlets* are generated using linear distortion technique and based on the idea of self-similarity seen in fractals.
- The addition of specific noise generators.
- The combination of these new techniques into a unified engine is extrapolated as being a contribution to the field of sound synthesis.

15.2.1.2 Waveholes

The Wavehole system contributes a singular technique with which to control and compose particles of sound through multiple time frames. This can be done using a variety of compositional approaches including automatic or generative techniques, manual intervention or through set procedures such as being able to program the Wavehole environment using number systems or patterns together with the addition of external sound and ecological analysis modulators. Not all these have been implemented in the current version of the ESE but given time they will be included.

As discussed previously, the Wavehole core functionality is conceived from a subtractive approach because it can exclude individual cycle events generated by the PWS system. By employing such a strategy it expands the generally established concept of subtractive synthesis and in the process, demonstrates that wave-orientated synthesis techniques can indeed be transposed to time domain synthesis. This contributory notion is further expanded by adapting the technique in order to control other ESE properties including:

Wavepan - spatialisation

Wavefield - event formant frequency

Waveshadow – processing

Each of these features has a separate interface which can be programmed and controlled individually whilst synced to the other Wavehole processes. This comprehensive approach to organising particles in time as well as other domains, offers a new approach to composing and designing sound particles across time levels.

Furthermore, it offers an alternative and new approach to the composition of FOF like sound classes with greater control over the particle organisation mechanism liberating this from the octave switching, as in *octaviation*, to any arbitrary interval groupings. The morph scheduling system controlling all the Wavehole techniques contribute to a new concept in electronic music composition and is inclusive of the formalist and interventionist compositional approaches together with the proceduralist (as in "programming steps" rather

than just the algorithmic / generative) approach. As previously stated, the composition of microsound music using the flutter region is still in its early stages. The Wavehole filter is presented as a contribution to this area because of its multiple levels of control, that is, its facility to easily design particle groupings using any of the approaches summarised in the above paragraphs in order to control, for example, *flutter* streams. The controllable morphing of these offers new compositional avenues to artists especially in conjunction with spectral analysis.

15.2.1.3 ESE software release

The ESE was released as alpha software on February 2014. It was subsequently released as beta on April 2014. Since making it available, it has been officially by downloaded globally 1418 times (Villez, 2014d). It is being actively used by many different types of users including professional sound designers and composers. Since publishing the ESE support blog (Villez, 2014c) it is being followed by over 300 readers. The Soundcloud examples (Villez, 2014e) have had over 6200 plays since published in early 2014.

References

- Anon. (2012). Cognitive Dimensions of Information Artefacts: a tutorial, Version 1.2 October 1998. Available at:
<http://www.ndirect.co.uk/~thomas.green/workStuff/Papers/> Retrieved on Tuesday 28th August 2012 at 16:32.
- Anon. (2012). Live. Available: <http://www.Ableton.com/live-8> retrieved 04-03-2012 at 12.30.
- Anon. (2013). Csound for Live. Available at:
<http://www.csoundforlive.com/FAQ.html> retrieved 10_04_2013 at 18.30.
- Anon. (2013). MIDI. Available: http://www.amei.or.jp/index_e.html retrieved 24-06-2013 at 21.34.
- Anon. (2013). OSC. Available: <http://opensoundcontrol.org/> retrieved 24-06-2013 at 21.37.
- Anon. (2013). Tempophone. Available: <http://www.sfu.ca/sonic-studio/handbook/Tempophone.html> retrieved 23-06-2013 at 18.54.
- Anon. (2014) BTDSys. Available at:
<http://buzz.robotplanet.dk/machineinfo.php?id=7497> retrieved on 02 Oct 2014 at 18.07.
- Ash. M. (2013) Mac OS X Process Memory Statistics. Available:
<http://www.mikeash.com/pyblog/friday-qa-2009-06-19-mac-os-x-process-memory-statistics.html> retrieved on the 17/07/2013 at 18.25.
- Ashish. (2010). Software Product Development | Software Testing Tutorial | Software Process: Software requirements - Functional requirements [WWW Document], 2012. Posted by Ashish at 8/09/2010 12:03:00 AM. Available at:

<http://productdevelop.blogspot.com.es/2010/08/software-requirements-functional.html> Retrieved on 11-08-2012 at 16.50.

Barrière, F and Bennett, G. (1999) Eds. (Editions Mnemosyne, Bourges, France.

Behles, G., Starke, S., Röbel, A. (1998). Quasi-synchronous and pitch-synchronous granular sound processing with stampede ii. *Computer Music Journal*. vol. 22 (2) pp. 44-51).

Bencina, R. (2010). AudioMulch music performance and composition software. Available at: <http://www.audiomulch.com/> Retrieved on August 20, 2010.

Berndtsson, G. (2010). CMJ_VOL20_The KTH Rule System for Singing Synthesis. *Computer Music Journal*, pp.1–17.

Blackwell, A.F. (2001a) Cognitive Technology: Instruments of Mind: 4th International Conference, CT 2001 Blackwell, UK, August 6-9, 2001 Proceedings, p331.

Blackwell, A.F. et al. (2001b). Cognitive dimensions of notations: Design tools for cognitive technology. In M. Beynon, C.L. Nehaniv, and K. Dautenhahn (Eds.) *Cognitive Technology 2001 (LNAI 2117)*. Springer-Verlag, pp. 325-341.

Blackwell, T. M., and Young, M. W. (2004). Swarm Granulator. *Applications of Evolutionary Computing, Proc. Evo Workshops 2004 (LNCS 3005)*, pp. 399-408

Boulanger, R. (2000). *The Csound Book* MIT Press, p653-656, Publisher: MIT Press; Pap/Cdr edition (10 April 2000), Language English, ISBN-10: 0262522616, ISBN-13: 978-0262522618.

Boulanger, R. (2012). Boulanger Labs | Home Available at: <http://boulangerlabs.com/> Retrieved on 19-02-2013 at 11.31.

Boulez, P. (1971) *Boulez on Music Today*. Cambridge, Mass: Harvard University Press.

Brandt, E. (2001). "Hard Sync without Aliasing." *Proceedings of the 2001 International Computer Music Conference* . San Fransisco: International

Computer Music Association. Available: <http://www-2.cs.cmu.edu/~eli/L/icmc01/hardsync.html> retrieved on 20-02-2010.

Brandtsegg, Ø., Saue, S. & JOHANSEN, T., (2011). Particle synthesis—a unified model for granular synthesis. Available at <http://lac.linuxaudio.org/2011/papers/39.pdf> retrieved 21st April 2013 at 13.36.

Byrne Villez, P. (2000). Processing Samples with Csound's FOF Opcode, chapter 15, pages 307-320. In *The Csound Book* MIT Press, p653-656, Publisher: MIT Press; Pap/Cdr edition (10 April 2000), Language English, ISBN-10: 0262522616, ISBN-13: 978-0262522618.

Cascone, K. (2000). The aesthetics of failure: post-digital tendencies in contemporary computer music. *Computer Music Journal* 24(4): 12–18.

Cascone, K. (2009) *Microsound, Vague Terrains*. Available: <http://vagueterrain.net/journal15/> retrieved 21st April 2013 at 13.50.

Castagne, N., Cadoz, C. (2003). 10 criteria for evaluating physical modelling schemes for music creation. *Proceedings of the 6th Conference on Digital Audio Effects (DAFX-03)*.

Chafe, C., (2001). *A Short History of Digital Sound Synthesis by Composers in the USA*. unpublished. Available: <http://www-ccrma.stanford.edu/~cc/lyon/historyFinal.pdf> Retrieved on 20-03-10 at 17.50.

Chamberlain, H. (1980) *Musical Applications of Microprocessors*, Hadyn Book Company 1980.

Chowning, J. (1977) "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation." *Journal of the Audio Engineering Society* 21(7):526-534. Re- printed in *Computer Music Journal* 1(2):46-54.

Clark, J. J. (2010). Villez, P. FOF (Fonction d'Onde Formantique) Synthesis, Available at: http://www.cim.mcgill.ca/~clark/nordmodularbook/nm_oscillator.html#FOF on the 2010-07-31 at 16:24:30.

Clarke, M. (1996). Composing at the intersection of time and frequency. *Organised Sound*, 1(2), pp 107-117.

- Clarke, M. (1998). VOCEL. New implementations of the FOF synthesis method. In: Ch. Lischka and J. Fritsch, eds. Proceedings of the 1988 International Computer Music Conference. San Francisco: International Computer Music Association.
- Clarke, M. (2000). FOF and FOG synthesis in Csound, chapter 14, pages 293-306. In [Boulanger, 2000].
- Clarke, M. (2006). Chapter 6: Jonathan Harvey's *Mortuos Plango, Vovos Voco*. In *Analytical Methods of Electroacoustic Music*, Mary Simoni, ed. New York: Routledge, Taylor & Frahcis Group.
- Clarke, M. (2013) A FOF Synthesis Tutorial. Available: <http://www.csounds.com/tootsother/FOF/FOF.html> retrieved on the 11 May 2013 at 18.03.
- Cooper, A., Reimann, R., Dubberly, H. (2003). *About Face 2.0: The Essentials of Interaction Design* (1 ed.). John Wiley & Sons, Inc., New York, NY, USA.
- Cooper, S. (2000), Pulsar Studies review. <http://www.propheticdesire.us/microsound/html/2000/2000-04/msg00001.html> retrieved 8th April 2013 18.20
- Cornicello, A. (2010). Interview With Curtis Roads. pp. 1-17. *Seamus Online » Curtis Roads*. SEAMUSONLINE. P1-17. Available at: <http://www.seamusonline.org/?tag=curtis-roads> retrieved: August 8, 2010].
- Dack, J. (1999). Karlheinz Stockhausen's *Kontakte* and Narrativity. Available at: <http://cec.concordia.ca/econtact/SAN/Dack.htm> [Accessed August 13, 2010].
- De Poli, G., Piccialli, A., Roads, C. (1991). *Representations of musical signals*. MIT Press. ISBN-10: 0262041138.
- De Poli, G., Piccialli, A., Roads, C. (Eds.). (1991). *Representations of Musical Signals*. MIT Press, Cambridge, MA, USA. p119.
- De Poli, G., Piccialli, A., Roads, C. (Eds.). (1991). *Representations of Musical Signals*. MIT Press, Cambridge, MA, USA. p140.
- Deleuze, G (1998). *Boulez, Proust and time: "Occupying without counting"*. *Angelaki*. vol. 3 (2) pp. 69-74.

- Dubberly, H. (2001). Alan Cooper and the Goal Directed Design Process, *Gain* ALGA Journal of Design for the Network Economy Volume 1, Number 2, 2001
- Dufourt, H. (1991). *Musique, pouvoir, écriture*. Paris: C. Bourgois.
- Duignan, M., Noble, J., and Biddle, R. (2010). Abstraction and activity in computer-mediated music production. *Comput. Music J.* 34, 4 (Dec. 2010), 22-33. DOI= http://dx.doi.org/10.1162/COMJ_a_00023
- E. R. Miranda, E. (2002). *Computer Sound Design: Synthesis techniques and Programming*, Oxford: Focal Press, pp 101-102.
- Erkut, C. (2006). Towards physics-based control and sound synthesis of multi-agent systems: Application to synthetic hand clapping. *Nordic Music Technology Conference 2006*. pp. 12-14.
- Ernst, D. (1977) *The Evolution of Electronic Music*. London: Collier Macmillan Publishers.
- Everest, F. (2009). *The Master handbook of Acoustics 5th Ed -*, Publisher: Tab Electronics; 5 edition (1 July 2009), ISBN-10: 0071603328, ISBN-13: 978-0071603324.
- Feldstein, S. (2001). *Roland Drum machine rhythm dictionary*. Alfred Publishing Company ISBN-10: 0739027263, p 9-10.
- Feller, R. <http://www.computermusicjournal.org/reviews/36-3/feller-jaffe.html>. (retrieved 9th April 2013 17.46)
- Filter_7 (username). (2013). Available at: <https://forum.Ableton.com/viewtopic.php?f=1&t=176141> retrieved 10 April 2013 at 18.17.
- Fischman, R. (2003) *Clouds, Pyramids, and Diamonds: Applying Schrödinger's Equation to Granular Synthesis and Compositional Structure*. *Computer Music Journal*.vol. 27 (2) pp. 47-69.
- Fourier, J. (1807). *Mémoire sur la propagation de la chaleur dans les corps solides* p.215-221 Présenté Le 21 décembre 1807 à l'Institut national - *Nouveau Bulletin des sciences par la Société philomatique de Paris*, t. I, p. 112-116, n°6; mars 1808. Paris, Bernard. Available:

<http://gallica.bnf.fr/ark:/12148/bpt6k33707.image.f220> Retrieved 9th March 2010.

Gabor, D. (1946) "Theory of communication," J. IEE (London), vol. 93, pp. 429-457, 1946.

Gerhardt-Powals, J. (1996). Cognitive engineering principles for enhancing human-computer performance. *Int. J. Hum.-Comput. Interact.* 8, 2 (April 1996), 189-211.

Giovanni De Poli , Aldo Piccialli, Pitch-synchronous granular synthesis, Representations of musical signals, MIT Press, Cambridge, MA, (1991, pp 140-141.

Goodall, H. (2001). Big bangs-The story of five discoveries that changed musical history. London: Vintage.

Green, T. R. G. (1989) Cognitive dimensions of notations. In A. Sutcliffe and L. Macaulay (Eds.) People and Computers V. Cambridge, UK: Cambridge University Press, pp 443-460.

Green, T. R. G. (2000) Instructions and descriptions: some cognitive aspects of programming and similar activities. Invited paper, in Di Gesù, V., Levialdi, S. and Tarantino, L., (Eds.) Proceedings of Working Conference on Advanced Visual Interfaces (AVI 2000). New York: ACM Press, pp21-28.

Gross, R et-all. (1983). Program for the 1983 International Computer Music Conference, October 7-10, 1983

Harvey, J. Computer Music Journal, Vol. 5, No. 4 (Winter, 1981), pp. 22-24.

Harvey, J. et al. (1984). Notes on the realization of Bhakti. Contemporary Music Review, 1(1), pp.111–129, p77.

Hordijk, R. (2010). VOSIM, an advanced application of oscillator sync. Available at: <http://www.clavia.se/nordmodular/Modularzone/VOSIM.html> retrieved on the 31-07-2010 at 16:16:09.

Jaffe, D. (1995). Ten Criteria for Evaluating Synthesis Techniques. Computer Music Journal Vol. 19, pp.76–87.

- Jehan, T (2004) "Perceptual segment clustering for music description and time-axis redundancy cancellation," in Proceedings of the 5th International Conference on Music Information Retrieval, Barcelona, Spain.
- Johnson, J., Henderson, A. (2012). Conceptual Models Core to Good Design, Morgan & Claypool publishers. pp 9-10.
- Kaegi, Werner, and Tempelaars. (1978). VoSim - a new sound synthesis system. *Journal of the Audio Engineering Society*, 26(6):418--425, 1978.
- Kaner, C., Bach, J., Pettichord, B. (2001). *Lessons Learned in Software Testing*. John Wiley & Sons, Inc., New York, NY, USA.
- Karplus K, Strong A (1983). Digital Synthesis of Plucked String and Drum Timbres. *Computer Music Journal*, vol 7/2 – MIT Press.
- Keller, D. & Rolfe, C. (1998). The corner effect. Paper presented at the XII Colloquium on Musical Informatics, University of Udine, Gorizia, Italy. Retrieved October 26, 2006, from <http://www.sfu.ca/~dkeller/CornerEffect/CornerEffect.html>
- Krasner, G. (1980). Machine tongues VIII: The design of a Smalltalk music system. *Computer Music Journal*. pp. 4-14.
- Le Brun, M. (1978). Digital waveshaping synthesis *J. Audio Eng. Soc.*, 27:250-266.
- Lidwell, W., Holden, K., Butler, J., & Elam, K. (2010). *Universal principles of design: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Beverly, Mass, Rockport Publishers.
- Loy, D. G. (2006). *Musimathics, Volume 1 : A Guided Tour of the Mathematics of Music*, Cambridge, MA, USA: MIT Press, 2006.
- Lynn, P, A and Fuerst, W. (1999) *Introductory Digital Signal Processing with Computer Applications*, SOL 2 Rev t/a (Paperback).
- Manning, P. (2004). *Electronic and Computer Music*. New York, NY: Oxford University Press.

Miranda, E. (2001). *Composing Music with Computers (Music Technology)*. Focal Press.

Miranda, E. (2002), *Computer Sound Design: Synthesis techniques and Programming*, Oxford: Focal Press, pp 101-102.

Moore R. (1988). The dysfunctions of MIDI. *Computer Music Journal* (1988) vol. 12 (1) pp. 19-28

Moore, F. R. (1990). *Elements of Computer Music*, Upper Saddle River, NJ: PTR Prentice Hall

Mosc (2003) . *Modular 2003, London - Concert Review and Videos*. Available at: <http://www.electro-music.com/forum/topic-473.html> retrieved on 21st, April, 2013 at 13:49.

Nielsen, J. (1994a). *Guerrilla HCI: Using Discount Usability Engineering to Penetrate the Intimidation Barrier*. Available at: http://www.useit.com/papers/guerrilla_hci.html Retrieved on 07-08-2012 at 15:43.

Nielsen, J. (1994b). *Heuristic evaluation*. In Nielsen, J., and Mack, R.L. (Eds.), *Usability Inspection Methods*, John Wiley & Sons, New York, NY.

Nielsen, J. (1995a). *10 Heuristics for User Interface Design..* Available at: http://www.useit.com/papers/heuristic/heuristic_list.html Retrieved on 09-08-2012 at 10:22.

Nielsen, J. (1995b). *How to conduct a Heuristic evaluation*. Available at: http://www.useit.com/papers/heuristic/heuristic_evaluation.html Retrieved on 08-08-2012 at 23:11.

Nielsen, J. (1995c). *Technology Transfer of Heuristic Evaluation and Usability Inspection* Available at: http://www.useit.com/papers/heuristic/learning_inspection.html Retrieved on 08-08-2012 at 22:10.

Pearson, M. (2000). *Synthesis of organic sounds for electroacoustic music*, Thesis for Degree of DPhil in Music Technology Department of Electronics, University of York.

- Powley, E. (2004) BTDSys, BUZZ_pulsar_10.Manual. Available at:
<http://btdsys.lazytrap.com/remository?func=startdown&id=7> retrieved on 02-08-2010 at 17:07:00.
- Puckette, M. (2003) Theory and Techniques of Electronic Music. Available:
<http://crca.ucsd.edu/~msp/techniques/latest/book.pdf>, December 8. Retrieved 7-10-2008.
- Reybrouck. M. (1989). Music and the higher functions of the brain. Journal of New Music Research vol. 18 (1) pp. 73-88.....Page 82.
- Reynolds, C (1987), "Flocks, herds and schools: A distributed behavioral model.", SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques (Association for Computing Machinery): 25--34, doi:10.1145/37401.37406, ISBN 0-89791-227-6.
- Risset, J.C. (2005) Horacio Vaggione: Towards a syntax of sound, Contemporary Music Review, 24:4,287 — 293)
- Roads, C. (1978a). An Interview with Gottfried Michael Koenig. Computer Music Journal 2(3):11,15,29.
- Roads, C. (1978b). Automated granular synthesis of sound. Computer Music Journal. vol. 2 (2) pp. 61-62).
- Roads, C. (1978c). Computer Music Journal Automated Granular Synthesis of Sound. Volume 2, Number 2, p61.
- Roads, C. (1988) Computer Music Journal Introduction to Granular Synthesis. Volume 12, Number 2, p11-13.
- Roads, C. (1996). The Computer Music Tutorial. The MIT Press.
- Roads, C. (1999), "Time Scales of musical structure," in Actes V. Academie Internationale de musique electro-acoustique, F Barrière and G. Bennett, Eds. (Editions Mnemosyne, Bourges, France.
- Roads, C. (2001a). Microsound. MIT Press. ISBN 0-262-18215-7
- Roads, C. (2001b). Composing with Pulsars, AES Journal of the Audio Engineering Society, Volume 49 Number 3, March pp 134-147.

- Robindoré, B and Xenakis, I. (1996) *Eskhaté Ereuna: Extending the Limits of Musical Thought: Comments on and by Iannis Xenakis*. *Computer Music Journal* vol. 20 (4) pp. 11-16.
- Rodet, X.(1984). "Time-domain formant wave-function synthesis". *Computer Music Journal*, 8(3): 9–14.
- Ruch, A, (2004). *Joyce - Music: Pierre Boulez. The Modern World*. Available at: <http://www.themodernword.com/joyce/music/boulez.html> .Retrieved on August 9, 2010.
- Salvendy, G. (2012). *Handbook of Human Factors and Ergonomics*, 4th Edition. Gavriel Salvendy. ISBN: 978-0-470-52838-9. Hardcover. 1752 pages. March 2012, p306.
- Schoenberg, A. 1951. *Style and Idea*. London: Williams and Norgate.
- Schwarz, D. (2006). Concatenative sound synthesis: The early years. *Journal of New Music Research*,35:1,3 — 22.
- Shannon, C.E. (1948) "A Mathematical Theory of Communication", *Bell System Technical Journal*, vol. 27, pp. 379-423, 623-656, July, October.
- Schiff, D. 1995. *Unreconstructed Modernist*. Available at: <http://www.theatlantic.com/past/docs/issues/95sep/boulez.htm> Retrieved on 21-08-2013 at 17.24
- Skvorc, B. (2013). *Activity Monitor Demysified*. Available: <http://mac.tutsplus.com/tutorials/os-x/activity-monitor-demystified/> on the 18/07/2013 at 18.28.
- Smith, J. (2004) *Virtual acoustic musical instruments: Review and update*. *Journal of New Music Research*. vol. 33 (3) pp. 283-304.
- Stelkens, J. (2010). *accSone - audio - video - network software development*. Available at: <http://www.accsone.com/content/blogcategory/33/127//lang,english/> Retrieved on August 20, 2010.
- Stockhausen, K.H. (1959). "how time passes. . . ." *Die Reihe* (English Edition) p10-40.

Strange, A. (1984) *Electronic Music, Systems, Techniques and Controls*. William C Brown Pub.

Sturm, B. (2001) Synthesis and algorithmic composition techniques derived from particle physics. Eighth Biennial Symposium on Technology and the Arts.

Sundberg, J. (1978) "Synthesis of Singing." *Swedish Journal of Musicology* 60[1]:107-112.

Sundberg, J. (1989). "Synthesis of Singing by Rule." In Mathews, M. and J. Pierce, eds., *Current Directions in Computer Music Research*. Cambridge, Massachusetts: The MIT Press, pp. 45-56.

Sundberg, J. (2009). The KTH synthesis of singing. *Advances in Cognitive Psychology*, 2(2), 131–143.

Tarekith (username). (2013). Available at: <https://forum.Ableton.com/viewtopic.php?f=1&t=111880> retrieved on 10th April 2013 at 18.09.

Tempelaars, S. (1977). The VOSIM signal spectrum. *Journal of New Music Research*, Volume 6, Issue 2 September, pages 81 - 96.

Kaegi, W. (1973): A Minimum Description of the Linguistic Sign Repertoire (part I); *Interface*, 2, 141-156.

Truax, B. (1988). Computer Music Journal Real-Time Granular Synthesis with a Digital Signal Processing Computer Barry Truax p 14-26 Volume 12, Number 2 September.

Vaggione, H. (1993) Determinism and the false collective about models of time in early computer-aided composition. *Contemporary Music Review* vol. 7 (2) pp. 91-104).

Vaggione, H. (2001). Some ontological remarks about music composition processes. *Computer Music Journal*. vol. 25 (1) pp. 54-61.

Vercoe, B. (1986) *Csound: A manual for the audio processing system and supporting programs*. Technical report, MIT Media Lab.

Vercoe, B. (2013a). csound_manual. Available at:
<http://www.csounds.com/manual/html/index.html> retrieved on 18 May 2013 at 17.47.

Vercoe, B. (2013b). FOF. Available at:
<http://www.Csounds.com/manual/html/FOF.html> retrieved 8th April 2013 17.46.

Vercoe, B. (2013c). opcode. Available at: <http://www.csounds.com/docs>
retrieved on 18 May 2013 at 17.42.

Vercoe, B. (2013d). Partikkel. Available at:
<http://www.Csounds.com/manual/html/Partikkel.html> retrieved 8th April 2013 17.40.

Vercoe, B. (2013e). VoSim. Available at:
<http://www.csounds.com/manual/html/VoSim.html>, retrieved 10 April 2013 at 19.11.

Verwoest, A. (2013). Delay Lama. Available at: <http://www.audionerdz.com/>
retrieved 27_04_2013 at 18.00.

Villez, P. (2008) Composing with Waveholes and Microsounds. Proceedings from Reproduced Sound 24, Vol 30. Pt6, Immersive Audio, 20-21 November 2008. ISBN 1 901656 95 0, ISBN 1478-6095.

Villez, P. (2009) Elementary Signal Engine, Microsound, Vague Terrains. Available: <http://vagueterrain.net/journal15/pere-villez/02> retrieved 21st April 2013 at 13.54.

Villez, P. 2014a. Soundcloud. Primes. Available at:
<https://soundcloud.com/perevillez/prime-morphs> retrieved on 07, July, 2014 at 15.20

Villez, P. 2014b. Soundcloud. Inversions. Available at:
<https://soundcloud.com/perevillez/inversion-morphs> retrieved on 07, July, 2014 at 15.30

Villez, P. 2014c. Blog "The Villez Design. Electroacoustic Instrument Design. Available at: <http://perevillez.wordpress.com> retrieved 17.50 on 17 October 2014.

Villez, P. 2014d. Google Url Shortener. Available at: <http://goo.gl/Xm7Q0T> and <http://goo.gl/NLHzsP> retrieved on 25, October 2014 at 19.40

Villez, P. 2014e. ESE Soundcloud. Available at: <https://soundcloud.com/perevillez/> retrieved on 07, July, 2014 at 15.30

Warren, R.M. (2008). Auditory perception : an analysis and synthesis 3rd ed. Cambridge University Press. ISBNs:9780521868709, 052186870X.

Weinschenk, S and Barker, (2000) Designing Effective Speech Interfaces. Wiley.

Wiener, N (1948), Cybernetics: Or Control and Communication in the Animal and the Machine. Paris, France: Librairie Hermann & Cie, and Cambridge, MA: MIT.

Wierckx M. (2010). Real-Time Granular Synthesiser X. Available at: <http://www.lownorth.nl/software/rtgs/rtgs-info.html> Retrieved on August 20, 2010.

Wishart, T. (1996) On Sonic Art (Contemporary Music Studies).

World, P. (2012). Laptop Reviews: Specifications, Prices, Features and Rating of latest Budget, Mainstream, Ultraportable, Business Laptops in India with Best Buy products and Top 5 Rankings < PC World India Reviews < PCWorld. Available at: <http://www.pcworld.in/reviews/laptops> Retrieved on 12-08-2012 at 11.31.

Wright, M., Beauchamp, J., Fitz, K., Rodet, X., Röbel, A., Serra, X., and Wakefield, G. (2000). Analysis/synthesis comparison. Org. Sound 5, 3 (December 2000).

Xenakis, I. (1971). Formalized Music, Bloomington, IN: Indiana University Press

Xenakis, I. (1985). Music composition treks."Composers and the Computer pp. 170–92.

Xenakis, I. (1992). Formalized Music (Revised Edition),Pendragon Press, Stuyvesant NY.

Figures References

Anon. (2002). Fig.23. Audiomulch screenshot. Available at:
<http://www.audiomulch.com/> Retrieved October 26, 2006.

Software References

Anon. (2013). GyroOSC. <http://www.bitshapesoftware.com/instruments/gyrosc/>
retrieved on 02 May 2013 at 16.50.

Anon. (2013). Sonic Visualiser. Available at:
<http://www.sonicvisualiser.org/index.html> retrieved 02_05_2013 at 16.07.

Anon. (2013). Wavefront. Available at:
http://wavefrontlabs.com/Wavefront_Labs/Accelerometer_Data.html retrieved
02 May 2013 at 16.45.

Klingbeil, M. (2013). Spear. Software for spectral analysis, editing, and
synthesis. Available at: <http://www.klingbeil.com/papers/spearfinal05.pdf>
retrieved 01_05_2013 at 18.07.

Petrolati, A. (2013). Pulsaret. apeSoft. Available at: http://www.alessandro-petrolati.it/Manual/Pulsaret_user_manual.pdf retrieved on the 18 of May 2013 at
17.01.

Verwoest, A. (2013). Delay Lama. Available at: <http://www.audionerdz.com/>
retrieved 27_04_2013 at 18.00.

Discography references

Anon. IRCAM. (1983) IRCAM: un portrait," IRCAM 0001.

Anon. VoSim_b. (2013) Available at:
<http://www.youtube.com/watch?v=7GetTjx96D0>. Retrieved on 09_02_2012 at
14.31

Baptiste Barrière, J.P. (1983) Chreode 1, Unpublished, Festival International
des musiques expérimentales de Bourges, 1 June 1983.

Charles, Daniel. John Cage: Works for Percussion liner notes. Wergo WER
6203-2.

Clarke, M. (1987) Mälarsång, Refractions (MPSCD003).

Eimert, H. (1963). Elektronische Musik. WER 9000-6

Harvey, J (1990) Mortuos Plango Vivos Voco. Erato 2292-45409-2 Germany
1990

Roads Prototype (1975).

Stockhausen, K. 1964. Kontakte. WER 60009, WER 60 009

Xenakis Analogique (1958).

Bibliography

Amdahl, K. (1991) *There Are No Electrons: Electronics for Earthlings*. Broomfield, CO: Clearwater Publishing, 1991.

Anon. (1977). *The Evolution of Recordings...From Cylinder to Video Disc*," pamphlet published by the Audio Engineering Society.

Anon. (1987). *Le Corbusier: Architect of the Century*. Arts Council of Great Britain. Catalogue for the Le Corbusier centenary exhibit held at the Hayward, Gallery, London.

Anon. (2011). *Electronica primer*. Available: <http://phobos.plato.nl/e-primer/> Retrieved 23-06-11 at 12.31. This resources is now a dead-link.

Appignanesi, R and Garratt, C. (1995). *Introducing Postmodernism*. London, England: Icon Books UK.

Ballora, M. (1995). *History and Current Developments in Stereophonic Audio*. Master's thesis in Music Technology, New York University.

Becker, C., Wiens,A. eds. (1995). *The Artist in Society: Rights, Roles, and Responsibilities*. Chicago, IL: New Art Examiner Press.

Borio, G. (2011). 'Nono, Luigi', *The New Grove Dictionary of Music Online*. Available at: <http://www.grovemusic.com>, ed. L. Macy. Retrieved 21-06-2011 at 12.34

Buxton, W. (2013). *Structured Sound Synthesis Project (SSSP) Videos*. Available at: <http://www.billbuxton.com/buxtonSSSPVideos.html>. Retrieved 14-08-2013.

- Carlos, W. (2010). Source of information on a variety of topics including synthesis, stereophony, and microtonality. Available at: www.wendycarlos.com. Retrieved 07-11-2010 at 15.49.
- Chadabe, J. (1997). *Electric Sound: The Past and Promise of Electronic Music*. Upper Saddle River, NJ: Prentice Hall
- Chusid, I., Gurewitz, J. (1997). Liner notes for *Soothing Sounds for Baby: Electronic Music by Raymond Scott*. Reissued by BASTA Audio/Visuals.
- Debussy, C, et al. (1962). *Three Classics in the Aesthetic of Music*. New York: Dover Publications, Inc..
- Deleuze, G., Félix G. (1980). *A Thousand Plateaus*. Trans. Brian Massumi. London and New York: Continuum, 2004. Vol. 2 of *Capitalism and Schizophrenia*. 2 vols. 1972-1980. Trans. of *Mille Plateaux*. Paris: Les Editions de Minuit. ISBN 0-8264-7694-5.
- Gann, K. (1997). *American Music in the Twentieth Century*, New York, NY: Schirmer Books,.
- Giroux, H. (1994). *Disturbing Pleasures: Learning Popular Culture*, New York, NY: Routledge, 1994.
- Gluck, R.J. (2007) "The Columbia-Princeton Electronic Music Center: Educating International Composers." *Computer Music Journal* 31 (2).
- Hawking, S. (1996). *A Brief History of Time*. New York: Bantam Books, 1988, 1996.
- Helmuth, M. (2006). Chapter 8: Barry Truax's *Riverrun*. In *Analytical Methods of Electroacoustic Music*, Mary Simoni, ed. New York: Routledge, Taylor & Frahcis Group.
- Holman, T. (2005). *The History and Future of Surround Sound*. Presentation given to the PSU Audio Engineering Society, April 15, 2005.
- Matossian, N. (1986). *Xenakis*, London : Kahn & Averill; New York : Taplinger.
- McClary, S. (2000). *Conventional Wisdom*. Berkley, CA: University of California Press.

- Meneghini, M. (2007). An Analysis of the Compositional Techniques in John Chowning's Stria. *Computer Music Journal* 31(3), Fall 2007.
- Neill, B. (2002). Pleasure Beats: Rhythm and the Aesthetics of Current Electronic Music. *Leonardo Music Journal* 12.
- Ouellette, F. (1981). *Edgard Varèse*. Translated from the French by Derek Coltman. New York: Da Capo Press, 1968.
- Pasler, J. (2012). Postmodernism. *The New Grove Dictionary of Music Online* ed. L. Macy. Available at: <http://www.grovemusic.com>. Retrieved 23-05-2012.
- Pinch, T., Trocco, F. (2002). *Analog Days*. Cambridge, MA: Harvard University Press.
- Rowe, R. (1993). *Interactive Music Systems*. Cambridge, MA: MIT Press, 1993.
- Rowe, R. (2001). *Machine Musicianship*. Cambridge, MA: MIT Press, 2001.
- Smith, D.O. 'Berio, Luciano', *Grove Music Online* ed. L. Macy. Available at: (<http://www.grovemusic.com>). Retrieved on 24-05-2012.
- Treib, M. (1996). *Space Calculated in Seconds: The Philips Pavilion*. Princeton, NJ: Princeton University Press.
- Upton, M. (1957). *Electronics for Everyone*, New York: Devin-Adair.
- Vaughan, D. (1997). *Merce Cunningham: Fifty Years*. New York, NY: Aperture